



HAL
open science

Actes des 19es Rencontres des Jeunes Chercheurs en Intelligence Artificielle

Abir-Beatrice Karami

► **To cite this version:**

Abir-Beatrice Karami. Actes des 19es Rencontres des Jeunes Chercheurs en Intelligence Artificielle. Plate-Forme Intelligence Artificielle, Association Française pour l'Intelligence Artificielle, 2021. hal-03331492

HAL Id: hal-03331492

<https://hal.science/hal-03331492v1>

Submitted on 1 Sep 2021

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

*Rencontre des Jeunes Chercheurs
en Intelligence Artificielle*

PFIA 2021



Crédit photo : [Flicier/xlibber](#)

Table des matières

Abir KARAMI	
Éditorial	5
Comité de programme	6
V. Antuori, E. Hébrard, M.-J. Huguet, S. Essodaigui, A. Nguyen	
Étude de méthodes arborescentes de Monte-Carlo pour un problème de déplacement de pièces dans un atelier d'assemblage	7
A. Audic, L. Thevin	
COrHA : Architecture multi-agent pour la collaboration Utilisateurs-Objets connectés dans une tâche en domotique	14
S. Beaussant, S. Lengagne, O. Stasse	
Delay Aware Universal Notice Network : Real world multi-robot transfer learning	22
M. Cazes, N. Franiatte, A. Delmas, J.-M. André, M. Rodier, I. Chraïbi Kaadoud	
Evaluation of the sensitivity of cognitive biases in the design of artificial intelligence	30
H. Fargier, P. Jourdan, R. Sabbadin	
Jeux bayesiens hypergraphiques	38
H. Fargier, É. Martin-Dorel, P. Pomeret-Coquot	
Jeux incomplets algébriques	46
J. Ferry, U. Aïvodji, S. Gambs, M.-J. Huguet, M. Siala	
Améliorer la généralisation de l'équité en apprentissage grâce à l'Optimisation Distributionnellement Robuste	54
M. Guery, G. Blot, F. Rousseaux, E. Soulier, S. Cormier	
Je relie système complexe, combinatoire, biologie des systèmes et apprentissage profond, qui suis-je ? Hyper-réseau, définitions et cas d'usage	58
L. Pinos Ullauri, W. Van den Noortgate1, D. Debeer	
Modelling response time and impact of instructional level of support	65
J. Bernardo, A. Rangel Trejo, L. Werling, W. Uhring, L. Hebrard, C. Gontrand, L. Fakri-Bouchet	
Implantable NMR Microprobe Design using a Machine learning approach	73
C. Bout, A. Karami, A. Fleury	
L'habitat intelligent et sa capacité à s'adapter : vers une approche centrée multi-résidents	75
M. Collery, P. Bonnard, F. Fages	
Apprentissage de règles appliquée à la détection de fraudes	77
C. Juvigny, J. Baste, G. Lozenguez, A. Doniec, L. Jourdan	
Digitalisation de la déconstruction sélective : simulation et optimisation des filières	79
M. Lopez	
Prédiction d'erreurs critiques pour l'aide à la maintenance industrielle	82
C. Ostertag, M. Beurton-Aimar, M. Visani, T. Urruty	
Réseau de Neurones Siamois Multimodal pour la prédiction d'évolution de maladies neurodégénératives	84
K. Réby, M. Beurton-Aimar	
Reconnaissance d'émotions à partir de la posture par LSTM	86
J. Signe	
Extraction de sous-groupes exceptionnels de séries temporelles	89
A. Rangel Trejo, J. A. Bernardo, L. Konadu, L. Fakri-Bouchet	
AI's potential for Artificial Phantom Mimicking Tissue of the Human Breast Electrical Properties	91

J.-A. Vayssade, J.-N. Paoli, C. Gée, G. Jones

DeepIndices : Une nouvelle approche des indices de télédétection basée sur l'optimisation et l'approximation de fonctions par DeepLearning. Application aux indices de végétation sur des données non calibrées 94

Éditorial

Rencontre des Jeunes Chercheurs en Intelligence Artificielle

Les dix-neuvièmes Rencontres des Jeunes Chercheur.ses en Intelligence Artificielle (RJCIA 2021) se sont déroulées les deux derniers jours de l'édition 2021 de la Plate-Forme Intelligence Artificielle (PFIA), qui a eu lieu du 28 juin au 2 juillet.

Les RJCIA sont destinées aux jeunes chercheurs en IA, doctorants ou titulaires d'un doctorat depuis moins d'un an. L'objectif de cette manifestation est de permettre aux jeunes chercheurs préparant une thèse en Intelligence Artificielle, ou l'ayant soutenue depuis peu, de se rencontrer et de présenter leurs travaux, et 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.

Pour cette année 2021 de la conférence, nous avons eu l'honneur de recevoir professeur Manuela Veloso – Head de J.P. Morgan AI Research et professeur à Carnegie Mellon University –, qui a donné une conférence invitée intitulée « AI in Finance : Scope and Examples ».

Concernant les contributions scientifiques, 22 articles ont été soumis. Au total 19 articles ont été acceptés et constituent le contenu de ces actes. Ces articles sont de plusieurs types. 9 articles longs présentent des contributions originales dans les thèmes de la conférence. 10 articles courts qui présentent les sujets de recherche des jeunes chercheurs.

RJCIA'21 comme tout les conférences et événements hébergés de PFIA'21 se sont déroulés de manière virtuelle. Cela n'a pas affecter le niveau d'intérêt et de participation à RJCIA'21. Je remercie l'ensemble d'acteurs ; auteurs, participants et membres du comité de programme, pour leur implication dans la réussite de RJCIA'21.

Abir KARAMI

Comité de programme

Président

- Abir Karami - Université Catholique de Lille

Membres

- Grégory Bonnet - Université de Caen Normandie
- Zied Bouraoui - Université d'Artois
- Olivier Champalle - Université Paris Est
- Mathieu Chollet - IMT Atlantique
- Anthony Fleury - IMT Lille Douai
- Emmanuelle Grislin - Université de Valenciennes
- Julie Jacques - Université Catholique de Lille
- Alexis Lebis - IMT Lille Douai
- Guillaume Lozenguez - IMT Lille Douai
- Jean-Guy Mailly - Université de Paris Descartes
- Laetitia Matignon - Université de Lyon 1
- Anastasia Paparrizou - Université d'Artois
- Alexandre Pauchet - INSA Rouen
- Karim Sehaba - Université Lumière Lyon 2
- Nicolas Verstaevel - Université Toulouse 1 Capitole

Étude de méthodes arborescentes de Monte-Carlo pour un problème de déplacement de pièces dans un atelier d'assemblage

V. Antuori^{1,2}, E. Hébrard^{1,3}, MJ. Huguet¹, S. Essodaigui², A. Nguyen²

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

² Renault, France

³ ANITI, Université de Toulouse, France

{vantuori,hebrard,huguet}@laas.fr
{siham.essodaigui,alain.nguyen}@renault.com

Résumé

La recherche arborescente Monte-Carlo (MCTS) connaît un développement important pour la résolution de problèmes d'optimisation combinatoire, en particulier, lorsque les mécanismes d'inférence ne passent pas à l'échelle, ou sont trop faibles pour réduire l'espace de recherche. Dans cet article, nous appliquons la méthode MCTS à un problème de voyageur de commerce avec fenêtres de temps et contraintes de capacité, issu d'une chaîne de montage dans la construction automobile. Des adaptations du MCTS de base sont proposées et analysées via une étude expérimentale afin de dégager des pistes génériques pour la résolution de problème d'optimisation combinatoire.

Mots-clés

Recherche arborescente de Monte-Carlo, Ordonnancement, Voyageur de Commerce.

Abstract

The development of the Monte-Carlo Tree Search (MCTS) for combinatorial optimisation has grown significantly, in particular when inference mechanisms do not scale, or are too weak to reduce the search space. In this paper we apply the MCTS to a traveling salesman problem with time windows and a capacity constraint from an assembly line in car manufacturing. Adaptations of the basic MCTS are proposed and analysed via an experimental study in order to identify generic components for the resolution of combinatorial optimisation problems.

Keywords

Monte-Carlo Tree Search, Scheduling, Traveling Salesman Problem

1 Introduction

Pour de nombreux problèmes combinatoires, explorer de façon exhaustive un arbre de recherche serait trop coûteux. En effet, l'espace de recherche peut rarement être suffisamment réduit par des méthodes d'inférence exactes pour que ces méthodes passent à l'échelle.

La méthode Monte-Carlo Tree Search (MCTS) [4, 3] offre une stratégie générique pour s'attaquer à ces problèmes en ne nécessitant que très peu de connaissances spécifiques du problème. Cet algorithme a été proposé dans le contexte des jeux et a contribué aux récents progrès de l'intelligence artificielle, par exemple pour le Go [11].

Dans cette méthode, des simulations guident l'expansion de l'arbre de recherche et maintiennent un compromis entre des phases d'exploitation et d'exploration. Plus précisément, une simulation est l'action d'extension d'un noeud jusqu'à un état final. Cette extension est généralement faite par une heuristique "randomisée". Le résultat obtenu par simulation est ensuite rétro-propagé à ce noeud et à tous ses ancêtres jusqu'à la racine. C'est cette information qui sert de guide pour les itérations futures de l'algorithme via un mécanisme de bandit manchot.

On trouve dans la littérature plusieurs applications de la méthode à des problèmes d'optimisation combinatoire. Dans [7], les auteurs appliquent l'algorithme à un problème de voyageur de commerce avec drone, ils considèrent leur problème comme un jeu où une victoire est l'amélioration de la borne supérieure (amélioration de la meilleure solution déjà obtenue). [2] applique la méthode à deux problèmes stochastiques de gestion des incendies et de contrôle de réseau de files d'attente. Enfin la méthode est aussi appliquée à des problèmes de type ordonnancement [5, 9]. Différentes hybridations de la méthode MCTS avec des méthodes d'optimisation combinatoire ont été proposées. Dans [10], l'algorithme de bandits manchots du MCTS a été utilisé en particulier dans la programmation mixte en nombres entiers (MIP), en remplaçant les simulations de Monte-Carlo par une borne inférieure de la relaxation linéaire. Dans [6], la méthode MCTS a été combinée avec la programmation par contraintes (CP). Pour obtenir une méthode exacte et permettre des redémarrages, les auteurs ont dû modifier de manière significative plusieurs aspects de l'algorithme. La combinaison avec la satisfiabilité booléenne (SAT) proposée par [8] est plus fidèle aux principes de MCTS, cependant, dans ce cas elle n'inclut pas les caractéristiques des solveurs SAT modernes (apprentissage de clause, VSIDS, etc.).

Nous présentons une étude du MCTS appliquée à un problème industriel de type *voyageur de commerce*.

Pour ce problème, étudié dans [1], les méthodes de propagation de contraintes couplé à une heuristique adéquate obtiennent rapidement une solution réalisable pour une grande majorité d'instances industrielles. Néanmoins, sur des instances plus contraintes¹ l'obtention d'une telle heuristique n'est pas triviale

Pour cela une méthode d'apprentissage par renforcement a été utilisée, et l'heuristique ainsi découverte a permis, en conjonction avec d'autres méthodes combinatoires, de surpasser les méthodes employés par l'entreprise. Cependant, pour certains horizons temporels cette approche ne permet pas de résoudre l'ensemble des instances. Au vu de l'essence même de la méthode MCTS qui consiste à guider la recherche par de nombreuses simulations heuristiques, nous nous intéressons à son application pour notre problème, en s'appuyant sur nos précédents travaux.

Nous proposons alors de définir un cadre générique de l'application de la méthode pour les problèmes d'optimisation combinatoires, et d'utiliser ce cadre pour résoudre notre problème. Nous proposons également trois *composants additionnels*, que l'on ne retrouve habituellement pas dans cette méthode, qui aident à la résolution du problème étudié tout en restant génériques.

2 Description du problème

Le problème étudié consiste à déplacer un ensemble de m composants de véhicule à travers une chaîne d'assemblage d'un constructeur automobile, de leur point de production vers le lieu de consommation. Chaque composant est produit et consommé par deux machines uniques, et le déplacement de l'une à l'autre machine est effectué au moyen de chariot, spécifique à chaque composant. Lorsqu'un chariot est plein à son lieu de production, un opérateur doit l'amener à son lieu de consommation, et de manière symétrique, lorsqu'un chariot est vide au lieu de consommation, il doit être ramener au lieu de production correspondant. Un cycle de production est le temps c_i nécessaire pour produire (resp. consommer) le composant i , ou plutôt pour remplir (resp. vider) le chariot correspondant. La fin d'un cycle de production marque le début du suivant, et il y a donc $n_i = \frac{H}{c_i}$ jusqu'au à l'horizon temporel H . Pour chacun des cycles k de chaque composant i , il y a deux opérations de collectes, et deux opérations de livraison : la collecte pe_i^k et la livraison de_i^k du chariot vide du point de consommation vers le le point de production, et la collecte pf_i^k et la livraison df_i^k du chariot plein de la production vers la consommation. Chaque opération a prend un temps p_a pour être effectué, et le temps de déplacement entre l'opération a est b prend $D_{a,b}$ unités de temps.

On note A l'ensemble de toute les opérations. Le problème est alors de trouver une permutation σ des $|A| = n$ opérations, respectants les contraintes suivantes² :

1. Un jeu de données académiques s'inspirant de données industrielles a été développé pour palier le manque de données réelles.

2. Voir [1] pour une description plus détaillé

Routes : Pour tout $1 < j \leq n$, l'opération $\sigma(j)$ doit étant donné une date de début $s_{\sigma,j}$ (et une date de fin $e_{\sigma,j} = s_{\sigma,j} + p_{\sigma(j)}$) prendre en compte la durée et le temps de trajet : $s_{\sigma,j} \geq s_{\sigma,j-1} + p_{\sigma(j-1)} + D_{\sigma(j-1),\sigma(j)}$ (et $s_{\sigma,1} = 0$).

Fenêtre de temps : Une opération a de la période k du composant i a une date de disponibilité $r_a = (k-1)c_i$ et une date d'échéance $d_a = kc_i$, avec $r_a \leq s_a$ et $e_a \leq d_a$.

Précédences : Les collectes doivent précéder leur livraison.

Longueur du train : L'opérateur peut accrocher les chariots les uns aux autres de manière à former un train de chariot, mais la longueur total du train ne doit pas excéder une longueur donnée T_{\max} .

Ce problème est un cas particulier du problème de collecte et livraison à un véhicule avec contrainte de capacité, et fenêtres de temps. Cependant il n'y a pas de fonction objectif à optimiser, et au contraire la faisabilité est difficile. De plus, les cycles de production-consommation impliquent une structure très particulière : les quatre opérations de chaque composant à chaque cycle doivent être effectué durant la même fenêtre de temps, et ceci se répétant à chaque cycle.

Nous travaillons ici avec une relaxation du problème original, en relâchant les contraintes sur les dates d'échéances, nous cherchons à minimiser le retard maximum :

$$L(\sigma) = \max(0, \max_{1 \leq j \leq |\sigma|} (e_{\sigma(j)} - d_{\sigma(j)}))$$

Pour ce faire nous devons ajouter une contrainte de précédence additionnelle : pour toute paire d'opération a, b des cycles $k < p$ respectivement, d'un même composant, on a nécessairement a qui doit précéder b .

3 MCTS en optimisation combinatoire

3.1 Principes généraux

La méthode MCTS explore un espace de recherche avec une structure arborescente et peut être appliquée à l'optimisation combinatoire avec seulement quelques hypothèses faibles sur le modèle du problème.

Dans l'arbre de recherche développé, un noeud correspond à un état du problème. Chaque noeud est associé à un ensemble de décision menant à des noeuds enfants dans l'arbre. Le but est de trouver un chemin du noeud racine (état initial) à un état final (une solution du problème). A chaque solution, il est possible de calculer la valeur de la fonction objectif et sans perte de généralité, nous supposons ici que nous cherchons une solution de valeur minimale. La méthode MCTS repose sur une succession de quatre phases : (1) une phase de sélection, généralement basé sur un bandit manchot, pour guider l'expansion de l'arbre, et assurer un compromis entre l'exploitation (sélectionner le noeud le plus prometteur) et l'exploration (acquérir de nouvelles connaissances sur d'autres parties de l'arbre); (2) une

phase d'expansion ou des noeuds sont ajoutés dans l'arbre ; (3) une phase de simulation, également appelé "rollout" ou "payout", utilisé pour produire rapidement un chemin d'un noeud donné vers un état final (par exemple basé sur un échantillonnage aléatoire) ; et (4) une mise à jour des informations sur les noeuds pour prendre en compte la nouvelle simulation, informations utiles aux phases de sélections futurs.

Notations. Soit \mathcal{A} un ensemble de décisions, un état $\sigma \in \mathcal{A}^*$ représente une séquence de décisions et $p(\sigma)$ l'état parent de σ . De plus, on note $\sigma; a$ l'état atteint en appliquant la décision a dans l'état σ , et $\mathcal{A}(\sigma)$ l'ensemble des décisions possibles dans l'état σ . A chaque étape de l'algorithme, de nouveaux états sont ajoutés dans un arbre \mathcal{T} qui ne contient initialement que l'état initial en tant que racine. Pour chacun des états de l'arbre le triplet suivant est mémorisé : $\{N(\sigma), Pr(\sigma), V(\sigma)\}$ où $N(\sigma)$ est le nombre de fois que σ a été visité, $Pr(\sigma)$ est une probabilité a priori (une préférence issue de connaissances spécifique sur le problème) de prendre la décision qui mène à l'état σ à partir de son état parent $p(\sigma)$, et $V(\sigma)$ est une évaluation du noeud calculée à partir des résultats des simulations issues de σ . Habituellement en optimisation combinatoire, l'évaluation du noeud est la moyenne des valeurs de la fonction objectifs des solutions obtenues par les simulations des itérations antérieures passées par ce noeud. Nous proposons une autre évaluation également basé sur la valeur de la fonction objectif dans la section 3.2

Sélection. La phase de sélection commence au noeud racine, et se termine lorsqu'une feuille de \mathcal{T} est atteinte. À chaque noeud σ , une décision est sélectionnée en fonction des valeurs de chaque triplet des noeuds fils :

$$a^* = \arg \max_{a \in \mathcal{A}(\sigma)} \tilde{V}(\sigma; a) + c * U(\sigma; a) \quad (1)$$

où $\tilde{V}(\sigma; a)$ est le terme d'exploitation, $U(\sigma; a)$ le terme d'exploration, et c est un paramètre qui représente le compromis entre les deux termes. Le terme d'exploitation $\tilde{V}(\sigma; a)$ peut être directement $V(\sigma)$, $V(\sigma)$ étant basé sur la valeur de la fonction objectif, cette valeur peut être très différente d'une instance d'un problème à l'autre, ainsi pour faciliter le réglage du paramètre c , nous proposons de normaliser cette valeur dans $[-1, 1]$:

$$\tilde{V}(\sigma; a) = \begin{cases} 2 * \frac{V^+(\sigma) - V(\sigma; a)}{V^+(\sigma) - V^-(\sigma)} - 1 & \text{if } N(\sigma; a) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Avec :

$$V^+(\sigma) = \max_{a \in \mathcal{A}(\sigma), N(\sigma; a) > 0} V(\sigma; a)$$

$$V^-(\sigma) = \min_{a \in \mathcal{A}(\sigma), N(\sigma; a) > 0} V(\sigma; a)$$

Le terme d'exploration $U(\sigma; a)$ est celui utilisé dans [11] :

$$U(\sigma) = Pr(\sigma) \frac{\sqrt{N(p(\sigma))}}{N(\sigma) + 1} \quad (3)$$

Le compromis effectué vise à choisir l'action qui maximise $\tilde{V}(\sigma; a)$ auquel s'ajoute un bonus qui diminue à chaque visite afin de promouvoir l'exploration. La probabilité a priori biaise l'exploration par des connaissances initiales sur l'état (si elles existent). Ce processus se répète jusqu'à arriver à une feuille de l'arbre \mathcal{T} .

Expansion. Soit σ le noeud retourné par la phase de sélection. Pour toute décision $a \in \mathcal{A}(\sigma)$, la phase d'expansion ajoute un noeud enfant à σ et initialise chaque triplet $\{N(\sigma; a), Pr(\sigma; a), V(\sigma; a)\}$. Le nombre de visites $N(\sigma; a)$ et la valeur $V(\sigma; a)$ de chaque enfant de σ sont fixés à 0. Il est également possible d'initialiser les probabilité a priori $Pr(\sigma; a)$, si cette distribution n'est pas connue, la distribution uniforme $1/|\mathcal{A}(\sigma)|$ peut être utilisée. Dans cette phase d'expansion, nous pouvons empêcher la création de certains noeuds menant nécessairement à un échec, par exemple si l'on peut évaluer la borne inférieure d'un noeud que l'on veut ajouter, et que celle ci dépasse la borne supérieure (meilleure solution connue). Il se peut alors que la phase d'expansion supprime un noeud lorsqu'il n'a plus d'enfant possible.

Simulation Le noeud σ obtenu par la phase de sélection est étendu jusqu'à une solution en appliquant généralement un échantillonnage aléatoire des actions possibles jusqu'à un état final. Par exemple, on peut utiliser la distribution de probabilité donnée par $Pr(\sigma; a) \mid a \in \mathcal{A}(\sigma)$ ou toute heuristique gloutonne "randomisée" adaptée au problème traité.

Rétropropagation Pour chaque noeud σ' traversé durant la phase de sélection de la racine à σ , le nombre de visites est incrémenté et la valeur $V(\sigma')$ est alors mise à jour en prenant en compte le résultat de la simulation. Pour des problèmes de type jeux, le résultat d'une simulation est généralement -1 (partie perdue), 1 (partie gagnée) ou 0 (partie nulle). En revanche, pour un problème d'optimisation nous ne disposons pas d'un tel résultat, mais de la valeur de fonction objectif. Dans [7], les auteurs considèrent une amélioration de la meilleur solution courante comme une partie gagnée, pour se rapprocher du MCTS utilisé dans les jeux. Bien souvent, le résultat d'une simulation est la valeur de la fonction objectif elle même [5, 9], et comme évoqué précédemment, dans ce cas, $V(\sigma)$ représente alors l'espérance de l'objectif au noeud σ .

3.2 Adaptations

Dans un contexte d'optimisation combinatoire, différentes adaptations de la méthode MCTS peuvent être considérées.

Evaluation. Dans un problème d'optimisation combinatoire, nous disposons généralement d'une borne inférieure sur la valeur de la fonction objectif qui croit de façon monotone pour chaque décision gloutonne prise. Par conséquent, l'évolution de cette borne pendant la simulation nous permet d'avoir une bonne idée de la qualité de la décision initiale. De plus, dans une procédure gloutonne, on peut conjecturer qu'à mesure que la procédure avance, la corrélation entre la qualité de la décision initiale, et l'accroissement de la borne inférieure diminue. On propose alors

d'évaluer la valeur $V(\sigma)$ d'un noeud σ comme l'espérance de la somme des accroissements marginaux de la borne inférieure obtenus après ce noeud, pondéré par un coefficient γ , décroissant exponentiellement. L'algorithme 1 décrit la procédure d'évaluation et de rétropropagation. Cette procédure prend en entrée la séquence des accroissements marginaux de la borne inférieure obtenue par la simulation R (R_i est alors égal à la différence de borne inférieure entre l'étape i et $i - 1$ lors de la simulation), et calcule une évaluation pour le noeud sélectionné par la procédure de sélection. Ensuite tous les noeuds σ jusqu'à la racine sont mis à jour pour que $V(\sigma)$ reflète l'espérance de son impact dans l'accroissement de la borne inférieure future. Dans l'algorithme, $LB(\sigma)$ est l'évaluation de la borne inférieure au noeud σ .

Algorithme 1 : Algorithme de rétropropagation

Données : R : séquence des accroissements marginaux de la borne inférieure obtenue par la simulation, σ : noeud sélectionné par la phase de sélection, γ : facteur de décroissance

```

1 // Somme pondérée des accroissements marginaux de la simulation
2  $val \leftarrow \sum_{i=1}^{|R|} \gamma^{i-1} R_i$  // Rétropropagation jusqu'à la racine incluse
3 répéter
4    $val \leftarrow \gamma * val + LB(\sigma) - LB(p(\sigma))$ 
5    $N(\sigma) \leftarrow N(\sigma) + 1$ 
6    $V(\sigma) \leftarrow V(\sigma) + \frac{val - V(\sigma)}{N(\sigma)}$ 
7    $\sigma \leftarrow p(\sigma)$ 
8 jusqu'à  $\sigma = Nil$ ;
```

Compromis dynamique exploitation-exploration. Les expériences préliminaires nous montrent qu'il est possible d'adapter dynamiquement le paramètre contrôlant le compromis entre l'exploration et l'exploitation, en fonction de la profondeur de l'arbre, afin de forcer l'exploitation sur la partie haute de l'arbre au fur et à mesure de sa croissance. En notant $d(\mathcal{T})$ la hauteur de l'arbre \mathcal{T} , le coefficient de compromis lors de l'itération t de la phase de sélection s'écrit alors :

$$\beta^{d(\mathcal{T})-t} * c \quad (4)$$

où $\beta < 1$ est un paramètre. Ce compromis dynamique est similaire à une phase de "commit" au noeud racine qui est généralement effectué dans le MCTS, [2, 7] mais faite de manière plus douce : plus la hauteur de l'arbre augmente, moins la décision à la racine est remise en cause.

Approfondissement de simulations. Afin d'améliorer la phase de simulations, nous proposons d'utiliser un Depth First Search (DFS) à budget limité en guise de processus d'intensification. Le principe est de remplacer la procédure gloutonne par un DFS, et de décider d'un budget à accorder lors du premier échec détecté lors de la simulation, par

exemple lorsque la borne inférieure dépasse la borne supérieure. Pour cela, on évalue la pertinence de cette première branche (du noeud sélectionné, jusqu'à l'échec). Si cette branche est prometteuse alors nous accordons un budget de recherche plus ou moins important en fonction de la qualité de la branche et un DFS est lancé. Si elle ne l'est pas, alors nous basculons vers la procédure gloutonne pour étendre cette solution partielle jusqu'à une feuille, sans DFS. Lorsque le budget alloué est totalement dépensé deux cas sont possibles. Soit on a obtenu une solution améliorante, et cette solution est retenue. Soit on sélectionne la meilleure solution partielle (selon le même critère que pour définir le budget) et on l'étend jusqu'à une solution par la procédure gloutonne. Plus de détail sur ce mécanisme sont donnés dans la section 4, notre proposition n'étant pas assez générique pour figurer dans cette section.

4 MCTS pour le problème de collecte et livraison

Dans cette section nous donnons des précisions sur la manière dont nous utilisons la méthode pour notre problème de collecte et de livraison

Dans ce problème, la racine de l'arbre consiste en une séquence vide d'opération et une décision consiste à choisir une opération de collecte ou de livraison que l'on ajoute en fin de la séquence d'opérations déjà réalisées. Comme l'objectif *in fine* est d'obtenir une solution sans aucun retard, lors de la phase d'expansion un noeud dont un des enfants est en retard sera supprimé car cette branche ne mènera à aucune solution. Durant les phases de simulation et de rétro-propagation, l'accroissement de la borne inférieure correspond à l'accroissement du retard maximal lorsque l'on ajoute une opération.

Enfin, nous utilisons une heuristique gloutonne stochastique dédiée, configurée par apprentissage par renforcement pour les "rollouts" [1]. Cette procédure détermine une distribution de probabilité sur l'ensemble des opérations disponibles dans un état donné. C'est cette même distribution qui est également utilisée lors de la phase d'expansion, pour obtenir la distribution de probabilité sur les enfants d'un noeud, et qui est ensuite utilisée lors de la phase de sélection (3). Dans un état donné σ , on décrit chaque opération $a \in \mathcal{A}(\sigma)$ par un vecteur de 4 critères :

1. *L'urgence* d'une opération : $l_{st}(a, \sigma) - \max(r_a, e_{\sigma, | \sigma |} + D_{\sigma(| \sigma |), a})$, avec $l_{st}(a, \sigma)$ la date de début au plus tard de l'opération a pour satisfaire sa date d'échéance, compte tenu des opérations déjà présentes, et des contraintes de précédences;
2. *Le temps de trajet/temps d'attente* de l'opération : $\max(D_{\sigma(| \sigma |), a}, (r_a - e_{\sigma(| \sigma |)}))$;
3. *La longueur* du chariot;
4. *Le type* de l'opération, 1 pour les opérations de collecte et 0 pour les opérations de livraison.

Chaque critère est ensuite normalisé entre $[0, 1]$, et une fonction de *fitness* pour chaque opération est alors obtenu

par une combinaison linéaire de ces critères. On utilise ensuite la fonction `softmax` pour obtenir une distribution de probabilité sur les opérations disponibles, avec un paramètre β qui contrôle la forme de cette distribution. Pour une valeur "basse" de ce paramètre, la probabilité de choisir la meilleure action sera plus grande, quand une valeur plus "haute" permet plus de diversité. Nous utilisons $\beta = 0.005$ pour la simulation ce qui nous permet d'obtenir de bonne solution tout en maintenant la diversité et $\beta = 0.1$ pour la probabilité à priori. Le vecteur de poids de la combinaison linéaire a été calculé par apprentissage par renforcement sur l'ensemble des instances générées aléatoirement, avec un horizon temporel d'une journée (l'horizon temporel intermédiaire de nos catégories d'instance).

Pour le DFS, nous cherchons à la fois une séquence sans retard, mais également à améliorer notre borne supérieure. Nous prenons en compte ces deux critères pour décider d'un nombre de "backtrack" entre 0 et \mathcal{B} autorisés.

Lors de la procédure gloutonne initiale, nous stoppons l'heuristique lorsque notre retard courant dépasse notre meilleure solution. Si cette condition n'arrive pas, alors nous avons trouvé une solution améliorante, et notre budget est maximum (\mathcal{B}) afin de chercher une solution voisine améliorant à nouveau la fonction objectif. Sinon, on se base sur la position ϕ du premier retard détectable dans la séquence d'opérations (i.e., le rang auquel la borne inférieure devient positive) pour définir le budget. Soit ϕ^* la position du premier retard dans la meilleure solution obtenue jusqu'à présent. Le budget est accordé comme suit :

$$\begin{cases} \mathcal{B} & \text{si } \phi \geq \phi^* \\ \mathcal{B} \left(\frac{\phi^* - \phi}{\phi^* - \alpha * \phi^*} \right)^2 & \text{si } \phi^* > \phi > \alpha * \phi^* \\ 0 & \text{sinon} \end{cases} \quad (5)$$

Avec $\alpha \leq 1$, un paramètre qui permet de définir un seuil. Notre DFS s'appuie sur l'heuristique utilisée dans la procédure gloutonne, à la fois pour l'ordre d'exploration, mais aussi pour limiter le facteur de branchement. En effet, cette procédure détermine une distribution de probabilité sur les actions disponibles, et un paramètre de seuil permet de se limiter aux seules actions les plus probables. Enfin, il est généralement admis qu'une stratégie de *restart* lors du DFS permet d'obtenir de meilleure solution. Suivant ce principe nous permettons alors au DFS (dans la limite de son budget) de restaurer l'état du noeud sélectionné par la phase de sélection au bout d'un certain nombre d'échecs, ce nombre va croître géométriquement à chaque *restart*, et sera réinitialisé pour la prochaine itération de la méthode.

5 Expérimentation

Nous proposons d'évaluer la méthode MCTS sur notre problème de collecte et de livraison de pièce. Nous considérons un ensemble d'instances partitionné en quatre classes (A, B, C, D), générées aléatoirement sur la base des caractéristiques du problème industriel. Chaque classe se différencie par le nombre de composant, allant de 15 composants pour les instances de la classe A, jusqu'à 30 pour celles de la classe D) Pour chacune des classes, nous disposons de

TABLE 1 – Valeur des paramètres

γ	0.9977
c	1
β	0.995
α	0.9
\mathcal{B}	50000
redémarrage (base)	100
redémarrage (facteur)	1.2

10 instances déclinés en trois horizons temporel ("Shift", "Jour", "Semaine"), pour un total de 120 instances au total³.

Nous proposons 5 versions différentes de la méthode MCTS intégrant progressivement les adaptations proposées. Nous appelons MCTS la méthode qui utilise la valeur de la fonction objectif directement, sans utiliser l'accroissement marginal pondéré de la borne inférieure, sans DFS, et sans décroissance du paramètre c de compromis exploitation/exploration. La méthode DSMI ajoute l'utilisation de l'accroissement marginal de la borne inférieure. Les méthodes DSMI+DFS et DSMI+DFS+DC ajoutent toutes deux l'intensification de la simulation par DFS, la seconde ajoute également le mécanisme de décroissance du paramètre c comme expliqué dans la section 3.2. Enfin la colonne DSMI+SAT-DFS+DC est une variation de DSMI+DFS+DC, pour laquelle nous fixons à 1 la borne supérieure pour la partie simulation, ce qui a pour effet de transformer la phase de simulation avec DFS en résolution d'un problème de satisfaction (un échec sera levé dès qu'un retard est rencontré). Cependant une vraie borne supérieure est tout de même disponible, la meilleure branche étant étendue jusqu'à une feuille à la fin du DFS. Les valeurs des paramètres utilisés dans ces méthodes sont donnés dans la table 1. Le paramètre γ est le coefficient de pondération de l'accroissement marginal (algorithme 1). Les paramètres c et β sont respectivement le compromis d'exploitation-exploration et sa variation (équation 4). Les paramètres α et \mathcal{B} sont utilisés pour fixer le budget de "backtrack" de la procédure de DFS (équation 5). Ce choix de valeurs s'appuie sur des expérimentations préliminaires, non détaillées ici.

Nous nous comparons avec deux précédentes méthodes décrites dans [1], l'une basée sur un solveur de programmation par contraintes (CP) pour la version satisfaction du problème et l'autre sur une recherche locale multistart (GRASP) pour la version en minimisation du retard.

Les résultats sont présentés dans la table 2 où #S représente le nombre d'instances résolues et L_{max} la valeur de l'objectif moyennée sur les 10 instances de la classe lancées 10 fois chacune. Une limite de temps d'une heure a été utilisée pour chacune des méthodes. Enfin l'expérience a été menée sur un cluster de calcul composé de processeurs Xeon E5-2695 v3 @ 2.30GHz et Xeon E5-2695 v4 @ 2.10GHz.

³ Instances disponibles sur <https://gepgitlab.laas.fr/vantuori/trolley-pb>

TABLE 2 – Comparaison des méthodes

CI	H	CP	GRASP		MCTS		DSMI		DSMI+DFS		DSMI+DFS+DC		DSMI+SAT-DFS+DC	
		#S	#S	L_{max}	#S	L_{max}	#S	L_{max}	#S	L_{max}	#S	L_{max}	#S	L_{max}
A	shift	90	90	10	100	0	100	0	100	0	100	0	100	0
	day	90	90	193	90	135	90	115	90	77	100	0	98	0
	week	80	70	2433	68	1996	70	1800	80	1839	77	1840	80	1858
B	shift	60	60	467	80	420	90	372	82	353	81	349	87	439
	day	52	46	3218	50	2954	60	2522	60	2439	63	2121	70	2134
	week	35	10	26915	10	21070	10	20572	32	20541	31	20355	36	20635
C	shift	40	40	1941	49	1676	40	1901	45	1727	40	1824	40	2012
	day	10	10	9498	10	9503	10	8683	26	8656	36	8747	35	9022
	week	10	0	71104	0	64442	0	64480	9	64584	9	64474	10	64445
D	shift	19	16	2677	40	2154	30	2338	33	2018	30	2304	30	2621
	day	0	0	13994	0	13659	0	12657	0	12723	13	12225	11	12340
	week	0	0	107186	0	101474	0	100533	0	100760	0	100954	0	100840
Moyenne		40.5	36	19969	41	18290	42	17998	46	17976	48	17933	50	18029

On peut voir dans ce tableau, que dans sa version de base, le MCTS nous permet de résoudre plus d'instances pour les horizons courts mais dégrade les résultats pour les horizons plus longs. L'utilisation de l'accroissement de la borne inférieure (DSMI) permet d'améliorer légèrement au global le nombre d'instances résolues et la valeur moyenne de l'objectif. Par contre, moins d'instances sont résolues pour les horizons "shift" des classes d'instances C et D. L'ajout du DFS (DSMI+DFS) nous permet surtout de résoudre plus d'instances au global, particulièrement sur les horizons longs ("day" et "week"), mais ne domine aucune des méthodes précédentes si l'on regarde ligne par ligne (excepté GRASP, déjà dominée par DSMI). Le mécanisme de décroissance du compromis exploitation/exploration couplé au DFS (DSMI+DFS+DC) permet encore d'améliorer les résultats globaux, et plus particulièrement sur un horizon d'une journée, sans dominer les méthodes précédentes encore une fois. Enfin, la version DSMI+SAT-DFS+DC permet de résoudre plus d'instances au global, contre une dégradation de la fonction objectif, ce qui est attendu, étant donné que le DFS ne cherche plus à améliorer la borne supérieure de l'objectif. Cette dernière version est la plus performante en matière de nombre d'instances résolues : 9.5% d'instances en plus par rapport à la méthode CP que nous avons avant.

6 Conclusion

Nous avons présenté des adaptations de la méthode MCTS pour la résolution d'un problème d'optimisation combinatoire issu de l'industrie automobile. Les évaluations expérimentales montrent un gain de performance par rapport à des méthodes classiques.

Les adaptations que nous proposons sont efficaces pour ce problème, d'autres combinaisons de ces adaptations sont en cours d'évaluation afin d'affiner l'apport de chacune d'entre elles. Les travaux futurs se focaliseront sur l'étude de la généralité de ces adaptations, notamment l'affectation du budget du DFS, en les appliquant à d'autres problèmes académiques et industriels.

Références

- [1] Valentin Antuori, Emmanuel Hebrard, Marie-José Huguet, Siham Essodaigui, and Alain Nguyen. Leveraging Reinforcement Learning, Constraint Programming and Local Search : A Case Study in Car Manufacturing. In *Principles and Practice of Constraint Programming - 26th International Conference, CP 2020*, pages 657–672, 2020.
- [2] Dimitris Bertsimas, J. Daniel Griffith, Vishal Gupta, Mykel J. Kochenderfer, and Velibor V. Misić. A comparison of Monte Carlo tree search and rolling horizon optimization for large-scale dynamic resource allocation problems. *European Journal Operational Research*, 263(2) :664–678, 2017.
- [3] Cameron Browne, Edward Jack Powley, Daniel Whitehouse, Simon M. Lucas, Peter I. Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez Liebana, Spyridon Samothrakis, and Simon Colton. A Survey of Monte Carlo Tree Search Methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(1) :1–43, 2012.
- [4] Rémi Coulom. Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search. In *Computers and Games, 5th International Conference, CG 2006*, pages 72–83, 2006.
- [5] Ryota Furuoka and Shimpei Matsumoto. Worker's knowledge evaluation with single-player monte carlo tree search for a practical reentrant scheduling problem. *Artificial Life and Robotics*, 22(1) :130–138, 2017.
- [6] Manuel Loth, Michèle Sebag, Youssef Hamadi, and Marc Schoenauer. Bandit-Based Search for Constraint Programming. In *Principles and Practice of Constraint Programming - 19th International Conference, CP 2013*, pages 464–480, 2013.
- [7] Minh Anh Nguyen, Kazushi Sano, and Vu Tu Tran. A monte carlo tree search for traveling salesman problem with drone. *Asian Transport Studies*, 6 :100028, 2020.

- [8] Alessandro Previti, Raghuram Ramanujan, Marco Schaerf, and Bart Selman. Monte-Carlo Style UCT Search for Boolean Satisfiability. In *AI*IA 2011 : Artificial Intelligence Around Man and Beyond - XIIIth International Conference of the Italian Association for Artificial Intelligence*, pages 177–188, 2011.
- [9] Thomas Philip Runarsson, Marc Schoenauer, and Michèle Sebag. Pilot, rollout and monte carlo tree search methods for job shop scheduling. In *Learning and Intelligent Optimization - 6th International Conference, LION 6*, pages 160–174, 2012.
- [10] Ashish Sabharwal, Horst Samulowitz, and Chandra Reddy. Guiding combinatorial optimization with UCT. In *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems - 9th International Conference, CPAIOR 2012*, pages 356–361, 2012.
- [11] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Vedavyas Panneshelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P. Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587) :484–489, 2016.

COrHA: Architecture multi-agent pour la collaboration Utilisateurs-Objets connectés dans une tâche en domotique

A. Audic¹, L. Thevin²

¹ Fondation Université Bretagne Sud, Chaire M@D "Maintien à Domicile"

² Université Catholique de l'Ouest, équipe MAI

al.audic.a@gmail.com, lauren.thevin@uco.fr

Résumé

*Les systèmes domotiques sont décentralisés et permettent idéalement des interactions riches et flexibles (dispositifs techniques et utilisateurs). Dans ce cadre, nous nous intéressons à la manière de coordonner à un ensemble varié de (i) **dispositifs** (inputs, outputs) (ii) **utilisateurs** (préférences d'interaction), et (iii) **tâches** soutenues par le système interactif. En étudiant la littérature scientifique, les systèmes multi-agents ont été proposés pour coordonner l'interaction objets-utilisateurs. Peu de systèmes explicitent la coordination de la tâche réalisée entre l'utilisateur et le système informatique. L'architecture multi-agent COrAH, reprend les modèles OrA et JaCaMo pour la collaboration (multi-utilisateur, accessibilité, objets autonomes).*

Mots-clés

SMA, IHM, domotique, collaboration, accessibilité.

Abstract

Smart home systems are distributed and (ideally) enable rich and flexible interaction (devices and users). The challenge is to coordinate diverse (1) devices (inputs and outputs), (2) users'usage and (3) tasks supported by the system. We found multiple work about devices coordination, and interaction with the users. However, we found few publications about managing the task regarding to the users and the devices. The multi-agent architecture COrAH, from OrA and JaCaMo frameworks, enables collaboration (multi-user, accessibility and autonom objects).

Keywords

MAS, HCI, domotics, collaboration, accessibility.

1 Introduction

Dans le cadre domestique, le nombre d'objets connectés (IoT) et de dispositifs de domotique continuent d'augmenter [1] (dans le monde : 3.6 milliards en 2015, 13.8 milliards en 2021¹). Outre l'accès pour les profils technophiles (e.g. *early adopters*), ces dispositifs devraient être accessibles au plus grand nombre. Cela garantit d'une égalité d'accès à

la technologie, l'illectronisme concernant 17% de la population en France en 2019 selon l'Insee [22]. De plus, les technologies d'assistance et de maintien à domicile constituent des champs d'application pour l'IoT et la domotique (e.g. [3, 5, 19]), considérant qu'une personne sur sept était en situation de handicap en 2011 [29]. On observe qu'une grande complexité peut naître de la **diversité des objets connectés** (inputs, outputs, équipements) et de la **diversité des utilisateurs** (préférences, accessibilité, personnes présentes). L'intelligence artificielle (IA) et, en particulier, les Systèmes Multi-Agent (SMA) semblent adaptés pour répondre à ces problématiques [9]. Nous proposons d'exploiter une architecture orientée multi-agent JaCaMo[4], indiquée pour la coordination complexe et robuste de composants modulaires [9]. Nous incluons les utilisateurs comme des Agents du SMA [33], car ils réalisent des tâches domotiques au même titre que les objets connectés et autonomes. Un SMA pour le cadre domotique s'explique par : (i) la présence de réseaux d'objets connectés et d'utilisateurs, disparaissant et apparaissant, (ii) la vision parcellaire et la distribution des tâches (agents humains et artificiels). Nous évoquons en discussion la complémentarité d'autres IA.

Dans l'état de l'art, les SMA sont proposés comme solution pour organiser des systèmes d'objets connectés décentralisés (section 2.1), et prendre en compte les utilisateurs (section 2.2). En conséquence, nos contributions sont (fig.1) :

- l'analyse d'un SMA pour la domotique **sous l'angle d'une collaboration humains-machines autour de tâches**, en plus de gérer le cycle de communication humains-machines (perception, action).
- **une proposition préliminaire d'une architecture multi-agent** adaptée du modèle OrA. Quelque soit la tâche, la perception des utilisateurs par les dispositifs techniques s'organise en espace de *production*, la perception des retours informés par les utilisateurs s'organise en espace de *communication*. La répartition des tâches entre humains et dispositifs techniques s'organise en espace de *coordination*,
- **trois scénarios illustratifs de collaboration autour des tâches domotiques**.

¹. Source : statista le 26 février 2021 <https://www.statista.com/statistics/1101442/iot-number-of-connected-devices-worldwide/>

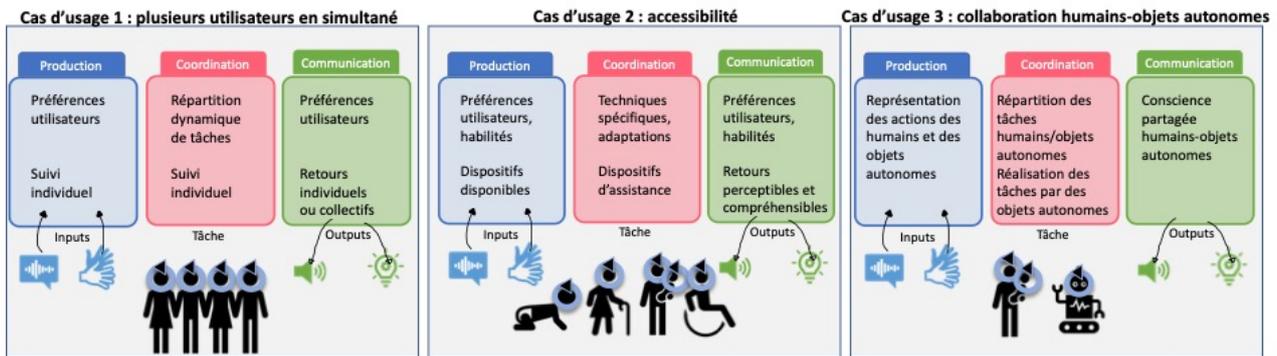


FIGURE 1 – SMA JaCaMo (dimensions VOWELS non représentées) selon trois espaces issus des collecticiels (production, coordination et communication) pour organiser la collaboration en domotique. Cas d’usages illustratifs : 1) utilisateurs présents collaborant ou non sur une tâche de cuisine, 2) prise en compte de stratégies d’accessibilité spécifiques pour les inputs, les outputs et la tâche et 3) collaboration humains-objets autonomes pour la réalisation conjointes de tâches.

2 Etat de l’art

Les dispositifs domotiques lèvent plusieurs difficultés en internet des objets (IoT), en interaction humain-machine (IHM) et en intelligence ambiante (AmI), pour l’adaptation (i) à la diversité des dispositifs [2], (ii) au caractère hétérogène des utilisateurs en limitant les technologies d’assistance mono-personnelles (risque d’isolement dans l’utilisation) [5], et (iii) aux défis pour créer un «*milieu ayant la faculté de percevoir, de raisonner, d’agir et d’interagir afin de fournir des services améliorant la qualité de vie*» [8]. Nous abordons dans cette section comment les SMA y répondent.

2.1 Utilisation des SMA pour organiser des systèmes décentralisés en IoT

Les SMA sont exploités pour organiser des systèmes décentralisés, notamment pour l’IoT et la domotique. Charif et Sabouret [7] composent un système de services domotiques en se reposant sur des protocoles d’interaction multi-agents avec le modèle de programmation VDL (View Design Language). Cela permet de raisonner, sous forme de requêtes formalisées, sur des services Agents-Agents [7]. Les agents répondent individuellement à un ensemble de requêtes pour réaliser tout ou partie d’une tâche. Les agents peuvent collaborer entre eux ou prendre des initiatives dans la création de fonctionnalités [7]. Un des avantages de ce système de requêtes est l’intelligibilité de celles-ci pour un utilisateur humain. De manière générale, les SMA amènent une puissance de calcul et de représentation, au service d’une vision d’ensemble de l’utilisation d’objets interactifs [18]. Cela gère l’activité d’une série d’objets virtuels et numériques qui peuvent apparaître ou disparaître à la volée [18]. Ce principe a été appliqué à la détection, au suivi et à la mise en place d’interactivité pour des objets tangibles dans le projet TTT. Les objets connectés tangibles sont tracés de manière unique (présence et localisation) en temps réel grâce à un tag RFID, et peuvent stocker des informations. La couche applicative est composée d’un SMA intégrant une vue globale des objets présents, gérant la détection d’objets et l’af-

fichage de retours virtuels. Ce SMA organise hiérarchiquement et dynamiquement les objets interactifs entre eux et dans leur environnement [18]. L’interaction centrée objet permet paradoxalement de créer une interaction collaborative et multi-utilisateur. En domotique, la réorganisation des composants est également un challenge étudié par Hamoui *et al.* [15]. Dans leurs travaux, les auteurs proposent de contrôler un environnement domotique par un SMA. Les agents contrôlent les équipements de l’environnement et les services afin d’exécuter des scénarios définis par les utilisateurs, dans le but de satisfaire leurs besoins [15]. Ce SMA modifie son architecture pour déployer des composants et des scénarios utilisateurs [15]. Dans ce cadre [14], quatre propriétés sont identifiées pour un système domotique organisant des composants intelligents et des scénarios [17, 27] :

- **configurable** : pour définir des scénarios.
- **décentralisé** : pour être fiable en évitant les pannes et optimisant la performance du système.
- **autonome** : limiter le travail de configuration par l’utilisateur au moment du paramétrage initial.
- **dynamique** : selon changements d’environnement.

Ce système se structure autour de deux types d’agents : les agents «*responsables de l’interaction avec les utilisateurs*» GUIA (Graphical User Interface Agent) et les agents «*de contrôle des équipements*» DCA (Device Control Agent) qui font le lien entre l’environnement et les scénarios [14]. En plus de faire fonctionner différents composants entre eux, les SMA permettent une optimisation de l’utilisation des objets connectés. Martins *et al.* [23] présentent un modèle de maison intelligente contribuant à l’amélioration de la consommation et la distribution énergétique dans un foyer tout en maintenant le confort de l’habitat. Les agents autonomes individuels contrôlent les appareils au sein du foyer et un agent global coordonne des agents individuels selon un compteur d’énergie. Le modèle proposé utilise la plateforme JaCaMo pour la programmation multi-agent :

- **organisation Moise** avec spécifications structurales (e.g. rôles), fonctionnelles (e.g. plans), et normatives (e.g. obligations contextuelles)

- **agents autonomes BDI** (Belief-Desir-Intention) implémentés en **Jason**
- **environnement** basé sur des artefacts en **CArtAgO**
- **interactions** entre ces éléments

L'organisation générale est gérée au niveau Moise, y compris la hiérarchisation des objectifs de confort et d'économie. Les appareils sont gérés individuellement par des agents Jason en lien avec leurs objectifs énergétiques. Deux types d'artefacts CArtAgO permettent de définir l'environnement en termes (i) de consommation énergétique globale, (ii) selon les cycles horaires. Les SMA permettent d'optimiser un système domotique selon les critères des utilisateurs (préférences) en fonction de l'architecture particulière du système domotique (opportunisme) [10]. L'application d'agents cognitifs de Dujardin *et al.* optimisent et contrebalancent ces valeurs d'opportunisme et de préférence pour chaque action possible d'un système domotique selon des critères. La préférence «écologie» favorisera l'extinction d'appareils [10]. Ce SMA peut également suggérer des actions à l'utilisateur [10] en explicitant la raison. Une alternative d'optimisation porte sur le diagnostic [25] et l'organisation du réseau lui-même [31]. Les SMA peuvent être résiliants jusqu'à la panne ou déconnexion de k objets physiques [31]. Cela implique : (1) de détecter les composants physiques ne fonctionnant pas, (2) d'identifier les opérations numériques qui étaient prises en charge par ces composants, (3) de répliquer ces opérations numériques, et (4) de les relocaliser au niveau d'un noeud du réseau (i.e. un composant encore fonctionnel dans le réseau IoT), en optimisant les communications nécessaires (*load*) [31]. En résumé, on peut faire «migrer», en cas de panne, les opérations numériques de manière optimisée sur le réseau physique. Les objets connectés ont une dimension «réelle» (ce sont de vrais objets) et une dimension numérique qu'il faut articuler entre elles (niveau objet puis niveau réseau). Schmitt *et al.* [32] explorent ces logiques d'interaction de deux manières dans la plateforme Triskell3S : en répondant aux problématiques d'interaction entre un agent (appelé IoT-a) et un objet connecté, et en étudiant l'exploitation de données hétérogènes et distribuées entre les objets grâce à un SMA. Les interactions du SMA sont affichées sur un mur écran : interactions au niveau des données (agents d'interface), du matériel (IoT-a), des usages (agents humains), et entre ces niveaux [32]. Les retours écrans et les agents IoT-a augmentent l'interaction utilisateur et l'optimisation, aux deux niveaux : objets et réseau.

En étudiant les SMA dédiés à l'organisation de systèmes décentralisés, nous avons pu noter leur utilisation pour :

- le suivi de la présence ou non d'objets interactifs
- la réorganisation dynamique (hétérogénéité, suppression et ajout) des composants selon des scénarios
- l'optimisation de la performance d'un réseau d'objets connectés (ex : consommation)
- l'optimisation de l'organisation distribuée d'objets connectés entre eux, e.g. couplé au diagnostic
- l'articulation des dimensions «réelle» et «numérique» des objets connectés

2.2 Utilisation des SMA pour prendre en compte les utilisateurs en domotique

En domotique, un enjeu important est l'intégration de l'activité des habitants [12]. Cela passe par l'identification et le suivi automatique des activités, et par l'adaptation aux habitudes et préférences des habitants, à la dynamique de l'environnement, aux sources de perception ainsi qu'à leurs changements [12, 2]. Le système *Smart Kitchen* [3] organise des composants domotiques grâce à un ensemble de règles et de scénarios d'utilisation. Il est alors possible de faire la même action via une télécommande, une tablette ou interface vocale. Cela peut être étendu ou modifié en ajoutant de nouveaux composants et leurs règles associées, et de nouveaux scénarios [3]. *Smart Kitchen* stocke dans une base de données les traces de l'utilisation du système par l'utilisateur (interaction contextuelle). En comparant l'évolution de ces données dans le temps (e.g l'heure d'utilisation d'un frigo ou la fréquence d'utilisation de la machine à laver), il est possible d'extraire un contexte utilisateur (insomnie, activités domestiques) [3]. Ces données sont transmises à des intervenants (soignants par exemple). L'organisation des composants domotiques évolue selon ce principe.

La sensibilité au contexte est un élément important dans la mise en place de services interactifs (au sens IHM) «adaptatifs» aux habitudes, aux activités, et à l'environnement dans les environnements intelligents [2], par exemple en s'auto-configurant. Benazzouz [2] spécifie les termes d'*adaptation* comme la personnalisation et la recommandation tandis que l'*adaptation automatique* se distingue par l'analyse du contexte et de l'environnement de l'utilisateur [2]. Le système proposé, découvre et suit des situations en analysant des «données de contexte» [2]. L'analyse algorithmique identifie les situations qui se répètent afin de définir des contextes (habitudes). Par ailleurs, la localisation des habitants est primordiale pour la mise en place d'une interaction contextuelle multi-utilisateurs [13]. Giroux *et al.* offrent ces fonctionnalités avec une architecture multi-agent utilisant des normes Moise. Les agents de localisation se basent sur un système bayésien (Domino librairie Dynamo [30]) pour la reconnaissance d'une suite d'événements et de la présence d'individus spécifiques. Un système de négociation attribue chaque événement à un agent [13]. Des agents d'environnement sont associés à chaque objet et gèrent les interfaces utilisateurs. Un agent personnel est associé à chaque individu, et intègre son profil, ses activités et ses préférences. Ce profil est transmis aux agents d'environnement proches. Des notions de priorité et d'urgence gèrent les injonctions paradoxales (e.g. avec plusieurs utilisateurs actifs) [13]. Sur le même principe, le système ICEMAS permet la collaboration de multiples utilisateurs, pour «l'amélioration de la vie quotidienne des personnes atteintes de déficiences cognitives» ayant besoin d'une assistance ponctuelle [5]. L'objectif est de «personnaliser» l'habitat en fonction des besoins, de l'assister selon son profil et la présence d'autres utilisateurs, en maintenant une discrétion et une automatisation optimale [5]. Les conflits entre individus sont réglés en comparant préférences et priorités,

e.g. en favorisant la personne avec le plus haut niveau d'assistance à un instant donné [5]. La particularité de ce système réside dans la prise en compte des contraintes de gestion globales et individuelles. Par exemple, un son fort la nuit (règle globale) déclenchera des actions hors du simple cadre de l'assistance (règle individuelle)[5]. Kuijpers [19] propose un SMA appliquant des règles de sécurité personnalisées selon le profil médical de l'utilisateur. Le système est structuré par des agents fonctionnant sur le modèle BDI. Des *DeviceAgents* coordonnent les appareils de cuisine et récupèrent des informations par le biais des agents de données (capteurs), selon trois granularités : un *DeviceAgent* par équipement, un pour chaque fonctionnalité de chaque équipement, et un pour coordonner l'ensemble des équipements. Si un *DeviceAgent* détecte un changement d'activité, il communique avec les *RiskAgents* (agents chargés des risques), enregistre un changement d'état et une trace d'activité dans un fichier log. Le *UserAgent* identifie les utilisateurs dans l'environnement, charge les profils, déverrouille les appareils de cuisine en fonction de l'autorisation et transmet ces profils aux *RiskAgents*. Un agent *WatchdogAgent* verrouille les équipements en cas de danger et agit directement sur le matériel en cas de contexte critique. Le système présenté ici (StoveMAS) utilise la plateforme JADE et suit les agents via une interface visuelle sur Android. Toujours sur JADE et proposé par Kuijpers, PADIS adapte les services aux utilisateurs selon leur handicap [20] via le dossier médical rédigé par le médecin (niveau d'assistance et services disponibles), les préférences de l'utilisateur (confort et habitude de vie), le contexte (e.g. heure de la journée). Un système de boucle MAPE-K (monitor, analyze, plan, execute - knowledge) permet de vérifier les services correctement exécutés. La gestion est dite autonome selon quatre concepts : (i) auto-configuration d'appareils lors de leur intégration au système domotique, via des agents BDI, (ii) auto-optimisation de paramètres, (iii) auto-réparation en isolant les services lors d'une défaillance durant la réparation, et (iv) auto-protection, c'est à dire l'autonomie dans la protection contre les dangers externes[20]. Les agents, structurés en groupes représentant les pièces de l'habitat, ont des buts différents. Les agents migrent entre les groupes pour garantir la continuité d'un service. Ce SMA se connecte sur un serveur, une base de données et une application mobile en guise d'interface[20]. La base de données crée une image du contexte des appareils répartis dans l'environnement. Les données des capteurs des dispositifs mobiles connectés sont mises à jour dans la base de données. L'interface propose les services des agents en fonction de la personne et de l'emplacement. Des conflits surviennent notamment lorsque plusieurs utilisateurs se trouvent dans la même pièce[20]. Les SMA étudiés soutiennent l'IHM à travers différents contextes :

- réaliser une même action par plusieurs moyens
- suivre les habitudes de vie (et changements) au travers des traces d'activité des utilisateurs
- identifier des utilisateurs concernés (localisation)
- prendre en compte plusieurs utilisateurs
- déployer des services d'accessibilité, d'assistance.

3 Approche

3.1 Problématique

Nous tentons de répondre à la question suivante : *Quelles sont les exigences auxquelles doit répondre un système dédié à l'organisation d'objets connectés dans un contexte domotique ?* Dans un premier temps, nous identifions les verrous adressés dans la littérature, et indispensable à la bonne gestion d'une application domotique :

1. **coordination et réorganisation à la volée des modalités d'interaction en fonction des objets connectés.** Un système domotique cohérent doit se ré-organiser, avec deux impacts majeurs : (i) les moyens de percevoir les utilisateurs, et (ii) les moyens de fournir des retours informés aux utilisateurs (selon les dispositifs disponibles)
2. **coordination et réorganisation à la volée des modalités d'interaction en fonction des utilisateurs présents.** Un système domotique cohérent doit se ré-organiser en termes de préférences et d'accessibilité, avec deux impacts majeurs : (i) les moyens d'interagir avec les dispositifs, et (ii) les moyens de fournir des retours informés en fonction des utilisateurs (selon les utilisateurs présents)

Si l'on se penche sur la définition des termes *coordination* et *collaboration*, nous pouvons mettre en lumière la place centrale de la planification et de la répartition des actions entre les personnes/agents participants [34]. En *coordination*, les tâches sont réparties en avance entre les membres impliqués, alors que la *collaboration* se caractérise par une répartition dynamique des tâches selon les actions en cours [34]. Nous ajoutons donc cette troisième exigence :

3. **la coordination et la réorganisation à la volée des tâches et de la répartition des tâches à réaliser.** L'utilisation d'un système domotique par un utilisateur se fait dans un but donné, autrement dit pour réaliser une tâche. Un système domotique doit pouvoir ré-organiser la tâche en fonction des utilisateurs et des dispositifs présents : prise en charge une partie des activités (automatisation), réorganisation intra et inter-personnelle de la tâche.

Nous avons identifié plusieurs cas d'usages mettant en oeuvre la prise en compte de la tâche dans un cadre domotique : (i) s'adapter à plusieurs utilisateurs en simultané, (ii) s'adapter à différents utilisateurs dans le cadre de l'accessibilité, et (iii) prendre en compte les objets autonomes dans la réalisation de tâches en domotique. Nous présentons, section suivante, l'adaptation du modèle OrA[33] pour la coordination humains-machines en domotique.

4 Proposition préliminaire

4.1 Programmation orienté multi-agent pour la coordination et la collaboration

Nous l'avons vu précédemment, la programmation orientée multi-agent VOWELS [9] indiquée pour la coordination de composants modulaires définit un SMA selon les

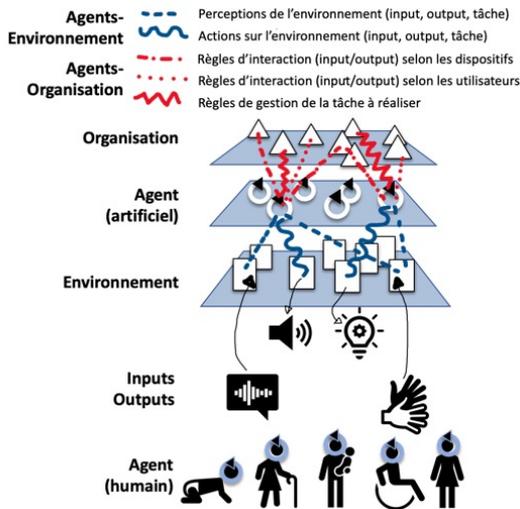


FIGURE 2 – Environnement CArTAgo : les agents humains interagissent (input); les Agents Jason perçoivent puis mettent en oeuvre des retours informés perceptibles par les utilisateurs (output). Agents Jason : perçoivent, analysent et génèrent ces informations. Organisation Moise : gestion des différents objets (input, output), des tâches en cours et des utilisateurs. L’architecture s’organisera en trois espaces.

niveaux <A,E,I,O (,U)>, pour Agent, Environnement, Interaction et Organisation; dans certains cas, le U est ajouté pour les Utilisateurs. JaCaMo implémente ces différents niveaux, qui permettent d’utiliser différents mécanismes directs et indirects de collaboration et de coordination [4]. Les Agents Jason peuvent interagir entre eux, via l’Environnement représenté à travers des artefacts CArTAgo, ou via des organisations normatives Moise, le tout s’organisant autour d’Interactions (figure 2). JaCaMo a été appliqué à l’IHM et à la collaboration dans le modèle OrA [33]. Dans OrA, l’Environnement permet de suivre l’activité en cours en créant des traces d’activité. Les Agents analysent et organisent les modalités d’interaction et de retours informés, et permettent par ailleurs le suivi des actions coordonnées des utilisateurs. Les Organisations permettent aux agents d’adopter des rôles afin de suivre des règles d’interaction, de mettre en place des retours informés et de représenter l’activité collaborative [33]. Dans ce modèle, les Humains sont considérés comme des Agents, même s’ils n’appartiennent pas au système technique, *a contrario* des agents Jason. En particulier, dans le cadre domotique, les agents, qu’ils soient des utilisateurs ou des objets connectés, réalisent une partie des tâches. Le fait de représenter les agents (artificiels et humains) distincts de l’organisation Moise permet la répartition dynamique (spécifications normatives) des tâches représentées dans l’organisation (spécifications fonctionnelles). Les tâches sont déclarées sous forme de sous-butts dans la spécification fonctionnelle Moise, indépendamment des agents. Les agents (humains et artificiels) s’attribuent ces sous-butts à la volée et au fur et à mesure

de l’évolution de l’activité et de leurs rôles (figure 4, agents 1,2 et 3). Si un agent disparaît, ces sous-butts sont “libérés” dans l’organisation, et peuvent être ré-attribués. Tout nouvel agent (artificiel ou humain) peut prendre en charge un des sous-butts. Cette vision permet d’intégrer les objets connectés et autonomes comme des acteurs dans les tâches, au même titre que les utilisateurs, et répond partiellement à l’exigence 3 section 3.1. Cependant, l’organisation VOWELS ne permet pas à elle seule de gérer la complexité des échanges Humains-Machines (inputs, outputs et tâche).

4.2 COrAH : Adaptation du modèle OrA

Pour répondre aux exigences de la section 3.1, le SMA s’organise en trois espaces ayant tous la structure VOWELS présentée précédemment : production, coordination et communication. Ces espaces sont issus des modèles de collectifs Clover [21] et SMA OrA déjà évoqué [33].

Les espaces s’articulent selon le cycle utilisateur (figure 3). L’espace de Production permet à l’utilisateur d’interagir avec le système (exigence 1.i et 2.i section 3.1). L’espace de Communication permet au système d’interagir avec l’utilisateur via des retours informés (exigence 1.ii et 2.ii section 3.1). Entre les deux, l’espace de Coordination est dédié à la tâche attribuée aux agents utilisateurs ou aux agents du système technique (troisième exigence section 3.1). Les appellations peuvent être questionnées (e.g. *production*, et non pas *perception*). Ces appellations respectent la syntaxe du modèle OrA, et Clover. De plus, l’espace de production concerne l’espace de *perception* des agents humains par les agents artificiels; l’espace de communication est l’espace de *perception* des agents artificiels par les agents humains. L’architecture SMA présentée, nommée COrAH (Collaboration d’Organisations Artificielles et Humaines) comporte plusieurs différences avec le modèle OrA originel. En espace de collaboration OrA, les agents sont des agents “mimes” intégrant et exécutant dans un scénario virtuel la tâche demandée par les humains. Par exemple, l’agent *Maire(sse) virtuel* réalisait une *évacuation virtuelle* lorsque l’utilisateur *Maire(esse)* demandait une *évacuation* dans le jeu de rôle interactif. Ici, les agents et les humains se distribuent et réalisent les tâches en espace de Collaboration. En résumé, dans COrAH, les agents artificiels et humains coopèrent via le système. Le lien entre les différents espaces est assuré par la création d’artefacts : la fin de l’activité en production initie une activité en coordination avec un nouvel artefact de coordination créé par un agent de production en espace de coordination (voir [33], p102). Le mécanisme est identique de la coordination vers la communication.

4.3 Scénarios d’illustration en Domotique

Nous illustrons l’utilisation de l’architecture COrAH dans trois scénarios, avec l’apport de la troisième exigence *représentation de la tâche* (coordination), en plus des inputs (production) et des outputs (communication).

Plusieurs utilisateurs en simultané. Le premier scénario (figures 1 gauche et 4 haut) présente le fonctionnement de COrAH dans un contexte multi-utilisateurs. Deux agents humains H1 et H2 collaboreront sur la même recette et un

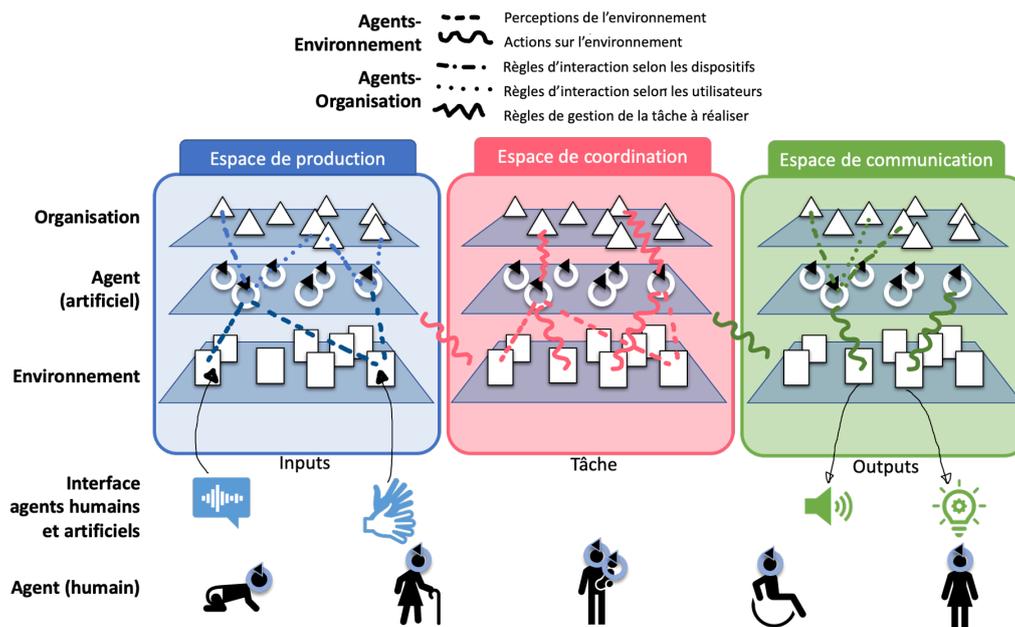


FIGURE 3 – Architecture CORHA : SMA VOWELS présentée est organisée en 3 espaces : production (input), coordination (tâche), et communication (output). **Espace de production** : les inputs utilisateurs, des artefacts dans l'Environnement, sont analysés en fonction des dispositifs disponibles et des utilisateurs. Pour cela, les Agents utilisent les Organisations normatives dédiées. Une action de coordination (artefact) est créée. **Espace de coordination** : ces artefacts dans l'Environnement (actions utilisateurs), sont analysés en termes de tâches à réaliser. Pour cela, les Agents utilisent les Organisations de coordination pour répartir et suivre les tâches. Une action de communication (output) est créée. **Espace de communication** : les retours informés à donner, représentés sous forme d'artefacts dans l'Environnement, sont générés selon les dispositifs et utilisateurs destinataires. Pour cela, les Agents utilisent les règles dans les Organisations de communication pour générer les outputs appropriés. Les outputs sont mis en oeuvre dans l'environnement.

troisième H3 fait une autre activité de nettoyage. **Étape 1** : H1 suit une recette, les buts du plan *Moise Recette* (Coordination) lui sont attribués. H3 est un intégré à un plan de nettoyage *Ménage*. **Étape 2** : H2 s'inclut dans la tâche de cuisine. Les buts du plan *Recette* en Coordination sont redistribués entre H1 et H2. Prioritairement, les tâches en cours restent attribuées à H1, et les buts parallèles sont répartis entre H1 et H2 pour limiter les interdépendances. H3 ne note pas de changement.

Accessibilité. Ce second scénario (figures 1 centre et 4 haut et bas) présente des adaptations en Coordination pour l'accessibilité via deux mécanismes : l'adaptation de plans *Moise*, et l'intégration des outils d'assistance. Suite au scénario 1, il se trouve que H2 a des déficiences visuelles et met en oeuvre des stratégies particulières dites d'AVJ (autonomie de la vie journalière). Par exemple H2 organise spatialement un plan de travail avant de commencer une recette. **Étape 1** : le but "Blancs en neige" de H2 active le plan "Blancs en neige A" (figure 4), intégrant les techniques d'AVJ. Le plan "Blanc en neige B" aurait été utilisé par H1 sur cette même tâche. **Étape 2** : Une technologie d'assistance peut réaliser une partie des tâches. Un agent artificiel A4 surveille la cuisson (cf scénario suivant). Ici, le système active en coordination des stratégies d'AVJ, puis collabore avec l'utilisateur en réalisant une partie des

tâches de la recette (surveillance optique de la cuisson). Afin d'adapter le système au contexte utilisateur, l'accessibilité pour les déficiences visuelles est optimisée via les modalités d'interaction cohérentes en Production (interaction vocale par exemple) et en Communication (réglage de l'écran, son spatialisé, description adaptée).

Coordination entre agents humains et objets autonomes

Ce troisième scénario (figures 1 centre et 4 haut) présente la prise en charge de buts en Coordination entre H1, H2 et A4, en attribuant la réalisation des tâches aux agents humains et artificiels sans distinction. **Étape 1** : H1 et H2 réalise une recette selon le plan *Recette* dans l'organisation de Coordination. **Étape 2** : Certains sous-butts sont réalisables par des agents, par exemple surveiller la cuisson. Ce but est attribué à l'agent artificiel A4 au lieu de H1 et H2. Ici, en plus d'une répartition des tâches, le système met en place une conscience partagée en informant les agents humains (Communication).

Synthèse. L'apparition et la disparition de dispositifs ou d'utilisateurs sont gérés à la volée dans les trois espaces. La réorganisation des plans en temps réel étant dissociés pour chaque espace (de production, de communication, de coordination), permettant une réorganisation des inputs, des outputs et des tâches indépendamment. La boucle dialogique (la communication entre agents humains et artificiels) s'ar-

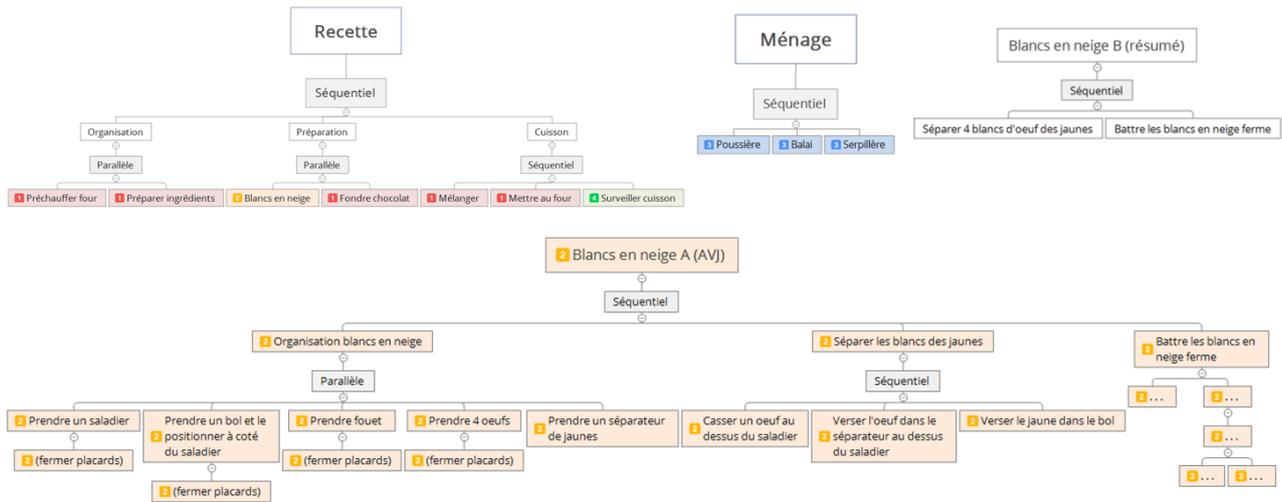


FIGURE 4 – Spécifications fonctionnelles des scénarios illustratifs. Les numéros représentent les agents (1, 2, 3 et 4) à qui seront attribués les sous-buts via les normes. La spécification *Recette* sera répartie entre deux utilisateurs (1, 2) et un agent artificiel (4). Le but *Blancs en neige* sera déclinée en version adaptée *Blancs en neige A (AVJ)*, au lieu de la version résumé *Blancs en neige B (Résumé)*. L'utilisateur 3 fait une autre tâche *Ménage*.

ticule en trois temps. La perception des agents humains par les agents artificiels est ainsi intégrée dans l'espace de production. La prise en compte de l'activité des agents se fait en espace de coordination. Enfin, les retours informés des agents artificiels vers les agents humains sont gérés en espace de communication. L'agent humain perçoit l'information des agents artificiels (retour informé), comprend ce retour et communique aux agents artificiels (retour à l'espace de production). Par exemple, les interactions vocales avec l'utilisateur passent par la captation sonore de l'agent humain. Cette captation sonore est analysée en espace de production en speech-to-text. L'analyse contextuelle de l'information sonore intégrée en espace de production se fait en espace de coordination. Le retour audio (ou autre modalité) vers l'humain est fait en espace de communication. Il est possible d'instancier le paradigme de l'intervention [16] de l'utilisateur via un tableau de bord qui agit directement sur les normes. Plus spécifiquement, un système domotique est intéressant s'il permet un haut niveau d'automatisation, tout en permettant à l'utilisateur d'intervenir. Ici, les agents BDI du SMA peuvent réaliser leurs objectifs, et intégrer les objectifs des agents humains lors de leurs interactions. L'activation et la désactivation de ces normes via un tableau de bord permet la modification du fonctionnement ou de l'analyse de la situation par le système (pour un exemple, voir [33], p 173 figure 4.16 ou p 169 figure 4.12).

5 Discussion et Perspectives

Les systèmes domotiques mobilisent de nombreux concepts à articuler : des objets connectés avec inputs, outputs, et services ; ainsi que des agents humains et artificiels. Nous tentons d'y répondre avec le modèle COrAH, articulant les inputs en espace de Production, indépendamment des outputs en Communication et la tâche pour laquelle est conçu

le système en Coordination. Le modèle COrAH actuel permet de représenter explicitement des spécifications normatives, mais n'en apprend pas de nouvelles. Les agents n'ont pas non plus de mécanismes d'apprentissage. D'autres approches en IA pourront être intégrées. Le système *IRON* permet de générer automatiquement des normes sans conflit [26]. Mazac [24] propose en domotique une architecture multi-agent qui applique les principes d'apprentissage sensorimoteur (observation et intégration des signaux de l'environnement). Chaouche [6] présente des mécanismes de sélection de plans existants et d'exploration de nouveaux plans par des agents BDI selon le contexte spatio-temporel en domotique. Une autre perspective est de ré-organiser les plans existants, en appliquant les principes d'IA traditionnelles. Par exemple, la méthode formelle Universal Method-Composition Planner [11] permettrait de redéfinir une planification (selon le matériel, les utilisateurs, le contexte) des tâches primitives en production, en coordination et communication, parmi l'ensemble des plans et de leurs décompositions disponibles. Les méthodes de planification automatique sont typiquement applicables aux tâches avec des contraintes d'ordre d'exécution et d'interdépendance, avec notamment les méthodes de type SHOP [28]. Enfin, nous n'abordons pas dans ce travail les questions du diagnostic, de la migration de services et de la modification de règles en fonction de pannes ne sont pas explicitement traitées ; ni l'identification du profil utilisateur.

Le modèle COrHA fonctionne théoriquement. La réalisation d'un simulateur est en cours pour valider expérimentalement le système proposé, à travers des cas d'usage.

Remerciements

Nous remercions la Plateforme M@D, la Fondation UBS et l'UCO-BS pour le soutien humain et financier.

Références

- [1] T. Alam. A reliable communication framework and its use in iot. *CSEIT*, 10 :450–456, 2018.
- [2] Y. Benazzouz. *Découverte de contexte pour une adaptation automatique de services en AmI*. PhD thesis, Saint-Etienne, EMSE, 2011.
- [3] R. Blasco, A. Marco, R. Casas, D. Cirujano, and R. Picking. A smart kitchen for ambient assisted living. *Sensors*, 14(1) :1629–1653, 2014.
- [4] O. Boissier, R. H Bordini, J. Hübner, A. Ricci, and A. Santi. Multi-agent oriented programming with jacamo. *Sc. of Computer Prog.*, 78(6) :747–761, 2013.
- [5] M. Castebrunet. *Etude et conception d’un système de personnalisation et d’aide fonctionnelle multi-agents permettant d’assister simultanément de manière transparente les activités de vie quotidienne de multiples personnes dans un Habitat Intelligent pour la Santé*. PhD thesis, Univ. de Sherbrooke, 2011.
- [6] A.C. Chaouche. *Une approche multi-agent pour la conception de systèmes d’AmI : Un modèle formel intégrant planification et apprentissage*. PhD thesis, Univ. P-M. Curie et Constantine 2, 2015.
- [7] Y. Charif and N. Sabouret. Protocole d’interaction pour la composition de services dans l’intelligence ambiante. *JFSMA*, pages 253–266, 2006.
- [8] J. Coutaz and J. Crowley. Plan “intelligence ambiante” : Défis et opportunités. document de réflexion conjoint du comité d’experts informatique ambiante st2i et du gt intelligence ambiante gcs3, 2008.
- [9] J. Da Silva and Y. Demazeau. Vowels co-ordination model. In *AAMAS*, pages 1129–1136, 2002.
- [10] T. Dujardin, J. Rouillard, J-C. Routier, J-C. Tarby, et al. Gestion intelligente d’un contexte domotique par un sma. *JFSMA*, pages 137–146, 2011.
- [11] K. Erol, J. A Hendler, and D. S Nau. Umcp : A sound and complete procedure for hierarchical task-network planning. In *Aips*, volume 94, pages 249–254, 1994.
- [12] M. Gallissot, J. Caelen, F. Jambon, and B. Meillon. Une plate-forme usage pour l’intégration de l’informatique ambiante dans l’habitat : Domus. *Techniques et sciences informatiques*, 2013.
- [13] S. Giroux, M. Castebrunet, O. Boissier, and V. Rialle. A multiagent approach to personalization and assistance to multiple persons in a smart home. In *Workshops at AAAI*, 2014.
- [14] F. Hamoui, M. Huchard, C. Urtado, and S. Vauttier. Specification of a component-based domotic system to support user-defined scenarios. In *SEKE*, 2009.
- [15] F. Hamoui, M. Huchard, C. Urtado, and S. Vauttier. Un système d’agents à base de composants pour les environnements domotiques. In *LMO*, page 35, 2010.
- [16] T. Herrmann, A. Schmidt, and M. Degeling. From interaction to intervention : An approach for keeping humans in control in the context of socio-technical systems. In *STPIS@ CAiSE*, pages 101–110, 2018.
- [17] T Kirste. Smart environments and self-organizing appliance ensembles. *True Visions*, Springer, 2005.
- [18] S. Kubicki, S. Lepreux, C. Kolski, C. Perrot, and J. Caelen. Tangisense : présentation d’une table interactive avec technologie rfid permettant la manipulation d’objets tangibles et traçables. In *IHM*, 2009.
- [19] N. Kuijpers. *Système autonome de sécurité lors de la préparation d’un repas pour les personnes cognitivement déficientes dans un habitat intelligent pour la santé*. PhD thesis, Univ. Bretagne Sud, 2017.
- [20] N. Kuijpers, S. Giroux, F. de Lamotte, and J-L. Philippe. Proposal of an adaptive service providing system for a multi-user smart home. In *AAAI*, 2016.
- [21] Y. Laurillau and L. Nigay. Clover architecture for groupware. In *ACM CSCW*, pages 236–245, 2002.
- [22] S. Legleye and A. Rolland. 1 personne sur 6 n’utilise pas internet, plus d’un usager sur 3 manque de compétences numériques de base. *Insee*, N° 1780, 2019.
- [23] R. Martins and F. Meneguzzi. A smart home model using jacamo framework. In *INDIN*, page 94, 2014.
- [24] S. Mazac, F. Armetta, and S. Hassas. Approche décentralisée pour un appr. constructiviste en env. continu : application à l’intelli. ambiante. In *JFSMA*, 2015.
- [25] N. Meliti. Architecture basée agents pour le diagnostic d’un système d’iot. *Mémoire de Master en Info.*, Univ. d’Oum Bouaghi Larbi Ben M’hidi, 2017.
- [26] J. Morales, M. Lopez, J. A Rodriguez, M. Wooldridge, and W. Vasconcelos. Automated synthesis of normative systems. In *AAMAS*, pages 483–490, 2013.
- [27] M. Nakamura, H. Igaki, H. Tamada, and K-I. Matsumoto. Implementing integrated services of networked home appliances using service oriented architecture. In *ISOC*, pages 269–278, 2004.
- [28] D. Nau, T-C Au, O Ilghami, U. Kuter, D. Wu, F. Yaman, H. Munoz, and J. Murdock. Applications of shop and shop2. *Intelligent Systems*, 20(2) :34–41, 2005.
- [29] OMS. *World report on disability*. 2011.
- [30] Y. Rahal, Ph. Mabillean, and H. Pigot. Bayesian filtering and anonymous sensors for localization in a smart home. In *AINAW vol.2*, pages 793–797. IEEE, 2007.
- [31] P. Rust, G. Picard, and F. Ramparany. Resilient distributed constraint optimization in physical mas. In *ECAI*, 2020.
- [32] A. Schmitt, F. Carlier, V. Renault, and P. Leroux. Communication multi-niveaux pour des iot-a. In *RJ-CIA*, 2017.
- [33] L. Thévin. *Un SMA normatif pour le soutien évaluatif à la collaboration humain-machine : application à la gestion de crise*. PhD thesis, Grenoble Alpes, 2016.
- [34] X. Yao, M. Turoff, and R. Hiltz. A field trial of a collaborative online scenario creation system for emergency management. In *ISCRAM*, volume 1, 2010.

Delay Aware Universal Notice Network: Real world multi-robot transfer learning

S. Beaussant¹, S. Lengagne¹, B. Thuilot¹, O. Stasse²

¹ Université Clermont Auvergne, CNRS, Clermont Auvergne INP,

² LAAS-CNRS, Université de Toulouse, CNRS

¹firstname.surname@uca.fr

²olivier.stasse@laas.fr

Résumé

Un problème majeur lors du passage de la simulation à la réalité est le retard sur le robot physique qui peut détériorer les performances de l'agent déployé. En outre, une fois qu'une politique de contrôle basée sur l'apprentissage a été entraînée avec succès, la réaffectation des connaissances acquises par l'agent pour permettre à un agent structurellement distinct d'effectuer la même tâche est dangereuse si elle est effectuée naïvement. Dans ce travail, nous abordons les problèmes ci-dessus avec une seule méthode, la DA-UNN (Delay Aware Universal Notice Network), qui décompose les connaissances en modules spécifiques au robot et à la tâche pour un transfert rapide. Notre méthode traite les retards immanents aux systèmes physiques afin d'améliorer le transfert sim2réel. Nous évaluons l'efficacité de notre approche en utilisant des robots simulés et réels sur une tâche de manipulation dynamique où la gestion des délais est cruciale.

Abstract

One major issue for the simulation to real world transfer is the delay on the physical robot that may deteriorate the performance of the deployed agent. Furthermore, once a successfully trained learning-based control policy is available, re-purposing the knowledge acquired by the agent to enable a structurally distinct agent to perform the same task is hazardous if done naively. In this work, we address the above issues with a single method, the DA-UNN (Delay Aware Universal Notice Network), which decomposes the knowledge into robot-specific and task-specific modules for fast transfer. Our framework deals with delays immanent to physical systems in order to improve sim2real transfer. We evaluate the efficiency of our approach using simulated and actual robots on a dynamic manipulation task where delay management is crucial.

Keywords

Transfer learning, Sim2real, Reinforcement learning, Robotic task, delay

1 Introduction

The Reinforcement Learning (RL) field has been successfully applied to a wide range of problems in the past years, demonstrating both its versatility and efficiency. The robotic domain in particular benefited from the tremendous progresses made in RL as shown in recent work for quadruped robots[1] even succeeding in very intricate manipulation tasks such as solving a rubik's cube with a shadow hand [2]. However, despite great achievements, RL suffers from very low sample efficiency, which means that a large amount of interactions with the environment is needed to obtain a high-performance policy. One way to mitigate this issue is through the use of transfer learning which alleviates the burden of training models from scratch by re-purposing knowledge acquired on another domain.

Nevertheless, in the context of RL, the neural network architectures are too shallow and do not encourage knowledge segmentation as it is often the case in large Computer Vision models [3]. As a consequence, if no particular precaution is taken, the unconstrained backpropagation procedure may result in an entangled knowledge representation. In this setting, it is difficult to determine which part of the network is relative to the task or the robot, making a partial or total transfer of the policy network hazardous if done naively and with very low chances of success.

The UNN (Universal Notice Network) framework introduced in [4] tackles this issue by implementing the idea of knowledge segmentation between the agent and the task. More precisely, the purpose of the UNN framework is to enable multi-task and multi-robot transfer by creating a reusable and robot-agnostic module of skills. Previous work [4] already demonstrated the efficiency of UNN transfer in simulation on a wide variety of tasks and robots morphology, exhibiting zero-shot performance in some cases. By applying the same transfer method on real world robots, we wish to find out if results obtained in simulation generalize well to the real world.

However, simulation to real world transfer (also called sim2real in the literature) is still an open problem. In general, simulations are imperfect and difficult to calibrate. The resulting modeling discrepancies cause a reality gap, which

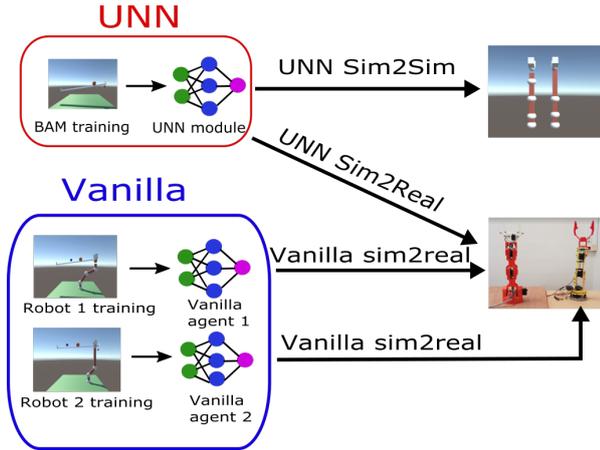


FIGURE 1 – Transfers considered. UNN module is trained with the BAM robot and then transferred to all robots. Vanilla agents are trained directly on the simulated robots and then transferred on the corresponding physical robots.

makes the transfer of RL policies from simulation to the real-world non-trivial. Sim2real is especially appealing as offers an efficient alternative to expansive real world data collections for learning complex robotic skills by training in a general-purpose simulator. However, most of the time, methods focus solely on domain adaptation between the real world and the simulation [5],[6]. They tend to ignore troublesome hardware specific issues such as control latency. Consequently, the policy obtained by training in simulation could be drastically disturbed once transferred in the real world if the task requires short reaction time.

In this paper, we consider the time delay associated with the physical system as another model’s input by including it in the observed state. At training time, we randomize the value of the delay and show that the agent is able to adapt to multiple delays on a dynamic and delay-sensitive manipulation task.

Our contributions are as follows :

1. We present and evaluate a delay-aware method to deal with the immanent delay on real hardware, thus furthering the adaptation capabilities of the UNN.
2. We evaluate the benefits of the UNN multi-robot transfer method over a vanilla transfer on real world robots. A pool containing four differently shaped real and virtual robots will solve a dynamic manipulation task they have not been trained on, by using the knowledge created by another agent as depicted in Figure 1.

2 Related Work

Transfer learning in RL has been recognized as an important direction towards building more sophisticated agents. For instance, multi-task learning aims at improving robots versatility via methods such as meta-learning [7],[8]. Another interesting approach is to hierarchically decompose

complex problems into tractable, simpler and reusable modules of skills through the use of concept networks [9].

Transfer between morphologically distinct robots on the other hand, is currently less studied in the scientific literature. A method proposed in [10] encapsulates and leverages skills learned by a task expert by using GAN’s discriminators as support for the knowledge transfer. Work by Gupta and Devin [11] presents a method to learn an invariant feature space for transferring skills between different robots. Other work by the same authors [12] uses a modular approach by training policy modules that are decomposed over robots and tasks. The UNN method used in this work, while very similar, differs from the prior method by explicitly defining the state shared between the task-specific and robot-specific modules. This choice suppresses the need of relying on a shared latent space between robot and task which may be prone to over-fitting if the number of available modules is too low. Furthermore, in the UNN framework, the task module can be trained separately and only once, thus creating a truly robot-agnostic module while saving the time and trouble of having to train multiple possible pairs of the training set.

Several methods has been proposed to deal with delays in robotic. In [13], the authors proposed a neural network based method to address the control delay issue. A predictor, approximated by a neural network, must infer the current state of a fast moving robot by observing a vector of stacked outdated states. To account for the imperfect actuators of the real hardware, authors of [2] introduced action delays with a probability of 0.5 at the beginning of every simulation training episode of a neural network with a LSTM layer. By doing so , they force the memory enhanced neural network to adapt to action delays. A cornerstone paper in RL with delay [14] learns a model of the undelayed Markov Decision Process (MDP) to simulate the most likely state in which the agent currently is, given the last observed (delayed) state and the k last actions taken since, k being the delay in timestep. This allows the agent to take decisions based on the expected current state rather than an outdated state, effectively undoing the harmful effect of delays. They also introduced the concept of Constant Delayed MDP used in section 4.1.

3 Universal Notice Network

3.1 The UNN pipeline

Instead of learning a single policy that will have to handle both robot control and task resolution, the UNN method [4] relies on an explicit decomposition between task-specific and robot-specific knowledge as depicted in Figure 2. In this modular approach, a model of the task, the *UNN* (Figure 2) is created in such a way that any robot, regardless of its morphology, number of articulations or actuators can efficiently benefit from it. This is similar to creating a notice containing a set of high level instructions, that any kind of robot could follow to solve a given task. It is primary to

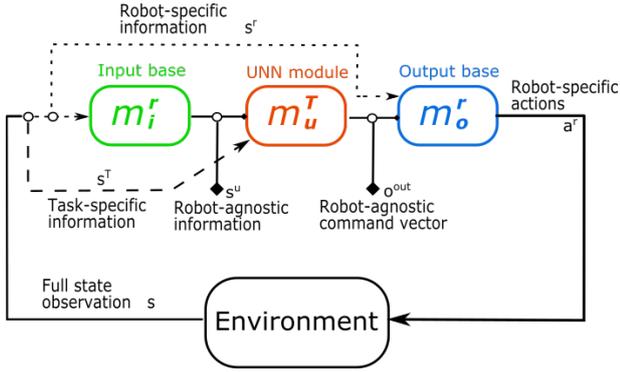


FIGURE 2 – Schematic representation of the UNN[4]

ensure that the robot has the mobility and capacities required to comply with the UNN instructions and accomplish the task. Defining R the set of feasible actions the robot can produce and U the set of actions required by the UNN to perform a task, we assume in this paper that the robot can perform the required actions, as sum up in :

$$U \subseteq R \quad (1)$$

Once this assumption is verified, two conditions are essential :

- First, the UNN module must be robot-agnostic to enable multi-robot transfer. This implies that robot-specific observations have to be translated into a feature space shared by the considered robots before being fed to the UNN module.
- Secondly, it is necessary to design a controller that will map the UNN commands from the shared feature space into low-level, robot-specific actions that the robot can execute.

These two requirements are handled by two additional modules inherent to the robot morphology called the *bases*. They are paired with the UNN and serve as an interface between the robot and the task module. More formally, the three modules form a pipeline composed of the input base m_i^r and the output base m_o^r , specific to the robot, which handle respectively the first and the second conditions, and the UNN m_u^T robot-agnostic and specific to the task only. In this setting, the state vector that is provided by the environment at each timestep can be split into two parts s^r, s^T , respectively holding data intrinsic to the considered robot and task-related information, independent from the agent. The input base m_i^r receives s^r to compute :

$$s^U = m_i^r(s^r) \quad (2)$$

which can be considered as robot-agnostic. The input base m_i^r is thus responsible for mapping the robot space to the shared feature space where the UNN operates. The next processing stage is the UNN module (i.e. the task module), conditioned by the task related observation s^T and the processed agent representation vector s^U . It then computes :

$$o^{out} = m_u^T(s^T, s^U) \quad (3)$$

where o^{out} is the command vector in the shared feature space. Finally, o^{out} is then re-mapped to the robot space by the output base with the following transformation :

$$a^r = m_o^r(o^{out}, s^r) \quad (4)$$

which yields a^r the effective action taken by the robot. In other words, the UNN module focuses solely on solving the task at hand, ignoring low level considerations such as the robot's DoF and shape, handled by the bases. This approach makes it possible to create a reusable module of skills that can be transferred to structurally different robots as long as their bases are available. It is then possible to build a library of UNN modules and robot's modules, draw any subset of interest from it and combine a UNN/Bases pair into a novel fully functional policy.

3.2 Modules training

In practice, each of the three sub-modules m_i^r, m_u^T, m_o^r can be either learned or obtained via analytical methods.

3.2.1 Bases modules

In the case where bases are obtained using neural networks, they can be trained on a suitable primitive task to acquire basic motor skills. Another alternative is to collect a dataset of trajectories of the robot and fit a regression model with supervised learning techniques. A last alternative, used in this work, consists to use analytical models for the robots bases.

3.2.2 UNN module

The UNN module (or task module) can be trained with or without the bases modules. In the first case, the UNN is coupled with a robot and its associated bases. The UNN interacts with the environment through the bases and its error on the task is back propagated through the network. In this case, we affect only the UNN module weights. However, the UNN module may then take advantage of the robot hardware structure to achieve the task (for instance, blocking an object between two articulations). As a consequence, the UNN may favor certain body configurations which may be detrimental for transfer. This issue is solved by using the Base Abstracted Modeling (BAM) method [15]. It assimilates the robot to its effector by setting m_i^r and m_o^r to identity mappings, thus making no assumption on the robot's constitution and preventing any bias related to the bases. This is equivalent to considering a purely virtual and free-flying robot. Using BAM enables faster convergence of the policy and a more defined knowledge segmentation, which in turns improves UNN transfer.

4 Delay Aware Universal Notice Network

4.1 Constant Delayed Markov Decision Process

The standard UNN proved its efficiency and versatility on a broad panel of tasks in simulation. However, these results

were obtained with perfect robots (e.g no offset and no delay) acting in a standard Markov Decision Process (MDP). Traditionally it is assumed in RL that at every timestep, the environment pauses while the agent receives the current observation, in order to derive an action that will be executed without delay. Of course, things do not behave this way in the real world. All agent observations and actions are delayed by an amount depending on the hardware used for the task. Therefore, an agent trained in simulation without exposition to delays will perform worse or even fail in the real world if no precaution is taken.

This brings up the need to adopt a different decision process modeling to solve tasks in the presence of delay. As we consider the delay to be constant, we found the Constant Delay MDP formulation introduced in [14] to be well suited. A CDMDP defines the delay d as the number of timesteps between an agent occupying a state and receiving its feedback from the environment, where $d \in \mathbb{R}^+$. The delay is assumed to be part of the environment. A known result in CDMPD is that observation delay and action delay are equivalent from the agent’s point of view [16]. Hence, we treated the total delay as being entirely caused by observation delay (see Figure 3).

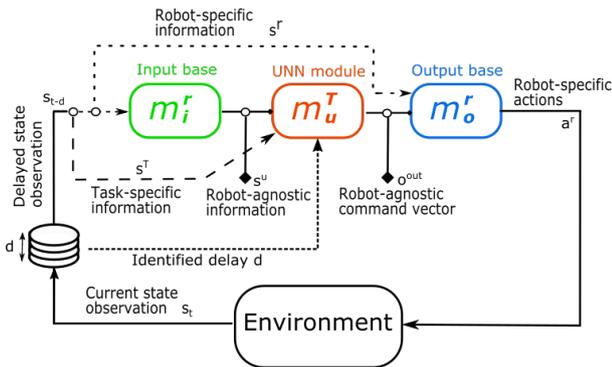


FIGURE 3 – Schematic representation of the delay aware UNN. Observations are queued into a pile of length d and each timestep, the observation at the top is fed to the agent (first in, first out).

4.2 Delay Aware UNN

A CDMDP can be transformed into a regular MDP by enlarging the state space with a history of the d last actions taken since the last observation. This transformation allows theoretically to derive an optimal policy for the CDMDP considered [17]. However, this approach does not allow direct transfer between systems with different delays as the input dimension depends on d . In this work we address the delay issue by augmenting the state space of the UNN module with the estimated delay of the system and by training the agent on a corresponding delayed environment as depicted in Figure 3. A key feature of the UNN is its ability to adapt to any robot regardless of its morphology. To keep this idea of “universality”, the delay was randomized during training to ensure that the UNN can adapt to a wide range of delay. By giving it access to the immanent delay, we enable

the UNN to act accordingly and to develop predictive capabilities. Thus, we add d to the task specific observations.

During training, the delay is sampled regularly from a discrete uniform distribution as $\mathcal{U}(d_{min}, d_{max})$ where d_{min} and d_{max} are respectively the minimum and the maximum delay considered for the environments. Since there is no assumption about the systems, we assumed a uniform distribution of the delay. But any knowledge could be used to deduce a better delay distribution. When deployed, the identified delay of the system is fed to the UNN, so it can act accordingly. While our approach can only yield sub-optimal CDMDP policies due to the incomplete state space considered, we believe that it represents an interesting trade-off between optimality and flexibility. This very simple method can improve drastically the performance of an agent on a delayed MDP as presented in section VI, given that the delay has been accurately determined and is suited for transfer on systems with different delays.

5 Experimental setup

5.1 System Architecture and Robots

In this section, we briefly present the different robots adopted throughout these experiments. We tested our method on both physical and simulated robots to demonstrate its efficiency and versatility. The physical robots used were a serial arm braccio robot with 5 DoF and a 4 DoF serial arm. These DIY robots are cheap and usually hard to work with, given their low reliability. Still, we manage to use them efficiently in our experiments. We also considered their simulated counterparts.

To sum things up, 5 different kinds of robots were used :

- **BAM** : the virtual BAM robot with the identity bases.
- **Robot 1** : the virtual braccio robot.
- **Robot 2** : the virtual 4 DoF robot.
- **Robot 3** : the physical braccio robot.
- **Robot 4** : the physical 4 DoF robot.

A fixed webcam was used to obtain the required pose estimations with OpenCV. The control frequency was 10 Hz, which means the agent was observing the environment state and acting every 0.1 second. The nominal delay was in average 300 ms on the physical systems. We identified the delay by measuring the time between a command send to the robot and the observation by the agent that the robot moved. On the simulation side, agent’s training was performed in simulation using the Unity physic simulator with the ML-agent package introduced in [18], a set of convenient tools for RL with a complete and reliable implementation of several RL algorithms. The PPO algorithm[19] was used to create the neural network policies, as it provides a monotonous performance improvement while being perfectly adapted to continuous action spaces. On-policy algorithm are also known to deal better with delays. We trained four kinds of agents :

- **Delay Aware UNN Agent** : The BAM virtual robot is trained in simulation with exposition to randomized delays to create the UNN.

- **Delay Aware Vanilla Agent** : The agent is trained from scratch directly on the simulated robot, with exposition to randomized delays.
- Finally, we also considered their **delay unaware** counterparts, trained without exposition to delays, in order to display the benefits of our delay management approach.

These agents will be used for the transfers detailed in section 6.2 (see Figure 1).

5.2 Task description

We display our method benefits on a 2D manipulation task (planar task), where a robot needs to keep a ball at a desired position on a gutter. In this regard, only 3DoF were required for the physical robots (base rotation and wrist roll unused). To further increase the gap between both robots, the Robot 4 was used as a 2 DoF robot (wrist pitch unused). The gutter is fixed at one end and held at the other end by the robot’s effector which therefore decides of its orientation and, as a consequence, of the position of the ball (see Figure 4). This task can be formalized with the following MDP :

State : $s_t \in \mathbb{R}^{4+1}$: the ball position and velocity on the gutter, the effector height, the desired ball position and the system delay d for the delay aware agents.

Action : $a_t \in \mathbb{R}^n$ is the target joints position (n being the number of considered joints). However, the vanilla agent was not making any progress with a full access to the action space. Indeed, to balance the ball on the gutter, it is first needed to hold it properly. These desired body configurations are just a fraction of the full state space and it is very unlikely to discover them without any prior knowledge of the task. To ease the vanilla agent learning process, its action space was constrained to output joints offsets values w.r.t a reference joints position which maintained the gutter in an equilibrium position.

Reward :

$$r_t = \begin{cases} r - \beta|\theta_e| & \text{if } d_{des,b} < \delta \\ -\alpha d_{des,b} - \beta|\theta_e| & \text{else} \end{cases} \quad (5)$$

where r is a small positive reward, δ is the positive reward area and $d_{des,b}$ is the distance between the ball and the desired ball position. $|\theta_e|$ is the angle between the effector pose and the vertical plane, α and β a weighting constant. This penalty ensures that the effector is in the right orientation to hold the gutter properly for the vanilla agent. The effector orientation constraint for the UNN is handled by the output base, which means that β is set to zero when training the UNN.

5.3 Delay Aware UNN creation

Creating the UNN module means training an agent to balance the gutter and keep the ball at the desired position. As the BAM method showed better transfer results [15], we decided to use it to create the UNN module. In this setting the robot is assimilated to its effector and the UNN output $o^{out} \in \mathbb{R}$ is a single value indicating at what height below or above the horizontal reference position of the gutter the effector should be. To take actions, the agent observes s^T , specific task information as well as the effector

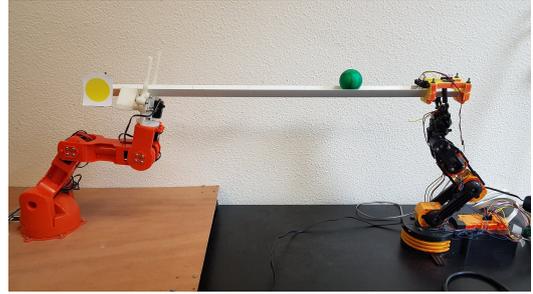


FIGURE 4 – Physical experiments setup. Left robot will perform the task, while the right robot is used only to hold one end of the gutter.

height given by $m_i^r(s^r)$. For this task, we chose the intermediate state s^u and o^{out} shared between the UNN module and the bases to be the effector position. The UNN module was receiving an extra input $d \sim \mathcal{U}(0.1, 1)$ representing the current delay of the system during training. A delay range between 0 and 1 is recommended as it corresponds to a normalized input. In our case, it also corresponds to our actual delay in second, with 0.1s being the smallest delay possible for our control frequency. The delay was created on the simulator by stacking the observations in a FIFO buffer, before feeding them to the UNN module.

6 Results

In this section, we present our results both on training and transferring on the chosen manipulation task. In particular, we compare the UNN agents with the vanilla agents with and without delay awareness. For further experiments, delay was added artificially to the real system with the same FIFO method seen in section 5.3. Code can be found at github.com/sabeaussan/DelayAwareUNN. Videos showing our results are available here.

6.1 Training

During the training, the desired ball position and system delay (for delay aware agents) were regularly changed to improve the adaptive capabilities and re-usability of the UNN. More precisely, a new delay d was sampled from $\mathcal{U}(0.1, 1)$ every 15 episodes. The desired ball position given to the model, varying between 20% and 80% of the gutter length, was also sampled from a uniform distribution $\mathcal{U}(0.2, 0.8)$ every 1000 training steps. Both the BAM agents and the Vanilla agents were trained for 4 millions steps. Figure 5 shows the cumulative reward obtained per episode. Only the term $d_{des,b}$ (distance between the ball and the desired position) common to both reward functions was considered for the comparison, as it reflects the agent overall progression on the task. As shown in Figure 5, the BAM agents in both settings converge slightly faster than their vanilla counterparts. The BAM agents focus solely on the task, leaving robot specific considerations to their bases. This decomposition of the learning problem similar to hierarchical RL eases the learning process. It is also worth noting that introdu-

cing varying delay during training reduces the convergence speed, as the task becomes more challenging. However, in the UNN framework, this training overhead is outweighed by the increased reusability of the UNN module.

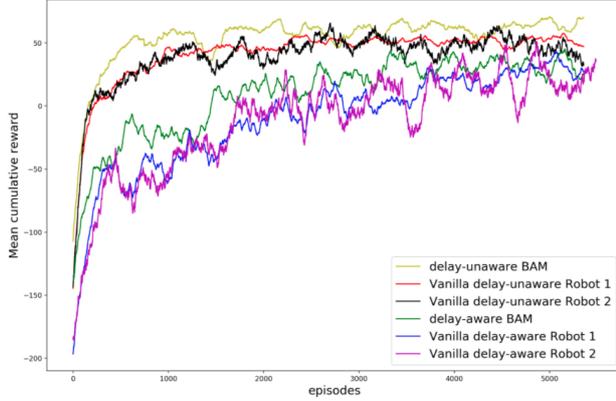


FIGURE 5 – Training curves. All the agents were trained for 400000 steps.

6.2 Transfer

There is two kinds of transfer to consider : simulation to real robot transfer and robot to robot transfer. The UNN framework mitigates the sim2real transfer problem by considering the real robot and the simulated one as two different robots, each one with its own bases, thus partially addressing the sim2real transfer as a robot to robot transfer. In this section we evaluate two methods of transfer

- **UNN transfer** : Once trained to convergence with the BAM robot, the UNN module is transferred to each robot of the set.
- **Vanilla transfer** : The vanilla agents trained on the simulated robot are directly transferred to their physical counterpart. This will serve as a baseline to study the UNN benefits for sim2real transfer.

The performance metric used was the integral of the absolute value of the error between the ball position and the desired ball position over time. This metric has the advantage of taking into account both settling time and the steady state error (the closer to 0, the better). For a fair comparison, each experiment has been conducted with the same settings (same initial ball position and desired ball position). Performances displayed in Tables 1 and 2 were averaged over 50 episodes.

6.2.1 Influence of delay

In this part, we evaluate the first contribution of this work : our delay management method, on both simulated and physical robots. Three delays were considered for the experiments : 300 ms (corresponding to the delay on the physical system), 500 ms and 700 ms. Figure 6 shows ball trajectories for the three delays considered, obtained by the UNN

Robots/Delays	0.3	0.5	0.7
BAM	3.12 / 15.57	3.94 / 22.57	4.91 / 25.28
UNN Robot 1	3.26 / 11.84	3.96 / 20.12	4.98 / 23.48
UNN Robot 2	3.43 / 12.21	4.19 / 18.47	5.10 / 21.19

TABLE 1 – Sim2sim transfer. Performances obtained for the UNN transfer on the simulated robots. Results are displayed with delay aware method on the left / delay unaware method on the right.

agents on robot 1 (virtual braccio robot) and 3 (physical braccio robot). On the simulation side, we added an optimal trajectory obtained with the delay unaware UNN agent acting on an undelayed environments to serve as a reference (see Figure 6a). The same agent was then exposed to the delays considered to study how quick performance deteriorates for unaware agents as the delay increases. As shown, agents not exposed to delays during training completely failed and systematically overshoot when trying to get the ball at the required position in delayed environment. Figure 6b emphasizes the inability of the delay unaware agents to cope with the physical system immanent delay (300 ms) as the ball starts oscillating. Moreover, as the delay increases, the delay unaware agents tend to become unstable. As for delay aware agents, in the simulation, they still manage to follow closely the optimal trajectory.

Table 1 shows the performances obtained in sim2sim transfer with the UNN agents on both delay aware and unaware settings. It is shown that delay aware agents perform from 3.5 to 5.72 times better than their unaware counterparts. It is also clear from looking at Figure 6b and Table 2, which shows the average performance after sim2real transfer, that dealing with delay in simulation greatly improves the results of the UNN agents once deployed on the physical robots. Vanilla agents also benefited from this delay management method, as shown in Table 2b, demonstrating the versatility of the proposed method.

Robots/Delays	0.3	0.5	0.7
BAM	3.12 / 15.57	3.94 / 22.57	4.91 / 25.28
UNN Robot 3	4.78 / 9.88	5.05 / 22.32	8.02 / 24.43
UNN Robot 4	5.86 / 15.72	7.45 / 22.42	8.76 / 24.41

(a) UNN transfer : BAM \rightarrow robot 3 and BAM \rightarrow robot 4

Robots/Delays	0.3	0.5	0.7
Vanilla Robot 1	3.32 / 17.33	4.22 / 26.17	5.56 / 31.48
Vanilla Robot 3	5.43 / 19.65	5.97 / 28.22	9.33 / 33.43
Vanilla Robot 2	3.78 / 18.63	4.81 / 26.45	6.16 / 32.48
Vanilla Robot 4	7.58 / 21.13	9.55 / 27.05	10.62 / 33.82

(b) Vanilla transfer : robot 1 \rightarrow robot 3 and robot 2 \rightarrow robot 4

TABLE 2 – Sim2real transfer. Performances obtained for the vanilla transfer and the UNN transfer on the physical robots. Results are displayed as delay aware method on the left / delay unaware method on the right.

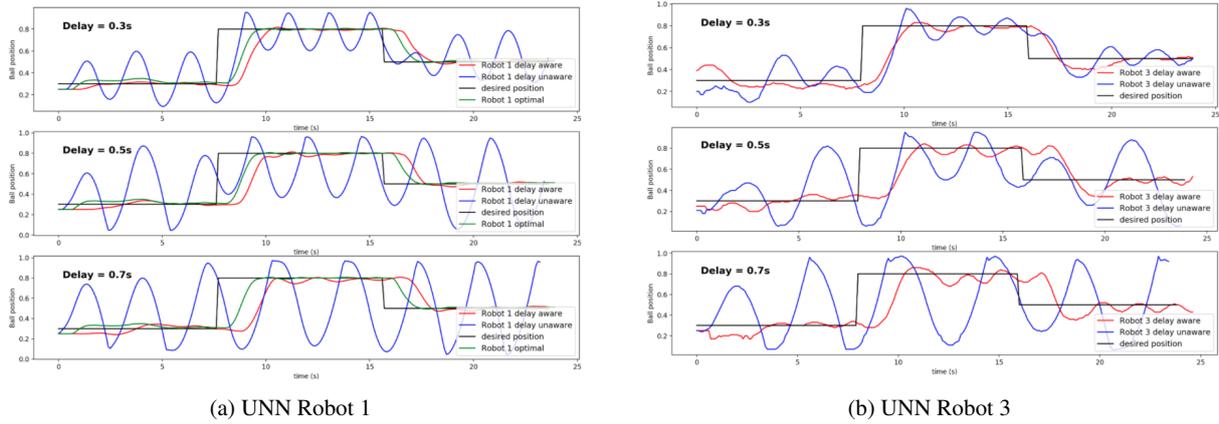


FIGURE 6 – Ball trajectory with 0.3, 0.8 and 0.5 as desired ball position.

6.2.2 sim2sim transfer

In this paragraph, we discuss the results obtained when transferring the delay aware UNN module from the BAM robot to robots 1 and 2 in simulation. We also compare the performance obtained against delay aware vanilla agents which learned the task from scratch on robots 1 and 2. As shown in Tables 1 and 2b, UNN-based approaches slightly outperform the policy of the vanilla agents for the robots and delays considered. We want to emphasize that the UNN module has been trained only once and on only one robot, the BAM robot, but still performs better than the vanilla agents specifically trained on robots 1 and 2. These results demonstrate the appealing re-usability and effectiveness of the UNN module. In some cases, the delay aware UNN agents achieve zero-shot performances (e.g robot 1 on delay 0.5). In the worst case, the transfer efficiency is 90.9% (3.12/3.43), 100% being the performance obtained by the UNN module on the BAM robot. In average, the transfer efficiency is 97.7 % for robot 1 and 93.7% on robot 2. Ideally, the UNN module paired with any of the robots would yield similar performance as with the BAM robot if equation (1) was respected. However, in some cases the body configurations required to comply with the UNN commands are not precisely achievable by the robot. For instance, the desired effector position need some of the joints to rotate beyond their limits. This also explain why the UNN transfer is less efficient on the 2 DoF robot, as it is less expressive and has a harder time following UNN commands.

6.2.3 sim2real transfer

In this paragraph, we study the UNN methodology as a sim2real transfer tool. More specifically, we compare the performance obtained after transfer on the physical robots for Vanilla agents and UNN agents. In this case, both the UNN module obtained on the BAM robot and the vanilla agents obtained on the simulated robots, were transferred to the physical robots. As shown in Figure 6, the UNN agent on robot 3 still manages to put the ball at the desired po-

sitions without too much overshooting. Table 2 shows the results obtained. As usual, the agents trained in simulation and transferred to the real world show lower performance than their virtual counterparts due to the reality gap. However, they still manage to obtain decent performances. One notable result is that the delay unaware agents transferred to the physical robot obtain very poor performance unlike their delay-aware counterpart. Once again, UNN based agents outperform vanilla agents. Moreover, the UNN based transfer reaches up to 78% (100% corresponds to the BAM performance) in the best case, while the vanilla transfer reaches 70.6% (100% corresponds to the vanilla agent on robot 1). In average, the UNN sim2real transfer efficiency is 68% on robot 3 and 54% on robot 4, against 63% on robot 3 and 52.7% for robot 4 for the vanilla sim2real transfer. As mentioned earlier, this slight sim2real improvement can be attributed to the robot-agnostic nature of the UNN module. Indeed, even if the vanilla agents were trained in simulation with a virtual copy, it remains an inaccurate model of the physical robot. The UNN on the other hand ignores those discrepancies by considering the physical robot and the virtual one as two different robots, each with their own bases.

6.3 Discussion and perspectives

From the previous results, it appears clearly that the delay management method used considerably improves the performances when working with delayed environment, as its often the case on the real world. Moreover, the UNN approach not only achieves very efficient transfer between robots in simulation, but slightly improves sim2real transfer over vanilla transfer. However, the zero-shot sim2real transfer efficiency is nowhere near what was obtained for the sim2sim transfers but further training could be done on the physical robots to achieve better performance. As aforementioned, the UNN mitigates the sim2real transfer by considering the physical system as just another robot that can be interfaced with the UNN module. Nevertheless, the UNN module which was trained in simulation can still overfit on its environment. As a result, the instruc-

tions given can be unsuitable if it is placed in a new domain with a slightly different state distribution, e.g the real world. Fortunately, the UNN approach can be combined with state-of-the-art sim2real methods such as automatic domain randomization[2] to improve sim2real transfer.

7 Conclusion

In this work, we studied the benefits of the UNN transfer for a sim2real application. More specifically, we addressed the delay management problem that occurs when working with a physical system by making the UNN “aware” of the latency of the system it is working with. By doing so, we extended the versatility of the UNN method and the range of compatible systems. We demonstrated this method efficiency by solving a dynamic manipulation task where delay management is paramount and showed that transfer across systems with heterogeneous delays and structurally distinct robots is possible. However, the UNN approach only is not sufficient for efficient sim2real transfer, but could be enhanced with other sim2real methods. This work empirically demonstrated the feasibility of our approach on a low dimensional task. Future work will investigate the efficiency of our delay-management method on a higher dimensional task.

Remerciements

This research was supported supported by the French Research Agency ANR through the AIM project.

Références

- [1] V. Tsounis et al. DeepGait : Planning and Control of Quadrupedal Gaits using Deep Reinforcement Learning. *RA-L*, 2020.
- [2] OpenAI et al. Solving Rubik’s Cube with a Robot Hand. *arXiv :1910.07113*, 2019.
- [3] J. Donahue et al. Decaf : A deep convolutional activation feature for generic visual recognition. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 647–655, 2014.
- [4] M. Mounsif et al. Universal Notice Network : Transferable Knowledge Among Agents. In *6th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 563–568, 2019.
- [5] K. Bousmalis et al. Using Simulation and Domain Adaptation to Improve Efficiency of Deep Robotic Grasping. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4243–4250, Brisbane, QLD, May 2018.
- [6] K. Arndt et al. Meta Reinforcement Learning for Sim-to-real Domain Adaptation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2725–2731, Paris, France, May 2020.
- [7] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135, 2017.
- [8] Y. Duan et al. RL² : Fast Reinforcement Learning via Slow Reinforcement Learning. *arXiv :1611.02779*, November 2016. *arXiv : 1611.02779 version : 2*.
- [9] A. Gudimella et al. Deep Reinforcement Learning for Dexterous Manipulation with Concept Networks. *ICRA*, 2020.
- [10] M. Mounsif et al. CoachGAN : Fast Adversarial Transfer Learning between Differently Shaped Entities . In *Proceedings of the 17th International Conference on Informatics in Control, Automation and Robotics*, pages 89–96, 2020.
- [11] A. Gupta et al. Learning Invariant Feature Spaces to Transfer Skills with Reinforcement Learning. *ICLR*, 2017.
- [12] C. Devin et al. Learning modular neural network policies for multi-task and multi-robot transfer. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2169–2176, Singapore, 2017.
- [13] S. Behnke et al. Predicting away robot control latency. In *RoboCup 2003 : Robot Soccer World Cup VII*, pages 712–719, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.
- [14] T.J. Walsh et al. Learning and planning in environments with delayed feedback. *Autonomous Agents and Multi-Agent Systems*, 18(1) :83–105, February 2009.
- [15] M. Mounsif et al. Bam! base abstracted modeling with universal notice network : Fast skill transfer between mobile manipulators. In *7th 2020 International Conference on Control, Decision and Information Technologies (IEEE-CoDIT)*, July 2019.
- [16] K.V. Katsikopoulos and S.E. Engelbrecht. Markov decision processes with delays and asynchronous cost collection. *IEEE Transactions on Automatic Control*, 48(4) :568–574, April 2003.
- [17] Dimitri P. Bertsekas. *Dynamic programming and optimal control. Vol.1.* Number 1 in Athena scientific optimization and computation series. Athena Scientific Publ, Belmont, Mass, 2. ed edition, 2000. OCLC : 833754683.
- [18] A. Juliani et al. Unity : A General Platform for Intelligent Agents. *arXiv :1809.02627*, May 2020. *arXiv : 1809.02627*.
- [19] J. Schulman et al. Proximal Policy Optimization Algorithms. *arXiv :1707.06347*, August 2017. *arXiv : 1707.06347 version : 2*.

Evaluation of the sensitivity of cognitive biases in the design of artificial intelligence.

M. Cazes^{1*}, N. Franiatte^{1*}, A. Delmas², J-M. André¹, M. Rodier³, I. Chraïbi Kaadoud⁴

¹ ENSC-Bordeaux INP, IMS, UMR CNRS 5218, Bordeaux, France

² Onepoint - R&D Department, Bordeaux, France

³ IBM - University chair "Sciences et Technologies Cognitives" in ENSC, Bordeaux, France

⁴ IMT Atlantique, Lab-STICC, UMR CNRS 6285, F-29238 Brest, France

Abstract

The reduction of algorithmic biases is a major issue in the field of artificial intelligence. Despite the diversity of sources and temporalities at the origin of these biases, the current solutions to achieve ethical results are mostly technical. The consideration of human factors and more particularly cognitive biases remains incomplete. Nevertheless, the task of designing artificial intelligence systems is conducive to the emergence of cognitive biases. The aim of our study is to test the awareness of individuals who design artificial intelligence systems of the impact of their cognitive biases in their productions. The study focuses on conformity bias, confirmation bias and illusory correlation bias. The first results of this pre-experimentation show that these individuals believe that their decisions are subject to the cognitive biases of conformity, illusory correlation and confirmation.

Keywords

cognitive bias, artificial intelligence, decision making, human factors, human-IA interaction

Introduction

The question of biases, and their reduction, is a major issue in artificial intelligence (AI), especially since the raise of all the ethical issues related to deep learning algorithms application [39, 27, 42, 13]. These lead to more research on human-AI interaction field that studies human impact on AI systems [27, 6], and inversely, on the question of the impact of those systems on human [15, 36, 29].

Our work is in the field of human-AI interaction, not only machine learning, and questions the impact of human on AI systems during their design.

We define an AI system as a computational software, an algorithm or a scientific methodology using a set of algorithms, that assists humans to make a decision based on data analysis and data processing (mining, clustering, classification, prediction, recommendations, object recognition, etc.) whether in industrial or scientific context. We consider as an

AI system any system that involves at least one of the following elements: an AI symbolic system (e.g. decision trees, expert systems) or a machine learning or deep learning algorithm (e.g. shallow and deep neural networks). From a software management point of view, the conception of an AI system goes through different steps that can be summarized as follows [3]: design, implementation of the code, test and production. We will refer in our work to all these steps as AI design. By doing so, we aim to group all the actors that can impact the product life cycle of an AI system. We will refer to these actors as AI professionals: AI researchers, AI manager products, data scientists, data analysts, data architects, data engineers and AI developers. Thus, the purpose of the current work is to propose a pre-experiment in the shape of a survey submitted to AI professionals in order to identify whether they are aware of being influenced in their work by cognitive biases classically known in human factors: conformity bias, confirmation bias, and illusory correlation bias. We therefore propose to conduct a study to test the following hypotheses (Figure 5):

- H0: Actors involved in the design of AI systems are aware that the biases of illusory correlation, conformity and confirmation influence their task.
- H1: Feedback collected during the qualitative interviews shows that AI systems design choices are influenced by these cognitive biases.

We adopt a different approach for works in literature by using human factors knowledge to address the issue of the presence of biases in AI systems by focusing first on the AI professionals' sensitivity of the cognitive biases. We will test H0 in this pre-experiment. This work is thus a first step in a larger question of the evaluation of cognitive biases on AI systems from a human factor perspective.

As the subject of biases in AI systems is much debated and studied within the AI community, we are concerned about clarifying the contribution and position of the current work: we do not aim to establish any causal effect of relation between cognitive biases and algorithmic biases.

*these authors contributed equally to this work

These concepts are clearly distinct. However there will be cited in our work since they both intervene in the larger research field of human-AI interaction. Table 1 provides definitions and non-exhaustive examples of the two distinct concepts according to the literature.

	Cognitive bias	Algorithmic bias
Definitions	<p>"Distortion (systematic deviation from a norm) that information undergoes when entering or leaving the cognitive system. In the first case, the subject selects the information, in the second, he selects the answers" [26]</p> <p>"Systematic pattern of deviations from the norm or rationality in judgment" [16]</p>	<p>"Algorithmic bias can occur when the data used to train a machine learning algorithm reflects the implicit values of the humans involved in collecting, selecting, or using that data" [38]</p> <p>"Problems related to the gathering or processing of data that might result in prejudiced decisions on the bases of demographic features such as race, sex, and so forth" [39]</p>
Examples	<p>Confirmation bias [49]</p> <p>Conformity bias [20]</p> <p>Illusory correlation bias [8]</p>	<p>Omitted variable bias [5]</p> <p>Data selection bias [31]</p>

TABLE 1 – Differences between cognitive and algorithmic bias: extraction of definitions from scientific studies and non-exhaustive examples [26, 38, 16, 39, 49, 20, 8, 5, 31]

1 The biases of artificial intelligence

[42] showed that Amazon’s facial recognition AI was very effective in performing facial recognition for light-skinned men (almost zero error rate) but had an error rate of 31.4% for dark-skinned women. This is one of the studies that highlights ethical issues related to discriminatory practices of AI. Their social consequences are concrete [39] as shown by the results of the NGO ProPublica’s survey showing that the software used to predict crimes in the USA (The Correctional Offender Management Profiling for Alternative Sanctions, COMPAS) had a ‘racist bias’ [13]. The explanation of these discriminatory practices requires an understanding of the emergence of bias in the results of AI algorithms. We consider algorithmic bias, as defined in Table 1, including for example ‘Problems related to the gathering or processing of data that might result in prejudiced decisions on the bases of demographic features such as race, sex, and so forth’ [39]. The sources of algorithmic biases are of various natures and occur at different temporalities. Algorithmic biases may appear upstream (selection of learning data), during (inclusion of certain variables) and downstream (interpretation bias) of the learning phase [44]. Moreover, it is essential to qualify the nature of the source of these biases: technical bias (omitted variable bias, database bias, selection bias...) or ‘society bias’ (emotional or cognitive biases) [31]. However, despite the identification of the diversity of the sources of algorithmic biases, the correction of these biases is considered only from a statistical and algorithmic performance point of view i.e. in relation to certain metrics such as the percentage of prediction error or the accuracy in machine learning [39] (Figure 2). Indeed, [39] denounces the sole search for technical performance, which is neither relevant nor desirable to deal entirely with algorithmic biases. The influence of the human characteristics of designers on programmes is known [36], as recommendations are made regarding the need to incorporate diversity

in the profiles of AI professionals [29, 34] (Figure 2, bottom right). However, there are no empirical studies investigating how human factors - and especially cognitive biases - can be concretely illustrated in the AI design process [40].

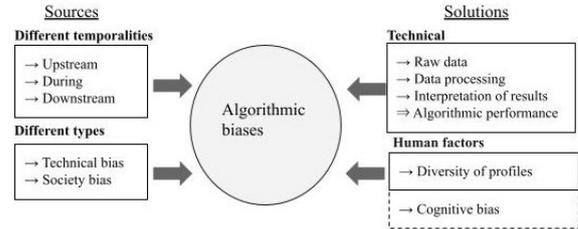


FIGURE 1 – Sources of algorithmic biases and solutions implemented [34] [31] [44]

We note that the debates around algorithmic biases bring into play problems of taking into account human factors in the actors of the design of artificial intelligence systems. We propose to carry out a study on this topic and more particularly the issue of the cognitive biases of the actors of this design chain.

2 Artificial intelligence design and cognitive biases

The conception of an AI system is a complex task. [32] develops the concept of complexity: a phenomenon is complex when ‘the whole is more than the sum of its parts’ i.e. there is an emergence of information. The conception of an AI system is a question of manipulating phenomena whose complexity is not mastered [1, 33]. In particular concerning the neural networks which make information emerge from which it is difficult to directly apprehend the chain of causality. In industry, designing AI systems is a task subject to constraints of time, means, objectives and expectations on the part of customers. These elements are part of a complex and uncertain context that is conducive to the use of heuristics on the part of designers and developers. [21] defined a heuristic as ‘a simple procedure that makes it possible to find adequate, although often imperfect, answers to difficult questions’. These heuristics reduce the complexity of the task but can lead to the appearance of cognitive biases. [23] wrote as early as 1974 : ‘In general, these heuristics are very useful, but sometimes they lead to severe and systematic errors’. The ‘severe and systematic errors’ considered in the quotation from Kahneman and Tversky [23] correspond to what will be referred to later as cognitive biases. It seems to us that they can be defined in this way as a systematic deviation of logical and rational thinking from reality. Some authors have sought to model the functioning of cognitive biases by considering the theory of dual processes [48, 21] (Figure 2). The latter is based on the dichotomy between two main modes of reasoning: unconscious, rapid and automatic processes on the one hand, and slow and deliberate processes on the other. These two modes are colloquially referred to as ‘intuitive’ and ‘analytical’ but are

also known as System 1 (intuitive) and System 2 (analytical). According to [21], these two systems should be 'seen as agents with their own capabilities, limitations and functions'. Nevertheless, these 'agents' interact with each other and thus divide up certain tasks related to information processing. For example, while system 1 will handle automatic activities such as 'orienting towards the source of a sudden noise' or 'making a face of disgust at a horrible image', system 2 will handle tasks such as 'focusing on the voice of a particular person in a crowded and noisy room' or 'checking the validity of a complex logical argument'.

Interactions / Division of tasks

	System 1	System 2
Functions	<ul style="list-style-type: none"> - Automatic operations - Innate and acquired skills (perception and understanding) - Combines ideas 	<ul style="list-style-type: none"> - Operations with attention control - Compelling mental activities - Develops thoughts in an orderly series of steps
Associated processes	<ul style="list-style-type: none"> - Fast and automatic - Little or no effort - No sense of deliberative control - Emotional component 	<ul style="list-style-type: none"> - Slow and deliberate - Continuous relative effort - Orderly mental work - Emotionally neutral

FIGURE 2 – Functions and processes associated with Systems 1 and 2, based on the work of Daniel Kahneman and Amos Tversky [23] [21]

Also, it appears that the notion of expertise is associated with the development of better performance of system 1. However, this 'intuitive expertise' is only valid when the task to be performed is part of a predictable environment in which the individual can learn from regularities [22]. In the case of developers of AI systems, the result of their actions on the neural network involves an element of uncertainty. In other words, their understanding of the impact of the modification of certain parameters is not complete. For example, [24] emphasised the fact that it seems unlikely that a single programmer would have a complete understanding of a large complex system designed by many teams around the world. Thus, despite the sense of expertise and mastery, the system of intuitive thinking cannot be fully effective.

In the field of computer science, the concrete bias impacts on developers has already been proved. Studies have shown that developers' cognitive confirmation bias has an influence on the quality of their programs. Indeed, [7] demonstrate that developers do unit testing - a software testing method to determine whether the written code is fit for use [19] - to show that their system works rather than testing it. More precisely, in the field of AI, research highlighted the concrete impacts of cognitive biases of the people involved in the design and implementation of AI programs. For example, the choice of the set of training variables for the creation of AI recruitment software has an important role in the quality and results of the program. However, this choice is made by managers and developers who are subject to their own cognitive biases such as generalization bias [45].

Finally, the task of designing an AI system has characteristics conducive to the emergence of cognitive biases. There

are a multitude of cognitive biases, and it will be a matter of identifying the most relevant ones in our context.

3 Impossible and irrelevant identification of all cognitive biases

There are multiple typologies that classify cognitive biases according to pre-established criteria [18, 4, 12]. From the seminal work of Kahneman and Tversky [23] to more recent research [43], there does not seem to be a consensus on the comprehensiveness and classification of cognitive biases. Moreover, cognitive biases raise several epistemological questions, which are important to consider as they directly influence the applications of research results [28]. In order to qualify 'diversion' i.e. a bias, a rational behaviour must be established, i.e. a norm from which the behaviour deviates [47]. But how can behaviour be described as irrational? Similarly, the nature of the task seems to influence the cognitive mechanism at work. [9] was thus able to demonstrate the existence of content effects for Wason's task: human beings use reasoning strategies appropriate to the nature of the problem they are facing. Cognitive biases would therefore not be 'mysterious irrationalities' but adaptations of the mind [16]. But how can cognitive processing be described as biased if it is context-dependent? These elements show the complexity surrounding the notion of cognitive bias, which often makes it difficult to identify and evaluate them for a specific task. [5] already identified cognitive biases that could have impacts during the design and implementation of an AI system. In accordance with this work, we have chosen to focus on the study of three cognitive biases: conformity bias, confirmation bias, and the bias of illusory correlation.

4 Conformity bias, confirmation bias, illusory correlation bias

According to the codex of cognitive bias [30], the conformity bias and illusory correlation are due to the problem of the lack of sense of the individual's environment. In order to create meaning, individuals extrapolate 'attributes on the basis of stereotypes, generalities or antecedents' (translation of [30]). This led, among other reasons, to the development of the explainable AI and interpretable AI fields. Indeed, the neural networks used create a combinatorial system can be difficult to understand for the human cognitive system [1] and by extent to understand how AI systems lead to the given result. The difficulty of apprehending the internal functioning of neural networks can thus lead to a lack of understanding on the part of the AI professionals [1, 33] on the behavior or inner mechanisms of their AI systems. Consequently, it is possible to assume that this task takes place in an environment conducive to the expression of the biases mentioned above (Figure 3). Moreover, designing an AI program is mainly by selecting the data that will be used as a basis for learning the neural network. Data from various sources, types and formats must be chosen so that AI can produce quality information [14]. Therefore, it is ne-

cessary to select information in an environment that is saturated with it. Indeed, according to the codex typology of cognitive biases, too much information leads to cognitive biases [30]. Faced with the need to filter this information, individuals are attracted by what confirms their own convictions. This mechanism is at the origin of confirmation bias. The design of AI systems would therefore be an environment conducive to the emergence of confirmation bias (Figure 3).

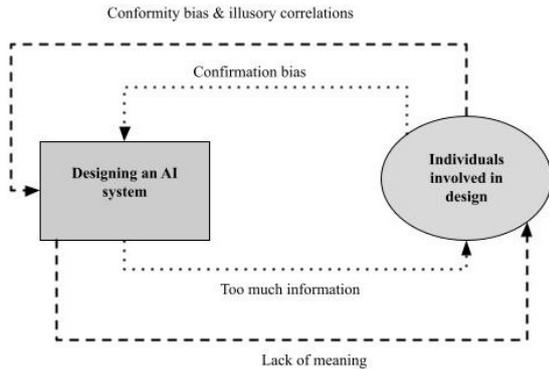


FIGURE 3 – Schematic representation of our hypothesis: link between elements of the context of the task and the cognitive biases they could encourage to appear

Finally, the characteristics of the AI systems design task seem to create an environment conducive to the emergence of the following cognitive biases: illusory correlation, conformity and confirmation bias. In order to understand more precisely how these biases operate, the following section will detail the cognitive mechanisms underlying their appearance.

4.1 Conformity Bias

Conformity bias can be defined as the ‘modification of an individual’s behavior or judgment to bring it into harmony with the behavior or judgment of the majority’ (translation of [20]). As Asch’s pioneering experience shows, it is a powerful mechanism in decision-making [2]. The consequences of conformity bias are ambivalent and contextual. It is, for example, one of the factors at the origin of collective intelligence phenomena [35] but also of ‘collective conservatism’ [46].

4.2 Confirmation bias

[49] was one of the first to point out the existence of a confirmation bias. [4] state that in forming a judgment, a large majority of individuals reason by trial and error on the basis of previous judgments, and tend to confirm the accuracy of their initial hypotheses. Thus, confirmation bias can be defined as the tendency to explain facts with consistent stories and neglect facts that contradict them [37, 11]. Confirmation bias can have different consequences depending on the context in which it is embodied. For example, in the run-up to the US presidential election, [25] studied

the evolution of political books, highlighting a confirmation bias that shows that people who buy these books do so not for information, but for confirmation of their opinion.

4.3 The illusory correlation bias

The first descriptions and explanations of the phenomenon of illusory correlation come from social psychology, and more particularly from the studies of the researchers Chapman and Chapman [8]. When two events are correlated, individuals attribute cause and effect relationships to one of them. The meaning of this correlation may be erroneous, or even totally illusory [41].

The three biases of conformity, confirmation and illusory correlation are therefore the focus of our study. After having identified them, we will transpose them into the context that actors in AI systems development may encounter.

5 Experimentation

5.1 Purpose and hypothesis

We focus our study on the impact of cognitive biases in the AI systems design process. Indeed, the nature and conditions of the AI systems design task are, in our opinion, clues to suppose that the biases of illusory correlation, conformity and confirmation would be at the origin of weaknesses in AI systems creation. There is a lack of empirical studies on how cognitive biases act precisely on decision making in an AI system design context. We therefore propose to conduct a study to test the following hypotheses (Figure 4). As a reminder, our assumptions are as follows:

- H0: Actors involved in the design of AI systems are aware that the biases of illusory correlation, conformity and confirmation influence their task.
- H1: Feedback collected during the qualitative interviews shows that AI systems design choices are influenced by these cognitive biases.

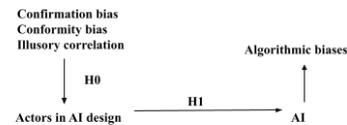


FIGURE 4 – Hypothesis on the impact of cognitive biases in AI systems design

5.2 Methodology for testing the H0 hypothesis

To test the H0 hypothesis, an online questionnaire was carried out. The purpose of this questionnaire is to perform a preliminary study to obtain the self-assessments of AI professionals of the impact of cognitive biases in their work, and to recruit target people to have more accurate feedback. The online tool FramForms was used, which anonymizes the data and removes it permanently after six months. The

questionnaire was sent via the bull-i3 mailing list on January 4, 2021, in the BAIA newsletter on January 10, 2021 and in the RISC newsletter on January 26, 2021. We have blocked the recording of the answers to the questionnaire on February 17, 2021. Bull-i3 is a mailing list of the IRIT (Toulouse Institute of Computer Science Research) which gathers all the members (industrialists, researchers, professors, PhD students,...) of the Information, Intelligence and Interaction communities concerned by the issues of these research fields. BAIA is the newsletter of the Bordeaux Artificial Intelligence Alliance and RISC is the information relay on cognitive sciences. These three mailing lists are French. This is a methodological choice. We wanted to avoid bias related to the translation of the questionnaire or cultural bias. The questionnaire is addressed to what we call AI professionals as defined above. The exclusion criteria set concern people whose expertise is not related to AI e.g. the answer of a journalist was discarded because his profile did not correspond to an AI professional. The questionnaire is structured in four main parts, each of which includes several multiple-choice questions: I) Sensitivity to cognitive bias; II) Cognitive biases and your profession; III) Your profile. The questions were originally in French (Figure 5).

* Mandatory response

Questions
I) Sensitivity to cognitive bias
Have you ever heard about the 3 cognitive biases mentioned in the email (conformity, confirmation, illusory correlation)? *
If yes, in what context?
II) Cognitive biases and your profession
Do you think that some decision making in your professional environment has already been influenced by one of the cognitive biases mentioned above? *
If yes, which one(s)?
III) Your profile
In which sector do you work? *
If private, how big is your business?
What is the field of activity of your company / laboratory / other? *
What is the title of your profession? *
How old are you?
How do you identify yourself?
What is your nationality? *

FIGURE 5 – Questions in the questionnaire sent

5.3 Results

5.3.1 Profile of sampled respondents

39 people responded to our questionnaire. Among them, 11 women (28.2%) and 28 men (71.8%). Regarding age (in years), 18 people surveyed (46.2% of the people surveyed)

were in the 35-55 age category. Next, in order of importance, come the 18-25 age group (10 people or 25.6%), then the 25-35 age group (7 people or 17.9%) and finally the 55+ age group (4 people or 10.3%).

5.3.2 Professional Profile

36 out of 39 people surveyed (92.3%) are in either the digital or artificial intelligence sector, or both. Four people combine one of these sectors with a specialty or do not belong to either one (i.e. 'Administration, data mining project', 'Social Robotics', 'Geomatics' and one person looking for work). The respondents are equally divided between the private sector (19 people) and the public sector (19 people), with one person not belonging to either of these two sectors ('In search of employment'). Of the 19 people working in the private sector, 9 (47.4%) belong to a large group, 6 (31.2%) to a start-up, 3 (15.8%) to a PME (Petite ou Moyenne entreprise in french, Small or Medium Factory) and 1 (5.3%) to neither of these categories. Finally, as regards the professions practiced by the respondents: 35.9% of them qualify as researchers/professors, the remainder are divided between student/doctoral students (17.9%), data related professions (25.6%) and other professions (including for example an application architect, a research engineer and an innovation manager) (Figure 6).

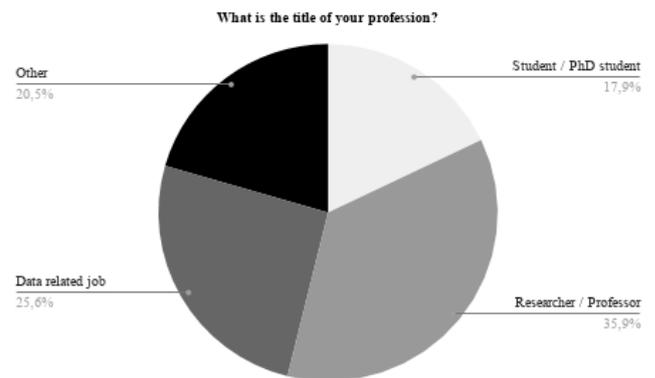


FIGURE 6 – Respondents' responses regarding the title of their professional designation.

5.3.3 Sensitivity to cognitive biases

In the questionnaire sent out, we briefly defined the biases of conformity, confirmation and illusory correlation. 66.7% of the people surveyed told us that they were aware of at least one of these three cognitive biases. Among the latter, this knowledge comes in 100% of cases from personal interest and/or professional training.

5.3.4 Potential impacts of cognitive biases on tasks in the work environment

Finally, all respondents believe that some decisions in their professional environments have already been influenced by at least one of the three cognitive biases. Among the answers obtained concerning the frequency of this influence : 56.4% answered 'Yes, regularly' and 43.6% 'Yes, rarely'.

According to the people surveyed, conformity bias seems to be the most frequent cognitive bias, followed by confirmation bias and then illusory correlation bias (Figure 7).

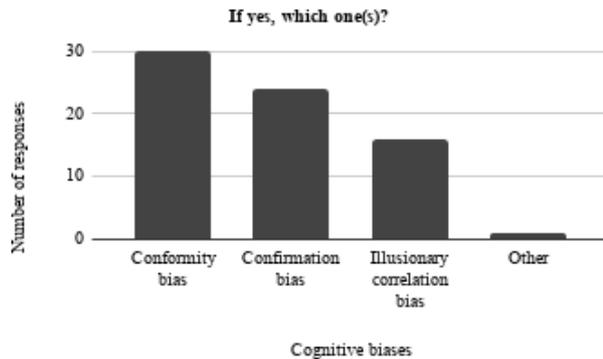


FIGURE 7 – Biases chosen by respondents who answered positively to the question 'Do you think that some decision making in your professional environment has already been influenced by one of the previously mentioned cognitive biases' (Question originally in French)

5.4 Discussion

First and foremost, the gender figures (28.2% women and 71.8% men) seem to reflect the current context, as women are still in the minority in the digital and artificial intelligence sectors. Indeed, according to INSEE, in 2017, the share of women in IT professions was 28%. Also, a study by the World Economic Forum and LinkedIn (2018) reports that only 22% of jobs in the field of artificial intelligence are held by women, and even less by senior managers.

This survey is a pre-experiment that establishes that all respondents believe that the cognitive biases studied in this article have (or have already had) an impact on their professional decision-making (39/39 or 100% of them). It confirms the need to take the human factors into account in the development of AI systems and to underline the relevance of the three cognitive biases mentioned in the article by Bertail et al [5] from which we drew inspiration. Also, these results allow us to note that the conformity bias is considered by respondents to be more influential in their professional tasks than the other two proposed biases (of confirmation and illusory correlation). It would therefore be possible to envisage that certain cognitive biases are more likely to influence decision making than others. However, this finding should be qualified in that the conformity bias may be better known to respondents than the other two biases. Also, another point to consider is the fact that it is generally easier for human beings to think that cognitive biases influence others before themselves (which would also bias our own analysis). This is an important point raised notably in the work of Fabrizio Butera [17]. We can summarize it here with the example of confirmation bias: 'The problem with confirmation bias is that it is itself subject to bias: people agree that confirmation bias exists in

others, but find it difficult to admit that it exists in them as well. A meta-bias if you like.' This notion of 'meta-bias' should therefore be taken into account in our research and should lead us to interpret any results very cautiously.

Besides, during the study, we realized the semantic - even conceptual - ambiguity that can exist around the term 'bias', whether cognitive or algorithmic. Initially, the connotation of the latter seemed to us rather negative, often leading us to wonder about the potential means of 'debiasing' the human being. Nevertheless, in the course of our work, we found it interesting to also question their possible usefulness. Consequently, the term 'bias' semantically poses a confusion, in the sense that it always refers to a distortion, often interpreted as an error, whereas this distortion is also an essential means of survival for the human being. Moreover, it is difficult to find appropriate solutions to these biases (which would imply that it would be a problem). For Fabrizio Butera, cited by Hernandez [17]: 'The term bias is confusing. It refers to a tunnel of the mind. It implies that one cannot get out of it. If you are in a tunnel, you have to go from the entrance to the exit without making a detour. This implies that we are all helpless in the face of the slippery slope of our biases. Yet this is not what we observe socially.' Thus, in this article we use the terms 'cognitive bias' and 'algorithmic bias' while trying to define as well as possible what they represent for us, in order to avoid any confusion.

A second element that can be discussed is the potential social desirability bias, which results from a tendency of the individual to want to present himself favourably in the eyes of society [10]. Indeed, although our survey was anonymized, this psychological mechanism is sometimes implicitly exercised without the subject being aware of it. It would therefore be relevant to conduct another study, not related to AI systems design, in order to verify whether the awareness of cognitive biases by individuals in their professional environments shows a trend similar to the one found in this article.

Finally, the interest of this article is to question the taking into account the human factor - and more particularly cognitive biases - in the whole AI system design cycle. The questionnaire provided initial insight into the place of cognitive biases among AI designers and developers. The study of the potential impact of cognitive biases on the actors of AI system development, and their incarnations in the resulting work, represents a problem that could also be raised in future work. This is what we would like to do next, by testing hypothesis H1 through individual interviews with people who have completed our questionnaire and agreed to be contacted. The objective is to have more qualitative and targeted feedback on the cognitive biases that can occur according to the different expertises related to the AI domain.

6 Conclusion

The aim of this work is to raise the human factors and, more particularly, cognitive biases issues in the task of designing

AI systems. Indeed, the appearance of biases from different sources [31] and at all stages of designing AI systems [44] raises the question of the impact of cognitive biases during the design of an AI system. In other words, we wish to raise the question of the impact of cognitive bias in individuals at the origin of the realisation of intelligent systems. The AI system and cognitive bias relationship is an open scientific question which tends to grow in importance, particularly thanks to the growing concern about explainable AI systems intrinsically linked to the question of responsibility and ethics in AI [1].

Our objective is to test the awareness of those involved in the design of AI systems of the influence that their cognitive biases may have in their professional decision-making. Our work suggests that people involved in AI system design processes believe that their decisions are subject to the cognitive biases of conformity, illusory correlation and confirmation.

Future work should try to understand whether there is an embodiment of these cognitive biases in the AI system, and if so, how they are embodied. This involves testing our second hypothesis (H1, Figure 4). Indeed, it is not so much a question of the cognitive biases of the individuals involved in the design of AI systems, which are natural and useful [16], but of their incarnations and therefore their consequences in artificial intelligence. Thus, it would be relevant to carry out a study using feedback, in order to evaluate how these cognitive biases are embodied, or not, in the work of individuals responsible for designing AI systems.

Authors contributions

*MC and NF contributed equally to this work. All authors contributed to manuscript revision, read and approved the submitted version.

Références

- [1] A. B. ARRIETA et al. “Explainable Artificial Intelligence (XAI) : Concepts, taxonomies, opportunities and challenges toward responsible AI.” In : *Information Fusion* 58 (2019), p. 82-115. DOI : 10 . 1016 / j . infus . 2019 . 12 . 012.
- [2] S. E. ASCH et H. GUETZKOW. “Effects of group pressure upon the modification and distortion of judgments.” In : *Organizational influence processes* 58 (1951), p. 295-303.
- [3] Marco BARENKAMP, Jonas REBSTADT et Oliver THOMAS. “Applications of AI in classical software engineering”. In : *AI Perspectives* 2.1 (2020), p. 1-15.
- [4] D.A. BAZERMAN M.H.and Moore. *The Principles of Quantum Mechanics*. John Wiley Sons., 2008.
- [5] Patrice BERTAIL et al. “Algorithmes : Biases , Discrimination et Équité”. In : *HAL* (2019). DOI : hal - 02077745.
- [6] Aylin CALISKAN, Joanna J BRYSON et Arvind NARAYANAN. “Semantics derived automatically from language corpora contain human-like biases”. In : *Science* 356.6334 (2017), p. 183-186.
- [7] Gül ÇALIKLI et Ayşe Başar BENER. “Influence of confirmation biases of developers on software quality : an empirical study”. In : *Software Quality Journal* 21.2 (2013), p. 377-416.
- [8] Loren J CHAPMAN et Jean P CHAPMAN. “Genesis of popular but erroneous psycho-diagnostic observations *”. In : *Journal of Abnormal Psychology* 72.3 (1967), p. 193-204.
- [9] Leda COSMIDES. “The logic of social exchange : Has natural selection shaped how humans reason? Studies with the Wason selection task.” In : *Cognition* 31.3 (1989), p. 187-276. DOI : 10 . 1016 / 0010-0277 (89) 90023-1.
- [10] Douglas P. CROWNE et David MARLOWE. “A new scale of social desirability independent of psychopathology”. In : *Journal of Consulting Psychology* 24.4 (1960), p. 349-354. ISSN : 00958891. DOI : 10 . 1037/h0047358.
- [11] Erica DAWSON et Dennis T REGAN. “Motivated Reasoning and Performance on the Wason Selection Task”. In : *Personality and Social Psychology Bulletin* 28.10 (2002), p. 1379-1387. DOI : 10 . 1177 / 014616702236869.
- [12] R DOBELLI. *The Art of Thinking Clearly*. 2013.
- [13] Julia DRESSEL et Hany FARID. “The accuracy, fairness, and limits of predicting recidivism”. In : *Science advances* 4.1 (2018), eaao5580.
- [14] Usama FAYYAD, Gregory PIATETSKY-SHAPIRO et Padhraic SMYTH. “From data mining to knowledge discovery in databases”. In : *AI magazine* 17.3 (1996), p. 37-37.
- [15] Diana F GORDON et Marie DESJARDINS. “Evaluation and selection of biases in machine learning”. In : *Machine learning* 20.1-2 (1995), p. 5-22.
- [16] Martie G. HASELTON, Daniel NETTLE et Paul W. ANDREWS. “The Evolution of Cognitive Bias”. In : *The Handbook of Evolutionary Psychology* (2015), p. 724-746. DOI : 10 . 1002 / 9780470939376 . ch25.
- [17] J. HERNANDEZ. *Biais de confirmation : pouvez-vous le contrer?* <https://www.futura-sciences.com/sante/actualites/psychologie-biais-confirmation-pouvez-vous-contrer-85040/>. 2021.
- [18] R.M. HOGARTH. *Judgement and choice : The psychology of decision*. John Wiley Sons., 1980.
- [19] Dorota HUIZINGA et Adam KOLAWA. *Automated defect prevention : best practices in software management*. John Wiley & Sons, 2007.

- [20] A. IONESCU S. and Banchet. *Psychologie sociale : Nouveau cours de psychologie, Licence*. Presses Universitaires de France - PUF, 2009.
- [21] D. KAHNEMAN. *Système 1 / Système 2 : Les deux vitesses de la pensée*. Flammarion, 2011.
- [22] Daniel KAHNEMAN et Gary KLEIN. “Conditions for Intuitive Expertise : A Failure to Disagree”. In : *American Psychologist* 64.6 (2009), p. 515-526. ISSN : 0003066X. DOI : 10.1037/a0016755.
- [23] Daniel KAHNEMAN et Amos TVERSKY. “Judgment under Uncertainty : Heuristics and Biases”. In : *Science* 185.4157 (1974), p. 1124-1131. DOI : 10.1126/science.185.4157.1124.
- [24] Rob KITCHIN. “Thinking critically about and researching algorithms”. In : *Information Communication and Society* 20.1 (2017), p. 14-29. ISSN : 14684462. DOI : 10.1080/1369118X.2016.1154087.
- [25] V. KREBS. *Political Polarization*. <http://www.orgnet.com/divided.html>. 2008.
- [26] Collectif LE NY. *Grand Dictionnaire de la Psychologie*. 1991.
- [27] Susan LEAVY. “Gender bias in artificial intelligence : The need for diversity and gender theory in machine learning”. In : *Proceedings of the 1st international workshop on gender equality in software engineering*. 2018, p. 14-16.
- [28] Jean-fabrice LEBRATY. “Biais cognitifs : quel statut dans la prise de décision assistée ?” In : *Systèmes d’information et management* 9.3 (2004), p. 1-27.
- [29] Bruno LEPRI et al. “Fair, transparent, and accountable algorithmic decision-making processes”. In : *Philosophy & Technology* 31.4 (2018), p. 611-627.
- [30] John MANOOGIAN III et B BENSON. “Codex des biais cognitifs”. In : *Penser critique*. (2016).
- [31] Institut MONTAIGNE. “Algorithmes : contrôle des biais svp”. In : (2020).
- [32] Edgar MORIN. “Le défi de la complexité”. In : *Chimères. Revue des schizoanalyses* (1988). DOI : <https://doi.org/10.3406/chime.1988.1060>.
- [33] Shane T MUELLER et al. “Principles of Explanation in Human-AI Systems”. In : *arXiv preprint arXiv :2102.04972* (2021).
- [34] Sarah MYERS-WEST, Meredith WHITTAKER et Kate CRAWFORD. *Discriminating systems*. April. 2019, p. 33. ISBN : 6153295159091.
- [35] Joaquin NAVAJAS et al. “Aggregated knowledge from a small number of debates outperforms the wisdom of large crowds”. In : *Nature Human Behaviour* 2.2 (2018), p. 126-132. ISSN : 23973374. DOI : 10.1038/s41562-017-0273-4. arXiv : 1703.00045.
- [36] Gregory S NELSON. “Bias in artificial intelligence”. In : *North Carolina medical journal* 80.4 (2019), p. 220-222.
- [37] Raymond S NICKERSON. “Confirmation Bias : A Ubiquitous Phenomenon in Many Guises”. In : *Review of General Psychology* 2.2 (1998), p. 175-220.
- [38] Helen NISSENBAUM. “How computer systems embody values”. In : *Computer* 34.3 (2001), p. 120-119.
- [39] Eirini NTOUTSI et al. “Bias in data-driven artificial intelligence systems—An introductory survey”. In : *Wiley Interdisciplinary Reviews : Data Mining and Knowledge Discovery* 10.3 (2020), e1356.
- [40] Tore PEDERSEN, Christian JOHANSEN et Johanna JOHANSEN. “Studying the Transfer of Biases from Programmers to Programs”. In : *arXiv* 2016 (2020). arXiv : 2005.08231.
- [41] Arun RAI et Charles STUBBART. “Can executive information systems reinforce biases ?” In : *Banking Technology*, 4.2 (1994), p. 87-106.
- [42] Inioluwa Deborah RAJI et Joy BUOLAMWINI. “Actionable Auditing : Investigating the Impact of Publicly Naming Biased Performance Results of Commercial AI Products”. In : *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society* (2019), p. 429-435. DOI : 10.1145/3306618.3314244.
- [43] Olivier SIBONY. “Comprendre et prévenir l’erreur récurrente dans les processus de décision stratégique : l’apport de la Behavioral Strategy”. Thèse de doct. PSL Research University, 2017.
- [44] Selena SILVA et Martin KENNEY. “Viewpoint algorithms, platforms, and ethnic bias”. In : *Communications of the ACM* 62.11 (2019), p. 37-39. ISSN : 15577317. DOI : 10.1145/3318157.
- [45] Melika SOLEIMANI et al. “Cognitive biases in developing biased Artificial Intelligence recruitment system”. In : *Proceedings of the 54th Hawaii International Conference on System Sciences*. 2021, p. 5091.
- [46] C. THALER R. and Sunstein. *Nudge*. Yale University Press., 2008.
- [47] Pascal WAGNER-EGGER. “Les canons de la rationalité : essai de classification des points de vue dans le débat sur les biais cognitifs et la rationalité humaine”. In : *L’Année psychologique* 111.01 (2011), p. 191. ISSN : 0003-5033. DOI : 10.4074/s0003503311001072.
- [48] P C WASON, J ST et T EVANS. “Dual processes in reasoning ?* P. C. WASON”. In : *Cognition*. 3.2 (1975).
- [49] P. C. WASON. “On the failure to eliminate hypotheses in a conceptual task”. In : *Quarterly Journal of Experimental Psychology* 12.3 (1960), p. 129-140. ISSN : 0033-555X. DOI : 10.1080/17470216008416717.

Jeux bayésiens hypergraphiques

H. Fargier¹, P. Jourdan^{1,2}, R. Sabbadin²

¹ IRIT, Université de Toulouse, Toulouse, France

² INRAE, Université de Toulouse, UR MIAT, Castanet-Tolosan, France

helene.fargier@irit.fr, {paul.jourdan, regis.sabbadin}@inrae.fr

Résumé

Cet article définit le cadre des jeux bayésiens hypergraphiques, qui permet de représenter succinctement des jeux bayésiens à interactions locales. Ce cadre généralise à la fois les jeux bayésiens sous forme normale et les jeux hypergraphiques (y compris les jeux polymatriciels). Nous montrons que les jeux bayésiens hypergraphiques (et polymatriciels) peuvent être transformés, en temps polynomial, en jeux hypergraphiques équivalents à information complète, en généralisant le théorème de Howson et Rosenthal. Cette démonstration a plusieurs conséquences. Elle permet de montrer que le problème de recherche d'un équilibre de Nash mixte dans un jeu bayésien hypergraphique ou polymatriciel est PPAD-complet tandis que le problème de recherche de l'existence d'un équilibre de Nash pur est NP-complet. Elle montre également que le problème de recherche d'un équilibre de Nash mixte dans un jeu bayésien sous forme normale est aussi PPAD-complet

Mots-clés

Théorie des jeux, jeux bayésiens, jeux hypergraphiques, décision sous incertitude

Abstract

This paper defines the framework of hypergraphical Bayesian games, which allows to concisely specify Bayesian games with local interactions. This framework generalizes both normal-form Bayesian games and hypergraphical games (including polymatrix games). Establishing a generalization of Howson and Rosenthal's Theorem, we show that hypergraphical (resp. polymatrix) Bayesian games can be transformed, in polynomial time, into equivalent complete-information hypergraphical (resp. polymatrix) games. This result has several consequences. It involves that finding a mixed Nash equilibrium in a hypergraphical or polymatrix Bayesian game is a PPAD-complete problem while the existence of a pure Nash equilibrium defines an NP-complete problem. It also implies that computing a mixed Nash-equilibrium in a standard normal-form, Bayesian game is PPAD-complete.

Keywords

Game Theory, Bayesian games, hypergraphical games, decision under uncertainty

1 Introduction

La théorie des jeux a pour but de représenter, expliquer et optimiser des situations où plusieurs agents, les joueurs, ont le choix entre plusieurs actions. L'utilité reçue par un joueur, qu'il souhaite maximiser, dépend de son action choisie ainsi que des actions choisies par les autres joueurs. Dans un jeu simultané à information complète, les joueurs ne peuvent pas coordonner leurs actions mais chacun a une connaissance totale du jeu : les joueurs, les actions possibles et les utilités.

L'hypothèse d'une information complète n'est pas toujours satisfaite : les joueurs ne connaissent que partiellement le jeu. Leur connaissance des utilités des autres joueurs peut, en particulier, être incomplète. Les jeux bayésiens [17], proposés dans les années soixante, permettent de modéliser la connaissance incomplète d'un jeu et la possibilité qu'ont les joueurs de disposer d'informations supplémentaires, représentées par la notion de "type". Le "type" d'un joueur regroupe les informations sur le jeu auxquelles il a accès.

La représentation sous forme normale d'un jeu bayésien est coûteuse. Pour un jeu à n joueurs, chacun d'entre eux disposant de t types possibles et de d actions, il faut définir et stocker n tables d'utilité de taille $t^n \times d^n$. Il existe plusieurs cadres de représentation succincte de jeux à information complète. En particulier les *jeux graphiques* [22], *jeux polymatriciels* [29] et *jeux hypergraphiques* [25] ont été proposés pour représenter des jeux à plusieurs joueurs dont les interactions sont locales. Dans ces cadres, l'utilité (dite globale) d'un joueur est la somme d'utilités (dites locales) obtenues dans différents sous-jeux (un seul, dans le cas graphique) auxquels il participe. Chacun de ces sous-jeux n'implique qu'une partie de l'ensemble des joueurs.

Ces représentations peuvent demander un espace exponentiellement inférieur à une représentation sous forme normale. Dans un jeu polymatriciel, par exemple, les interactions des agents sont représentées par un graphe où chaque agent (noeud) participe à plusieurs sous-jeux à deux joueurs sous forme normale (arêtes). Dans un cas hypergraphique les joueurs peuvent jouer à des sous-jeux avec plus de 2 joueurs. La taille de représentation d'un jeu hypergraphique est polynomiale dans le nombre d'actions possibles pour chaque joueur et dans le nombre total de joueurs et exponentielle en le nombre maximal de joueurs d'un sous-jeu. Ces cadres succincts disposent de plusieurs méthodes de ré-

solution exacte ou approchée : LCP [18], descente de gradient [14], propagation de message [22, 24, 8], programmation par contrainte [27, 16, 28]...

Dans cet article, nous proposons un cadre de représentation des jeux bayésiens plus concis que la forme normale habituelle : le cadre des jeux bayésiens *hypergraphiques*. La prochaine section présente les notions basiques sur lesquelles notre travail s'appuie. La section 3 définit les jeux bayésiens polymatriciels et hypergraphiques. Nous rappelons ensuite dans la section 4 que, grâce au théorème de Howson et Rosenthal [19], tout jeu bayésien polymatriciel peut être transformé en un jeu polymatriciel (à information complète) équivalent. Nous généralisons le théorème de Howson et Rosenthal et proposons une transformation en temps polynomial d'un jeu bayésien hypergraphique vers un jeu hypergraphique à information complète. On peut alors montrer que le problème de recherche d'un équilibre dans un jeu bayésien hypergraphique n'est pas plus difficile que dans un jeu hypergraphique "standard".

2 Contexte et notations

2.1 Jeux sous forme normale

Un jeu sous forme normale à information complète est défini par un triplet $G = \langle N, A, U \rangle$. $N = \{1, \dots, n\}$ est un ensemble fini de joueurs. $A = \times_{i \in N} A_i$ est un ensemble d'actions jointes, où A_i est l'ensemble fini des actions du joueur $i \in N$. Une action jointe est un vecteur $a \in A$ spécifiant une action a_i pour chaque joueur i . $U = \{(u_i)_{i \in N}\}$ est un ensemble de fonctions d'utilité associant à toute action $a \in A$ un réel $u_i(a)$ décrivant l'utilité du joueur i lorsque $a \in A$ est jouée. Dans un jeu sous forme normale (jeu "SNF"), u_i est donnée par une table à $|A|$ entrées.

Une stratégie mixte pour un joueur i est une distribution de probabilité sur A_i . Une stratégie mixte jointe est un n -uplet $\sigma = (\sigma_1, \dots, \sigma_n)$ où σ_i est une stratégie mixte du joueur i . L'utilité d'un joueur pour une stratégie mixte est son utilité espérée :

Définition 1 (Utilité d'une stratégie mixte). *Pour un jeu $G = \langle N, A, U \rangle$, σ une stratégie et i un joueur :*

$$EU_i(\sigma) = \sum_{a \in A} \left(\prod_{j=1}^n \sigma_j(a_j) \right) u_i(a)$$

Lorsque pour σ une seule action par joueur a une probabilité non nulle, σ est une stratégie *pure* et définit simplement une action jointe a . Dans ce cas, on a tout simplement $EU_i(\sigma) = u_i(a)$.

Dans la suite nous utilisons la notation suivante : pour tout vecteur $v = (v_1, \dots, v_n)$ dans le domaine produit $V = \times_{i \in N} V_i$ et pour n'importe quel $e \subseteq N$, v_e est la restriction de v à e et $V_e = \times_{i \in e} V_i$. Par abus de notation, nous utilisons v_i pour $v_{\{i\}}$. Pour tout i , $-i$ désigne l'ensemble $N \setminus \{i\}$, i.e. $v_{-i} = (v_1, \dots, v_{i-1}, v_{i+1}, \dots, v_n) \in V_{-i} = \times_{j \neq i} V_j$. v_{-i} est une restriction de v à tous les joueurs sauf i . Enfin, "." désignera la concaténation : $v'_i.v_{-i} = (v_1, \dots, v_{i-1}, v'_i, v_{i+1}, \dots, v_n)$. En particulier, pour deux

actions jointes $a, a' \in A$, $a'_i.a_{-i}$ désigne l'action jointe où a_i est remplacé par a'_i .

Un équilibre de Nash est une stratégie mixte jointe qui ne peut pas être modifiée de manière unilatérale par un joueur pour obtenir un gain d'utilité.

Définition 2 (Équilibre de Nash). *Une stratégie mixte σ est un équilibre de Nash ssi $EU_i(\sigma_i.\sigma_{-i}) \geq EU_i(\sigma'_i.\sigma_{-i})$, $\forall i \in N, \forall \sigma'_i$. Une action jointe a est un équilibre de Nash pur ssi $U_i(a) \geq U_i(a'_i.a_{-i})$, $\forall i \in N, \forall a'_i \in A_i$.*

Le modèle classique SNF est limité. Premièrement, l'hypothèse d'une connaissance complète n'est pas toujours satisfaite. Les joueurs peuvent ne pas être bien informés sur les propriétés du jeu : la connaissance du jeu est alors qualifiée d'*incomplète*. Les jeux bayésiens, proposés par Harsanyi [17] offrent un cadre adapté pour les jeux à information incomplète. Ils sont détaillés ci dessous. Deuxièmement, même en supposant une connaissance complète, la description d'un problème sous la forme normale est très coûteuse lorsque le nombre de joueurs augmente alors que les interactions entre les joueurs restent limitées. Le cadre des jeux hypergraphiques, qui permet un gain exponentiel en espace de cette description, est détaillé dans la sous-section 2.3.

2.2 Jeux bayésiens

Un jeu à information incomplète [17, 23] comprend un ensemble S de "mondes possibles", chaque monde correspondant à un jeu classique. L'utilité u_i d'un agent i ne dépend pas seulement des actions mais aussi du monde/jeu réel. Les joueurs ne savent pas précisément quel est le monde réel, mais ils ont une connaissance commune sur ce dernier. Juste avant de jouer, chaque joueur i reçoit une information privée $\tau_i(s)$ sur le monde réel. τ_i associe à chaque monde $s \in S$ un élément θ_i d'un ensemble Θ_i appelé l'ensemble des "types" du joueur i . Après avoir observé $\tau_i(s)$, l'agent i en sait plus sur le jeu réel, mais plusieurs jeux restent souvent possibles. L'agent conditionne ses connaissances par $\tau_i(s)$ et décide quelle action jouer. Les agents peuvent obtenir des informations différentes et par conséquent, ne partagent pas la même connaissance a posteriori sur le monde réel.

Harsanyi propose de décrire les jeux à information incomplète à partir de l'ensemble des types $\Theta = \Theta_1 \times \dots \times \Theta_n$ (les mondes sous-jacents sont omis). L'idée de Harsanyi en définissant les types était que la notion de type encapsule toutes les informations d'un jeu auxquelles les joueurs ont accès. Elle n'inclut pas seulement le statut du monde que les joueurs peuvent observer mais aussi leur croyance sur les autres joueurs et leur état mental introspectif¹.

Les jeux bayésiens sont des jeux à information incomplète où la connaissance a priori des joueurs est commune et est capturée par une distribution de probabilité :

Définition 3 (Jeu bayésien). *Un jeu bayésien (B-game) G est un n -uplet $\langle N, A, \Theta, P, U \rangle$ où :*

— N est un ensemble fini de n joueurs $\{1, \dots, n\}$;

¹ Voir [5, 2, 3, 4, 13] pour les liens entre les états de croyance et types, et pour plus de développements sur la théorie des jeux épistémiques.

- $A = \times_{i \in N} A_i$, où A_i est l'ensemble des actions du joueur i ;
- $\Theta = \times_{i \in N} \Theta_i$, où Θ_i est l'ensemble des types du joueur i , regroupe toute les configurations $\theta = (\theta_1, \dots, \theta_n)$ de types;
- $P : \Theta \rightarrow [0, 1]$ est une distribution de probabilité jointe sur les combinaisons de types;
- $U = \{(u_i)_{i \in N}\}$ où $u_i : A \times \Theta \rightarrow \mathbb{R}$ est la fonction d'utilité du joueur i .

P capture la connaissance a priori des joueurs. Cette connaissance est commune. L'information que les joueurs ont sur le monde réel correspond à un $\theta \in \Theta$, mais n'est pas commune. i n'a pas connaissance de θ mais seulement de θ_i . $P(\cdot|\theta_i)$ représente la connaissance a posteriori qu'a l'agent i de type θ_i à propos du jeu. D'autre part, l'utilité $u_i(a, \theta)$ est l'utilité pour i de l'action jointe a quand le monde réel est décrit par θ . Elle est reçue seulement après que tous les joueurs aient joué leurs actions et révélé leurs types : u_i dépend de θ et non pas seulement de θ_i .

Un joueur i , de type θ_i , joue une stratégie mixte $\sigma_i(\cdot|\theta_i)$, qui est une distribution de probabilité sur A_i . Comme sous information complète, une stratégie pure est une stratégie mixte particulière.

Dans un jeu bayésien, pour toute combinaison de types $\theta \in \Theta$, la probabilité qu'une action jointe $a \in A$ soit sélectionnée est $\sigma(a|\theta) = \prod_{i \in N} \sigma_i(a_i|\theta_i)$.

L'utilité espérée d'une stratégie σ pour un joueur i de type θ_i est l'espérance de l'utilité de i selon la distribution σ , conditionnée par le type θ_i reçu par i :

Définition 4 (Utilité d'une stratégie dans jeu bayésien).

$$EU_i(\sigma|\theta_i) = \sum_{\theta_{-i}} P(\theta_{-i}|\theta_i) \sum_{a \in A} \left(\prod_{j=1}^n \sigma_j(a_j|\theta_j) \right) u_i(a, \theta)$$

Comme dans un jeu à information complète, σ est un équilibre de Nash mixte ssi $EU_i(\sigma_i, \sigma_{-i}|\theta_i) \geq EU_i(a_i, \sigma_{-i}|\theta_i), \forall i \in N, \theta_i \in \Theta_i, \forall a_i \in A_i$.

Un jeu bayésien sous forme normale est décrit par une distribution de probabilité de taille t^n et par n tables d'utilité de taille $t^n \cdot d^n$ (pour une question de simplicité, nous considérons que tous les joueurs ont t types et d actions possibles). Même en tenant compte de cette représentation en extension coûteuse, le problème de recherche d'un équilibre de Nash est difficile : pour un jeu bayésien sous forme normale, déterminer s'il existe un équilibre de Nash pur est un problème NP-complet [10] et le problème de recherche d'un équilibre mixte est PPAD-difficile même s'il n'y a que deux joueurs [9, 12].

2.3 Jeux hypergraphiques

La question de la représentation succincte des jeux est importante pour les jeux classiques non bayésiens : la représentation d'un jeu classique sous sa forme SNF est coûteuse (n tables de taille d^n) et, comme mentionné par [15] :

"For large population of games (modeling for instance agents interactions over the Internet), the

SNF is practically unfeasible, while the more succinct graphical normal form works very well, and is actually a more natural representation."

En effet, dans beaucoup de situations, l'utilité d'un joueur ne va pas dépendre des actions de tous les autres – les interactions sont souvent *locales*. Pour les jeux classiques, [29] a proposé les *jeux polymatriciels* comme moyen de représentation des jeux à plusieurs joueurs avec des interactions par paires. Ce formalisme a été généralisé dans les années 2000 pour décrire les interactions locales comprenant plus de deux joueurs [22, 25] :

Définition 5 (Jeu hypergraphique). Un jeu hypergraphique est un n -uplet $G = \langle N, A, E, U \rangle$ où :

- $N = \{1, \dots, n\}$ est un ensemble fini de joueurs;
- $A = \times_{i \in N} A_i$, où A_i est l'ensemble fini d'actions du joueur $i \in N$;
- E est un ensemble de sous-ensembles de N , i.e. $\langle N, E \rangle$ est un hypergraphe décrivant les interactions locales;
- $U = \{u_i^e\}_{e \in E, i \in e}$ est un ensemble de fonctions d'utilité locales. Chaque u_i^e associe à $a_e \in A_e$ une utilité dans \mathbb{R} .

Chaque hyperarête e correspond à un jeu SNF classique $G^e = \langle e, A_e, U_e \rangle$: e est l'ensemble des joueurs, $A^e = \times_{i \in e} A_i$ et $U_e = \{u_i^e, i \in e\}$. S'il n'existe aucune hyperarête contenant la paire (i, j) , cela signifie que la fonction d'utilité du joueur i ne dépend pas de l'action choisie par le joueur j . Cependant, les stratégies de deux joueurs ne jouant pas ensemble peuvent tout de même être liées indirectement via les actions des autres joueurs dont ils dépendent. Dans un jeu hypergraphique, la fonction d'utilité globale d'un joueur est la somme des utilités récupérées par ce joueur dans les jeux locaux dans lesquels il est impliqué.

Définition 6 (Utilité d'une stratégie jointe pure).

$$u_i(a) = \sum_{e \in E \text{ tq } i \in e} u_i^e(a_e), \forall i \in N, \forall a \in A$$

En d'autres termes, un jeu SNF équivalent à un jeu hypergraphique $G = \langle N, A, E, U \rangle$ est un jeu $\langle N, A, \{u_i, i \in N\} \rangle$, u_i étant donnée par la définition 6.

L'utilité d'une stratégie mixte σ est définie par :

Définition 7 (Utilité d'une stratégie mixte).

$$EU_i(\sigma) = \sum_{a \in A} \sigma(a) \cdot \left(\sum_{e \in E, i \in e} u_i^e(a_e) \right)$$

Quand chaque jeu local est un jeu à 2 joueurs, G est un jeu *polymatriciel*. Les jeux graphiques, proposés à l'origine par [22], sont des jeux sous forme normale où la table d'utilité de chaque joueur ne dépend des actions que d'un sous-ensemble de joueurs. Ces jeux correspondent à des jeux hypergraphiques où $E = \{e_1, \dots, e_n\}$, où $i \in e_i, \forall i = 1..n$ et où $u_j^{e_i}(a_{e_i}) = 0, \forall j \neq i$. L'espace de représentation requis pour les jeux hypergraphiques, graphiques et polymatriciels peut être exponentiellement inférieur à celui demandé

pour une représentation équivalente sous forme normale [25, 28, 26]. Par exemple, un jeu polymatriciel demande de stocker $2 \cdot |E| \cdot d^2$ nombres ($|E| < n^2$) tandis que le jeu SNF correspondant demande de stocker $n \cdot d^n$ nombres.

3 Jeux bayésiens hypergraphiques

3.1 Définitions et propriétés

Les jeux hypergraphiques permettent une représentation succincte des jeux à informations complète. Nous allons les étendre au cas bayésien :

Définition 8 (Jeu bayésien hypergraphique). *Un jeu bayésien hypergraphique (raccourci en (H, B) -game), $G = \langle N, A, \Theta, E, P, U \rangle$, est défini par :*

- $N = \{1, \dots, n\}$ est un ensemble fini de joueurs ;
- $A = \times_{i \in N} A_i$, où A_i est l'ensemble fini des actions de i ;
- $\Theta = \times_{i \in N} \Theta_i$, où Θ_i est l'ensemble fini des types de i ;
- E est un ensemble de sous-ensembles de N , i.e. $\langle N, E \rangle$ est un hypergraphe ;
- $\{P_e, e \in E\}$ est un ensemble de distributions de probabilité locales $P_e : \Theta_e \mapsto [0, 1]$ où Θ_e est le type joint local dans e ;
- $U = \{u_i^e\}_{e \in E, i \in e}$ est l'ensemble des fonctions d'utilité locales : chaque u_i^e est une fonction de $A_e \times \Theta_e$ vers \mathbb{R} .

Lorsque $|e| \leq 2, \forall e \in E$, G est un jeu bayésien polymatriciel.

On notera $G^e = \langle e, A_e, \Theta_e, P_e, \{u_i^e, i \in e\} \rangle$ le jeu local correspondant à $e \in E$ dans le (H, B) game.

Définition 9 (Utilité globale dans un (H, B) -game).

$$u_i(a, \theta) = \sum_{e \in E \text{ tq } i \in e} u_i^e(a_e, \theta_e), \forall a \in A, \theta \in \Theta$$

Les définitions 8 et 9 capturent, en tant que cas dégénérés, les jeux hypergraphiques et polymatriciels (sans incertitude : $|\Theta| = 1$) d'un côté et de l'autre les jeux bayésiens (une seule hyperarête $e = N$). Mais à la différence des jeux bayésiens, $u_i(a, \theta)$ ne dépend pas forcément des types et des actions de tous les joueurs.

La condition de "connaissance commune" est moins naturelle dans le cas d'une série de jeux bayésiens locaux que dans un jeu bayésien SNF. En effet, il n'y a pas de raison pour que le type d'un joueur l'informe des types des joueurs avec lesquels il n'est pas lié par un jeu local. Ainsi, nous réduisons la condition de "connaissance commune" dans les (H, B) -game à une simple condition de cohérence des distributions de probabilité locales. En d'autres mots, nous supposons que les croyances des joueurs sont cohérentes avec une probabilité a priori jointe (inconnue) sur les types :

Hypothèse 1 (Cohérence des distributions de probabilités locales). *Il existe une distribution de probabilité jointe P sur Θ telle que :*

$$P_e(\theta_e) = \sum_{\theta_{-e}} P(\theta_e, \theta_{-e}), \forall e \in E, \theta_e \in \Theta_e.$$

Dans un (H, B) -game la connaissance a priori sur un groupe de joueurs $v \subseteq N$ est cohérente d'un jeu local à l'autre : $\forall v \subseteq N, \forall e, e'$ tels que $v \subseteq e \cap e'$,

$$\sum_{\theta_{e \setminus v}} P_e(\theta_v, \theta_{e \setminus v}) = \sum_{\theta_{e' \setminus v}} P_{e'}(\theta_v, \theta_{e' \setminus v}) = P_v(\theta_v).$$

Le (H, B) -game est donc équivalent au jeu bayésien $\langle N, A, \Theta, P, \{u_i, i \in N\} \rangle$ où u_i est donnée par la définition 9. Notons que dans la définition 4, EU_i dépend de P , qui porte sur les types de tous les joueurs et n'est pas totalement connue par le joueur i . Seule sa marginalisation aux jeux locaux auxquels i participe est connue. On peut montrer que :

Théorème 1 (Décomposition de l'utilité espérée). *Soit σ une stratégie mixte pour un (H, B) -game G :*

$$EU_i(\sigma|\theta_i) = \sum_{e \in E \text{ tq } i \in e} EU_i^e(\sigma_e|\theta_i), \forall i \in N, \theta_i \in \Theta_i,$$

où $EU_i^e(\sigma_e|\theta_i)$ est l'utilité espérée de σ_e dans le jeu bayésien local $G^e = \langle e, A_e, \Theta_e, P_e, \{u_i^e, i \in e\} \rangle$.

Donc, sous l'hypothèse 1, la connaissance des distributions locales sur les types est suffisante pour calculer l'utilité d'une stratégie mixte. Le cadre est donc bien fondé et opérationnel.

3.2 Exemple

Prenons un exemple inspiré de [26], qui est un jeu de coordination entre des joueurs choisissant un fournisseur Internet. Dans ce problème, la satisfaction d'un joueur dépend des choix de ses voisins proches, voisinage qui est représenté par un graphe $\langle N, E \rangle$. Plus le nombre de voisins d'un joueur utilisant le même fournisseur est élevé, plus la satisfaction de ce joueur sera élevée. L'incertitude du problème vient des possibilités d'offres que les joueurs peuvent recevoir des fournisseurs. Recevoir une offre peut augmenter la préférence pour ce fournisseur et donner une information sur les autres agents : *si je reçois une offre, mes voisins probablement aussi*.

Les jeux bayésiens polymatriciels fournissent une représentation appropriée à ce type de problème. Considérons un jeu bayésien polymatriciel à 4 joueurs dont la figure 1 représente le graphe.

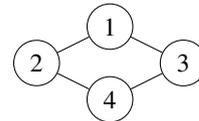


FIGURE 1 – Graphe d'un jeu bayésien polymatriciel à 4 joueur.

Chaque joueur i a deux actions, $A_i = \{b, c\}$, correspondant aux choix des fournisseurs, respectivement *Berizon* ou *C-Mobile*. *Berizon* a envoyé des offres à certains joueurs mais les joueurs n'ont pas une connaissance complète de qui a reçu ces offres. $\Theta_i = \{R, \bar{R}\}$, R et \bar{R} correspondant

respectivement à "a reçu" et "n'a pas reçu" une offre de Berizon. Pour chaque paire de joueurs possible (chaque arête) une fonction d'utilité est associée à cette paire et dépend des 4 combinaisons de paires de types possibles. Un joueur qui n'a pas reçu d'offre (joueur de type \bar{R}) préfère choisir c plutôt que b . Un joueur qui a reçu une offre (joueur de type R) préfère b plutôt que c (les offres fonctionnent !). Qu'ils aient reçu une offre ou non, les joueurs qui ne se coordonnent pas dans un jeu local avec leur voisin ne sont pas satisfaits. La figure 2 formalise un tel jeu bayésien SNF local à 2 joueurs. Tous les jeux bayésiens locaux sont supposés identiques.

		R		\bar{R}	
		b	c	b	c
R	b	3, 3	0, 0	3, 1	0, 0
	c	0, 0	2, 2	0, 0	2, 2
\bar{R}	b	1, 3	0, 0	1, 1	0, 0
	c	0, 0	2, 2	0, 0	2, 2

FIGURE 2 – Tables d'utilité pour une paire de joueurs (i, j) du jeu bayésien polymatriciel.

Les distributions de probabilité locales sont supposées identiques :

$$P(\theta_i = R, \theta_j = R) = 0.4, P(\theta_i = R, \theta_j = \bar{R}) = 0.1, \\ P(\theta_i = \bar{R}, \theta_j = R) = 0.1 \text{ et } P(\theta_i = \bar{R}, \theta_j = \bar{R}) = 0.4.$$

Nous pouvons calculer les probabilités conditionnelles et marginales des types des jeux bimatriels locaux :

$$\begin{aligned} &— P(\theta_j = R) = 0.5 \text{ et } P(\theta_j = \bar{R}) = 0.5, \forall j \in N, \\ &— P(\theta_j = R | \theta_i = R) = 0.8, P(\theta_j = \bar{R} | \theta_i = R) = 0.2, \\ &— P(\theta_j = R | \theta_i = \bar{R}) = 0.2, P(\theta_j = \bar{R} | \theta_i = \bar{R}) = 0.8, \forall (i, j) \in E \end{aligned}$$

Avec cette distribution, il est plus probable que le voisin d'un joueur reçoive une offre si ce joueur en a reçu une ou que le voisin ne reçoive pas d'offre si le joueur n'en a pas reçu.

Dans un jeu bayésien polymatriciel, pour tout joueur $i \in N$,

$$\begin{aligned} EU_i(\sigma | \theta_i) &= \sum_{e \text{ tq } i \in e} EU_i^e(\sigma_e | \theta_i), \text{ où} \\ EU_i^e(\sigma_e | \theta_i) &= EU_i^{(i,j)}(\sigma_i, \sigma_j | \theta_i) \\ &= \sum_{\theta_j} P(\theta_j | \theta_i) \times \\ &\quad \sum_{a_i, a_j} \sigma(a_i, a_j | \theta_i, \theta_j) u_i(a_i, a_j, \theta_i, \theta_j) \end{aligned}$$

Soit une stratégie σ^* consistant en quatre stratégies pures identiques σ_i^* , tel que $\sigma_i^*(\theta_i = R) = b$ et $\sigma_i^*(\theta_i = \bar{R}) = c$. Cela signifie que les joueurs suivent sans réfléchir l'offre quand ils en reçoivent une et préfèrent le fournisseur c lorsqu'ils n'en reçoivent pas. On peut calculer l'utilité espérée

de cette stratégie pour un joueur i :

$$\begin{aligned} EU_i^j(\sigma^* | \theta_i = R) &= P(\theta_j = R | \theta_i = R) u_i^j(b, b, R, R) \\ &\quad + P(\theta_j = \bar{R} | \theta_i = R) u_i^j(b, c, R, \bar{R}) \\ &= 0.8 \times 3 + 0.2 \times 0 = 2.4 \text{ et} \\ EU_i(\sigma^* | \theta_i = R) &= 2.4 + 2.4 = 4.8 \end{aligned}$$

De la même manière, nous obtenons

$$\begin{aligned} EU_i^j(\sigma^* | \theta_i = \bar{R}) &= 0.2 u_i^j(c, b, \bar{R}, R) + 0.8 u_i^j(c, c, \bar{R}, \bar{R}) \\ &= 1.6 \\ EU_i(\sigma^* | \theta_i = \bar{R}) &= 3.2 \end{aligned}$$

On peut vérifier facilement que σ^* est bien un équilibre de Nash, puisque modifier unilatéralement l'action choisie par un joueur n'augmentera pas son utilité locale. Les stratégies inconditionnelles où chaque joueur choisit c ou choisit b sont aussi des équilibres de Nash purs.

3.3 Complexité spatiale

Bien sur, tous les jeux bayésiens n'impliquent pas exclusivement des interactions locales. Mais c'est le cas pour certains et une représentation polymatricielle du jeu, quand elle est possible, est moins coûteuse que sa représentation SNF. Ceci est connu [25] pour les jeux à information complète et c'est aussi vrai pour les (H, B) -games.

En considérant le cas d'interactions par paires : un jeu bayésien polymatriciel contient $|E| = O(n^2)$ distributions de probabilité de taille $O(t^2)$, et $2 \cdot |E|$ fonctions d'utilités de taille $O(t^2 \cdot d^2)$ où t est le nombre maximum de types pour tout joueur et d est le nombre maximum d'actions. Plus généralement, notant ρ l'arité maximale de n'importe quelle hyperarête, nous avons :

Proposition 1 (Taille de représentation). *La complexité en espace de la représentation d'un (H, B) -game est $O(|E| \cdot \rho \cdot t^\rho \cdot d^\rho)$. Pour le cas particulier des jeux bayésiens polymatriciels, c'est $O(|E| \cdot t^2 \cdot d^2)$.*

Cette complexité en espace doit être comparée à la taille du jeu bayésien SNF équivalent à ce (H, B) -game. La forme normale comprend une distribution de probabilité de taille $O(t^n)$ et n fonctions d'utilité de taille $O(d^n \cdot t^n)$. Alors, quand ρ est borné par une constante (e.g. pour le cas polymatriciel, $\rho = 2$) la représentation graphique est exponentiellement moins coûteuse que la représentation SNF.

4 Des (H, B) -games aux jeux hypergraphiques à information complète

Au-delà de la proposition d'un nouveau cadre succinct, nous montrons que le gain de concision est obtenu sans augmentation de la complexité théorique du calcul d'un équilibre de Nash. Le principal résultat de cet article est de montrer que tout (H, B) -game peut être transformé en un jeu hypergraphique à information complète équivalent, en utilisant une transformation en *temps polynomial*. Donc, la complexité du calcul d'un équilibre de Nash dans ce cadre

n'est pas plus élevée que dans le cadre des jeux hypergraphiques à information complète. Notre travail s'inspire du résultat établi par Howson et Rosenthal [19], qui fait référence à une transformation originalement proposée par Selten [17]² pour les jeux bayésiens à 2 joueurs. Les auteurs proposent une transformation d'un jeu bayésien à 2 joueurs en un équivalent polymatriciel. Ce résultat fondamental est présenté de manière informelle par les auteurs. Nous le présenterons plus formellement ici.

Définition 10 (Transformation en jeu polymatriciel). *Soit $G = \langle N, A, \Theta, P, U \rangle$ un jeu bayésien à 2 joueurs. Nous définissons le jeu polymatriciel \tilde{G} comme :*

- $\tilde{N} = \{(i, \theta_i), i \in N = \{1, 2\}, \theta_i \in \Theta_i\}$;
- $\tilde{A} = \times_{(i, \theta_i)} \tilde{A}_{i, \theta_i}$ où $\tilde{A}_{(i, \theta_i)} = A_i, \forall i \in N, \theta_i \in \Theta_i$;
- $\tilde{E} = \{((i, \theta_i), (j, \theta_j)), i \neq j, \theta_i \in \Theta_i, \theta_j \in \Theta_j\}$;
- $\tilde{U} = \left\{ u_{(i, \theta_i)}^{(j, \theta_j)}, ((i, \theta_i), (j, \theta_j)) \in \tilde{E} \right\}$, où
$$u_{(i, \theta_i)}^{(j, \theta_j)}(\tilde{a}_{(i, \theta_i)}, \tilde{a}_{(j, \theta_j)}) = P(\theta_j | \theta_i) \cdot u_i^j(\tilde{a}_{(i, \theta_i)}, \tilde{a}_{(j, \theta_j)}, \theta)$$

On considère la bijection suivante entre les stratégies mixtes de G et celles de \tilde{G} :

Définition 11 (Transformation de stratégie). *Pour une stratégie mixte σ de G , $\tilde{\sigma}$ est la stratégie de \tilde{G} définie par $\tilde{\sigma}_{(i, \theta_i)}(a_i) = \sigma_i(a_i | \theta_i) \forall a_i \in A_i$.*

[19] montrent alors que :

Théorème 2 (Théorème de Howson et Rosenthal). *Si G est un jeu bayésien à 2 joueurs, alors le jeu polymatriciel \tilde{G} est tel que $EU_i(\sigma | \theta_i) = EU_{(i, \theta_i)}(\tilde{\sigma}), \forall i \in N, \theta_i \in \Theta_i, \forall \sigma$.*

En d'autres mots, tout jeu bayésien à 2 joueurs peut être transformé en un jeu polymatriciel équivalent (sans incertitude) et les équilibres de Nash de ces deux jeux sont en bijection. Notez aussi que, puisque N contient seulement deux joueurs, le graphe résultant (\tilde{N}, \tilde{E}) est biparti.

4.1 Extension du théorème de Howson et Rosenthal aux jeux bayésiens polymatriciels

Le théorème 2 concerne les jeux bayésiens à 2 joueurs sous forme normale. Nous pouvons l'étendre aux jeux bayésiens polymatriciels :

Théorème 3 (Transformation d'un jeu bayésien polymatriciel). *Pour tout jeu bayésien polymatriciel G , il existe un jeu polymatriciel \tilde{G} tel que :*

$$EU_i(\sigma | \theta_i) = EU_{(i, \theta_i)}(\tilde{\sigma}), \forall i \in N, \forall \theta_i \in \Theta_i, \forall \sigma$$

Ici, \tilde{G} est obtenu à partir des jeux polymatriciels locaux \tilde{G}^e construits à partir des jeux bimatriels locaux G^e , comme dans la définition 10. En effet, nous construisons $\tilde{N} = \cup_{e \in E} \tilde{N}^e$ et $\tilde{E} = \cup_{e \in E} \tilde{E}^e$ (\tilde{A} et \tilde{U} sont ensuite définis naturellement). $\tilde{\sigma}$ est défini selon la définition 11. Pour tous les jeux locaux de G , nous utilisons cette transformation

2. Il est intéressant de mentionner que Selten et Harsanyi sont co-tentateurs, avec Nash, du prix Nobel en science économique, pour leur "analyse innovante de l'équilibre en théorie des jeux compétitifs".

pour les transformer en jeux locaux de \tilde{G} . Donc, lorsque $n > 2$ le graphe du jeu obtenu n'est pas biparti.

En reprenant l'exemple de la section 3.2, le jeu bayésien polymatriciel à 4 joueurs devient un jeu polymatriciel à information complète à 8 joueurs. Le graphe de ce jeu est représenté dans la figure 3. Chaque arête $((i, \theta_i), (j, \theta_j))$ correspond à un jeu bimatriel, dont la table d'utilité sera différente en fonction de la paire (θ_i, θ_j) des joueurs. Ces tables sont représentées dans la figure 4.

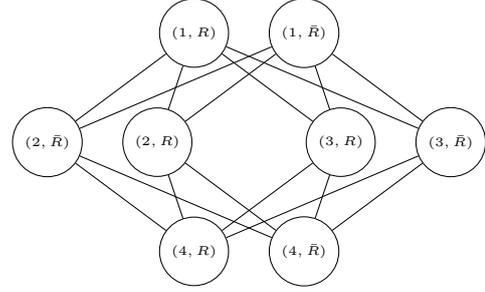


FIGURE 3 – Graphe du jeu polymatriciel à 8 joueurs correspondant au jeu bayésien polymatriciel à 4 joueurs de la figure 1.

		(j, R)		(j, \bar{R})	
		b	c	b	c
(i, R)	b	2.4, 2.4	0, 0	0.6, 0.2	0, 0
	c	0, 0	1.6, 1.6	0, 0	0.4, 0.4
(i, \bar{R})	b	0.2, 0.6	0, 0	0.8, 0.8	0, 0
	c	0, 0	0.4, 0.4	0, 0	1.6, 1.6

FIGURE 4 – Tables d'utilité du jeu polymatriciel à 8 joueurs.

4.2 Extension du théorème de Howson et Rosenthal aux jeux bayésiens hypergraphiques

Nous allons suivre, pour les (H, B) -games, la même approche que pour les jeux bayésiens polymatriciels. Nous allons d'abord définir la transformation :

Définition 12 (Transformation d'un (H, B) -game). *Soit $G = \langle N, A, \Theta, E, P, U \rangle$ un (H, B) -game. \tilde{G} est le jeu hypergraphique défini par :*

- $\tilde{N} = \{(i, \theta_i), i \in N, \theta_i \in \Theta_i\}$
- $\tilde{A} = \times_{(i, \theta_i)} \tilde{A}_{i, \theta_i}$ où $\tilde{A}_{(i, \theta_i)} = A_i$
- $\tilde{E} = \{\theta_e, e \in E, \theta_e \in \Theta_e\}$ où $\tilde{\theta}_e = \{(i, \theta_i), i \in e, \theta_i \in \Theta_e\}$
- $\tilde{U} = \{u_{(i, \theta_i)}^{\theta_e}, \theta_e \in \tilde{E}, (i, \theta_i) \in \tilde{\theta}_e\}$
où $u_{(i, \theta_i)}^{\theta_e}(a_e) = p(\theta_{e-\{i\}} | \theta_i) \cdot u_i^e(a_e, \theta_e)$

Cette définition généralise la définition 10. Considérons $\tilde{\sigma}$ la stratégie de \tilde{G} correspondant à σ (voir la définition 11). Nous pouvons montrer que :

Théorème 4 (Transformation d'un jeu bayésien sous forme normale). *Pour tout jeu bayésien G , il existe un jeu hypergraphique \tilde{G} tel que $EU_i(\sigma|\theta_i) = EU_{(i,\theta_i)}(\tilde{\sigma}), \forall i \in N, \theta_i \in \Theta_i, \forall \sigma$.*

Ce théorème généralise le théorème de Howson et Rosenthal aux jeux bayésiens à n joueurs. Bien sur, σ est un équilibre de Nash de G si et seulement si $\tilde{\sigma}$ est un équilibre de \tilde{G} . Notez que \tilde{G} dans le théorème 4 est obtenu à partir de la définition 12, en considérant une seule hyperarête e de taille n . Ainsi, le jeu hypergraphique obtenu a $\sum_{i \in N} |\Theta_i|$ noeuds et $|\Theta|$ hyperarêtes de taille n . Cette transformation n'est donc pas forcément très utile pour les jeux bayésiens sous forme normale. Cependant, grâce au théorème 4, nous obtenons :

Théorème 5 (Transformation d'un (H, B) -game). *Si G est un (H, B) -game et \tilde{G} un jeu équivalent comme défini dans la définition 12, alors $EU_i(\sigma|\theta_i) = EU_{(i,\theta_i)}(\tilde{\sigma}), \forall i \in N, \theta_i \in \Theta_i, \forall \sigma$.*

Il est donc possible de transformer n'importe quel (H, B) -game en un jeu hypergraphique à information complète. Dans la suite, nous montrons que la transformation en jeu hypergraphique à information complète est un résultat important, puisque qu'elle peut être effectuée en temps polynomial.

4.3 "The games world is (still) flat"

Puisque le problème de recherche d'un équilibre de Nash mixte dans les jeux polymatriciels ou hypergraphiques à information complète est PPAD-complet [11], la recherche d'un équilibre de Nash mixte dans un jeu bayésien polymatriciel/hypergraphique n'est pas plus facile (l'existence d'un équilibre est garantie, car tout (H, B) -game G est équivalent à un jeu bayésien sous forme normale $\langle N, A, \Theta, P, \{u_i\} \rangle$ où u_i est donné par la définition 9). Néanmoins, nous montrons que ce problème n'est en réalité pas plus dur. En effet, on peut montrer que :

Lemme 1 (Transformation en temps polynomial). *Pour tout (H, B) -game G , \tilde{G} peut être calculé en temps polynomial.*

La taille de \tilde{G} est comparable à celle de G . En notant t le nombre maximum de types, d le nombre maximum d'actions et ρ l'arité maximale des hyperarêtes, on constate que représenter G demande de stocker $|E|.t^\rho.(\rho.d^\rho + 1)$ nombres. D'un autre coté, \tilde{G} contient $|E|.t^\rho$ jeux SNF avec $\rho.t$ tables de d^ρ nombres, i.e. $|E|.t^{\rho+1}.\rho.d^\rho$ nombres. Nous dérivons du lemme 1 que :

Proposition 2 (Complexité). *Le problème de recherche d'un équilibre de Nash mixte dans un (H, B) -game est PPAD-complet. La PPAD-complétude est également vérifiée pour les jeux bayésiens polymatriciels.*

Cela signifie que résoudre un (H, B) -game n'est pas plus difficile que résoudre un jeu hypergraphique à information complète de taille similaire. En d'autres mots, dans les jeux

bayésiens comme dans les jeux classiques "the game world is flat", comme écrit par [11].

Maintenant, remarquons que les jeux bayésiens sous forme normale sont des (H, B) -games (dégénérés), avec une seule hyperarête. Donc la transformation en temps polynomial s'applique aussi à eux. Nous en déduisons :

Proposition 3 (Complexité des jeux bayésiens). *Calculer un équilibre de Nash mixte dans un jeu bayésien sous forme normale est un problème PPAD-complet.*

A notre connaissance ce résultat n'a pas encore été montré. Seule la NP-complétude du problème d'existence d'un EN pur dans un jeu bayésien a été montrée [10].

En ce qui concerne les équilibres de Nash purs dans les (H, B) -games, nous déduisons du théorème 5 :

Proposition 4 (Équivalence de l'équilibre Nash pur). *σ est un équilibre de Nash pur du (H, B) -game G ssi $\tilde{\sigma}$ est un équilibre de Nash pur de \tilde{G} .*

Ceci implique que décider si un (H, B) -game a un EN pur ou non appartient à NP. La NP-complétude du problème est déduite du résultat connu, que décider si un jeu bayésien à 2 joueurs admet un EN pur est NP-complet [10] :

Proposition 5 (Complexité d'un équilibre de Nash pur). *Déterminer s'il existe un équilibre de Nash pur dans un (H, B) -game est un problème NP-complet.*

Pour résumer, le gain en espace, potentiellement exponentiel, offert par les (H, B) -games par rapport aux jeux bayésiens sous forme normale n'augmente pas la complexité théorique de la résolution.

5 Conclusion

Dans cet article, nous proposons un nouveau cadre, les *jeux bayésiens hypergraphiques*, permettant de représenter des jeux à information incomplète de manière succincte. Notre principal résultat est de généraliser le théorème de Howson et Rosenthal et de montrer que n'importe quel (H, B) -game peut être transformé, en temps polynomial, en un jeu hypergraphique équivalent. De plus, la taille de description du jeu hypergraphique résultant est comparable à celle du (H, B) -game d'origine. Ceci est aussi vrai pour les jeux polymatriciels. Ce résultat a d'importantes conséquences, d'un point de vue algorithmique et de complexité théorique. Premièrement, nous avons prouvé que le calcul d'un équilibre de Nash mixte dans un (H, B) -game est PPAD-complet. Ceci fournit également la preuve que la recherche d'un EN dans un jeu bayésien sous forme normale est PPAD-complet.

Les *Bayesian Action-Graph Games (BAGG)* [20] sont une autre représentation succincte des jeux bayésiens, représentant leur fonction d'utilité comme dans le cadre des *Action-Graph Games (AGG)* [21]. Bien qu'elle puisse être exponentiellement plus compacte, la représentation de la distribution de probabilité sous forme d'un réseau bayésien peut être beaucoup plus coûteuse qu'un produit de distributions marginales. Le calcul d'utilité peut prendre un temps exponentiel (à cause de l'inférence). Le calcul d'un équilibre

de Nash mixte dans un BAGG est probablement plus complexe que pour les (H, B) -games pour différents types de distributions de probabilité possibles. Cependant il existe vraisemblablement des types de distributions de probabilité pour lesquels la transformation d'un BAGG en AGG peut être fait en temps polynomial. La comparaison des propriétés des cadres BAGG et (H, B) -games semble une perspective intéressante.

D'un point de vue algorithmique, puisque les (H, B) -games peuvent être transformés en temps polynomial en jeux hypergraphique, nous pouvons adapter aux (H, B) -games les algorithmes de résolution, exacte ou approchée, existant pour les jeux hypergraphiques, polymatriciels et graphiques [18, 24, 27, 28, 8]. Ceci est aussi intéressant pour les jeux Bayésiens classiques, même si leur transformation en jeu hypergraphique est en général coûteuse. En effet, la littérature sur l'algorithmique du calcul d'équilibres dans les jeux bayésiens est peu abondante [1, 7, 6].

Références

- [1] O. Armantier, J.P. Florens, and J.F. Richard. Approximation of bayesian nash equilibrium. *Journal of Applied Econometrics*, 23(7) :965–981, 2008.
- [2] R. Aumann and A. Brandenburger. Epistemic conditions for nash equilibrium. *Econometrica : Journal of the Econometric Society*, pages 1161–1180, 1995.
- [3] P. Battigalli and G. Bonanno. Recent results on belief, knowledge and the epistemic foundations of game theory. *Research in Economics*, 53(2) :149–225, 1999.
- [4] A. Brandenburger. Epistemic game theory : an overview. *The New Palgrave Dictionary of Economics, 2nd edition*, 2008.
- [5] A. Brandenburger and E. Dekel. Hierarchies of beliefs and common knowledge. *Journal of Economic Theory*, 59(1) :189–198, 1993.
- [6] G. Cai and P.R. Wurman. Monte carlo approximation in incomplete information sequential auction games. *Decision Support Systems*, 39(2) :153–168, 2005.
- [7] S. Ceppi, N. Gatti, and N. Basilico. Computing bayesian nash equilibria through support enumeration methods in bayesian two-player strategic-form games. In *IAT*, pages 541–548, 2009.
- [8] A.C. Chapman, A. Farinelli, E.M. de Cote, A. Rogers, and N.R. Jennings. A distributed algorithm for optimising over pure strategy nash equilibria. In *AAAI*, pages 749–755, 2010.
- [9] X. Chen and X. Deng. Settling the complexity of two-player nash-equilibrium. In *FOCS*, pages 261–272, 2006.
- [10] V. Conitzer and T. Sandholm. New complexity results about nash equilibria. *Games and Economic Behavior*, 63(2) :621–641, 2008.
- [11] C. Daskalakis, A. Fabrikant, and C.H. Papadimitriou. The game world is flat : The complexity of nash equilibria in succinct games. In *ICALP*, pages 513–524. Springer, 2006.
- [12] C. Daskalakis, P.W. Goldberg, and C.H. Papadimitriou. The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, 39(1) :195–259, 2009.
- [13] E. Dekel and M. Siniscalchi. Epistemic game theory. In *Handbook of Game Theory with Economic Applications*, volume 4, pages 619–702. Elsevier, 2015.
- [14] A. Deligkas, J. Fearnley, R. Savani, and P. Spirakis. Computing approximate nash equilibria in polymatrix games. In *WINE*, pages 58–71. Springer, 2014.
- [15] G. Gottlob, G. Greco, and F. Scarcello. Pure nash equilibria : Hard and easy games. *Journal of Artificial Intelligence Research*, 24 :357–406, 2005.
- [16] A. Grubshtein and A. Meisels. Finding a nash equilibrium by asynchronous backtracking. In *CP*, pages 925–940. Springer, 2012.
- [17] J.C. Harsanyi. Games with incomplete information played by “bayesian” players, i–iii part i. the basic model. *Management science*, 14(3) :159–182, 1967.
- [18] J.T. Howson. Equilibria of polymatrix games. *Management Science*, 18(5-part-1) :312–318, 1972.
- [19] J.T. Howson and R.W. Rosenthal. Bayesian equilibria of finite two-person games with incomplete information. *Management Science*, 21(3) :313–315, 1974.
- [20] A.X. Jiang and K. Leyton-Brown. Bayesian action-graph games. In *NIPS*, volume 23, pages 991–999, 2010.
- [21] A.X. Jiang, K. Leyton-Brown, and N.A.R. Bhat. Action-graph games. *Games and Economic Behavior*, 71(1) :141–173, 2011.
- [22] M. Kearns, M.L. Littman, and S. Singh. Graphical models for game theory. *UAI*, pages 253–260, 2001.
- [23] R.B. Myerson. Comments on “games with incomplete information played by ‘bayesian’ players, i–iii harsanyi’s games with incomplete information”. *Management Science*, 50(12_supplement) :1818–1824, 2004.
- [24] L.E. Ortiz and M. Kearns. Nash propagation for loopy graphical games. In *NIPS*, volume 15, pages 817–824, 2003.
- [25] C.H. Papadimitriou and T. Roughgarden. Computing correlated equilibria in multi-player games. *Journal of the ACM (JACM)*, 55(3) :1–29, 2008.
- [26] S. Simon and D. Wojtczak. Constrained pure nash equilibria in polymatrix games. In *AAAI*, pages 691–697, 2017.
- [27] D. Vickrey and D. Koller. Multi-agent algorithms for solving graphical games. In *AAAI*, pages 345–351, 2002.
- [28] M. Wahbi and K.N. Brown. A distributed asynchronous solver for nash equilibria in hypergraphical games. In *ECAI*, pages 1291–1299. IOS Press, 2016.
- [29] E.B. Yanovskaya. Equilibrium points in polymatrix games. *Litovskii Matematicheskii Sbornik*, 8 :381–384, 1968.

Jeux incomplets algébriques

H. Fargier¹, E. Martin-Dorel², P. Pomeret-Coquot²

¹ CNRS, IRIT, Université de Toulouse

² Université Toulouse III – Paul Sabatier, IRIT, Université de Toulouse

helene.fargier@irit.fr – erik.martin-dorel@irit.fr – pierre.pomeret@irit.fr

Résumé

Cet article propose une approche algébrique des jeux incomplets, ce qui permet de capturer des problèmes où des agents ont différents types de connaissances, des fonctions d'utilité complexes (bi-critères par exemple) et peuvent utiliser toute une gamme de règles de décision. Nous montrons que, lorsque la mesure de plausibilité est décomposable, de tels jeux peuvent être transformés en temps linéaire en jeux hypergraphiques (sans incertitude), généralisant ainsi de manière algébrique et à un nombre quelconque de joueurs le théorème de Howson et Rosenthal [10].

Mots-clés

Théorie des jeux - Jeu incomplet - Jeu hypergraphique - Approche algébrique

Abstract

This article proposes an algebraic approach to incomplete games allowing the modelling of problems where the agents have different types of knowledge, complex utility functions (bi-criteria for example) and can use a whole family of decision rules. We show that, when the plausibility measure is decomposable, such games can be transformed in linear time into hypergraphical games (without uncertainty), thus generalizing algebraically and to any number of players Howson and Rosenthal's Theorem [10].

Keywords

Game theory - Incomplete game - Hypergraphical game - Algebraic approach

1 Introduction

La théorie de la décision sous incertitude s'attache à modéliser le comportement d'agents ayant à choisir des décisions dont les conséquences ne sont pas connues avec certitude. Plusieurs règles de décision ont été établies et justifiées. La règle la plus connue est sans doute celle de l'utilité espérée [13] - l'agent a une connaissance probabiliste sur les conséquences de chaque décision et il cherche à maximiser l'utilité espérée des conséquences de l'acte choisi; d'autres règles ont été proposées *p.ex.* l'ordre de grandeur de l'utilité espérée [23] ou, pour les problèmes qualitatifs, les utilités possibilistes optimiste et pessimiste [7].

Nous nous intéressons ici à des problèmes de décision multi-agents issue de la théorie des jeux, où chacun prend une dé-

cision individuelle, dont la conséquence dépend à la fois des décisions individuelles et de l'état réel du monde. Chaque agent a un intérêt ou "utilité" (éventuellement différente de celle des autres agents) pour cette conséquence. Cette forme de problème de décision est l'objet du cadre des *jeux incomplets* (où les agents ignorent une partie du problème).¹

Deux types de jeux incomplets ont été décrits : les jeux bayésiens [9] (utilité espérée) et les jeux possibilistes [1] (utilité qualitative pessimiste). Dans cet article, nous proposons une définition algébrique généralisant ces deux cas particuliers et permettant d'utiliser d'autres règles de décision.

Par ailleurs, les jeux bayésiens et les jeux possibilistes à deux joueurs peuvent toujours être transformés en des jeux polymatriciels équivalents [10, 1]. Nous étendons ce résultat connu sous le nom de *théorème de Howson-Rosenthal* à notre cadre algébrique et pour un nombre *quelconque* d'agents. Cette transformation s'effectue en temps linéaire et produit un jeu hypergraphique de complexité spatiale inférieure ou égale à celle du jeu incomplet initial.

Pour les preuves, nous utilisons l'assistant de preuve COQ [2]. C'est un environnement logiciel permettant la spécification et la vérification de structures de données, d'algorithmes et de théorèmes. Son fonctionnement interne est basé sur la correspondance preuve-programme (ou correspondance de Curry-Howard). Nous utilisons COQ afin d'apporter une forte garantie de correction sur les preuves et les transformations, ainsi que pour faciliter leur réutilisation. La formalisation en COQ est disponible sur GitHub.²

2 Définitions préliminaires

Soient N un ensemble fini et $(X_i)_{i \in N}$ une famille d'ensembles indicés par N . Nous notons $X = \prod_{i \in N} X_i$ le produit cartésien des X_i et $x = (x_1, \dots, x_n) \in X$ un élément de X . Pour tout $M \subset N$, nous notons : $X^{(M)} = \prod_{i \in M} X_i$ et $x^{(M)} \in X^{(M)}$. Pour tout $i \in N$, nous notons : $X_{-i} = X^{(N \setminus \{i\})}$ et $x_{-i} \in X_{-i}$. Enfin, "." dénote la concaténation, ainsi $x = x_i.x_{-i}$.

1. Un glissement de sens semble avoir eu lieu. Ce qui est appelé aujourd'hui *jeu incomplet* a été défini en 1967 [9], puis probabilisé afin de d'introduire les *jeux bayésiens*. Harsanyi affirme qu'un jeu bayésien est un *jeu complet* - alors qu'aujourd'hui les jeux bayésiens sont vus comme des "jeux incomplets" - au sens de "jeux sous connaissance incomplète".

2. ppomco.github.io/coq-incomplete-games-rjcia2021

2.1 Décision mono-agent sous incertitude

Savage [20] introduit en 1954 les problèmes de décision sous incertitude en considérant qu'un acte (une décision) est une fonction qui associe à tout monde une conséquence, ce qui lui permet de caractériser la règle de l'utilité espérée. Plusieurs alternatives à l'utilité espérée ont été étudiées depuis, comme l'utilité dépendant du rang [18] ou les utilités qualitatives [7], etc. Nous présentons algébriquement les problèmes de décision mono-agent sous incertitude, en suivant les définitions de [6], qui recourent celles de [17, 16]. La généralisation algébrique porte sur la règle de décision, qui est encodée dans une *structure d'évaluation* contenant les opérateurs algébriques et leurs domaines.

Définition 1 (Structure d'évaluation) Une *structure d'évaluation* est un quintuplet $\mathcal{E} = (W, U, V, \oplus, \otimes)$ où :

- W est le domaine de plausibilité, muni de l'ordre \preceq_W ,³
- U est le domaine d'utilité, muni de l'ordre \preceq_U ,
- V est le domaine de valuation, muni de l'ordre \preceq_V , tel que $U \subseteq V$ et $\forall x, y \in U, x \preceq_U y \Rightarrow x \preceq_V y$,
- $\oplus : V \times V \rightarrow V$ est un opérateur associatif et commutatif, muni d'un élément neutre noté 0_V ,
- $\otimes : W \times U \rightarrow V$ est muni d'un élément absorbant noté 0_W et est distributif par rapport à \oplus .

Lorsque $U = V = W$ et $0_V = 0_W$, nous retrouvons tout simplement les semi-anneaux. Le domaine de plausibilité va capturer la croyance des agents, le domaine d'utilité leurs préférences et V est le domaine d'évaluation des actes.

Définition 2 (Problème de décision sous incertitude)

Un problème de décision sous incertitude est un quadruplet $\mathcal{D} = ((\Omega, X), \mathcal{E}, u, Pl)$ où :

- Ω est l'ensemble des états du monde,
- X est l'ensemble des conséquences,
- $\mathcal{E} = (W, U, V, \oplus, \otimes)$ est la structure d'évaluation,
- $u : X \rightarrow U$ est une fonction d'utilité,
- $Pl : 2^\Omega \rightarrow W$ une fonction d'ensemble monotone croissante par rapport à l'inclusion (une capacité).

Un problème de décision sous incertitude est décrit par trois composants : (i) les ensembles Ω et X qui décrivent la *situation* (pour reprendre les termes de [6]), les actes (ou décisions) étant représentés par des fonctions $a : \Omega \rightarrow X$; (ii) les préférences de l'agent, capturées par la fonction d'utilité et ses croyances capturées par la mesure Pl (dans cet article, nous nous restreignons aux mesures de plausibilité décomposables [22], c'est à dire telles que $\forall A, B$ disjoints, $Pl(A \cup B) = Pl(A) \oplus Pl(B)$: elles ont l'avantage de pouvoir être représentées succinctement, par des distributions $d : \Omega \rightarrow W$. C'est notamment le cas des mesures de probabilité et des mesures de possibilité); (iii) \mathcal{E} qui porte la règle de décision : l'agent maximise son utilité espérée généralisée (*GEU*, pour "Generalized Expected Utility").

3. Pour tout ensemble X muni d'un ordre \preceq_X, \sim_X désignera sa partie symétrique et \prec_X sa partie asymétrique : $(x \sim_X x') \iff (x \preceq_X x') \wedge (x' \preceq_X x)$ et $(x \prec_X x') \iff (x \preceq_X x') \wedge \neg(x' \preceq_X x)$.

Définition 3 (GEU [6]) Pour tout acte a dans un problème de décision \mathcal{D} fondé sur une structure d'évaluation \mathcal{E} :

$$GEU_{\mathcal{D}}(a) := \bigoplus_{\omega \in \Omega} d(\omega) \otimes u(a(\omega)).$$

En d'autres termes, l'agent évalue un acte en combinant, via \oplus et \otimes , ses préférences sur les conséquences de l'acte et ses croyances sur l'état du monde : l'acte a_1 est préféré à l'acte a_2 ssi $GEU(a_2) \preceq_V GEU(a_1)$.

Exemples La règle de l'utilité espérée est capturée par la structure $\mathcal{E}_{UE} = ([0, 1], \mathbb{R}, \mathbb{R}, +, \times)$, d étant une distribution de probabilité sur Ω . Dans ce cas en effet $GEU(a) = \sum_{\omega \in \Omega} d(\omega) \cdot u(a(\omega))$. Les règles possibiliste optimiste et pessimiste sont encodées dans les structures $\mathcal{E}_{OPT} = ([0, 1], [0, 1], [0, 1], \max, \min)$ et $\mathcal{E}_{PES} = ([0, 1], [0, 1], [0, 1], \min, \overline{\max})$.⁴ La règle de décision en ordre de grandeur de l'utilité espérée [23] est décrite par la structure $\mathcal{E}_{OOM} = (\mathbb{R}^o, \mathbb{R}^o, \mathbb{R}^o, \oplus, \otimes)$ où : $\mathbb{R}^o = \{(\sigma, m) \mid m \in \mathbb{Z}, \sigma \in \{+, -, 0\}\} \cup \{(0, \infty)\}$; $(\sigma, m) \oplus (\sigma', m')$ vaut (σ, m) si $m < m'$, (σ', m') si $m > m'$ et $(\sigma \boxplus \sigma', m)$ si $m = m'$ ⁵; enfin $(\sigma, m) \otimes (\sigma', m') = (\sigma \boxtimes \sigma', m + m')$.⁶ L'ordre $\preceq_{\mathbb{R}^o}$ est tel que $(\sigma, m) \preceq_{\mathbb{R}^o} (\sigma', m')$ ssi $(\sigma' = +) \wedge (m' < m)$, $(\sigma = -) \wedge (m < m')$, $(\sigma = -) \wedge (\sigma' = +) \wedge (m = m')$ ou $(\sigma = \sigma') \wedge (m = m')$.

2.2 Jeux simultanés à information complète

Considérons maintenant le cas multi-agents, tel que défini par la théorie des jeux : dans les jeux simultanés, chaque agent choisit individuellement une action, sans observer celles choisies par les autres ; à toute combinaison d'actions correspond une conséquence, et chaque agent a un intérêt (une utilité), éventuellement différent de celui des autres agents, pour cette conséquence. Lorsque chaque joueur connaît les actions possibles des autres joueurs et les utilités qui en résulteraient, le jeu est un *jeu simultané à information complète* (ou encore "jeu complet", "jeu classique") : l'état du monde est parfaitement connu de chacun - chaque agent maximise son utilité (on dit qu'il est "rationnel").

Définition 4 (Jeu simultané à information complète)

Un jeu est un quintuplet $G = (N, X, \chi, (A_i, v_i)_{i \in N})$ où :

- N est l'ensemble fini des n agents,
- A_i est l'ensemble des actions de l'agent i ,
- X est l'ensemble des conséquences,
- $\chi : A \rightarrow X$ décrit la conséquence d'une combinaison d'actions,
- $v_i : X \rightarrow U_i$ est la fonction d'utilité de l'agent i où U_i dénote le domaine d'utilité de i , muni de l'ordre \preceq_{U_i} .

Le jeu est dit sous forme normale lorsque les v_i sont décrites par des tables.

U_i est un domaine ordonné, et généralement $U_i = \mathbb{R}$ (en économie, on parle souvent de "payoff"). Un vecteur $a =$

4. $\forall w \in W, \forall x \in U, \overline{\max}(w, x) = \max(1 - w, x)$.

5. \boxplus est l'addition usuelle des signes, avec $+\boxplus+ = +, -\boxplus- = -, \sigma\boxplus\sigma' = 0$ sinon. \boxtimes est la multiplication des signes : $+\boxtimes+ = -\boxtimes- = +, +\boxtimes- = -\boxtimes+ = -$ et $\sigma\boxtimes 0 = 0\boxtimes\sigma = 0$.

6. L'addition $+$ est étendue sur $\mathbb{Z} \cup \{\infty\}$ par $m + \infty = \infty + m = \infty$.

$(a_1, \dots, a_n) \in A = \prod_{i \in N} A_i$ est appelé *profil de stratégies pures*, il correspond à un acte dont $\chi(a)$ est la conséquence dans l'unique état du monde.

Dans la suite, X sera rendu implicite en notant $G = (N, (A_i, u_i)_{i \in N})$, avec $u_i = v_i \circ \chi : A \rightarrow \mathbb{R}$.

Proposition 1 (Complexité spatiale) *La taille d'un jeu simultané G en forme normale est $S_G \in O(n|A|)$, i.e. si les n agents ont au plus α actions, $S_G \in O(n\alpha^n)$.*

En effet, les fonctions d'utilité sont représentées en extension par des tables listant leurs valeurs pour chacun des n agents et chacun des $|A| \leq \alpha^n$ profils de stratégies pures. Ainsi $S_G = \sum_{i \in N} |u_i| \leq n\alpha^n \in O(n\alpha^n)$.

Parmi les profils de stratégies couramment étudiés, nous nous intéressons aux équilibres de Nash [14].

Définition 5 (Équilibre de Nash) *Dans un jeu G , un profil de stratégies pures est un équilibre de Nash pur ssi aucun agent ne peut unilatéralement augmenter son utilité, i.e. ssi $\forall i \in N, \nexists a'_i \in A_i, u_i(a) \prec_{U_i} u_i(a'_i.a_{-i})$.*

Exemple 1 (Le repas) *Dans cette variante du jeu appelé "bataille des sexes", quatre collègues souhaitent manger ensemble, dans l'un des restaurants R_1 et R_2 , de préférence avec ceux qu'ils apprécient le plus - la satisfaction d'un agent est d'autant plus grande que beaucoup des collègues qu'il apprécie sont aussi présents. Les affinités de l'agent i sont décrites par une fonction $\text{aff}_i : N_{-i} \rightarrow \mathbb{R}^+$ ($\text{aff}_i(j)$ est l'affinité de i pour j). Sans pouvoir communiquer avec les autres, chacun va se rendre à l'un des deux restaurants. Cet exemple peut être modélisé par le jeu suivant :*

$$\begin{aligned} N &= \{1, 2, 3, 4\} \\ A_i &= \{R_1, R_2\} & \forall i \in N \\ u_i(a) &= \sum_{\substack{j \in N_{-i} \\ a_i = a_j}} \text{aff}_i(j) & \forall i \in N, \forall a \in A. \end{aligned}$$

Supposons $\text{aff}_i(j) = 1$ pour tout $i, j \in N$. Alors (R_1, R_1, R_1, R_1) et (R_2, R_2, R_2, R_2) sont des équilibres de Nash : aucun agent n'a intérêt à changer de stratégie.

Dans ce premier exemple, nous avons supposé que les utilités des agents sont cardinales - d'où un ordre complet sur les profils. Mais ce n'est pas toujours le cas. Dans certains travaux, les préférences sont complètes, mais seulement ordinales [24, 8]; dans d'autres, les U_i ne sont que partiellement ordonnés [21, 3, 4] - et cela suffit pour définir la notion d'équilibre de Nash, qui est une notion purement ordinale. La définition que nous avons donnée, à la suite de celle de [19], utilise des "utilités abstraites" et permet de capturer tous ces types de préférences, et en particulier des payoffs monétaires ou des préférences partielles.

Exemple 2 (Le repas - approche bi-critères) *Les quatre collègues préfèrent manger non seulement avec ceux qu'ils apprécient le plus, mais aussi avec ceux qu'ils n'ont pas vu*

depuis longtemps - ce jeu est bi-critères. Soit $\text{tps}_i(j) \in \mathbb{R}^+$ le temps écoulé depuis que i a vu j . Ce qui donne le jeu :

$$\begin{aligned} N &= \{1, 2, 3, 4\} \\ A_i &= \{R_1, R_2\} & \forall i \in N \\ V_i &= \mathbb{R}^+ \times \mathbb{R}^+ & \forall i \in N \\ u_i(a) &= \left(\sum_{\substack{j \in N_{-i} \\ a_i = a_j}} \text{aff}_i(j), \sum_{\substack{j \in N_{-i} \\ a_i = a_j}} \text{tps}_i(j) \right) & \begin{array}{l} \forall i \in N \\ \forall a \in A. \end{array} \end{aligned}$$

Pour tout agent $i \in N$, V_i est un ensemble partiellement ordonné. $\forall (x, y), (x', y') \in V_i, (x, y) \preceq_{V_i} (x', y') \iff x \leq x' \wedge y \leq y'$.

Si $\text{aff}_i(j) = 1$ pour tout $i, j \in N$, alors (R_1, R_1, R_1, R_1) et (R_2, R_2, R_2, R_2) sont des équilibres de Nash. De plus, si $\text{tps}_i(j) = 10$ lorsque $i, j \in \{1, 2\}$ ou $i, j \in \{3, 4\}$, et $\text{tps}_i(j) = 1$ sinon, alors (R_1, R_1, R_2, R_2) et (R_2, R_2, R_1, R_1) sont aussi des équilibres de Nash.

3 Jeux incomplets algébriques

Un jeu incomplet est un jeu où les agents ont une connaissance incomplète sur certains aspects du jeu - en fait sur l'état du monde $\omega \in \Omega$. La conséquence d'une action jointe a (et donc l'utilité des agents) dépend à la fois des actions choisies et de l'état du monde : il y a un jeu par état ω . Les agents ne connaissent pas exactement l'état réel du monde, et donc ne savent à quel jeu ils participent. Ils peuvent par contre avoir des connaissances a priori sur ω , représentées par une distribution de plausibilité. De plus, avant de choisir son action, chaque agent i peut recevoir un signal informatif $\tau_i(\omega)$. Il acquiert ainsi de la connaissance sur ω et sur les signaux corrélés reçus par les autres agents : il conditionne sa croyance a priori et obtient une connaissance personnelle a posteriori. La distribution de plausibilité a priori peut être commune, comme dans les jeux Bayésiens, mais il est aussi possible de considérer que les agents ont des croyances différentes (éventuellement incohérentes entre elles); nous supposons alors simplement que chaque agent connaît toutes les distributions de tous les agents.

Harsanyi propose de décrire les jeux à information incomplète non pas sur les mondes, mais sur les états de croyances possibles - typiquement sur les combinaisons de signaux possibles et plus largement sur les informations d'un jeu auxquels les joueurs ont accès (en ce qui les concerne ou en ce qui concerne les autres). Il appelle ces états de croyance des "types". Ainsi, un jeu incomplet décrit un jeu simultané pour chaque combinaison de types possible.

3.1 Définition

Deux formes de jeux incomplets ont été proposées : les jeux bayésiens [9], fondés sur la règle de l'utilité espérée, et les jeux possibilistes [1], fondés sur celle de l'utilité pessimiste. Dans cet article, nous les généralisons en en proposant une formulation algébrique.

Définition 6 (Jeu incomplet algébrique) *Un jeu incomplet algébrique est un tuple $G = (N, (A_i, \Theta_i, \mathcal{E}_i, d_i, u_i)_{i \in N})$ où :*

- N est l'ensemble fini des n agents,
- A_i est l'ensemble des actions de l'agent i ,
- Θ_i est l'ensemble des types de l'agent i ,
- $\mathcal{E}_i = (W_i, U_i, V_i, \oplus_i, \otimes_i)$ est la structure d'évaluation encodant sa règle de décision,
- $d_i : \Theta_i \rightarrow \Theta_{-i} \rightarrow W_i$ où $d_i(\cdot|\theta_i)$ est la distribution de plausibilité a posteriori de l'agent i , sachant qu'il est de type θ_i .
- $u_i : A \times \Theta \rightarrow U_i$ est sa fonction d'utilité.

Lorsque les fonctions u_i et les distributions d_i sont données sous formes de tables, le jeu est dit sous forme normale.

Dans un jeu incomplet, les agents choisissent une action en fonction du type qu'ils reçoivent : une stratégie pure de l'agent i est une fonction $\sigma_i : \Theta_i \rightarrow A_i$. Appelons Σ_i l'espace des stratégies pures de l'agent i et Σ l'ensemble des stratégies jointes. Un profil de stratégies pures sous incertitude spécifie alors l'action choisie pour chaque type de chaque agent : c'est un vecteur $\sigma = (\sigma_1, \dots, \sigma_n) \in \Sigma$. Pour toute combinaison de types $\theta \in \Theta$, nous notons $\sigma^\theta = (\sigma_1(\theta_1), \dots, \sigma_n(\theta_n)) \in A$ le profil de stratégies pures qui sera effectivement joué si ces types sont reçus. L'utilité d'un tel profil dépend de l'état du monde, et donc de la combinaison θ des types reçus. Si l'agent i est de type θ_i , son utilité est donnée par $GEU_{(i,\theta_i)} : \Sigma \rightarrow V_i$:

$$GEU_{(i,\theta_i)}(\sigma) = \bigoplus_{\theta_{-i} \in \Theta_{-i}} d_i(\theta_{-i}|\theta_i) \otimes_i u_i(\sigma^{\theta_i, \theta_{-i}}, \theta_i, \theta_{-i}).$$

Proposition 2 (Complexité spatiale) La taille d'un jeu incomplet sous forme normale G est $S_G \in O(n \cdot |A| \cdot |\Theta|)$, i.e. si les n agents ont au plus α actions et peuvent recevoir au plus β types, la taille est en $O(n(\alpha\beta)^n)$.

En effet, pour chacune des $|\Theta|$ combinaisons de types, il faut donner un jeu simultané de taille $n|A|$.

Un profil de stratégies est un équilibre de Nash sous incertitude pur ssi aucun agent ne peut augmenter unilatéralement son utilité en modifiant sa stratégie. Formellement :

Définition 7 (Équilibre de Nash pour un jeu incomplet) $\sigma \in \Sigma$ est un équilibre de Nash de G ssi $\forall i \in N, \forall \theta_i \in \Theta_i, \nexists \sigma'_i \in \Sigma_i, GEU_{(i,\theta_i)}(\sigma) \prec_{V_i} GEU_{(i,t)}(\sigma'_i, \sigma_{-i})$.

3.2 Cas particuliers

Soit un jeu algébrique $G = (N, (A_i, \Theta_i, \mathcal{E}_i, d_i, u_i)_{i \in N})$.

Jeu classique. Lorsqu'un seul état du monde est possible, chaque agent ne possède qu'un seul type ($\Theta = \{\theta\}$) : G décrit un jeu classique.

Jeu bayésien. Tous les agents utilisent la règle de l'utilité espérée en conditionnant une distribution de probabilité p commune : pour tout i , $\mathcal{E}_i = \mathcal{E}_{UE} = ([0, 1], \mathbb{R}, \mathbb{R}, +, \times)$ et pour tout θ , $d_i(\theta_{-i}|\theta_i) = p(\theta|\theta_i)$.

Jeu possibiliste pessimiste. Tous les agents utilisent la règle de l'utilité pessimiste en conditionnant une distribution de possibilité π commune : pour tout i , $\mathcal{E}_i = \mathcal{E}_{PES} = ([0, 1], [0, 1], [0, 1], \min, \overline{\max})$ et pour tout θ , $d_i(\theta_{-i}|\theta_i) = \pi(\theta|\theta_i)$.

3.3 Exemples

Nous illustrons l'utilisation de différentes règles de décisions sur l'exemple suivant :

Exemple 3 (Le repas - jeu incomplet) Trois agents souhaitent manger ensemble ce midi. Chacun des deux restaurants R_k ($k \in \{1, 2\}$) est ouvert (O_k) ou fermé (F_k). Habituellement, au moins un restaurant est ouvert. Le matin, l'agent 1 emprunte le chemin C_1 qui passe devant le restaurant R_1 et sait si R_1 est ouvert. De même, l'agent 2 emprunte le chemin C_2 et sait si R_2 est ouvert. D'habitude, l'agent 3 emprunte le chemin C_1 et rencontre l'agent 1 : ils savent tous deux si R_1 est ouvert. Parfois, l'agent 3 emprunte le chemin C_2 et rencontre alors l'agent 2.

Modélisons d'abord la situation. Par souci de clarté, nous expliciterons les états du monde sous-jacents.

- Les agents : $N = \{1, 2, 3\}$
- Leurs actions : $A_i = \{R_1, R_2\} \quad \forall i \in N$
- $\Omega = \{O_1, F_1\} \times \{O_2, F_2\} \times \{C_1, C_2\}$

Un état du monde est un triplet $(\omega_1, \omega_2, \omega_3)$ où :

- $\omega_1 \in \{O_1, F_1\}$ indique si R_1 est ouvert,
- $\omega_2 \in \{O_2, F_2\}$ indique si R_2 est ouvert,
- $\omega_3 \in \{C_1, C_2\}$ indique le chemin de l'agent 3.

- Les types : $\Theta = \Theta_1 \times \Theta_2 \times \Theta_3$ avec :

- $\Theta_1 = \{O_1, F_1\} \times \{C_1, C_2\}$
- $\Theta_2 = \{O_2, F_2\} \times \{C_1, C_2\}$
- $\Theta_3 = \{O_1, F_1, O_2, F_2\}$

Dans l'état du monde $\omega = (\omega_1, \omega_2, \omega_3) \in \Omega$, l'agent $i \in N$ est de type $\theta_i = \tau_i(\omega)$, où :

- $\tau_1((\omega_1, \omega_2, \omega_3)) = (\omega_1, \omega_3)$
- $\tau_2((\omega_1, \omega_2, \omega_3)) = (\omega_2, \omega_3)$
- $\tau_3((\omega_1, \omega_2, \omega_3)) = \begin{cases} \omega_1 & \text{si } \omega_3 = C_1 \\ \omega_2 & \text{si } \omega_3 = C_2. \end{cases}$

Nous proposons plusieurs variations autour de cet exemple afin d'illustrer l'utilisation de différentes règles de décision. Pour cela, notons $nb_i(a)$ le nombre d'agents ayant choisi le même restaurant que i ($nb_i(a) = \sum_{j \in N_{-i}, a_j = a_i} 1$) et $ouv_i : A \times \Theta \rightarrow \{0, 1\}$ la fonction telle $ouv_i(a, \theta) = 1$ ssi le restaurant a_i est ouvert.⁷

Jeu bayésien Les agents utilisent la règle de l'utilité espérée en conditionnant la loi de probabilité commune $p^* : \Theta \rightarrow [0, 1]$ (table 2). La modélisation est la suivante :

$$\mathcal{E}_i = \mathcal{E}_{UE} = ([0, 1], \mathbb{R}, \mathbb{R}, +, \times)$$

$$d_i(\theta_{-i}|\theta_i) = p^*(\theta|\theta_i)$$

$$u_i(a, \theta) = \begin{cases} \sum_{\substack{j \in N_{-i} \\ a_j = a_i}} aff_i(j) & \text{si } ouv_i(a, \theta) = 1 \\ -1 & \text{sinon.} \end{cases}$$

Les utilités sont cardinales. L'ordre est complet. Le jeu $G = (N, (A_i, \Theta_i, \mathcal{E}_i, d_i, u_i)_{i \in N})$ est un jeu bayésien.

⁷ Pour tout agent $i \in N$, l'ouverture du restaurant a_i est donnée par $\theta : a_i = R_1$ est ouvert ssi $\theta_1 \in \{(O_1, C_1), (O_1, C_2)\}$ et $a_i = R_2$ est ouvert ssi $\theta_2 \in \{(O_2, C_1), (O_2, C_2)\}$.

$\omega \in \Omega$	$p(\omega)$	$\pi(\omega)$	$p_{OOM}(\omega)$
(O_1, O_2, C_1)	0.2	1	(+, 7)
(O_1, O_2, C_2)	0.1	0.8	(+, 10)
(O_1, F_2, C_1)	0.2	1	(+, 7)
(O_1, F_2, C_2)	0.1	0.8	(+, 10)
(F_1, O_2, C_1)	0.2	1	(+, 7)
(F_1, O_2, C_2)	0.1	0.8	(+, 10)
(F_1, F_2, C_1)	0.075	0.4	(+, 12)
(F_1, F_2, C_2)	0.025	0.2	(+, 16)

TABLE 1 – Distributions de probabilité, d'ordre de grandeur et de possibilité sur les états du monde.

$\theta \in \Theta$	$p^*(\theta)$	$\pi^*(\theta)$	$p_{OOM}^*(\omega)$
$((O_1, C_1), (O_2, C_1), O_1)$	0.2	1	(+, 7)
$((O_1, C_2), (O_2, C_2), O_2)$	0.1	0.8	(+, 10)
$((O_1, C_1), (F_2, C_1), O_1)$	0.2	1	(+, 7)
$((O_1, C_2), (F_2, C_2), F_2)$	0.1	0.8	(+, 10)
$((F_1, C_1), (O_2, C_1), F_1)$	0.2	1	(+, 7)
$((F_1, C_2), (O_2, C_2), O_2)$	0.1	0.8	(+, 10)
$((F_1, C_1), (F_2, C_1), F_1)$	0.075	0.4	(+, 12)
$((F_1, C_2), (F_2, C_2), F_2)$	0.025	0.2	(+, 16)
Combinaisons impossibles	0	0	(0, ∞)

TABLE 2 – Distributions de probabilité, d'ordre de grandeur et de possibilité sur les types joints, déduites de la table 1.

Jeu en ordre de grandeur. Les agents utilisent la règle de l'utilité espérée en ordre de grandeur en conditionnant la loi de $p_{OOM}^* : \Theta \rightarrow \mathbb{R}^o$ (table 2). Dans ce cas :

$$\begin{aligned} \mathcal{E}_i &= \mathcal{E}_{OOM} = (\mathbb{R}^o, \mathbb{R}^o, \mathbb{R}^o, \oplus, \otimes) \\ d_i(\theta_{-i} | \theta_i) &= p_{OOM}^*(\theta | \theta_i) \\ u_i(a, \theta) &= \begin{cases} (+, -nb_i(a)) & \text{si } \text{ouv}_i(a, \theta) = 1 \\ (-, 0) & \text{sinon.} \end{cases} \end{aligned}$$

Le jeu $G = (N, (A_i, \Theta_i, \mathcal{E}_i, d_i, u_i)_{i \in N})$ est un jeu en ordre de grandeur de l'utilité espérée.

Jeu possibiliste pessimiste. Les agents utilisent la règle de l'utilité pessimiste en conditionnant la loi de possibilité commune $\pi^* : \Theta \rightarrow [0, 1]$ (table 2). Dans ce cas :

$$\begin{aligned} \mathcal{E}_i &= \mathcal{E}_{PES} = ([0, 1], [0, 1], [0, 1], \overline{\max}, \min) \\ d_i(\theta_{-i} | \theta_i) &= \pi^*(\theta | \theta_i) \\ u_i(a, \theta) &= \begin{cases} 1 - \max_{\substack{j \in N \\ a_j \neq a_i}} \text{aff}_i(j) & \text{si } \text{ouv}_i(a, \theta) = 1 \\ 0 & \text{sinon.} \end{cases} \end{aligned}$$

$G = (N, (A_i, \Theta_i, \mathcal{E}_i, d_i, u_i)_{i \in N})$ est un jeu possibiliste pessimiste.

Jeu bayésien Pareto bi-critères. Comme dans les jeux bayésiens la connaissance est capturée par une loi de probabilité commune p^* (table 2), mais ici les agents décident selon deux critères : l'affinité et la présence de collègues que l'on n'a pas vus depuis longtemps (cf. exemple 2) :

$$u_i(a, \theta) = \begin{cases} \left(\sum_{\substack{j \in N \\ a_i = a_j}} \text{aff}_i(j), \sum_{\substack{j \in N \\ a_i = a_j}} \text{tps}_i(j) \right) & \text{si } \text{ouv}_i(a, \theta) = 1 \\ (0, 0) & \text{sinon.} \end{cases}$$

L'ordre sur $U_i = \mathbb{R}^2$ est partiel, un agent ne peut pas dire s'il préfère les collègues qu'il n'a pas vu depuis longtemps ou ceux pour lesquels il a beaucoup d'affinité. Plus formellement, $(a, b) \preceq_{U_i} (a', b') \iff (a \leq a') \wedge (b \leq b')$. Nous utiliserons alors une utilité espérée par critère :

$$\begin{aligned} w \otimes_i (a, b) &= (w \times a, w \times b) \\ (a, b) \oplus_i (a', b') &= (a + a', b + b') \\ \mathcal{E}_i &= ([0, 1], \mathbb{R}^2, \mathbb{R}^2, \oplus_i, \otimes_i) \\ d_i(\theta_{-i} | \theta_i) &= p^*(\theta | \theta_i). \end{aligned}$$

Agents aux comportements différents. Chaque agent peut avoir une distribution différente et une règle de décision différente. Le cadre des jeux algébriques permet *p.ex.* de modéliser un problème où l'un utilise une utilité espérée, l'autre une utilité pessimiste et le troisième une utilité bi-critères en ordre de grandeur.

4 Une généralisation du théorème de Howson-Rosenthal

Le *théorème de Howson-Rosenthal* [10] permet de construire en temps et espace polynomiaux, un jeu polymatriciel équivalent à un jeu bayésien à deux agents. Un théorème similaire a été proposé pour les jeux possibilistes à deux agents [1]. Nous généralisons ce théorème dans notre cadre algébrique et l'étendons aux jeux à n agents ($n \geq 2$). Commençons par présenter et généraliser algébriquement le formalisme des jeux hypergraphiques

4.1 Jeux hypergraphiques algébriques

Certains jeux peuvent être représentés succinctement grâce au cadre des jeux hypergraphiques [11, 15], où un jeu global est décrit par plusieurs jeux locaux en forme normale standard, l'utilité d'un joueur étant la somme des utilités qu'il récolte dans les jeux locaux. C'est typiquement le cas du jeu de l'exemple 1. Sa représentation sous forme normale demande n tables de taille α^n . Il pourrait facilement être exprimé sous la forme d'un jeu dit "polymatriciel" [11], c'est à dire d'un jeu hypergraphique dont les arrêtes ne lient que deux joueurs : chacun joue la même action dans tous les jeux locaux auxquels il participe. En considérant des jeux locaux avec plus deux joueurs, on obtient un hypergraphe où les sommets correspondent aux agents et les hyper-arêtes correspondent aux jeux locaux.

Ici encore, nous adoptons une approche algébrique. Définissons d'abord une structure d'évaluation réduite (à une dimension, celle des préférences), dans laquelle sont effectués les calculs d'utilité globale à partir des utilités locales.

Définition 8 (Structure d'évaluation réduite) Une structure d'évaluation réduite est un couple $\mathcal{V} = (V, \boxplus)$ où :

- V est l'ensemble de valuation, muni de l'ordre \preceq_V
- $\boxplus : V \times V \rightarrow V$ est un opérateur commutatif muni d'un élément neutre 0_V .

Définition 9 (Jeu hypergraphique) Un jeu hypergraphique $G = (N, E, (A_i, \mathcal{V}_i)_{i \in N}, (u_i^e)_{e \in E; i \in e})$ où :

- N est l'ensemble des sommets (agents),
- $E \subseteq 2^N$ est l'ensemble des hyper-arêtes (jeux locaux),
- A_i est l'ensemble des actions de l'agent i ,
- $\mathcal{V}_i = (V_i, \boxplus_i)$ est la structure d'évaluation réduite de l'agent i ,
- $u_i^e : A^{(e)} \rightarrow V_i$ est la fonction d'utilité locale de l'agent $i \in N$ pour le jeu local $e \in E$.

L'utilité globale de i est définie par : $u_i(a) = \boxplus_{\substack{e \in E \\ i \in e}} u_i^e(a^{(e)})$.

Un jeu hypergraphique où toutes les arêtes sont de taille 2 est appelé jeu polymatriciel.

Proposition 3 (Complexité spatiale) La taille d'un jeu hypergraphique G où les hyper-arêtes ont au plus une taille k ($k = \max_{e \in E} |e|$) et où les agents ont au plus α actions ($\alpha = \max_{i \in N} |A_i|$) est $S_G \in O(|E|k\alpha^k)$.

En effet, pour décrire les $|E|$ jeux locaux sous forme normale, il faut au plus k tables de taille α^k . G est une représentation plus succincte du jeu $G' = (N, (A_i, u_i)_{i \in N})$.

Exemple 4 (Le repas (complet) hypergraphique)

Représentons de manière succincte le jeu de l'exemple 2 :

$$\begin{aligned} N &= \{1, 2, 3, 4\} \\ E &= \{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}, \{3, 4\}\} \\ A_i &= \{R_1, R_2\} \quad \forall i \in N \\ V_i &= \mathbb{R}^2 \text{ et } 0_{V_i} = (0, 0) \in V_i \quad \forall i \in N \end{aligned}$$

\boxplus_i est tel que $\forall i \in N, \forall (x, y), (x', y') \in V_i$,
 $(x, y) \boxplus_i (x', y') = (x + x', y + y')$

et u_i^e est telle que $\forall e = \{i, j\} \in E, \forall a \in A$,

$$u_i^e(a^{(e)}) = \begin{cases} (\text{aff}_i(j), \text{tps}_i(j)) & \text{si } a_i^{(e)} = a_j^{(e)} \\ (0, 0) & \text{sinon.} \end{cases}$$

Ce jeu polymatriciel est illustré par la figure 1. Les sommets ronds correspondent à un agent $i \in N$ et les sommets carrés à un jeu local, i.e. une hyper-arête $e \in E$.

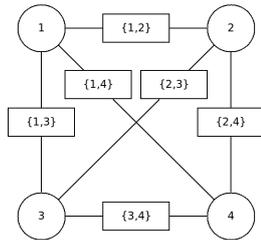


FIGURE 1 – Jeu hypergraphique polymatriciel (exemple 4)

Il est facile de vérifier que :

$$\begin{aligned} u_i(a) &= \boxplus_{\substack{\{i,j\} \in E \\ i \in e}} \begin{cases} (\text{aff}_i(j), \text{tps}_i(j)) & \text{si } a_i^{(e)} = a_j^{(e)} \\ (0, 0) & \text{sinon} \end{cases} \\ &= \left(\sum_{\substack{j \in N-i \\ a_i = a_j}} \text{aff}_i(j), \sum_{\substack{j \in N-i \\ a_i = a_j}} \text{tps}_i(j) \right). \end{aligned}$$

Les utilités étant identiques à celles de l'exemple 2, les équilibres de Nash le sont aussi.

4.2 Résultat principal

Partant d'un jeu incomplet G quelconque, nous allons construire un jeu hypergraphique \tilde{G} que nous appellerons *modèle de Selten* de G . Nous montrerons ensuite que cette transformation préserve les utilités, c'est à dire que l'on peut trouver une correspondance entre les agents, leurs actions et les profils de stratégies pures telles que les utilités dans \tilde{G} soit égales à celles dans G . Il est donc possible d'étudier G en étudiant \tilde{G} et réciproquement.

Définition 10 (Préservation des utilités) La transformation d'un jeu incomplet $G = (N, (A_i, \Theta_i, \mathcal{E}_i, d_i, u_i)_{i \in N})$ en un jeu classique $G' = (N', (A'_k, u'_k)_{k \in N'})$ préserve les utilités ssi :

$$\begin{aligned} \exists f_N : \{(i, \theta_i) \mid i \in N, \theta_i \in \Theta_i\} &\rightarrow N' \text{ bijective,} \\ \exists f_\Sigma : \Sigma &\rightarrow A' \text{ bijective,} \\ \forall i \in N, \forall t \in \Theta_i, \forall \sigma \in \Sigma, & \\ GEU_{(i, \theta_i)}(\sigma) &= u'_{f_N(i, \theta_i)}(f_\Sigma(\sigma)), \end{aligned}$$

i.e. la GEU du profil σ pour l'agent i de type θ_i est égale à l'utilité du profil $f_\Sigma(\sigma)$ pour l'agent $f_N(i, \theta_i)$.

Définition 11 (Modèle de Selten) Soit le jeu incomplet $G = (N, (A_i, \Theta_i, \mathcal{E}_i, d_i, u_i)_{i \in N})$. Soit :

- $\tilde{N} = \{(i, \theta_i) \mid i \in N, \theta_i \in \Theta_i\}$ l'ensemble des paires dépendantes (agent, type),
- $\Theta^* = \{\theta \in \Theta \mid \exists i \in N, \exists \theta_i \in \Theta_i, d_i(\theta_{-i} \mid \theta_i) \succ_{W_i} 0_{W_i}\}$ l'ensemble des combinaisons de types qui sont possibles pour au moins un agent,
- $\text{ens} : \Theta^* \rightarrow \tilde{N}$ telle que $\text{ens}(\theta) = \{(i, \theta_i) \mid i \in N\}$,
- $\tilde{E} = \{\text{ens}(\theta) \mid \theta \in \Theta^*\}$ l'ensemble des ensembles de paires dépendantes correspondants aux combinaisons de types possibles,
- $\tilde{A}_{(i, \theta_i)} = A_i$ les actions de l'agent i ,
- $\boxplus_{(i, \theta_i)} = \boxplus_i$ l'opérateur de l'agent i ,
- $\tilde{u}_{(i, \theta_i)}^{\text{ens}(\theta)}(\tilde{a}^{\text{ens}(\theta)}) = d_i(\theta_{-i} \mid \theta_i) \otimes_i u_i(\sigma^\theta, \theta)$, $\forall \theta \in \Theta^*$.

Le modèle de Selten \tilde{G} correspondant à G est le jeu hypergraphique :

$$\tilde{G} = \left(\tilde{N}, \tilde{E}, (\tilde{A}_{(i, \theta_i)}, \boxplus_{(i, \theta_i)})_{(i, \theta_i) \in \tilde{N}}, (\tilde{u}_{(i, \theta_i)}^e)_{e \in \tilde{E}, (i, \theta_i) \in e} \right)$$

Exemple 5 (Le repas - Modèle de Selten) Nous avons modélisé le problème de l'exemple 3 par le jeu incomplet G . Nous construisons son modèle de Selten \tilde{G} avec (pour l'approche bayésienne) :

$$\tilde{N} = \left\{ \begin{array}{l} (1, (O_1, C_1)), (1, (O_1, C_2)), \\ (1, (F_1, C_1)), (1, (F_1, C_2)), \\ (2, (O_2, C_1)), (2, (O_2, C_2)), \\ (2, (F_2, C_1)), (2, (F_2, C_2)), \\ (3, O_1), (3, F_1), (3, O_2), (3, F_2) \end{array} \right\}$$

$$\begin{aligned} \tilde{E} &= \{\text{ens}(\theta) \mid \theta \in \Theta^*\} ; \tilde{A}_{(i, t)} = \{R_1, R_2\} ; \boxplus_i = + \\ \tilde{u}_{(i, \theta_i)}^{\text{ens}(\theta)}(\tilde{a}^{\text{ens}(\theta)}) &= p^*(\theta \mid \theta_i) \times u_i(\sigma^\theta, \theta). \end{aligned}$$

Les ensembles $ens(\theta)$ sont construits pour toute combinaison de types $\theta \in \Theta^*$ possible (table 2).

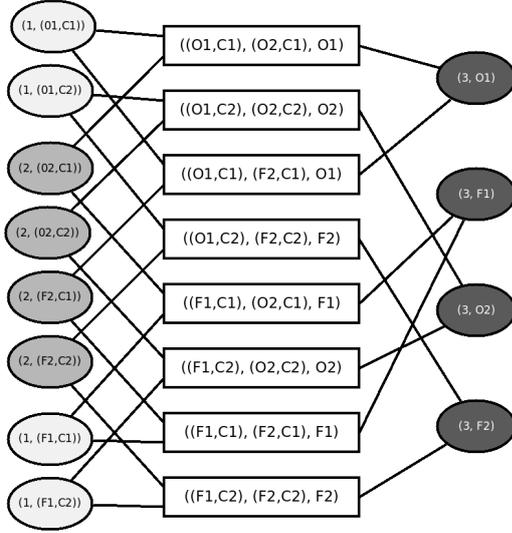


FIGURE 2 – Jeu hypergraphique (exemple 5) : un sommet rond désigne un couple $(i, \theta_i) \in \tilde{N}$ (une couleur par agent i), un sommet carré désigne une hyper-arête $ens(\theta)$ correspondant à une combinaison possible de types $\theta \in \Theta^*$.

\tilde{G} a 8 jeux locaux à 3 agents (figure 2); les fonctions d'utilité locales peuvent être listées dans 8×3 tables de taille 2^3 (i.e. 192 valeurs). La transformation effectue une opération $\otimes_i = \times$ en $O(1)$ pour chacune des 192 valeurs d'utilité; elle est donc linéaire. Le jeu dont \tilde{G} est la représentation succincte a 12 fonctions d'utilité, qui seraient listées dans 12 tables de taille $|\tilde{A}| = 2^{12}$ (i.e. 49152 valeurs).

Les utilités sont préservées en utilisant la fonction identité $f_N : \tilde{N} \rightarrow \tilde{N}$ et la fonction $f_\Sigma : \Sigma \rightarrow \tilde{A}$ telle que :

$$f_\Sigma(\sigma) = (\sigma_1(\theta_{1,1}), \dots, \sigma_n(\theta_{n,|\Theta_n|})) \in \tilde{A}$$

$$f_\Sigma^{-1}(\tilde{a}) = ((fun \theta_1 \rightarrow \tilde{a}_{(1,\theta_1)}), \dots, (fun \theta_n \rightarrow \tilde{a}_{(n,\theta_n)})) \in \Sigma$$

C'est-à-dire, $\forall \sigma \in \Sigma, \forall \tilde{a} \in \tilde{A}, \forall i \in N, \forall \theta_i \in \Theta_i$, la bijection $\tilde{a} = f_\Sigma(\sigma)$ met en correspondance $\sigma_i(\theta_i)$ et $\tilde{a}_{(i,\theta_i)}$. En d'autres termes la transformation d'un jeu incomplet algébrique en son modèle de Selten préserve les utilités - ce qui généralise le théorème de Howson-Rosenthal.

Théorème 1 (Howson-Rosenthal généralisé) On a : $\forall \sigma \in \Sigma, \forall i \in N, \forall \theta_i \in \Theta_i, GEU_{(i,\theta_i)}(\sigma) = \tilde{u}_{(i,\theta_i)}(f_\Sigma(\sigma))$.

La preuve est donnée en COQ sur GitHub.² Il s'agit principalement d'un jeu de réécriture des définitions, qui est possible d'une part car les agents, types et profils de stratégies sont en bijection (par f_N et f_Σ), et d'autre part car l'existence d'un élément O_{W_i} absorbant pour \otimes_i et neutre pour \oplus_i permet d'effectuer la sommation abstraite sur Θ^* (les combinaisons de types possibles pour au moins un agent) de la même façon que sur Θ (toutes les combinaisons).

Corollaire 1 (Correspondance des équilibres de Nash) Un profil σ est un équilibre du jeu G ssi son image $\tilde{a} = f_\Sigma(\sigma)$ est un équilibre de Nash du modèle de Selten \tilde{G} .

Ce résultat est direct par conservation des utilités : dans G , aucun agent i n'a intérêt à changer sa stratégie σ_i (quel que soit le type θ_i reçu) ssi dans \tilde{G} , aucun agent $f_N(i, \theta_i)$ n'a intérêt à changer sa stratégie $\tilde{a}_{(i,\theta_i)}$.

Proposition 4 (Complexité spatiale) Le modèle de Selten \tilde{G} a au plus la même taille que le jeu incomplet initial.

En effet, d'après la proposition 3, le modèle de Selten \tilde{G} a une taille $S_{\tilde{G}} \in O(|\tilde{E}|k\alpha^k)$ avec $|\tilde{E}| = |\Theta^*| \leq |\Theta| \leq \beta^n$ (où $\alpha = \max_{i \in N} |A_i|$ et $\beta = \max_{i \in N} |\Theta_i|$) et avec $k = n$ la taille de chaque hyper-arête. Ainsi, le modèle de Selten est de taille $S_{\tilde{G}} \in O(|\Theta^*|n\alpha^n) \subseteq O(n(\alpha\beta)^n)$.

Le modèle de Selten \tilde{G} , en tant que jeu hypergraphique, est la représentation succincte d'une forme normale, plus précisément du jeu $G' = (\tilde{N}, \tilde{A}, (GEU_{(i,\theta_i)})_{(i,\theta_i) \in \tilde{N}})$. Ce jeu est différent de la forme normale complète proposée par Harsanyi pour les jeux bayésiens (la transformation Harsanyi ne préserve pas les utilités). La complexité spatiale de G' est supérieure à celle du modèle de Selten : $S_{G'} \in O(n\beta\alpha^{n\beta}) = O(S_{\tilde{G}} \cdot \beta(\alpha^{\beta-1}/\beta)^n)$.

Proposition 5 (Complexité temporelle) La transformation de Selten d'un jeu incomplet algébrique en forme normale se fait en temps linéaire si pour tout $i \in N$ l'opérateur \otimes_i est linéaire.

En effet, pour chaque valeur $\tilde{u}_{(i,t)}^{ens(\theta)}$, nous effectuons une unique opération \otimes_i . Si cette opération est en $O(\gamma)$, la transformation se fait en $O(n\gamma(\alpha\beta)^n)$. Remarquons que les distributions a posteriori $d_i(\cdot|\theta_i)$ sont données (définition 6). Sans quoi il faudrait également considérer la complexité temporelle du conditionnement.

Par conséquent, la détermination d'un équilibre de Nash dans un jeu incomplet n'est pas plus difficile que dans un jeu hypergraphique basé sur \oplus . Ceci suggère que l'existence d'un équilibre dans de nombreux types de jeux incomplets est un problème NP-complet (et pas plus !).

5 Conclusion

Cet article apporte trois contributions. D'abord, nous proposons un cadre algébrique permettant de représenter des jeux incomplets basés sur différentes approches connues en théorie de la décision sous incertitude. Ce cadre généralise deux formes de jeux existantes : les jeux bayésiens et les jeux possibilistes ; nous en proposons d'autres en exemple (multi-critères, ordre de grandeur de l'utilité espérée).

Ensuite, nous généralisons le théorème dit de Howson et Rosenthal à ce modèle algébrique et à un nombre quelconque d'agents. À partir de toute instance de jeu incomplet, nous pouvons ainsi construire en temps linéaire son modèle de Selten : un jeu hypergraphique de complexité spatiale jamais supérieure et préservant les utilités. Les équilibres de Nash y sont en correspondance. L'étude d'un jeu incomplet peut donc se ramener à l'étude d'un jeu complet, soit sous sa forme succincte (hypergraphique), soit sous sa forme normale (dont la complexité spatiale est supérieure).

Enfin, les structures de données et les preuves ont été exprimées en COQ. D'une part, cela leur donne une très forte garantie de correction. D'autre part, cela permet leur réutilisation. En effet, cette formalisation en COQ est disponible sur GitHub et a vocation à être étoffée ultérieurement.

Ces travaux préliminaires ouvrent plusieurs voies de recherche. D'une part, il faudrait étendre notre résultat aux capacités quelconques, et définir des jeux utilisant des mesures de plausibilité non décomposables (par exemple, des jeux basés sur l'utilité dépendant du rang [18], et plus généralement sur des intégrales de Choquet [5]). D'autre part, il faut maintenant travailler du point de vue algorithmique : en dotant les jeux hypergraphiques d'une algorithmique efficace, nous espérons en doter également les jeux algébriques incomplets. Plusieurs pistes sont envisageables, par exemple l'utilisation de la programmation par contraintes (en transformant à son tour le jeu hypergraphique en CSP valué); une autre piste serait d'envisager l'algorithme de Lemke-Howson [12] dans un cadre algébrique. Enfin, nous visons à intégrer les définitions et structures de données définies en COQ dans une bibliothèque de preuves plus large. Ce travail, en plus d'amener de nouveaux résultats, permettrait de justifier la correction des algorithmes implémentés, voire de les générer automatiquement.

Références

- [1] Nahla Amor, Hélène Fargier, Régis Sabbadin, and Meriem Trabelsi. Ordinal polymatrix games with incomplete information. In *Principles of Knowledge Representation and Reasoning*, pages 99–108, 2020.
- [2] Yves Bertot and Pierre Castéran. *Interactive theorem proving and program development : Coq'Art : the calculus of inductive constructions*. Springer Science & Business Media, 2013.
- [3] David Blackwell. Equivalent comparisons of experiments. *The Annals of Mathematical Statistics*, pages 265–272, 1953.
- [4] Peter Borm, Freek van Meegen, and Stef Tijs. A perfectness concept for multicriteria games. *Mathematical Methods of Operations Research*, 49(3) :401–412, 1999.
- [5] Gustave Choquet. Theory of capacities. In *Annales de l'Institut Fourier*, volume 5, pages 131–295, 1954.
- [6] Francis C Chu and Joseph Y Halpern. Great expectations. Part II : Generalized expected utility as a universal decision rule. *Artificial Intelligence*, 159(1-2) :207–229, 2004.
- [7] Didier Dubois and Henri Prade. Possibility theory as a basis for qualitative decision theory. In *International Joint Conference on Artificial Intelligence*, volume 95, pages 1924–1930, 1995.
- [8] Jacques Durieu, Hans Haller, Nicolas Quérou, and Philippe Solal. Ordinal games. *International Game Theory Review*, 10(02) :177–194, 2008.
- [9] John C Harsanyi. Games with incomplete information played by “Bayesian” players, I–III. Part I : The basic model. *Management Science*, 14(3) :159–182, 1967.
- [10] Joseph T Howson Jr and Robert W Rosenthal. Bayesian equilibria of finite two-person games with incomplete information. *Management Science*, 21(3) :313–315, 1974.
- [11] Elena Janovskaja. Equilibrium points in polymatrix games. *Lithuanian Mathematical Journal*, (2) :381–384, 1968.
- [12] Carlton E Lemke and Joseph T Howson, Jr. Equilibrium points of bimatrix games. *Journal of the Society for industrial and Applied Mathematics*, 12(2) :413–423, 1964.
- [13] Oskar Morgenstern and John Von Neumann. *Theory of games and economic behavior*. Princeton university press, 1953.
- [14] John Nash. Non-cooperative games. *Annals of Mathematics*, pages 286–295, 1951.
- [15] Christos H Papadimitriou and Tim Roughgarden. Computing correlated equilibria in multi-player games. *Journal of the ACM*, 55(3) :1–29, 2008.
- [16] Patrice Perny, Olivier Spanjaard, and Paul Weng. Algebraic markov decision processes. In *International Joint Conference on Artificial Intelligence*, pages 1372–1377, 2005.
- [17] Cédric Pralet, Gérard Verfaillie, and Thomas Schiex. An algebraic graphical model for decision with uncertainties, feasibilities, and utilities. *Journal of Artificial Intelligence Research*, 29 :421–489, 2007.
- [18] John Quiggin. A theory of anticipated utility. *Journal of Economic Behavior & Organization*, 3(4) :323–343, 1982.
- [19] Stéphane Le Roux. Acyclic preferences and existence of sequential Nash equilibria : A formal and constructive equivalence. In *Theorem Proving in Higher Order Logics*, LNCS vol. 5674, pages 293–309. Springer, 2009.
- [20] Leonard J Savage. *The Foundations of Statistics*. Wiley, 1954.
- [21] Herbert A Simon. A behavioral model of rational choice. *The Quarterly Journal of Economics*, 69(1) :99–118, 1955.
- [22] Siegfried Weber. \perp -decomposable measures and integrals for Archimedean t-conorms \perp . *Journal of Mathematical Analysis and Applications*, 101(1) :114–138, 1984.
- [23] Nic Wilson. An order of magnitude calculus. In *Uncertainty in Artificial Intelligence*, pages 548–555. Morgan Kaufmann, 1995.
- [24] Yejun Xu, Ravi Patnayakuni, and Huimin Wang. The ordinal consistency of a fuzzy preference relation. *Information Sciences*, 224 :152–164, 2013.

Améliorer la généralisation de l'équité en apprentissage grâce à l'Optimisation Distributionnellement Robuste

J. Ferry¹, U. Aïvodji², S. Gambs², M-J. Huguet¹, M.Siala¹

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

² UQAM, Montréal, Canada

jferry@laas.fr

Résumé

Pour répondre aux enjeux de biais non-désirés en apprentissage machine, de nombreux travaux ont proposé des techniques d'amélioration de l'équité se basant sur des métriques statistiques. Cependant, l'expérience montre que la généralisation sur de nouvelles données n'est pas toujours au rendez-vous. Pour répondre à ce problème, nous proposons une technique d'Optimisation Distributionnellement Robuste permettant de générer des modèles d'apprentissage dont l'équité généralise mieux sur des nouvelles données. L'évaluation expérimentale de cette technique démontre son efficacité.

Mots-clés

Apprentissage supervisé, Équité, Généralisation, Optimisation Distributionnellement Robuste.

Abstract

Recent work have proposed fairness enhancement methods based on statistical metrics to address the problem of negative bias in machine learning. However, it has been shown that these methods do not always generalize well to new data. To address this issue, we propose a Distributionally Robust Optimization technique allowing for the generation of learning models whose fairness generalizes better to new data. Empirical evaluation demonstrates the effectiveness of the approach.

Keywords

Supervised Learning, Fairness, Generalization, Distributionally Robust Optimization

1 Introduction

Le déploiement croissant de systèmes d'intelligence artificielle soulève l'enjeu, de plus en plus étudié, de l'équité de ces systèmes. En particulier, l'objectif est souvent d'assurer que des systèmes, entraînés sur des données historiquement biaisées, ne reproduisent pas ces biais, parfois implicites. Considérant que certains attributs, dits sensibles, ne devraient pas (pour des raisons éthiques et/ou légales) influencer sur la décision, différentes métriques d'équité statistique ont été proposées. Elles reposent généralement sur le même principe : égaliser la valeur d'une certaine me-

sure (par exemple le taux de prédictions positives) entre des groupes différenciés par la valeur d'un ou plusieurs attribut(s) sensible(s). Plusieurs méthodes ont été proposées pour apprendre des modèles respectant ces contraintes d'équité sur leur ensemble d'entraînement. Ces méthodes peuvent être regroupées en trois grandes familles d'approches. D'une part, les approches de *preprocessing* [8] ou de *postprocessing* [6] interviennent respectivement avant (pour éliminer le biais des données d'entraînement) et après (en modifiant les prédictions pour les rendre équitables) la phase d'apprentissage qui conduit à la construction du modèle. Enfin, les approches *inprocessing* [11] consistent à modifier directement l'algorithme d'apprentissage afin de construire des modèles équitables à partir de données possiblement biaisées. La méthode que nous proposons s'applique à cette dernière famille de techniques. Cependant, quelle que soit l'approche choisie, la généralisation de l'équité de ces modèles sur de nouvelles données n'est souvent pas au rendez-vous.

Pour améliorer cette généralisation, [7] propose l'ajout d'un terme de régularisation (mesurant la stabilité) à la fonction objectif d'un problème d'apprentissage équitable. En s'assurant que les prédictions du modèle ne varient pas trop lorsque l'ensemble d'entraînement est perturbé, cette méthode permet de borner théoriquement l'erreur de généralisation.

Plusieurs approches améliorant la généralisation de métriques statistiques d'équité [3, 4, 9] sont basées sur la méthode de [1], qui formule le problème d'apprentissage équitable comme un jeu à deux joueurs. L'un des joueurs optimise la fonction objectif (incluant un terme d'équité) et l'autre cherche à approximer la relaxation Lagrangienne la plus difficile en modifiant les coefficients Lagrangiens associés à la violation de l'équité. Dans [3, 4], le second joueur met à jour les coefficients Lagrangiens associés aux contraintes d'équité en mesurant la violation de l'équité sur un ensemble de validation séparé (plutôt que sur l'ensemble d'entraînement lui-même), ce qui permet d'améliorer la généralisation de ces contraintes.

Enfin, plutôt que d'optimiser une fonction objectif f sur un ensemble \mathcal{D} , l'Optimisation Distributionnellement Robuste (ODR) consiste à optimiser f sur le "pire cas", parmi un ensemble de perturbations de \mathcal{D} [10] (et donc d'optimiser f

pour un ensemble de distributions voisines de \mathcal{D}). D'autres travaux récents utilisent le principe de l'ODR pour améliorer la généralisation de l'équité [9, 10]. Dans [10], un modèle est construit en minimisant l'erreur maximale sur un ensemble de groupes définis par la valeur d'attributs biaisés. L'approche d'ODR de [9] reprend la formulation de [1]. Le second joueur mesure alors la violation de l'équité "pire cas" en pondérant les instances d'entraînement (l'équité est ainsi optimisée pour un ensemble de pondérations des instances de l'ensemble d'entraînement).

2 Problématique

Nous considérons un algorithme d'apprentissage supervisé équitable de la littérature, produisant des modèles interprétables de type *rule list* : FairCORELS [2]¹. Cet algorithme se base sur une approche de type branch and bound explorant l'ensemble \mathcal{R} des *rule lists* à l'aide d'un arbre des préfixes. Dans cet arbre, chaque noeud correspond à une règle et chaque chemin (depuis la racine) est une solution possible. Ainsi, une étape préalable à l'utilisation de FairCORELS est le minage de règles. Ces dernières peuvent être n'importe quelle combinaison des attributs du jeu de données et doivent prendre une valeur binaire pour toutes les instances du jeu de données d'entraînement. L'objectif de FairCORELS est de déterminer la solution minimisant la fonction objectif f_{obj} (somme pondérée de l'erreur de classification et du nombre de règles) tout en respectant une contrainte d'équité ϵ donnée.

Soit $\mathcal{D} = (X, Y, A)$ l'ensemble de données d'entraînement, où X est l'ensemble des attributs non sensibles, Y l'ensemble des étiquettes et A l'attribut (ou l'ensemble des attributs) sensible(s). La fonction $\text{misc}(\cdot)$ mesure l'erreur de classification alors que $\text{unf}(\cdot)$ quantifie la violation de l'équité (selon la métrique choisie). La *rule list* recherchée r^* est la solution du problème suivant, où K_r désigne la longueur de r , et λ est un coefficient de régularisation :

$$\begin{aligned} \arg \min_{r \in \mathcal{R}} \quad & f_{obj} = \text{misc}(r, X, Y) + \lambda \cdot K_r \\ \text{s.t.} \quad & \text{unf}(r, X, Y, A) \leq \epsilon, \end{aligned}$$

A chaque itération, un préfixe r est évalué. S'il améliore f_{obj} et respecte la contrainte d'équité (sur l'ensemble d'entraînement), la meilleure solution courante est mise à jour. Plusieurs heuristiques peuvent être utilisées pour guider l'exploration, dont l'efficacité est améliorée par l'existence de différentes bornes. FairCORELS retourne théoriquement la solution optimale, c'est-à-dire la *rule list* pour laquelle f_{obj} est minimisée, et qui respecte la contrainte d'équité. Toutefois, la taille de l'arbre des préfixes augmentant exponentiellement avec le nombre de règles, un paramètre n_{iter} définit le nombre maximal de noeuds à explorer dans l'arbre des préfixes. Ce paramètre limite l'espace mémoire utilisé par le programme, permettant ainsi d'obtenir de bonnes solutions dans un délai maîtrisé.

1. <https://github.com/ferryjul/fairCORELS>

Il est possible d'utiliser FairCORELS pour générer un ensemble de solutions, en réalisant des appels successifs pour différentes valeurs d' ϵ . On obtient ainsi un ensemble de compromis précision/équité, qu'on peut représenter par un front de Pareto sur l'ensemble d'entraînement. Cependant, ces solutions définissent souvent un ensemble de compromis moins intéressants sur leur ensemble de test, en raison d'une mauvaise généralisation de l'équité, notamment lorsque les contraintes sont fortes (ϵ faible). Plusieurs travaux récents ont ainsi proposé des méthodes pour palier à ce problème [3, 4, 7, 9, 10].

3 Méthode proposée

Notre objectif est d'obtenir des modèles dont l'équité généralise bien sur l'ensemble de test (c'est-à-dire de nouvelles données non observées pendant l'entraînement de ces modèles). A l'inverse de la précision, l'équité statistique se mesure uniquement sur des ensembles d'instances (et pas sur des instances seules). En outre, il est possible qu'un modèle paraisse équitable sur un ensemble donné, tout en prenant des décisions non équitables localement [5]. Notre intuition est qu'assurer l'équité sur plusieurs sous-ensembles de l'ensemble d'entraînement peut forcer le modèle construit à une meilleure organisation de ses décisions, et par conséquent à une meilleure généralisation de l'équité. Chaque sous-ensemble aléatoire de taille suffisamment importante présente une distribution voisine de celle de l'ensemble global. Pour cette raison, notre méthode est directement inspirée de l'Optimisation Distributionnellement Robuste, qui vise à optimiser une métrique (ici l'équité) sur un ensemble donné (ici l'ensemble d'entraînement \mathcal{D}), mais également sur un ensemble de distributions voisines (ici les sous-ensembles de \mathcal{D}).

Dans le contexte de FairCORELS, l'approche proposée consiste à s'assurer que la *rule list* générée respecte la contrainte d'équité sur l'ensemble d'entraînement, et sur un certain nombre de sous-ensembles aléatoires de celui-ci, approximant les "distributions voisines" du cadre de l'ODR. Pour cela, n masques binaires aléatoires sont générés et utilisés afin de définir n sous-ensembles de l'ensemble d'entraînement. Lorsque la *rule list* évaluée r permet d'améliorer la fonction objectif, on ne met à jour la meilleure solution courante que si r respecte la contrainte d'équité sur l'ensemble d'entraînement et sur chacun des sous-ensembles définis par les n masques. La *rule list* recherchée r^* est donc la solution du problème suivant, où V_i est le sous-ensemble de V défini par le masque i , pour $V \in \{X, Y, A\}$:

$$\begin{aligned} \arg \min_{r \in \mathcal{R}} \quad & f_{obj} = \text{misc}(r, X, Y) + \lambda \cdot K_r \\ \text{s.t.} \quad & \text{unf}(r, X, Y, A) \leq \epsilon, \\ & \text{unf}(r, X_i, Y_i, A_i) \leq \epsilon \\ & \quad \forall i \in [n] \end{aligned}$$

4 Evaluation expérimentale

Afin d'évaluer notre méthode, nous calculons un ensemble de solutions non-dominées (en faisant varier la contrainte

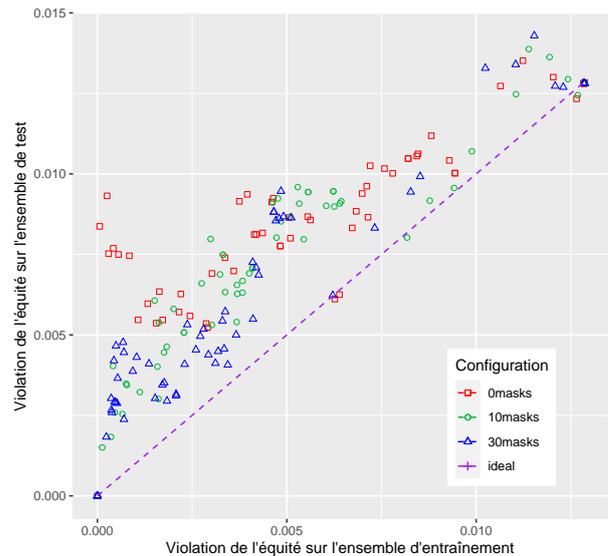
Jeu de données	Nombre d'instances	Attribut sensible	Nombre de règles minées	Prédiction
Adult Income	33 917	Homme/Femme	183	Salaire : haut/bas
COMPAS	5 273	African-American/Caucasian	165	Récidive : oui/non
Default of Credit Card	29 986	Homme/Femme	189	Paiement refusé le mois prochain : oui/non
Marketing	41 175	Age : entre 30 et 60 ans ou pas	179	Souscription : oui/non

TABLE 1 – Caractéristiques des différents jeux de données (après notre pré-traitement) utilisés pour évaluer notre méthode

d'équité) pour trois valeurs de masques : 0 (cas où notre méthode n'est pas utilisée), 10 et 30. Cette évaluation a été réalisée pour six métriques statistiques d'équité (*Statistical Parity*, *Predictive Parity*, *Predictive Equality*, *Equal Opportunity*, *Equalized Odds* et *Conditional Use Accuracy Equality*), sur quatre ensembles de données (*Adult Income*, *COMPAS*, *Default of Credit Card* et *Marketing*). Les caractéristiques (taille, attribut sensible, ...) de ces ensembles de données sont résumés dans le tableau 1.

Les quatre ensembles de données regroupent des données historiquement biaisées (notamment par rapport à l'attribut sensible considéré), et peuvent donc être utilisés pour générer des ensembles de compromis entre la précision et l'équité. Afin de pouvoir utiliser FairCORELS sur ces ensembles de données, nous les avons binarisés, avant de calculer, pour chacun d'entre eux, un ensemble de règles. Les règles minées sont des conjonctions d'au plus deux attributs (ou leur négation) capturant plus de $\min_{support}\%$ des instances de l'ensemble de données, où $\min_{support}$ est choisi de sorte à limiter le nombre de règles minées. Pour toutes nos expérimentations, n_{iter} est fixé à $25 \cdot 10^5$, $\lambda = 10^{-3}$ et l'heuristique d'exploration utilisée est le BFS *obj.-aware* (une recherche en largeur priorisant les solutions présentant une meilleure valeur de f_{obj} parmi celles de même profondeur). Les valeurs reportées (précision et équité) sont obtenues au moyen d'une *validation croisée 5-folds*. Tous les résultats obtenus confortent les observations présentées ci-après sur l'ensemble de données *Default of Credit Card* et pour la métrique d'équité *Predictive Equality*.

La figure 1 présente la généralisation des solutions construites, où les points les plus proches de la diagonale correspondent aux solutions dont l'équité généralise le mieux. Elle illustre l'amélioration apportée pour la généralisation de l'équité, en particulier pour des contraintes fortes d'équité. En effet, on peut voir que lorsque la violation de l'équité sur l'ensemble d'entraînement est faible, les solutions obtenues en utilisant la méthode basée sur l'ODR (10masks et 30masks) présentent une violation de l'équité en test plus faible que celles obtenues sans ODR (0masks). La figure 2 suggère qu'un compromis entre performance sur l'ensemble d'entraînement (équité et précision) et généralisation doit être fait. En effet, la méthode basée sur l'ODR permet de générer des solutions moins performantes sur l'ensemble d'entraînement, mais plus robustes (généralisant mieux). La figure 3 démontre qu'une conséquence de l'amélioration de la généralisation


 FIGURE 1 – Équité sur l'ensemble de test en fonction de l'équité sur l'ensemble d'entraînement pour la *Predictive Equality* sur l'ensemble de données *Default of Credit Card*

de l'équité est la production d'un front de Pareto sur l'ensemble de test plus garni. L'utilisation de notre méthode permet en effet de générer des solutions avec des violations d'équité très faibles, y compris sur de nouvelles données, sans affecter de manière trop importante la précision. Le nombre de masques conduisant à la meilleure généralisation varie selon la métrique d'équité et le jeu de données considérés. L'ensemble des résultats obtenus démontrent ainsi l'intérêt de l'approche proposée : générer des solutions plus robustes, et présentant de meilleurs compromis équité/précision sur de nouvelles données, au prix d'une performance dégradée sur l'ensemble d'entraînement.

5 Conclusion

Nous proposons une méthode inspirée de l'ODR améliorant la généralisation de l'équité en apprentissage supervisé. Une évaluation expérimentale en démontre l'intérêt, bien qu'elle ne s'accompagne pas de garanties théoriques. Les perspectives futures portent sur l'étude de l'impact du nombre de masques sur la généralisation ainsi que des extensions pour d'autres modèles d'apprentissage équitable.

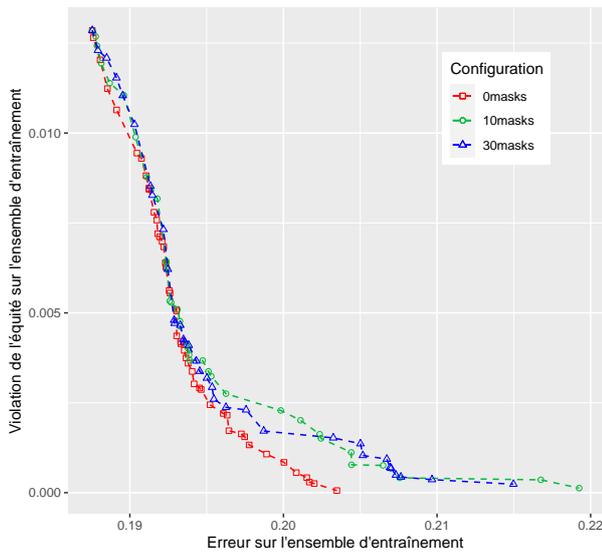


FIGURE 2 – Front de Pareto (violation de l'équité en fonction de l'erreur) sur l'ensemble d'entraînement, pour la *Predictive Equality*, sur l'ensemble de données *Default of Credit Card*

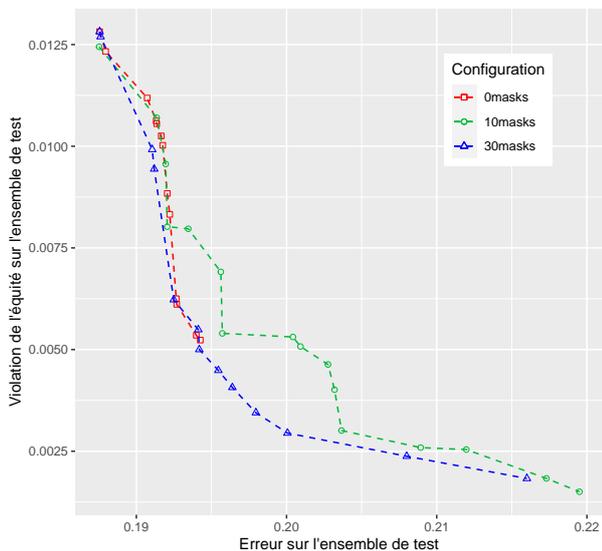


FIGURE 3 – Front de Pareto (violation de l'équité en fonction de l'erreur) sur l'ensemble de test, pour la *Predictive Equality*, sur le jeu de données *Default of Credit Card*

Références

- [1] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna M. Wallach. A reductions approach to fair classification. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 60–69. PMLR, 2018.
- [2] Ulrich Aïvodji, Julien Ferry, Sébastien Gambs, Marie-José Huguët, and Mohamed Siala. Learning fair rule lists. *arXiv preprint arXiv :1909.03977*, 2019.
- [3] Andrew Cotter, Maya Gupta, Heinrich Jiang, Nathan Srebro, Karthik Sridharan, Serena Wang, Blake Woodworth, and Seungil You. Training fairness-constrained classifiers to generalize, 2018.
- [4] Andrew Cotter, Maya Gupta, Heinrich Jiang, Nathan Srebro, Karthik Sridharan, Serena Wang, Blake Woodworth, and Seungil You. Training well-generalizing classifiers for fairness metrics and other data-dependent constraints. In *International Conference on Machine Learning*, pages 1397–1405. PMLR, 2019.
- [5] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.
- [6] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. *arXiv preprint arXiv :1610.02413*, 2016.
- [7] Lingxiao Huang and Nisheeth K. Vishnoi. Stable and fair classification. *36th International Conference on Machine Learning, ICML 2019, 2019-June :5130–5144*, 2 2019.
- [8] Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1) :1–33, 2012.
- [9] Debmalya Mandal, Samuel Deng, Suman Jana, Jeanette Wing, and Daniel J Hsu. Ensuring fairness beyond the training data. *Advances in Neural Information Processing Systems*, 33, 2020.
- [10] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts : On the importance of regularization for worst-case generalization. *arXiv preprint arXiv :1911.08731*, 2019.
- [11] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact : Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web*, pages 1171–1180, 2017.

Je relie système complexe, combinatoire, biologie des systèmes et apprentissage profond, qui suis-je ? Hyper-réseau, définitions et cas d'usage

M. Guery¹, G. Blot¹, F. Rousseaux², E. Soulier³, S. Cormier²

¹ Data Nostra, R&D lab

² Université de Reims Champagne Ardenne (URCA),
CReSTIC EA 3804, 51097 Reims, France, Equipe MODECO

³ Université de Technologie de Troyes (UTT), UMR CNRS 6281 "Institut Charles Delaunay",
Equipe Tech-CICO, Département Homme, Environnement,
Technologies de l'Information et de la Communication (HETIC)

Résumé

Les hyper-réseaux apparaissent dans de nombreux domaines et usages dont des problèmes de combinatoire, la théorie des systèmes complexes, la biologie des systèmes et l'apprentissage profond avec une sémantique différente. Il est important de comprendre les utilisations du terme hyper-réseau dans la littérature afin d'éviter toute confusion. L'objectif de cet article est de faire un état de l'art des définitions de l'hyper-réseau suivant le domaine d'utilisation. Nous nous attacherons notamment à situer les hyper-réseaux dans notre contexte de recherche sur l'intelligence artificielle.

Mots-clés

Hyper-réseau, intelligence artificielle, combinatoire, système complexe, apprentissage profond, biologie des systèmes

Abstract

Hypernetworks appeared in many fields and uses including combinatorics problems, the theory of complex systems, the systems biology and deep learning with different semantics. It is important to understand the uses of the term hypernetwork in the literature to avoid confusion. The objective of this article is to make a state of the art of the definitions of the hypernetwork according to the field of use. In particular, we will try to situate hypernetworks in our research context on artificial intelligence.

Keywords

Hypernetwork, artificial intelligence, combinatorics, complex system, deep learning, systems biology

1 Introduction

Selon le domaine d'utilisation, le terme *hyper-réseau* est défini et utilisé différemment dans : les problèmes de combinatoire, la théorie des systèmes complexes, la biologie des systèmes et le domaine de l'apprentissage profond. En ef-

fet, le terme hyper-réseau désignera un principe différent suivant le domaine dans lequel on l'utilise. Après avoir rappelé la définition de l'hyper-réseau dans chacun de ces domaines, cet article vise à mettre en lumière les développements en intelligence artificielle utilisant les hyper-réseaux des systèmes complexes.

Ce papier est organisé ainsi : dans la section 2, nous décrivons ce qu'est un hyper-réseau en combinatoire, dans la section 3, nous présentons les hyper-réseaux dans le sillage des systèmes complexes, dans la section 4, nous décrivons l'architecture de l'hyper-réseau en biologie des systèmes, dans la section 5, nous expliquons le fonctionnement et le rôle de l'hyper-réseau en apprentissage profond et enfin, dans la section 6, nous définissons les pistes de recherches en intelligence artificielle basé sur la théorie de l'hyper-réseau des systèmes complexes.

2 Combinatoire

Pour rappel, en mathématiques, la combinatoire est un domaine d'étude permettant d'étudier les configurations de collections finies d'objets, les combinaisons d'ensembles finis ainsi que les dénombrements.

2.1 Définitions

Définition d'un hypergraphe. En 1973, Claude Berge a publié un livre sur la théorie des graphes et des hypergraphes [3] et une nouvelle version de la deuxième partie de ce même livre sur les hypergraphes en 1987 [4]. Dans ces deux publications, il définit un hypergraphe non orienté H comme une paire $H = (X; E)$ où X est un ensemble d'éléments appelés nœuds ou sommets, et E est un ensemble de sous-ensembles non vides de X appelés hyper-relations ou arêtes (voir Figure 1). Contrairement aux relations des graphes ordinaires, une hyper-relation peut être liée à k nœuds tels que : $k \in [1, n]$ où $n = |X|$, nombre de sommets de l'ensemble X , alors qu'une relation d'un graphe classique ne peut être liée qu'à 2 nœuds maximum.

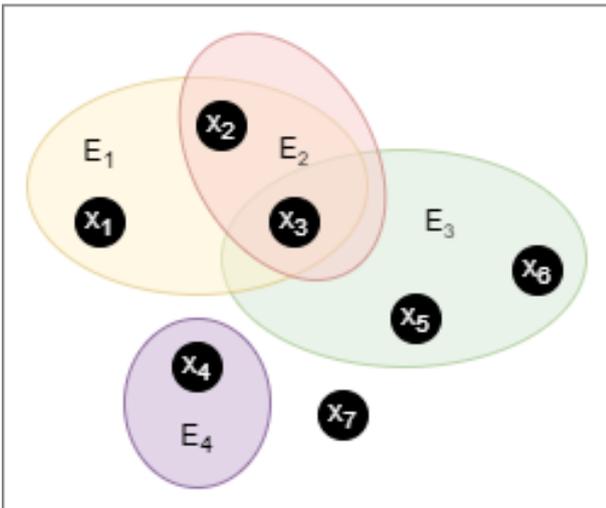


FIGURE 1 – Exemple d’hypergraphe $H = (X; E)$ possédant sept nœuds et quatre hyper-relations où $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ et $E = \{E_1, E_2, E_3, E_4\}$.

Définition d’un hyper-réseau. Un hyper-réseau quant à lui, est défini comme un hypergraphe qui est augmenté d’un poids pour chaque hyper-relation [19, 13]. Un hyper-réseau H est alors défini comme un hypergraphe pondéré constitué d’un ensemble X de sommets, d’un ensemble E d’hyper-relations et d’un ensemble W de poids tel que $H = (X; E; W)$ (voir Figure 2). Les poids représentent la force de l’attachement des sommets à l’intérieur de chaque hyper-relation et permettent à l’hyper-réseau de représenter un modèle probabiliste.

Le cas particulier du k -hyper-réseau. On peut ensuite définir un k -hyper-réseau. Ce dernier possède les mêmes propriétés qu’un hyper-réseau mais chaque hyper-relation connecte exactement k nœuds [13]. Cet hyper-réseau est alors dit k -uniforme.

En combinatoire, l’hyper-réseau est un objet mathématique.

2.2 Exemples de domaines et applications

2.2.1 Apprentissage cognitif et mémoire

Dans [19], Byoung-Tak Zhang propose d’utiliser l’hyper-réseau comme modèle d’architecture pour l’apprentissage cognitif et la mémoire. L’objectif est d’utiliser l’hyper-réseau pour apprendre des modèles de langage généraux à partir des dialogues de situations quotidiennes. Les données proviennent de séries comme : Friends, House, 24 ou encore Prison Break. Chaque hyper-relation représente une combinaison de mots formant une phrase. Le poids de l’hyper-relation est la probabilité d’apparition de ce modèle d’association de mots. Cet hyper-réseau permet de montrer la capacité de rappel linguistique, c’est-à-dire créer une phrase complète ou générer une nouvelle phrase à partir d’une liste partielle de mots. Plusieurs cas d’utilisation sont présentés dans cet article avec de très bons résultats : la complétion de phrases, la classification de phrases, la génération d’un texte pour une image et la génération d’une

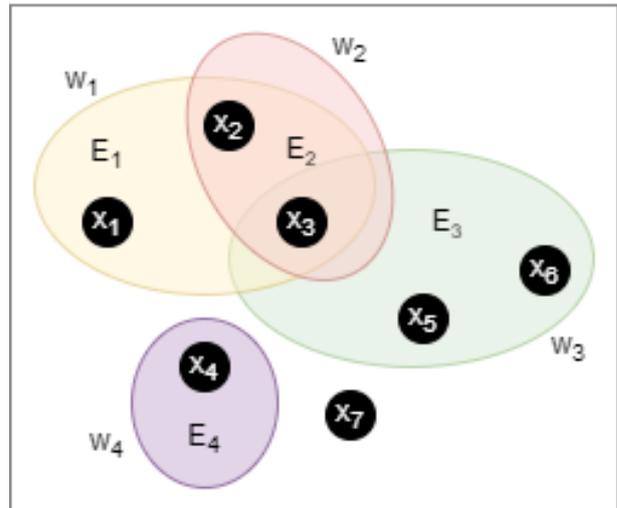


FIGURE 2 – Exemple d’hyper-réseau $H = (X; E; W)$ possédant sept nœuds et quatre hyper-relations avec quatre poids où $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$, $E = \{E_1, E_2, E_3, E_4\}$ et $W = \{w_1, w_2, w_3, w_4\}$.

image pour un texte.

2.2.2 Classification

Dans [13], les auteurs présentent une méthode de classification innovante pour identifier les modules de gènes associés aux cancers à partir de Micro-ARN. L’hyper-réseau joue donc le rôle de classifieur.

Soit un jeu de données avec un ensemble de variables explicatives X et une classe Y . L’hyper-réseau représente le jeu de données et chaque hyper-relation est une ligne du jeu de données (voir Figure 3). Un des avantages de l’hyper-réseau réside dans sa capacité à permettre à chaque hyper-relation de disposer d’un nombre différent de variables explicatives. En spécifiant une entrée X , le classifieur renvoie alors la classe dont la probabilité est la plus élevée. L’hyper-réseau représente un modèle probabiliste d’un ensemble de données utilisant une population d’hyper-relations et leurs poids.

Pour faire évoluer l’hyper-réseau, le poids d’une hyper-relation est exprimé en autorisant les doublons de cette même hyper-relation dans l’hyper-réseau. La tâche d’apprentissage d’un hyper-réseau consiste donc à ajuster le nombre d’hyper-relations afin de minimiser les erreurs de classification.

Les résultats présentés par les auteurs montrent que l’hyper-réseau classifieur surpassent les arbres de décision et la classification naïve bayésienne, tout en offrant des performances comparables aux réseaux de neurones et aux machines à vecteurs de support. Ils montrent également que l’hyper-réseau classifieur trouvent des blocs de Micro-ARN biologiquement significatifs. Ils concluent que les structures d’hyper-réseau sont efficaces car elles fournissent des solutions interprétables, ainsi que de bonnes performances de classification.

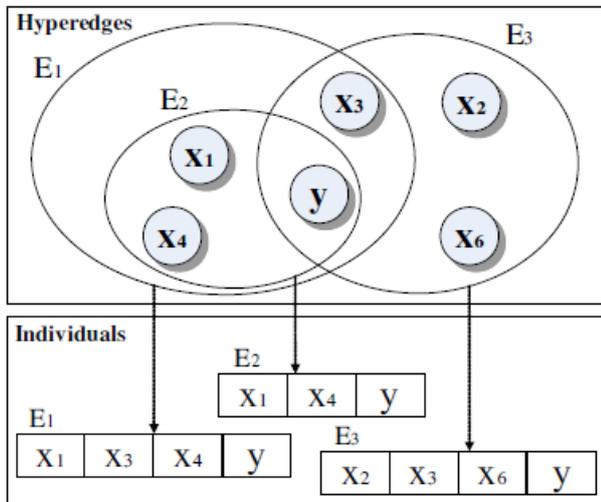


FIGURE 3 – Image issue de [13]. Chaque hyper-relation représente une ligne du jeu de données et possède deux informations : l'ensemble des variables explicatives (X) et le label (Y).

3 Système complexe

Un système complexe est un ensemble constitué d'un grand nombre d'entités en interaction dont il est difficile de prévoir les rétroactions, les comportements ou les évolutions par le calcul.

3.1 Définitions

Dans le domaine des systèmes complexes, les hyper-réseaux repose sur l'hypergraphe sauf que les relations jouent un rôle majeur contrairement au modèle probabiliste utilisé dans les exemples de la partie 1. Ronald Harry Atkin [1] et Jeffrey Johnson [11] ont une approche relationnelle de l'hypergraphe. Selon eux, les hyper-réseaux permettent de représenter un système complexe. Leur hyper-réseau utilise la théorie des complexes simpliciaux qui peut être vue comme un hypergraphe [12]. Afin de comprendre ce qu'est un hyper-réseau, il est important de définir quelques termes [11, 12] :

- le simplexe abstrait est un objet abstrait déterminé par des sommets et noté $\sigma = \langle v_0, v_1, \dots, v_p \rangle$, les sommets sont ordonnés et le simplexe possède une représentation géométrique ;
- le simplexe est un simplexe abstrait noté également $\sigma = \langle v_0, v_1, \dots, v_p \rangle$ mais qui possède une relation n-aire générale ;
- une famille simpliciale est un ensemble de simplexes ;
- un complexe simplicial est un ensemble de simplexes en considérant toutes leurs faces ;
- un hyper-simplexe (ou simplexe relationnel) est un ensemble ordonné de sommets avec une relation n-aire explicite ;
- un hyper-réseau est un ensemble d'hyper-simplexes ;

- le backcloth est la structure complexe simplicial ;
- le trafic représente l'activité sur le backcloth.

D'après Johnson [11], les hyper-réseaux possèdent trois caractéristiques principales :

- la première est celle de l'hyper-simplexe permettant de représenter une relation entre x objets ;
- la seconde est que les hyper-simplexes fournissent une manière non ambiguë de discriminer les niveaux dans des systèmes à plusieurs niveaux ;
- la troisième est que ces structures peuvent représenter des structures dynamiques à plusieurs niveaux (backcloth et trafic).

Pour lui, les hyper-réseaux ne sont pas en concurrence avec les hypergraphes ou les réseaux, ils généralisent les deux. Pour chaque hypergraphe, il existe un hyper-réseau unique, comme il y en a un pour chaque réseau. Les hyper-réseaux offrent donc une puissance de représentation supplémentaire.

On remarque aisément que l'hyper-réseau définie en combinatoire et l'hyper-réseau d'Atkin et Johnson possèdent des points communs. Ces définitions désignent deux choses différentes, mais les objets mathématiques sous-jacents sont identiques, mais sont nommés différemment par exemple : l'hyper-relation combinatoire est désignée par l'hyper-simplexe.

L'hyper-réseaux d'Atkin et Johnson est un outil permettant de modéliser, analyser et simuler des systèmes complexes. D'après Johnson [11], la structure (backcloth) et le comportement (trafic) sont fortement liés : la structure contraint la dynamique.

3.2 Exemples de domaines et applications

3.2.1 Match de football

L'étude [14] utilise les hyper-réseaux pour représenter les dynamiques de comportement pendant un match de football. Dans cette application, l'avantage de l'hyper-réseau est de pouvoir connecter plusieurs joueurs grâce aux simplexes. L'hyper-réseau permet d'analyser à plusieurs niveaux les interactions coopératives et compétitives entre les coéquipiers et les adversaires dans l'espace et le temps pendant un match. Les simplexes peuvent capturer les interactions d'ensembles de joueurs qui peuvent inclure un nombre arbitraire de coéquipiers et d'adversaires.

Les auteurs utilisent ces variables :

- l'étiquetage des ensembles de joueurs formés (par exemple, 2 contre 1 correspond à un ensemble avec deux attaquants et un défenseur) ;
- les informations spatio-temporelle des différents ensembles de joueurs en comptant leur fréquence et leur emplacement ;
- la vélocité des joueurs dans des événements spécifiques (arrêt du gardien, but, etc.) ;
- la formation et la dynamique des simplexes de niveau supérieur, notamment, les relations entre simplexes de simplexes.

Pour les auteurs, le backcloth est par exemple le nombre d'acteurs ou les rôles des joueurs et le trafic est le positionnement des joueurs par rapport aux adversaires, aux coéqui-

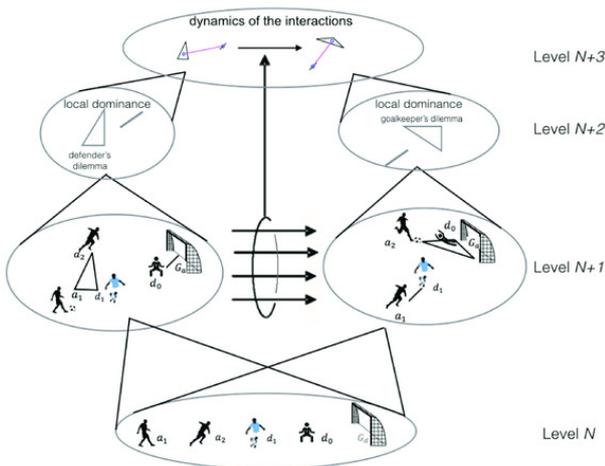


FIGURE 4 – Image issue de [14]. Représentation d’un hyper-réseau à plusieurs niveaux (de bas en haut). Chaque niveau correspond à un niveau d’abstraction différent (Niveau N, joueurs sur le terrain; Niveau N + 1, simplexes de proximité; Niveau N + 2, relation de dominance locale; Niveau N + 3, analyse dynamique via des simplexes de simplexes). Également représenté, le déplacement dans un match de football de 2 périodes de temps séquentielles (de gauche à droite).

piers et le but ou le ballon. L’hyper-réseau et les variables permettent de capturer des caractéristiques pertinentes pour les équipes et le match.

Voici les différentes questions auxquelles l’article essaye de répondre en utilisant l’hyper-réseau (voir Figure 4) :

- au niveau N : la structure a-t-elle changé pendant le match à cause d’événements tels que les remplacements, les expulsions ou les blessures ?
- au niveau N+1 : quels sont les relations les plus fréquentes dans les matchs de football ?
- au niveau N+1 : existe t-il des différences dans la structure et le nombre d’occurrences des simplexes entre les matchs à domicile et à l’extérieur ?
- au niveau N+1 : existe t-il des changements dans la structure des simplexes et la position sur le terrain au fur et à mesure que le score du match évolue ?
- au niveau N+2 : quelle est la dynamique des simplexes juste avant l’évolution du score ?
- au niveau N+3 : existe t-il une interaction entre les simplexes conduisant à l’émergence de nouvelles configurations d’équipes pouvant conduire à marquer un but ?

3.2.2 Sciences Sociales

Dans son article [10], Johnson explique que les hyper-réseaux généralisent les réseaux et les hypergraphes, permettant aux relations entre de nombreux éléments d’être modélisées par des hyper-simplexes avec une structure plus riche que les arêtes d’hypergraphes. Il explique aussi que les hyper-réseaux fournissent un moyen d’intégrer la dynamique ascendante et descendante aux niveaux micro, méso

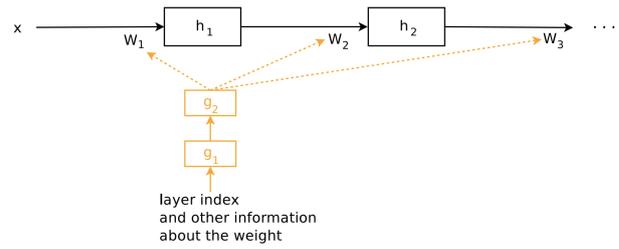


FIGURE 5 – Image tirée de l’article [6]. Les connexions et paramètres noirs sont associés au réseau principal tandis que les connexions et paramètres orange sont associée à l’hyper-réseau.

et macro dans des systèmes à plusieurs niveaux. Et donc que les hyper-réseaux fournissent un moyen naturel de représenter les structures sociales, permettant par exemple de simuler des actions politiques avant leur mise en œuvre.

Johnson prend comme exemple les politiques anti-gang de plusieurs pays. Il montre qu’il serait possible de tester une politique de manière informatique avant de la mettre en œuvre. Les hyper-réseaux permettent d’interpréter des structures de données et peuvent combler le fossé entre les modèles vernaculaires et le calcul des processus sociaux. D’après lui, les hyper-réseaux jouent un rôle important dans l’utilisation du Big Data dans l’informatique appliquée aux sciences sociales et politiques, et joueront un rôle central dans la formulation et la mise en œuvre des politiques futures.

4 Apprentissage profond

L’apprentissage profond est un type d’apprentissage automatique basé sur des réseaux de neurones artificiels. L’apprentissage automatique permet à des machines d’apprendre à résoudre des tâches, à partir de modèles de données, sans être explicitement programmés pour chacune. Ces ordinateurs utilisent des algorithmes et des modèles statistiques pour analyser et trouver des inférences.

4.1 Définition

En 2016, le terme hyper-réseau est apparu dans le domaine de l’apprentissage automatique [6]. L’hyper-réseau est le nom donné au réseau de neurones qui génère des poids pour le réseau de neurones principal plus grand. Le comportement du réseau principal est le même que n’importe quel réseau de neurones habituel : il apprend à faire correspondre les entrées aux valeurs de sorties. L’hyper-réseau prend un ensemble d’entrées contenant des informations sur la structure des poids et génère le poids pour cette couche (voir Figure 5).

En apprentissage profond, l’hyper-réseau est le nom donné au réseau de neurones qui génère des poids pour le réseau de neurones principal plus grand.

4.2 Exemples de domaines et applications

4.2.1 Apprentissage cognitif

Dans [6], les auteurs expliquent que les hyper-réseaux fournissent une abstraction similaire à ce que l'on trouve dans la nature : la relation entre un génotype (l'hyper-réseau) et un phénotype (le réseau principal). L'hyper-réseau est entraîné de bout en bout avec rétropropagation et est plus rapide. L'objectif de leur travail est de rendre les hyper-réseaux utiles pour les réseaux convolutifs profonds et les réseaux récurrents longs. Leur résultat principal est que les hyper-réseaux peuvent obtenir des meilleurs résultats que les méthodes actuelles sur une multitude de tâches de modélisation de séquence comme la génération d'écriture manuscrite et la traduction automatique. Leurs résultats montrent également que les hyper-réseaux appliqués aux réseaux convolutifs obtiennent des résultats respectables pour les tâches de reconnaissance d'image par rapport aux modèles de base tout en nécessitant moins de paramètres entraîna-

4.2.2 Apprentissage continu

L'apprentissage continu est la capacité d'un modèle à apprendre continuellement à partir d'un flux de données. En pratique, cela signifie soutenir la capacité d'un modèle à apprendre et à s'adapter de manière autonome en production à mesure que de nouvelles données arrivent.

Les auteurs ont introduit un nouveau modèle de réseau neuronal [18], l'hyper-réseau conditionné par des tâches. L'apprentissage continu est moins difficile pour les modèles utilisant un hyper-réseau, car au lieu de rappeler les relations d'entrée-sortie de toutes les données précédemment vues, les hyper-réseaux conditionnés par des tâches ne nécessitent que de répéter les réalisations de poids spécifiques à la tâche. Ce nouveau modèle est bien adapté aux problèmes d'apprentissage continu.

Leurs résultats sont excellents et suggèrent que les hyper-réseaux conditionnés par des tâches peuvent avoir une bonne mémoire, ainsi que transférer des informations vers des tâches futures, deux propriétés essentielles en apprentissage continu.

5 Biologie des systèmes

La biologie des systèmes est un domaine d'étude permettant de représenter les relations et les interactions entre les différents niveaux d'informations d'un organisme vivant afin de comprendre comment fonctionne réellement un système biologique.

5.1 Définition

En 1999, José L. Segovia-Juarez et Michael Conrad [16] définissent le terme hyper-réseau comme un modèle représentant hiérarchiquement les niveaux moléculaires, cellulaires et organismiques de l'organisation biologique (voir Figure 6). Cette définition peut être vue de deux manières différentes : la première comme un sous-ensemble des systèmes complexes tel que définie dans la partie 3 et la seconde comme une définition formelle d'un hyper-réseau. C'est-à-dire que l'hyper-réseau ne désigne plus l'ensemble des

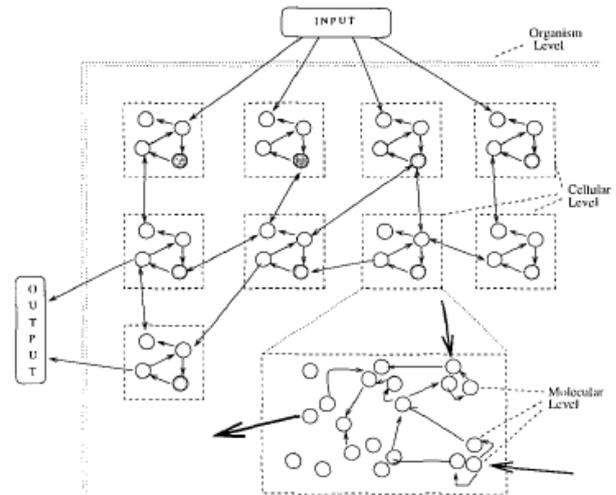


FIGURE 6 – Image tirée de l'article [16] représentant le modèle de l'hyper-réseau. Les cercles représentent les molécules et les flèches les interactions entre molécules. Les rectangles en pointillés autour des ensembles de molécules délimitent les unités cellulaires.

différents modèles hiérarchiques mais il représente cette organisation biologique précise (molécules, cellules et organismes). Les travaux de Segovia-Juarez et Conrad ne citent pas les travaux de Berge ou d'Atkin et Johnson, nous prenons donc comme hypothèse la seconde explication.

5.2 Exemples de domaines et applications

5.2.1 Classification

Dans [17], afin de valider le modèle de l'hyper-réseau, les auteurs utilisent le jeu trivial du morpion. Ils ont à disposition un jeu de données avec 958 configurations et l'objectif est de déterminer si la configuration est gagnante ou perdante pour le joueur qui joue avec les croix. Leur modèle obtient une précision de plus de 90%.

5.2.2 Identification des sites d'épissage d'ADN

L'identification des sites d'épissage d'ADN est une tâche principale de la recherche de gènes. Dans [15], les auteurs ont utilisé leur modèle d'hyper-réseau et développés une application, HyperExon, qui permet d'identifier les sites d'épissage d'ADN. Son apprentissage est basé sur l'évolution moléculaire. Leur base d'apprentissage provient de GenBank, chaque exemple d'ADN a été traduit en chaînes binaires et introduit dans un hyper-réseau pour l'entraînement. Les expériences ont été effectuées par validation croisée. Les performances de généralisation de l'hyper-réseau sont comparables à celles des algorithmes de classification. Le meilleur hyper-réseau obtenu a permis de construire l'application HyperExon pour obtenir des candidats de sites d'épissage. Le système HyperExon a surpassé les principaux systèmes de reconnaissance d'épissures dans la liste des séquences testées.

6 Hyper-réseau et intelligence artificielle

Maintenant que le terme hyper-réseau a été défini, nous souhaitons nous placer dans le domaine des systèmes complexes afin de développer des techniques d'intelligence artificielle utilisant les hyper-réseaux. Voici trois axes de recherche que nous avons identifié :

6.1 Système expert

En 1990, Jeffrey Johnson a présenté au travers de trois articles [7, 8, 9] un système expert basé sur la méthodologie et l'algorithme de la Q-analyse. La Q-analyse permet de trouver les composants q-connectés à chaque dimension d'un complexe simplicial. L'algorithme de la Q-analyse peut être utilisé de deux manières différentes : pour du partitionnement de données (clustering) ou pour l'analyse du trafic sur le backcloth.

La Q-analyse pourrait permettre de construire un système expert capable de détecter et de communiquer des points d'intérêts et d'enrichir la base de connaissance via des interactions avec l'analyste. La découverte des points d'intérêts est un problème majeur car il est subjectif.

6.2 Prédiction d'évènements

Les hyper-réseaux permettent de représenter des systèmes dynamiques. Atkin a suggéré que la formation de simplexes devrait être considérée comme des événements (p-event) dans un système. Ainsi, un événement est caractérisé par une discontinuité géométrique entre la non-existence d'un simplexe au temps $t - 1$ et l'existence de ce simplexe au temps t [11].

On pourrait en principe, développer un algorithme qui puisse prédire, dans un système dynamique, ces événements grâce aux informations du backcloth et du trafic.

6.3 Apprentissage hiérarchique

Au début de son ouvrage [2], Atkin présente deux exemples d'apprentissages hiérarchiques : l'apprentissage de la parole et l'apprentissage de l'écriture. Quand un enfant apprend à parler, il possède un vocabulaire restreint (quelques mots), mais plus l'enfant grandit, plus son vocabulaire se développe, devient plus riche, plus construit. Par opposition, lors de l'apprentissage de l'écriture, l'enfant commence par apprendre l'alphabet avant d'écrire des mots, puis des phrases, puis des articles.

On remarque au travers de ces deux exemples la hiérarchie et la profondeur grandissante de ces apprentissages. Développer des algorithmes d'apprentissage de concepts hiérarchiques utilisant les hyper-réseaux des systèmes complexes est donc une piste intéressante à explorer.

7 Conclusion

Selon le domaine, nous avons montré que le terme hyper-réseau désigne quelque chose de différent. Ainsi, dans le domaine de la combinatoire, l'hyper-réseau est un objet mathématique utilisé comme modèle pour représenter des

connaissances et supporter l'exécution d'algorithmes d'apprentissage automatique par exemple. Dans le domaine des systèmes complexes, l'hyper-réseau est utilisée pour représenter, analyser et simuler des systèmes complexes. Dans le domaine de la biologie des systèmes, l'hyper-réseau permet de représenter les niveaux moléculaires, cellulaires et organisationnels d'un organisme vivant. Dans le domaine de l'apprentissage automatique et plus précisément de l'apprentissage profond, l'hyper-réseau est un réseau de neurones qui génère des poids pour un autre réseau de neurones. Les axes de développement de nouvelles méthodes nous conduisent à explorer plusieurs stratégies pour unifier les démarches évoquées précédemment dans différents domaines.

De plus, cet article a montré que plusieurs pistes de recherches en intelligence artificielle reste à développer en utilisant les hyper-réseaux des systèmes complexes.

La première piste pour des travaux futurs serait d'identifier des points d'accroche entre les différentes définitions de ces différents domaines.

Le second point serait de compléter notre outil de modélisation et d'analyse d'hyper-réseaux "Hype", au sein duquel nous avons déjà pu expérimenter les travaux d'Atkin et de Johnson, dans le cadre d'une collaboration entre le centre de R&D d'Engie (CRIGEN) et l'université Technologique de Troyes [5], avec d'autres techniques présentées dans la partie 6 de cet article.

Remerciements

Nous remercions Yolène Berrou pour son aide dans le partage de ses connaissances sur la topologie algébrique et les hyper-réseaux d'Atkin et Johnson dans la théorie des systèmes complexes ainsi que les relecteurs anonymes pour leurs commentaires sur cet article afin de l'enrichir.

Références

- [1] R.H. Atkin, R. Bray, and I. Cook. A mathematical approach towards a social science. *The Essex Review, University of Essex*, (2) :3–5, 1968.
- [2] Ron Atkin. *Multidimensional Man : Can Man Live in 3-dimensional Space ?* Penguin Books Ltd, jan 1981.
- [3] Claude Berge. *Graphs and Hypergraphs*. North-Holland Publishing Company, 1973.
- [4] Claude Berge. *Hypergraphs - Combinatorics of Finite Sets*. North-Holland Publishing Company, 1987.
- [5] Yolène Berrou, Eddie Soulier, P. Calvez, Babiga Birregah, P. Vidal, E. Dupont, B. Carballa, G. Blot, and M. Guery. Smart city development strategy profile : Use case modeling based on simplicial complexes. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, VI-4/W2-2020 :25–32, 09 2020.
- [6] David Ha, Andrew Dai, and Quoc V. Le. *Hypernetworks*, 2016.
- [7] Jeffrey Johnson. Expert q-analysis. *Environment and Planning B-planning & Design*, 17(2) :221–244, 1990.

Je relie système complexe, combinatoire, biologie des systèmes et apprentissage profond, qui suis-je ?
Hyper-réseau, définitions et cas d'usage

- [8] Jeffrey Johnson. Interpretation and hierarchical set definition in q-analysis. *Environment and Planning B-planning & Design*, 17(3) :277–302, 1990.
- [9] Jeffrey Johnson. The rules of q-analysis. *Environment and Planning B-planning & Design*, 17(4) :475–486, 1990.
- [10] Jeffrey Johnson. Hypernetworks for policy design in systems of systems of systems. volume 126, pages 115–127, dec 2012.
- [11] Jeffrey Johnson. *Hypernetworks in the Science of Complex Systems*. Imperial College Press, 01 2014.
- [12] Jeffrey Johnson. Hypernetworks : Multidimensional relationships in multilevel systems. *The European Physical Journal Special Topics*, 225 :1037–1052, 09 2016.
- [13] Sun Kim, Soo-Jin Kim, and Byoung-Tak Zhang. Evolving hypernetwork classifiers for microrna expression profile analysis. In *2007 IEEE Congress on Evolutionary Computation*, pages 313–319, 2007.
- [14] J. Ramos, R. Lopes, P. Marques, and D. Araújo. Hypernetworks reveal compound variables that capture cooperative and competitive interactions in a soccer match. *Frontiers in Psychology*, 8, 2017.
- [15] Jose L. Segovia-Juarez, Silvano Colombano, and Denise E. Kirschner. Identifying dna splice sites using hypernetworks with artificial molecular evolution. *BioSystems*, 87(2) :117–124, 2007.
- [16] Jose L. Segovia-Juarez and Michael Conrad. Hypernetwork model of biological information processing. In *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, volume 1, pages 511–515, July 1999.
- [17] Jose L. Segovia-Juarez and Michael Conrad. Learning with the molecular-based hypernetwork model. In *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, volume 2, pages 1177–1182, 2001.
- [18] Johannes von Oswald, Christian Henning, João Sacramento, and Benjamin F. Grewe. Continual learning with hypernetworks, 2020.
- [19] Byoung-Tak Zhang. Hypernetworks : A molecular evolutionary architecture for cognitive learning and memory. *Computational Intelligence Magazine, IEEE*, 3 :49–63, 09 2008.

Modelling response time and impact of instructional level of support

L. Pinos Ullauri^{1,2}, W. Van den Noortgate¹, D. Debeer¹

¹ University of Leuven, ITEC an imec research group at KU Leuven

² IMT Lille-Douai, CERI-SN

luisalberto.pinosullauri@kuleuven.be

Abstract

This study investigates several approaches to modelling the response time a person, which has been given a certain level of support, requires to solve an item. Three different generic models are proposed explaining the involved latent variables and their interactions. The goal of this paper is to present various ways to model the instructional level of support and allow the reader to choose and extend the most suitable model in a particular dataset of interest. For illustrative purposes, the models are implemented within a Bayesian Framework for a specific situation.

Keywords

response time, modelling, level of support, bayesian framework

Résumé

Cette étude examine plusieurs approches pour modéliser le temps de réponse dont une personne, qui a reçu un certain niveau de soutien, a besoin pour résoudre un item. Trois modèles génériques différents sont proposés expliquant les variables latentes impliquées et leurs interactions. Le but de cet article est de présenter différentes manières de modéliser le niveau de soutien et de permettre au lecteur de choisir et d'étendre celui qui convient le mieux dans un jeu de données d'intérêt particulier. À des fins illustratives, les modèles sont implémentés dans un cadre Bayésien pour une situation spécifique.

Mots-clés

Temps de réponse, modélisation, niveau de soutien, cadre bayésien

1 Introduction

When a student is working on a test exercise the time it takes for the student to solve that exercise is a response time. This response time is measurable and relates that particular student with that specific exercise. Generally, the terms person and item can be used to describe the student and exercise, respectively. These abstract representations can be applied in various situations. For instance, if an employee working in an assembly line requires a certain amount of time to assemble a part, this part could be described as an item and the employee as the person, while the required time a response time. Similarly, if a student is

playing a virtual or real educational escape room, the puzzles within the room can be categorised as items, the students persons, and the time for the puzzles to be solved by the students response time.

Response time is observable and relates a person p with an item i . It has been studied under psychological research and a wide variety of models have been proposed. Response times can serve as outcomes to help explain the underlying factors that are involved in the item solving process [4], which allow researchers understand and possibly improve measurement techniques. A type of response time models has been inspired by Item Response Theory (IRT) [10] [14], where a similar structure is used [16]. In this framework, response times can be explained by latent parameters related to either items or persons.

A learning environment is a system where participants can interact with exercises, puzzles, challenges or other participants. Its main objective is to foster the conditions the participants' learning. The term e-learning environment as used by [1], encompasses a wide range of applications such as web-based training, Virtual Learning Environments and massively open online courses (MOOCs), among others.

With these technological advancements it is possible to provide assistance to the learners via tips, hints, chatbots or by giving more instructions. This assistance or support can influence not only the probability of correctly answering an item, but also of the time an item demands to be solved by persons. This effect is crucial for assessing the impact and accurately predicting the response time, which can help estimate latent variables related to person and item characteristics. These latent estimates can in turn serve as input for adaptive algorithms or recommender systems in personalised learning. To the best of our knowledge there are no studies regarding the impact of the instructional level of support on response time.

In the next section, the methodology describes the typical behaviour of response time distributions, their transformations through natural logarithms, and their relation with the underlying variables of interest. Three different generic models are proposed to explain the effects of the instructional level of support on log-transformed response times.

In section 3, an analysis with both generated and real-life data is performed. A more visual approach is taken with the generated data navigating through the possibilities of the impact of the level of support in the response time

distribution. On the other hand, with the real-life data a bayesian analysis is performed with the extensions of the generic models to a specific data set.

In section 4, a discussion of the results and limitations is done, laying the ground for future work, later presented in section 5.

2 Methodology

Response time distributions usually follow positively skewed patterns such as Gamma, exGaussian or Weibull distributions. Figure 1 shows a common example of a response time distribution as it decreases its density while the time increases. [9] showed that actual log-transformed response times can be approximated by normal distributions. The logarithmic transformation drastically changes the scale. For instance, if the unit of measure of raw response times is in seconds, then 1 second would become 0 in a log-scale. The equivalent of 18.27 minutes in seconds would be 7, and the conversion of 6.11 hours to seconds would turn to 10.

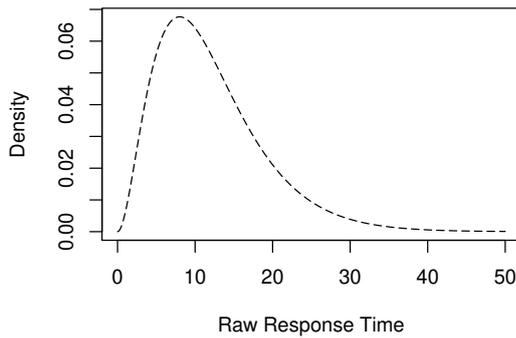


Figure 1: Common example of a raw response time distribution

Response time describes the amount of time a person p requires to solve an item i , which does not imply the accuracy or correctness of the response, but just the time. Nonetheless, the time itself can help measure latent variables or study the relation between them. These variables have particular meanings and interpretations and can generally be classified as time characteristics of items or persons.

From the item side, the time intensity can be interpreted as the time length a particular item requires to be solved, which is not the same as its difficulty. For instance, an item could be both easily solved and time-consuming. Moreover, there are certain cases, such as solving puzzles or mazes, where accuracy does not provide as much information as response time. The main person time characteristic is the speed, which relates to how fast or slow can the person be in solving items.

Following the work by [16], let us define the log response time of a particular item i and person p as a normal distribu-

tion with a mean μ_{ip} and an error variance σ_{error}^2 as shown in Equation 1. This variance would correspond to the residual differences between the predictions and actual values of log response time.

$$\log T_{ip} \sim \mathcal{N}(\mu_{ip}, \sigma_{error}^2) \quad (1)$$

In general, the difference between the time intensity and speed can explain the mean μ_{ip} . For those familiar with Item Response Theory, this subtraction is comparable with the 1 Parameter Logistic or Rasch model [14].

Another item characteristic is the time discrimination of an item i . This variable describes how some items are more or less sensitive towards variability of speed than others. In other words, if a person has a constant speed τ , instead of expecting the same linear reduction of τ for all items, it would vary depending on the item. For illustrative purposes, suppose there are two test exercises where the first one involves writing a paragraph and the second one selecting a multiple choice answer. Even if the student is quite fast, solving the first exercise will take at least the time to type or write the words, meanwhile for the second exercise, the student can just select the answer. In this case, the second exercise can be more sensitive to speed in comparison to the first one. The inclusion of time discrimination makes the model analogous with the 2 Parameter Logistic model from IRT.

The new player in these relations of variables is the instructional level of support, which can be interpreted as not belonging to either item or person time characteristics, but rather from the system side. This level of support guides the person in solving an item promoting learning. This new characteristic can be represented in a variety of ways ranging from automatic chatbots in e-learning environments to the manual inclusion of more instructional detail in assembly task training or the provision of hints in educational escape rooms. There may also be other variables valid for specific cases, although special care must be taken to interpret these variables, since these constructs can become intertwined in models.

Considering the support effect as an additional time characteristic, let us define, similarly to the models that stem from Equation 1, the log response time of a particular item i , person p and level of support l as shown in Equation 2.

$$\log T_{ipl} \sim \mathcal{N}(\mu_{ipl}, \sigma_{error}^2) \quad (2)$$

Maintaining the convention in literature by [11] [16] [8], let us define in this paper:

- λ_i as the time intensity of an item i
- ϕ_i as the time discrimination of an item i towards speed
- τ_p as the speed of a person p
- α_l as the instructional support effect of a level l

This mean μ_{ipl} can be expressed as function of characteristics from the person, item and system side as it can be

seen in Equation 3. This conveys that depending on the estimates and nature of the relation of these latent variables, the mean μ_{ipl} can be steered to the left or right moving the distribution of the log time response. Figure 2 shows an example of how the level of support can help displace the mean. For instance in the figure, the blue dashed distribution could be considered using a medium level of support, so that if there is a change to a higher level of support, the distribution would be the dotted green one. Following the same logic, if the level of support would decrease, the overall mean and distribution would turn into the red one.

$$\mu_{ipl} = f(\lambda_i, \phi_i, \tau_p, \alpha_l) \quad (3)$$

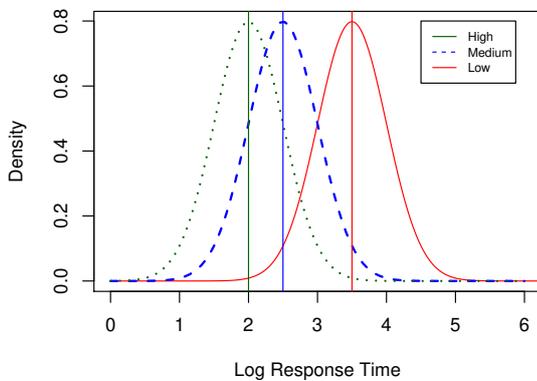


Figure 2: Example of the impact of the level of support

Taking into account the lognormal structure, let us propose three different generic models to express the relation between these latent variables with response time. The first model has a mean that describes a linear relation of the predictor variables with a sole interaction of the item discrimination with the person speed as shown in Equation 4. This sole interaction can be interpreted as the working speed for that particular item [11]. This model can serve as a benchmark to compare with the other models. There is a subtraction between the time intensity, the working speed and the level of support decreasing the value of the mean and steering the response time distribution to the left.

The effect of the level of support in this model is the same for all items and persons, which is not the same as its practical impact. For instance, a situation where the time intensity is much larger than the working speed would not be similar to a case where the difference between them is not as large. The level of support could have a bigger impact in the former case, while a modest or perhaps insignificant impact on the latter. Moreover, the effect of the change would not be the same when returning to raw scales due to the logarithmic transformation. Let us imagine an support effect with a magnitude of 0.5. If the log-response time mean is reduced from 3.5 to 3, the effect on raw response time would translate in 13.03 seconds. Meanwhile,

that same difference in a log-response time change from 5 to 4.5 would translate to 58.40 seconds.

$$\mu_{ipl} = \lambda_i - \phi_i \tau_p - \alpha_l \quad (4)$$

The second model's mean is shown in Equation 5, which describes a linear relation of the predictor variables with an interaction of the time intensity with the level of support. This interaction can be thought as the item final time-consuming characteristic. For instance, if the system offers high support, then the time intensity would be expected to be lower and with it the overall response time. Similarly, if the support effect is lowest, then the time intensity would be highest and the result from the subtraction would displace the mean to the right. In addition, the multiplicative nature of the relation would make the effect of the level of support even stronger in raw scales.

$$\mu_{ipl} = \lambda_i \alpha_l - \phi_i \tau_p \quad (5)$$

The third and final model's mean describes an overall interaction of the level of support with the subtraction of the time intensity and the working speed, as shown in Equation 6. The level of support acts as an increasing or decreasing factor depending on its level, which can steer the response time distribution to either side. The impact of the support would vary depending on the difference of the item and person variables. This means that if an item would ask much time for a given person, the effect of the support would be larger. Also, if a given person is fast enough for an item, the effect would be smaller.

$$\mu_{ipl} = (\lambda_i - \phi_i \tau_p) \alpha_l \quad (6)$$

Each of these models can be further extended by including additional predictors related to the specific problem context, characteristics of persons or items.

3 Analysis

3.1 Visualisation with Generated Data

In order to have a clearer view of the models, specifically the impact of the level of support, generated data is used to simulate raw response time distributions. A data grid is expanded using multiple combinations of plausible estimates as shown in Table 1. The combinations are a mixture of increasing arithmetic sequences and fixed values, which allow a broader visualisation of the effects of each of these variables towards raw response time. The generated data for model 1 consider a discrimination equal to 1, which is assumed for simplicity in IRT as in the 1P Model [14]. The generated data for model 2 and 3 utilises a fixed time intensity while taking into account varying sensitivities. It is important to remark that these mixture of fixed and varying values is used only for visualisation purposes in order to stress the impact of the level of support.

The first model can be visualised in Figure 3, where it shows a lattice of plots with the possible combinations of the latent variables of interest. These are raw response time density plots with different support effects (0, 0.15 and 0.3).

Model	λ	ϕ	τ	α
1	2,2.5,3	1	0.5,1,1.5	0,0.15,0.3
2	2.5	0.5,1,1.5	0.5,1,1.5	0.7,0.85,1
3	2.5	0.5,1,1.5	0.5,1,1.5	1,1.1,1.2

Table 1: Sample values used to generate artificial data

In this case, the discrimination is fixed at 1. It can be seen that when the highest level of support is given (green curves) the distributions concentrate more density near the beginning. Following the same logic, when the lowest level of support is given (blue curves) the distributions flatten decreasing their peaks and spreading probability mass. The vertical lines represent the mean of the distributions. The impact of the support effect can be visualised through the distance between the means, which becomes more significant when the speed decreases and even more so when the time intensity increases.

Similarly to the first model, a lattice of plots representing model 2 is shown in Figure 4. There are three different support effects (0.7, 0.85 and 1). Given the multiplicative nature of the interaction, the support effect of 1 shown with the green curve represents the response time with the lowest level of support while the blue curve the highest. It can be appreciated that the distance between the vertical lines (means) is more considerable when the speed and sensitivities are lowest. It can also be seen that when the speed and discrimination is highest, the overall response time means approach to zero.

A lattice of plots help visualise the behaviour of the third model in Figure 5. There are three different support effects (1, 1.1 and 1.2). In this case, the larger values of support effect refer to lower levels of support, since by increasing the overall interaction, the response time grows. The blue curve represents the highest level of support while the green curve the lowest. It can be seen that in this model, the distance between the means is wider when both the discrimination and speed are lower.

3.2 Data Set

For illustrative purposes, the models are implemented with real-life data. The data set was collected within the imec.icon project COSMO, co-partnered by imec itec KU Leuven, among others [7]. The data set consists of:

- 96 participants using VR-supported technology to train for five different assembly tasks with a total of 7161 data points.
- Each task consists from 6 to 12 steps. There are a total of 45 steps for the 5 tasks.
- The number of attempts the participants train at a certain step range from 1 to 4.
- The previous experience in AR/VR is measured with a four-point Likert-type scale and later standardised for modelling.
- The response time of the step is measured in seconds.

- 3 levels of instructional support: Low (L), Medium (M) and High (H) were implemented, and at least one of those levels was always used.

An example of the data set structure is shown in Table 2.

PartID	Step	Support	Attempt	Prev AR/VR	Response time
1	1	H	1	2	22.3
1	2	H	1	2	15.7
1	3	H	1	2	45.3
..
1	1	M	2	2	24.6
1	2	M	2	2	19.21
..
96	45	L	3	1	48.1

Table 2: Example of the dataset structure

A brief descriptive analysis of the step log response time is shown that:

- The minimum and maximum values are 1.1 and 6.7 respectively
- The mean is 3.31 and the standard deviation is 0.91

In addition, Figure 6 shows the kernel density plot of the step log response time, which visually suggests it can be approximated by a log normal distribution.

3.3 Model Extension

In this particular data set there are certain characteristics that can be included in the previous models to explain the behaviour of response time. Given the longitudinal nature of data, where several attempts were performed by persons assembling parts, an effect related to learning can be defined. This construct could be interpreted from both the person side, as the learning rate from a person in completing assembly steps, and the item side, as being a time intensity decreasing factor. Therefore, this effect is treated as a generalised fixed effect rather than individual effects specific to persons or items. In addition to the learning effect, an effect related to the previous experience with Augmented and Virtual Reality is taken into account. Following the previous logic, the previous AR and VR experience effect is considered as generalised fixed effect for all persons and items. These new effects are thus represented with γ and ρ for the learning and previous experience characteristics, respectively.

Since for this data set there was always a level of support present in the measurements, a fixed reference is needed to register the change of the level of support. In this case, the High level of instructional support is taken as overall reference for all the models, which means there are two support effects, α_1 and α_2 for Medium and Low level of support respectively. In the case of the first model, given its additive linear relation with the support effect, the High level is fixed at 0. Therefore, the values of α_1 and α_2 are expected to be greater than zero increasing the response time and moving the mean to the right. On the other hand, the

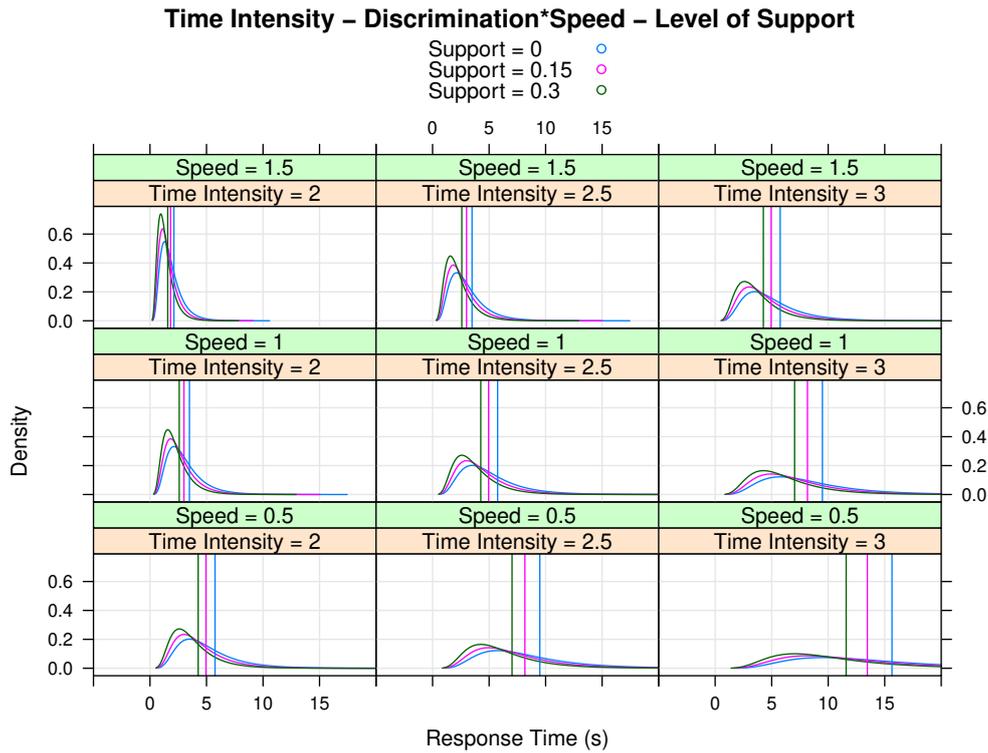


Figure 3: Lattice of Model 1 Response Time Distribution

second and third models, having multiplicative interactions with the support effect, the High level is fixed at 1. This means that the values α_1 and α_2 can be greater or smaller than 1 depending on the relation. For models 2 and 3, if the support effect values are greater than the High support reference of 1, then the response time would increase, which follows the fact the these values represent lower levels of support. Therefore, any support estimates smaller than 1 for models 2 and 3 would not be expected.

3.4 Bayesian Framework

As mentioned previously, the models are estimated through Bayesian methods, in this particular case with Stan by [2] and package RStan by [15], that employs Hamiltonian Monte Carlo to effectively provide a posterior distribution of the log-response time for steps. The number of iterations chosen for these models are 10000 with 2000 burn-in samples with 4 different chains, having a total of post warm-up 36000 samples.

The initialisation of the chains is fixed to stress reproducible results, using preliminary analysis of the models to help define sensible starting points of the variables. Considerable distance is left between the starting points of the different chains, which improves the robustness of the models.

The models suffer from identifiability issues similarly to those of IRT as described in [3]. In general, with the additive identifiability problems for the benchmark model, if a certain constant c is added and subtracted, the mean μ would not be affected. In the same way, for models 2 and 3,

if a certain constant c is multiplied and divided, there would not be a displacement of the mean μ . There are several ways to solve these issues, where the chosen one in this work is to set the mean of the speeds to 0 with the priors.

The models use both non-informative and weakly informative priors for the estimation process. Weakly informative priors help regularise and stabilise the chains accumulating probability mass in reasonable regions, meanwhile non-informative priors sparse the probability from $-\infty$ to $+\infty$ specifying no prior knowledge over the measures. As shown in Table 3, a sufficiently wide standard deviation of 100 (in log-scale which corresponds to roughly $8.64e+35$ years) is set for the support effects, learning and previous experience effects. On the other hand, the other variables ($\mu_\lambda, \sigma_\lambda$ and σ_τ) are estimated through the non-informative prior $\mathcal{U}(-\infty, +\infty)$.

Given the large number of parameters and the complex high dimensional space through which the estimation occurs, the discrimination parameter is set to 1 for all items. This paper’s goal is to stress the impact of the level of support on response time, nevertheless the readers are encouraged to take into account all of the parameters and use the estimates however they see fit.

3.5 Results

Overall the model parameters reached convergence. A visual case is shown in Figure 7, where the trace plot from Model 1 depicts the different chains that despite starting in different points eventually converge on the estimation of

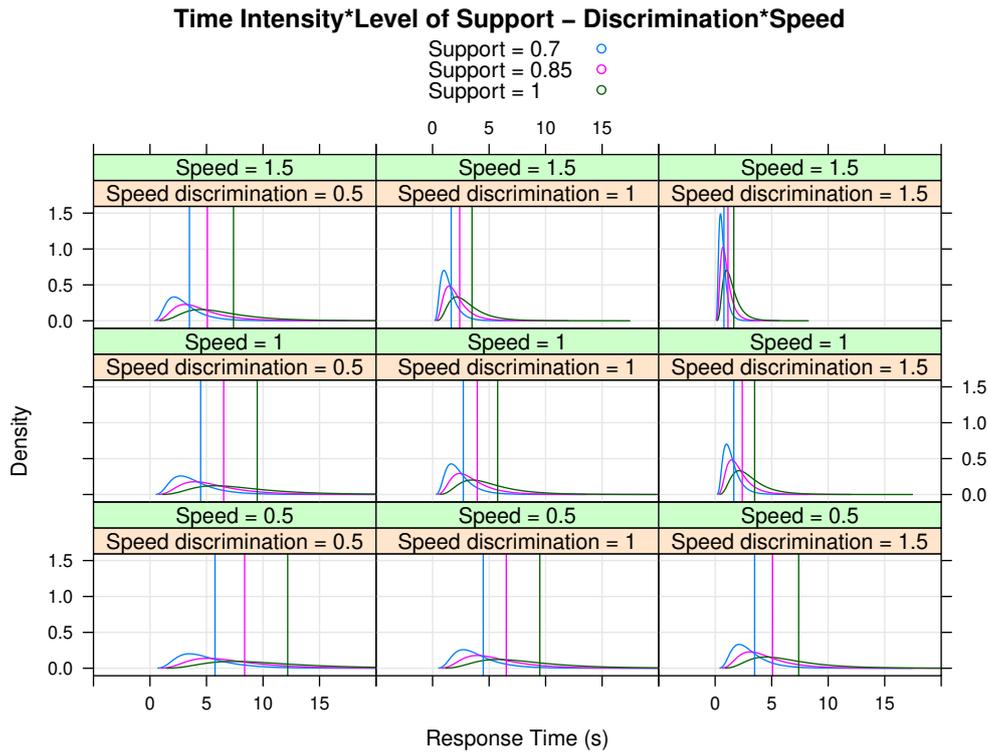


Figure 4: Lattice of Model 2 Response Time Distribution

Priors	Model 1	Model 2	Model 3
Time intensity	$\mathcal{N}(\mu_\lambda, \sigma_\lambda)$		
Speed	$\mathcal{N}(0, \sigma_\tau)$		
L. Support(Low)	$\mathcal{N}(1, 100)$	$\mathcal{N}(1.5, 100)$	
L. Support(Medium)	$\mathcal{N}(2, 100)$		$\mathcal{N}(2, 100)$
Learning rate	$\mathcal{N}(1, 100)$		
Previous Experience	$\mathcal{N}(1, 100)$		

Table 3: Weakly informative and non-informative priors chosen for this data set

the support parameters. Moreover, the empirical estimator of geometric ergodicity \hat{R} is equal to 1 in all cases, which further suggests convergence.

Table 4 show the mean estimates of the model parameters. It can be seen that in general the estimates are somewhat similar with the exception of the support effects of the benchmark model with the other models. There is not a considerable difference between the prior mean time intensity with the sample mean (3.31). On the other hand, the sample standard deviation is 0.91 while the residual standard deviation is 0.56, which means part of the variance is explained through that residual, but also probably with the variations of the parameters. The standard deviation of speed is around 0.2 standard deviations from the zero mean. The support effects show that the difference between the High and Medium level of support is not as considerable as with the low level. This can be seen taking into account the fixed references, in the case of Model 1 the High level

was fixed at 0 while for Models 2 and 3 it was fixed at 1. The learning rate has considerable effect on response time. Its effect varies depending on the number of attempts the participant repeats the step. If a participant attempted a particular step for a couple of times, the effect would be a displacement of around 0.26 in log-scales, meanwhile if the participant tried it for a third time the change would be equal to $0.26(2)=0.52$. If it happened for a fourth time, the effect would $0.26(3)=0.78$. The previous experience has an effect of approximately 0.08, however it cannot be compared with the other parameters since it was standardised prior to the estimation.

Means	μ_λ	σ_λ	σ_τ	α_1	α_2	γ	ρ	σ_{error}
Model 1	3.35	0.63	0.21	0.10	0.65	0.27	0.08	0.56
Model 2	3.37	0.61	0.20	1.03	1.18	0.27	0.08	0.56
Model 3	3.35	0.61	0.20	1.03	1.19	0.26	0.07	0.56

Table 4: Mean Estimate Results

Furthermore, in order to compare the best fitting model for this particular data set, the bayes factor is used. Bayes Factor provides a statistical way to measure the support of a model in favor of another [12]. The bridge sampling algorithm allows to iteratively calculate the bayes factor through samples of the posterior distribution [5], however it requires sufficiently large number of samples in order to create the bridge models [6].

Table 5 show the log bayes factors of the models. The convention for log bayes factor comparison dictates that if a

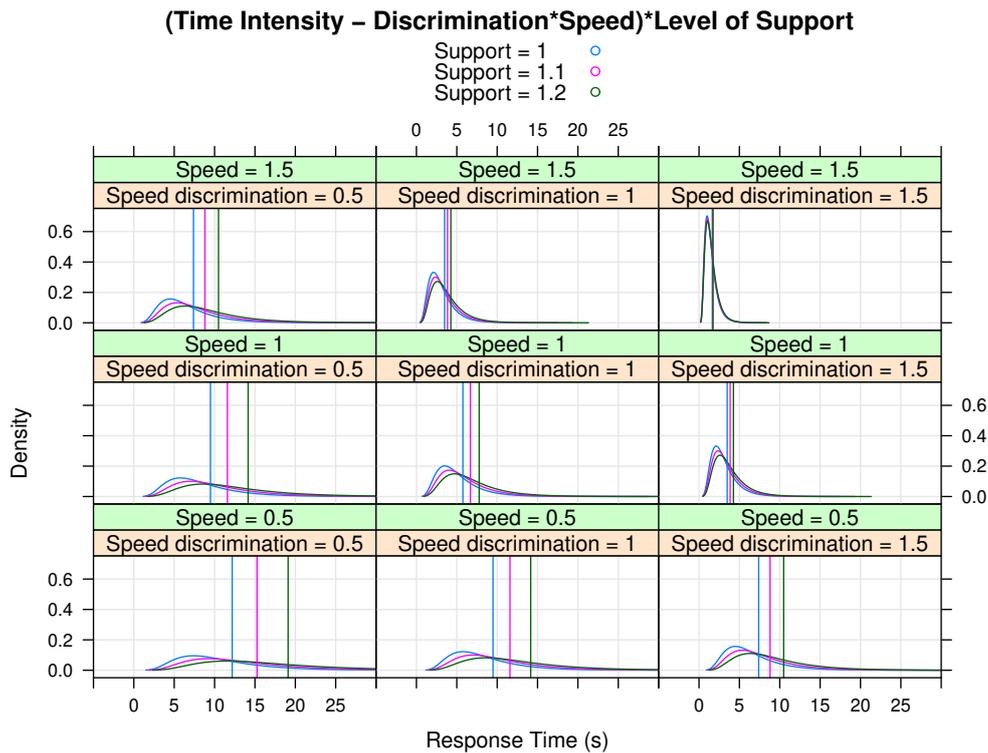


Figure 5: Lattice of Model 3 Response Time Distribution

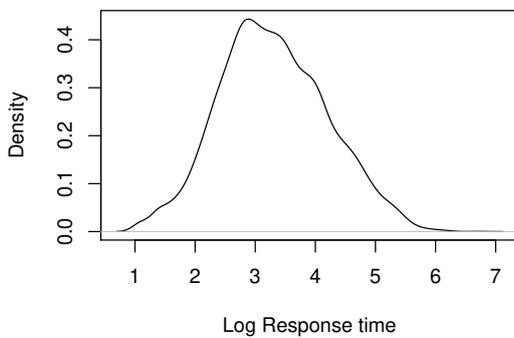


Figure 6: Log Response Time Kernel Density Plot

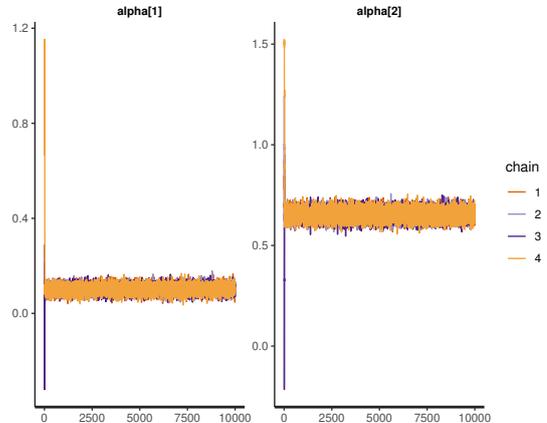


Figure 7: Trace plot support parameter Model 1

Model A has a factor larger than 2 in favor of a Model B, then Model A is preferred. In this case, the benchmark model fares better than the other two possibilities.

4 Discussion and Conclusions

Depending on the complexity and size of the data set, bayesian estimation can become a computationally expensive procedure. Moreover, by including more variables, the parameter space can grow large enough to need an immense amount of iterations to reach convergence. The models are

implemented considering an arbitrary fixed value for the discrimination of 1 for all items. This was done for simplicity in calculation as it would add 45 more parameters(one for each item in the data set) considerably increasing the necessary resources to reach convergence.

The results indicate that although all models achieved convergence, the benchmark model fits better than the other alternatives. This behaviour may not necessarily present itself with other data sets from other experiments or remain if more variables were to be included such as an individ-

Bayes Factor	Model 1	Model 2	Model 3
Model 1	-	49.22	40.59
Model 2	-49.22	-	-8.64
Model 3	-40.59	8.64	-

Table 5: Models Bayesian Factor

ual discrimination parameter for each item. Nevertheless, the goal of this paper is to explore and propose models to explain the variability of response time and impact of instructional level of support, and encourage the reader to try, implement and extend the models to other scenarios and choose the best fitting model to the data.

The applications of response time modelling are vast, being the first step towards adaptivity in learning environments. The estimation of the time a person should require to solve a particular item is the key to finding the optimal support, and the algorithm behind this decision can most certainly profit from models that explain the support effect. Works by [13] with the use of Elo-Rating system in learning environments can be adapted to include response times in order to track the growth and current speed of participants as they progress and learn. Moreover, the Elo-Rating system can provide information for personalised item selection, so that items with appropriate time intensities for the persons' current speeds are selected.

5 Future Work

It is important to remark that there are different possibilities to introduce the support effect into the models. Some assumptions that are not considered in this paper (due to the potential of increase of number of parameters, growth in complexity and computational expensiveness), but can be taken into account for future work are:

- Each person has a different speed parameter for each level of support, and that speed dimensions are correlated.
- Each item possesses a different time intensity parameter for each level of support, and that the various time intensity dimensions are correlated.

References

- [1] ul H. Anwar, George Magoulas, Jamal Arshad, Asim Majeed, and Diane Sloan. Users' perceptions of e-learning environments and services effectiveness. *Journal of Enterprise Information Management*, 31(1):89–111, 2018. Copyright - © Emerald Publishing Limited 2018; Last updated - 2021-02-19.
- [2] Bob Carpenter, Andrew Gelman, Matthew D. Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software, Articles*, 76(1):1–32, 2017.
- [3] S. McKay Curtis. Bugs code for item response theory. *Journal of Statistical Software, Code Snippets*, 36(1):1–34, 2010.
- [4] Paul De Boeck and Minjeong Jeon. An overview of models for response times and processes in cognitive tests. *Frontiers in Psychology*, 10:102, 2019.
- [5] Quentin F. Gronau, Alexandra Sarafoglou, Dora Matzke, Alexander Ly, Udo Boehm, Maarten Marsman, David S. Leslie, Jonathan J. Forster, Eric-Jan Wagenmakers, and Helen Steingroever. A tutorial on bridge sampling, 2017.
- [6] Quentin F. Gronau, Henrik Singmann, and Eric-Jan Wagenmakers. bridgesampling: An r package for estimating normalizing constants. *Journal of Statistical Software, Articles*, 92(10):1–29, 2020.
- [7] imec. Project COSMO cognitive support for manufacturing operations. <https://www.imec-int.com/en/what-we-offer/research-portfolio/cosmo>. Accessed: 2021-04-29.
- [8] Konrad Klotzke and Jean-Paul Fox. Modeling dependence structures for response times in a bayesian framework. *Psychometrika*, 84, 05 2019.
- [9] Wim J. Van Der Linden, David J. Scrams, and Deborah L. Schnipke. Using response-time constraints to control for differential speededness in computerized adaptive testing. *Applied Psychological Measurement*, 23(3):195–210, 1999.
- [10] Frederic M Lord. *A theory of test scores*. Psychometric Society, 1952.
- [11] Sukaesi Marianti, Jean-Paul Fox, Marianna Avetisyan, Bernard P. Veldkamp, and Jesper Tijmstra. Testing for aberrant behavior in response time modeling. *Journal of Educational and Behavioral Statistics*, 39(6):426–451, 2014.
- [12] Bruno Nicenboim and Shrahan Vasissth. Statistical methods for linguistic research: Foundational ideas—part ii. *Language and Linguistics Compass*, 10(11):591–613, 2016.
- [13] Radek Pelánek. Applications of the elo rating system in adaptive educational systems. *Computers & Education*, 98, 04 2016.
- [14] Georg Rasch. *Probabilistic models for some intelligence and attainment tests*. Danish Institute for Educational Research, 1960.
- [15] Stan Development Team. RStan: the R interface to Stan, 2020. R package version 2.21.2.
- [16] Wim J. van der Linden. A lognormal model for response times on test items. *Journal of Educational and Behavioral Statistics*, 31(2):181–204, 2006.

Implantable NMR Microprobe Design using a Machine learning approach.

José A. Bernardo^{1,2}, Abel Rangel Trejo¹, Lucas Werling⁴, Wilfried Uhring⁴, Luc Hebrard⁴, Christian Gontrand^{2,3}, Latifa Fakri-Bouchet^{1,3}.

jose.bernardo@etu.univ-lyon1.fr, Latifa.fakri-bouchet@insa-lyon.fr

¹ Université de Lyon, CNRS, Université Claude Bernard Lyon 1, Institut des Sciences Analytiques (ISA), UMR 5280, 5 rue de la Doua, 69100 Villeurbanne, France, ² INL (Institut de Nanotechnologies de Lyon), 69621 Villeurbanne, ³ INSA Lyon, France, ⁴ Icube ICube Laboratory, Université de Strasbourg, 67037 Strasbourg, France.

Abstract

In this work, we propose the use of an Artificial Neuronal Network (ANN) to achieve the design process of an implantable Nuclear Magnetic Resonance (NMR) microprobe. Before the micromanufacturing process, we need to optimize the microprobe i.e. micro-coil + transmission line and the connecting path (wire-bonding, underpass&vias or airbridge) to get an optimal performance (high Q -factor and Noise signal Ratio (NSR)). An ANN model is developed from the simulated dataset and used to predict the optimal microprobe performance for a working frequency and the used substrate.

Keywords

Artificial Neural Network, Machine Learning, MSE (mean square error), Matlab, Advanced Design System (ADS) Software, Nuclear Magnetic Resonance Implantable Micro-antenna, 3D-TLE (Transmission Line Extractor).

1 Introduction

One of the main challenges of the Nuclear Magnetic Resonance (NMR) technique is, to improve the sensitivity and spectral resolution in the case of weak metabolites concentration in a small region of interest (ROI $\sim 2\mu\text{L}$ - $3\mu\text{L}$). One way to increase this sensitivity is, to miniaturize the NMR receiver. Thus, the micro-coil part will be adapted to the sample dimension that improves the Signal to Noise Ratio (SNR) performance criteria.

The NMR sensitivity has been the subject of numerous studies for years. Thus, several approaches were already proposed for its optimization: use of higher field [1], and improvement of the sensitivity of the Radio Frequency (RF) coil receiver and transmitter (Microprobe) [2]. In that case, simulation work to optimize the microprobe is required before the manufacturing process. This work remains complex because several parameters must be taken into account to obtain an accurate microprobe Q factor and SNR evaluation (multi-parameters problem). It requires the use of several software, a long computational time. Nowadays artificial intelligence (AI) has been widely used in several application areas, more particularly artificial neural networks (ANN). Recently, they have been used to solve complex problems such as antenna parameters

extractions, structural design and data interpretation for radar, and Multiple-Input and Multiple-Output (MIMO) systems [3], also for optimizing the gain and directivity of a microstrip antenna [4]. ANN are extremely useful in modelling problems where the relationship between inputs of the system is not easily modelled. The advantages of ANN assisted antenna modelling system is reducing computational complexity, saving computational time, decreasing the cost compared to traditional computational modelling, and lead to highly accurate data set analysis [5]. All these advantages represent a great asset to deal with the complexity of our microprobe structure design and expected performances optimization [6][7]. ANN approach helps to establish a relationship between several microprobe geometric parameters, material properties and the working frequency with the microprobe performances parameters (resistance R and Q factor). Then, it makes easy the generalization of microprobe optimization problem. In this work, we demonstrate how, ANN succeed predicting the microprobe performances according to its geometrical parameters, material properties and working frequency. Thanks to the microprobe, generated Data from simulations, we proposed an accurate model.

2 Materials and methodology

2.1 Data generation

A trained ANN model requires a large dataset quantity corresponding to inputs data and their target data (outputs). In our case, the microprobe dataset was generated performing a numerical simulation of microprobe Resistance and Q factor

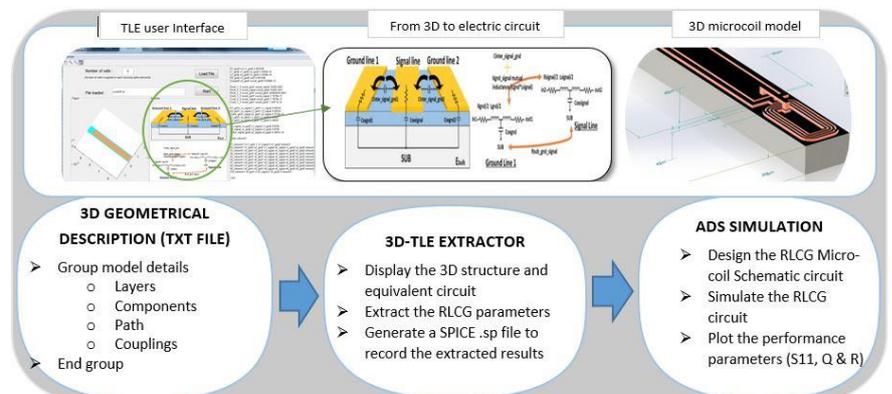


Fig. 1. Block diagrams of the simulation methodology: TLE user interface to 3D micro-coil model

thanks to 3D-TLE (Transmission Line Extractor) platform, and ADS (Advanced Design System) Keysight software. First, we simulated the micro-coil part, where four geometries were designed. Then 3D-TLE, homemade software helped to extract the transmission line electrical parameters and then propose an equivalent electrical circuit of the complete microprobe.

The electrical performances of the microprobe are obtained through the electrical circuit simulated using ADS. By that, we deduce the microprobe resistive losses. Figure 1, summarizes by blocks diagrams the steps from the 3D-TLE platform to ADS software simulations. Thanks to this, we generated 6528 data samples, used for training, validations and test set. The microprobe dataset includes four micro-coil geometries (Rectangular, Ellipsoidal, square and circular), eight working frequency (from 200 MHz to 900 MHz), and two substrate types (glass and silicon).

2.2 ANN Modelling

The Artificial Neuronal Network (ANN) was conceived using the MATLAB Toolbox, Deep Learning (DL). In figure 2, we show the ANN shallow architecture. Note that, between the input layer (Predictors) and the output layer (Targets), there is only one hidden layer that we optimized making trial testing. The data sample was divided into three sets: 70% for training,

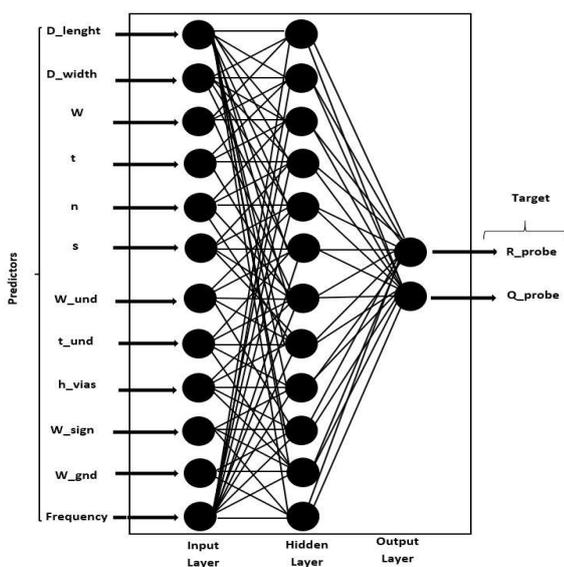


Fig. 2. ANN architecture for Microprobe dataset training model

15% for validation and 15% for test usage. To train our model, we used a custom-made script to automatically test a different number of the neuron from the hidden layer. That script allows choosing the optimal amount of neuron from the hidden layer in terms of MSE (mean square error) Test dataset performance.

3 Results and Conclusions

Figure 3 shows the plot of the microprobe Q-factor from the prediction of the ANN model and the one from the true data value. We can note that the prediction and true data value are in good concordance, which means that we can predict accurately the Q factor of the microprobe, based on its geometrical

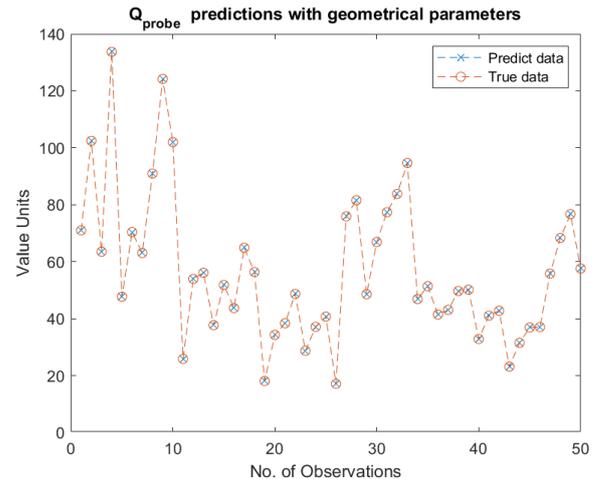


Fig. 3. The microprobe Q factor prediction and the true value

parameters, working frequency and materials properties.

As it is shown, our artificial intelligence algorithms (ANN) were validated to predict well (RMSE = 0.0316 Test dataset), the Q - factor of the microprobe even for a dataset of 6528 samples, which correspond to 99.8% for accuracy of the model prediction in term of to the Test performance.

Acknowledgement

The authors would like to thank the National Research Agency (ANR) for financial support through the ANR-16-CE19-0002-01 project and Dr Y. Zaim Wadghiri from *Grossman School of Medicine, New York University*, for his help in simulating micro-coils using CST MWS® software.

References

- [1] D. I. Hoult *et al.*, “The signal-to-noise ratio of the nuclear magnetic resonance experiment,” *Journal of Magnetic Resonance (1969)*, vol. 24, no. 1, pp. 71–85, Oct. 1976.
- [2] R. R. Ernst *et al.*, “Application of Fourier Transform Spectroscopy to Magnetic Resonance,” *Review of Scientific Instruments*, vol. 37, no.1, pp.93–102, Jan. 1966.
- [3] K. Sharma and G. P. Pandey, “Designing a Compact Microstrip Antenna Using the Machine Learning Approach,” *JTIT*, vol. 4, pp. 44–52, Jan. 2021.
- [4] M. Sağık *et al.*, “Optimizing the Gain and Directivity of a Microstrip Antenna with Metamaterial Structures by Using Artificial Neural Network Approach,” *Wireless Pers Commun*, Jan. 2021.
- [5] D. Erricolo *et al.*, “Machine Learning in Electromagnetics: A Review and Some Perspectives for Future Research,” in *2019 International Conference on Electromagnetics in Advanced Applications (ICEAA)*, Granada, Spain, Sep. 2019, pp. 1377–1380.
- [6] T. Truong-Cong, “Optimisation par approche physique des micro-antennes RMN fabriquees par Techniques Microelectroniques : Etude Theorique et Experimentale,” PhD Thesis, INSA de Lyon, 2014.
- [7] J. T. Rosillo, “Contribution à l’amélioration de la sensibilité d’un micro-récepteur RMN implantable,” Ph.D Thesis, Université Claude Bernard - INSA Lyon, 2014.

L'habitat intelligent et sa capacité à s'adapter: vers une approche centrée multi-résidents

C. Bout^{1,2}, A. Karami¹, A. Fleury²

¹ FGES, Université Catholique de Lille, Smart and Sustainable Cities, 59016 Lille, France.

² IMT Lille Douai, Institut Mines-Télécom, Université de Lille, CERI SN, F-59000 Lille, France.

clement.bout@univ-catholille.fr, Abir.Karami@univ-catholille.fr, Anthony.Fleury@imt-lille-douai.fr

Résumé

Cet article présente les travaux qui serviront de base à la réalisation d'un modèle d'habitat intelligent adaptatif multi-résidents centré utilisateur. Nous y présentons en premier lieu le contexte qui nous amène à proposer ce projet, puis nous dressons un état de l'art des travaux reliés à notre projet. Nous concluons enfin sur le positionnement de notre démarche par rapport à l'existant ainsi que sur une discussion sur les perspectives en cours et envisagées.

Mots-clés

Habitat intelligent, environnement intelligent, approche centrée utilisateur, pattern mining, multi-résident.

Abstract

This paper presents the work addressing the realization of a multi-user-centered adaptive smart home model. We first present the context and the state-of-the-art around the thematic of the project. Finally, we conclude with the positioning of our approach in relation to existing work and a discussion of current and planned perspectives.

Keywords

Smart home, intelligent environment, user-centered approach, pattern mining, multi-resident.

1 Introduction

Les habitats intelligents ont vu croître leur intérêt ces dernières années. Initialement destinés à décrire les bâtiments ayant des architectures et méthodes de constructions dites intelligentes, le périmètre du bâtiment intelligent s'est élargi à la suite des avancées dans les domaines de l'électronique, des technologies de l'information et de la communication. L'accessibilité grandissante à internet, aux capteurs et actionneurs a permis la démocratisation de certains usages comme le pilotage centralisé des chauffages, des éclairages ou encore l'assistance à la personne.

En parallèle, nous observons un vieillissement de nos populations. L'Organisation Mondiale de la Santé (OMS) estime que 25% de la population européenne seront âgés de plus de 65 ans en 2050 (14% en 2010). En France, 96% des Hommes et 93% des Femmes de plus de 65 ans vivraient à domicile (INSEE, 2016). Même si les avancées en ma-

tière de médecine permettent de maintenir une personne en meilleure santé plus longtemps, l'âge s'accompagne souvent d'une perte d'autonomie. Cependant, le maintien à domicile semble limiter les déclinés cognitifs.

Nos travaux s'inscrivent dans ce contexte en proposant une étude sur l'habitat intelligent et sa capacité à s'adapter. Nous nous intéressons plus particulièrement aux modélisations de comportements intra et inter utilisateurs ainsi qu'aux systèmes d'apprentissage par renforcement. Ces travaux se placent dans le domaine de l'Assisted Living et visent à proposer un nouveau système permettant au logement de s'adapter aux besoins de plusieurs utilisateurs. Pour ce faire, nous devons prendre en compte les retours implicites et explicites de ceux-ci afin d'adapter la prise de décision. Les problématiques soulevées au travers de cette démarche sont :

- Estimer les profils de comportement
- Adapter le comportement du système aux utilisateurs
- Adapter le système à un environnement dynamique
- Traiter l'ambiguïté des retours
- Aborder la problématique du cold start

La suite de ce papier est composée d'une présentation des travaux de références, suivi du positionnement de la problématique par rapport à ces travaux puis des perspectives et des premiers résultats des travaux en cours.

2 État de l'art et problématique

L'adaptation intelligente des habitats à leurs utilisateurs a déjà été abordée dans la littérature dans le cadre d'un domicile avec un seul occupant. Le centre de recherche CASAS de la WSU est pilote dans les avancées en système adaptatif et notamment dans les smart home. Dans le cadre d'un projet de CASAS, Rashidi et Cook [4], présentent un modèle de découverte de modèles d'activités, ou Activity Discovery (AD), prenant en compte les retours explicites d'un utilisateur. La force de ce modèle réside dans son abstraction de la représentation des activités. Les auteurs proposent une architecture composée de deux grandes parties : la première étant la découverte non-supervisée de modèles d'activité, la seconde, le choix et l'apprentissage de l'action à réaliser à l'aide ou non d'un retour utilisateur. Ainsi, au bout de plusieurs itérations (jours de vie) le système est capable d'identifier des besoins et des patterns chez l'habi-

tant.

Plus récemment, la revue de Ali et al. [1] expose un panorama de la recherche sur les approches centrées utilisateur dans le développement de logements intelligents et la problématique de transfert de l'apprentissage ainsi qu'une proposition pour fournir un habitat adapté à son utilisateur en intégrant l'utilisateur dans le processus de développement. Pour l'instant, l'architecture adaptative du modèle repose sur l'architecture SEArch (Smart Environments Architecture) [2]. Elle est basée sur le système LFPUBS (Learning Frequent Behaviors of the Users in Intelligent Environments), associant règles et modèles de comportement.

Pour permettre l'adaptation du logement à un ou plusieurs utilisateurs, il est nécessaire d'identifier leurs habitudes. Différentes approches sont possibles : l'activity recognition (AR), l'activity discovery (AD) [4]. L'AD est soumis aux problèmes de cold-start et l'AR dépend de la quantité de données disponibles en amont. Les deux peuvent être combinées pour améliorer l'estimation des profils utilisateurs. Une approche proposée actuellement est d'intégrer l'utilisateur dans le processus de développement du système [1]. Celle-ci permettrait de réduire les problèmes de cold-start du modèle. Cette approche est chronophage et nécessite de produire un système par utilisateur.

Ces approches s'accompagnent généralement d'un système de décision basé sur les habitudes de l'habitant et se focalisent sur un seul habitant. Différentes méthodes proposent de suivre chaque utilisateur et d'identifier des préférences propres à chaque habitant. Nous cherchons en premier lieu à proposer une approche basée sur l'analyse de séquences et de pattern dans l'environnement intelligent pour apprendre des comportements globaux indépendamment des utilisateurs. L'objectif est d'utiliser ces patterns comme états observés et l'activité des actionneurs présents dans ces patterns comme retour implicite pour l'apprentissage par renforcement.

3 Perspectives

Afin d'étudier la faisabilité d'un tel système, nous travaillons sur une base de données du projet CASAS [5], nommée « Kyoto ». Dans cette base, 26 couples d'utilisateurs ont été amenés à réaliser un ensemble de tâches prédéfinies dans un environnement équipé de capteurs : de présence, d'ouverture de porte et d'utilisation d'objet, les données capteurs sont événementielles. Les tâches peuvent être réalisées par l'un des résidents (ex. remplir le distributeur à médicaments), ou en coopération entre les deux (ex. déplacer un canapé).

La Figure 1 permet de visualiser un ensemble de représentations spatio-temporelles des activations de capteurs pour la réalisation d'une tâche. Chaque séquence correspondant à une expérimentation réalisée par différents sujets, le décalage temporel des débuts de séquences n'est pas un marqueur de différenciation. Cependant, la séquence d'activation permet de mettre en avant un comportement différent entre utilisateurs effectuant la même tâche.

En parallèle, nous souhaitons intégrer un modèle d'appren-

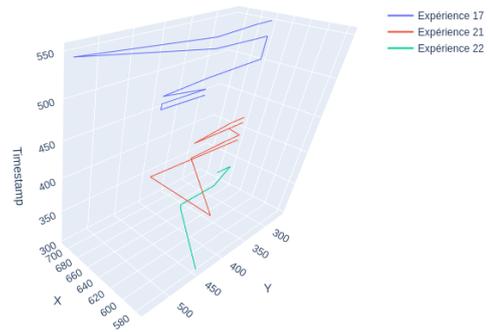


FIGURE 1 – Représentation spatio-temporelle de séquences d'activation de capteurs de présence pour la réalisation d'une tâche par différents utilisateurs.

tissage par renforcement basé sur le framework TAMER [3]. Cette méthode est utilisée pour apprendre la réalisation de tâches à des agents à l'aide du feedback positif/négatif d'un tuteur. Nous prévoyons d'utiliser les séquences d'activation comme retour implicite des utilisateurs.

Remerciements

Ce travail bénéficie du soutien de l'Agence Nationale de la Recherche, via le programme doctoral AI@IMT pour une IA responsable dans la société et dans l'industrie.

Références

- [1] S. M. Murad Ali, Juan Carlos Augusto, and David Windridge. A Survey of User-Centred Approaches for Smart Home Transfer Learning and New User Home Automation Adaptation. *Applied Artificial Intelligence*, 33(8) :747–774, 2019. Publisher : Taylor & Francis _eprint : <https://doi.org/10.1080/08839514.2019.1603784>.
- [2] Juan Carlos Augusto, Mario Quinde, Jose Gines Gimenez Manuel, S M Murad Ali, Chimezie Leonard Oguego, and Carl James-Reynolds. The SEArch Smart Environments Architecture. In *2019 15th International Conference on Intelligent Environments (IE)*, pages 60–63, Rabat, Morocco, June 2019. IEEE.
- [3] W. Bradley Knox and Peter Stone. TAMER : Training an Agent Manually via Evaluative Reinforcement. In *2008 7th IEEE International Conference on Development and Learning*, pages 292–297, Monterey, CA, August 2008. IEEE.
- [4] P. Rashidi and D.J. Cook. Keeping the Resident in the Loop : Adapting the Smart Home to the User. *IEEE Trans. Syst., Man, Cybern. A*, 39(5) :949–959, September 2009.
- [5] Geetika Singla, Diane J. Cook, and Maureen Schmitter-Edgecombe. Recognizing independent and joint activities among multiple residents in smart environments. *J Ambient Intell Human Comput*, 1(1) :57–63, March 2010.

Apprentissage de règles pour la détection de fraudes

M. Collery^{1,2}, P. Bonnard¹, F. Fages²¹ IBM France Lab² Inria Saclay Ile-de-France

marine.collery@ibm.com

Résumé

Dans le domaine bancaire, la détection de fraudes doit être explicite et le modèle de décision compréhensible. Cela justifie un apprentissage direct de règles. CN2 et RIPPER sont comparés sur différents datasets.

Mots-clés

Apprentissage de règles, détection de fraudes, explicabilité, règles métier.

Abstract

In financial environment, fraud detection must be explainable and the decision model understandable. This justifies a rule-based learning approach. CN2 and RIPPER are compared on different datasets.

Keywords

Rule learning, fraud detection, explainability, business rules.

1 Introduction

Les fraudes bancaires ont aujourd'hui un impact financier important et nécessitent d'être détectées au plus vite. Ces difficultés s'accompagnent de complexités additionnelles dues au caractère temporel et très déséquilibré des données ainsi qu'à l'évolution constante de patterns ou motifs de fraudes. De plus, dans ce domaine particulier non seulement les décisions doivent pouvoir être expliquées, mais le modèle derrière la prise de décision doit être compréhensible. Les modèles d'apprentissage dits "black boxes" même expliqués à posteriori, ne peuvent donc être envisagés. C'est pourquoi, l'utilisation d'un langage symbolique, via un apprentissage supervisé de règles de décision est privilégiée.

$$\begin{aligned} & \text{if } \text{oldbalanceorigin} - \text{amount} - \text{newbalanceorigin} = 0 \\ & \text{and } \text{newbalanceorigin} = 0 \text{ then } \text{fraud} = \text{true} \quad (1) \end{aligned}$$

L'objectif de ce projet est d'induire un ensemble de règles métier dites *business rules* (exemple Eq. 1), à partir de données labellisées (exemple partiel de données inspiré de *Synthetic Financial Datasets For Fraud Detection* [1] (SFD) ci-dessous) et de modèles de machine learning. Ceci permettra d'avancer dans la détection et prédiction de fraudes dans le contexte challenging de la finance et du monde bancaire où une complète transparence est nécessaire.

amount	type	old balance origin	new balance origin	fraud (label)
5324	payment	5324	0	true
695	cashout	6870	6175	false

2 Détection de fraudes bancaires

La détection de fraudes est le problème d'apprentissage qui vise à classer automatiquement des événements comme frauduleux ou honnêtes. Dans le cadre d'un travail de recherche, la limitation principale réside dans le fait qu'il n'y a pas de données réelles disponibles publiquement pour étudier la détection de fraudes [12]. Dans le cadre d'une application réelle, ce sont les contraintes opérationnelles qui prennent le dessus. Comme relevé par [12], dans ce cas, le choix de la technique d'apprentissage dépend plus de contraintes techniques issues d'obligations opérationnelles et computationnelles que de contraintes techniques liées aux données.

3 Induction de règles

Comme expliqué par J. Fürnkranz dans [9], il existe deux familles principales de méthodes pour induire des ensembles de règles : l'extraction de règles à partir d'arbre de décision (CART [2], C4.5 [13]) et le recouvrement séquentiel, c'est-à-dire l'apprentissage de règles à partir de données (CN2 [3], RIPPER [4]). Des extensions et de nouvelles approches basées sur ces modèles ont été proposées depuis. Nous pouvons mentionner FURIA, extension de RIPPER, qui construit des conditions floues ("fuzzy") apportant de la flexibilité aux règles, au détriment d'une augmentation du temps d'apprentissage [10]. Sood et al. présente en 2020, BiLevCSS, une approche basée sur deux phases d'apprentissage rassemblant règles d'association et RIPPER qui surpasse en terme de justesse (*accuracy*) les modèles existants [14]. Les algorithmes sur lesquels nous nous concentrons sont CN2 et RIPPER.

CN2 [3]. Cet algorithme a pour idée générale de créer une nouvelle règle tant que les données d'entraînement ne sont pas toutes couvertes par les règles trouvées.

RIPPER [4]. Cet algorithme dont la particularité est de proposer une étape de post-processing des règles, est toujours aujourd'hui l'état de l'art pour l'apprentissage de règles génériques. Il présente certaines limites en ce qui concerne la complexité du langage des règles générées. Plus

le nombre de conditions possibles est élevé, plus l'apprentissage sera long. La génération de conditions comportant des agrégats ou des structures plus complexes est donc compliquée dans un temps raisonnable.

4 Expériences et résultats

Framework. Pour mener à bien ce projet, un environnement de travail spécifique basé sur des conteneurs est mis en place. Il comprend des modules distincts permettant :

- le tracking des expériences (MLFlow)
- l'apprentissage de modèles (Orange [7], scikit-learn [11], module personnalisés etc),
- l'utilisation d'un langage de règle commun (module R2L implémenté pour le projet)
- une utilisation interactive (Jupyter Notebook)

Données. Avant toutes expérimentations sur des données réelles, un dataset open-source et fictif de données bancaires intitulé : *Synthetic Financial Datasets For Fraud Detection* (SFD) [1] est choisi. Ce dataset comprend 6 millions d'observations dont 8213 transactions frauduleuses ($\approx 1\%$). Découpage training/testing choisi : 10/90%. Pour valider nos résultats sur des datasets de références, 9 datasets complémentaires disponibles dans la base de données UCI [8] sont sélectionnés (*iris*, *adult*, *wine*, *heart disease cleveland* (Robert Detrano, M.D., Ph.D.), *breast cancer wisconsin*, *car evaluation*, *abalone*, *forest fires* [6], *wine quality (red & white)* [5]). Seuls les datasets adaptés pour la classification sont considérés dans la suite de ce document. Découpage training/testing choisi : 67/33%. Les premières expériences consistent à observer quelles sont les performances des algorithmes de l'état de l'art sur ces datasets.

Pre-processing. Des premiers tests sont réalisés sans et avec une étape de pré-processing manuel des données pour SFD. Celle-ci se révèle être indispensable, un module automatique de *feature generation* et *selection* est créé avec un accent particulier sur la conservation de la compréhensibilité des règles. Ce module est basé sur une analyse dimensionnelle des données et sur l'information mutuelle apportée avec la classe objectif.

Premiers résultats. De manière générale RIPPER dépasse CN2, comme le montrent les métriques présentées dans la Table ci-dessous. De plus, le nombre de règles générées est comparable pour la plupart des datasets, excepté pour *adult* où CN2 a généré 916 règles contre 35 pour RIPPER. Cependant une analyse plus poussée sur la qualité de l'ensemble des règles générées ainsi que sur la complexité des règles est nécessaire. Moyennant un coût computationnel, la qualité des résultats pourraient être améliorée en modifiant le protocole expérimental avec de la validation croisée. Par ailleurs, travailler à partir d'un dataset synthétique généré avec des règles connues (avec et sans ajout d'erreurs) permettrait d'évaluer plus précisément la qualité de l'apprentissage.

Remerciements

Ce travail a été soutenu par IBM France Lab / PSPC AIDA 2019-PSPC-09.

dataset	model	acc	bal_acc	f1	precision	recall
raw SFD	cn2	0.999	0.798	0.691	0.822	0.596
raw SFD	ripper	1	0.873	0.839	0.956	0.748
processed SFD	cn2	1	0.994	0.993	0.997	0.988
processed SFD	ripper	1	0.998	0.996	0.997	0.996
car evaluation	cn2	0.816	0.52	0.828	0.884	0.816
car evaluation	ripper	0.837	0.699	0.855	0.874	0.837
breast cancer wisconsin	cn2	0.65	0.65	0.788	1	0.65
breast cancer wisconsin	ripper	0.965	0.959	0.965	0.965	0.965
heart disease cleveland	cn2	0.515	0.318	0.525	0.567	0.515
heart disease cleveland	ripper	0.556	0.478	0.574	0.619	0.556
wine	cn2	0.889	0.914	0.889	0.902	0.889
wine	ripper	0.933	0.929	0.933	0.934	0.933
adult	cn2	0.782	0.747	0.839	0.934	0.782
adult	ripper	0.843	0.815	0.857	0.884	0.843
iris	cn2	0.98	0.979	0.98	0.981	0.98
iris	ripper	0.98	0.979	0.98	0.981	0.98

Références

- [1] Synthetic Financial Datasets For Fraud Detection. Library Catalog : www.kaggle.com.
- [2] L. Breiman et al. *Classification and Regression Trees*. Taylor & Francis, January 1984.
- [3] P. Clark and T. Niblett. The CN2 Induction Algorithm. *Machine Learning*, 3(4) :261–283, March 1989.
- [4] W. W. Cohen. Fast Effective Rule Induction. In *In Proceedings of the Twelfth International Conference on Machine Learning*, pages 115–123. Morgan Kaufmann, 1995.
- [5] P. Cortez et al. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4) :547–553, November 2009.
- [6] P. Cortez and A. Morais. A Data Mining Approach to Predict Forest Fires using Meteorological Data. page 12, 2007.
- [7] J. Demšar et al. Orange : Data mining toolbox in python. *Journal of Machine Learning Research*, 14 :2349–2353, 2013.
- [8] D. Dua et al. UCI machine learning repository, 2017.
- [9] J. Fürnkranz et al. *Foundations of Rule Learning*. Springer Science & Business Media, November 2012.
- [10] J. Hühn et al. FURIA : an algorithm for unordered fuzzy rule induction. *Data Mining and Knowledge Discovery*, 19(3) :293–319, December 2009.
- [11] F. Pedregosa et al. Scikit-learn : Machine learning in Python. *Journal of Machine Learning Research*, 12 :2825–2830, 2011.
- [12] C. Phua et al. A Comprehensive Survey of Data Mining-based Fraud Detection Research. *Computers in Human Behavior*, 28(3) :1002–1013, May 2012. arXiv : 1009.6119.
- [13] J. R. Quinlan. *C4.5 : Programs for Machine Learning*. Elsevier, June 1993.
- [14] N. Sood et al. Bi-Level Associative Classifier Using Automatic Learning on Rules. In *Database and Expert Systems Applications*, pages 201–216. Springer International Publishing, 2020.

Digitalisation de la déconstruction sélective : simulation et optimisation des filières

C. Juvigny^{1,2}, J. Baste², G. Lozenguez, A. Doniec¹, L. Jourdan²¹

¹ IMT Lille Douai, CERI Système numérique, 59650 Villeneuve-d'Ascq, France

² Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRISTAL, F-59000 Lille, France

Résumé

Le domaine de la déconstruction des bâtiments produit un grand nombre de déchets industriels qui ne sont, pour l'heure, que peu valorisés, alors qu'ils pourraient être soit recyclés, soit directement réutilisés dans d'autres chantiers. L'objectif de cette étude est de fournir un outil permettant de modéliser et d'optimiser les différents flux de matériaux issus des déconstructions pour une meilleure réutilisation de ces derniers, ou le cas échéant un meilleur recyclage. Pour cela, des techniques de modélisations à base d'agents seront utilisées afin de visualiser les dynamiques des différents acteurs de la filière, ainsi que des techniques de méta-heuristiques afin d'optimiser l'ordonnancement des actions de ces acteurs.

Mots-clés

Modélisation multi-agents, Recherche opérationnelle, Déconstruction sélective

Abstract

The field of building deconstruction produces a large amount of industrial waste which are not currently recovered. It represents a huge waste of resources as a large part of this waste could be either recycled or directly reused in other construction sites. The objective of this study is to provide a tool to model and optimize the different material flows from deconstruction for better reuse and recycling. For this purpose, agent-based modeling techniques will be used in order to visualize the dynamics of the different actors of the sector, as well as metaheuristic techniques in order to optimize the scheduling of the actions of these actors.

Keywords

Agent-based model, Operations research, Selective deconstruction

1 Introduction

Le domaine du BTP est le principal générateur de déchets en France (70% de la production annuelle totale en France en 2015, soit 228 millions de tonnes). Plus particulièrement le secteur du bâtiment représente 46 millions de tonnes de déchets annuellement. La loi sur la transition énergétique de 2015 ambitionne un taux de recyclage de 70% de ces déchets industriels. On est à l'heure actuelle bien loin de

cet objectif. En effet, l'Agence de la transition écologique (ADEME) estime que seuls 35% de ces déchets sont valorisés, alors même que des filières de valorisation existent dans la majorité des cas. Ainsi, l'ADEME a produit en 2016 le rapport DÉMOCLÈS qui a étudié les causes de la non-valorisation des matériaux issus des déconstructions. Les deux grandes causes révélées dans ce rapport sont d'une part la méconnaissance par les maîtres d'ouvrage du coût de la gestion des déchets et d'autre part une très mauvaise coordination des nombreux acteurs du domaine.

Une modélisation à base d'agents est adaptée pour étudier ce genre de système complexe, définie par un grand nombre d'acteurs interagissant ensemble selon leurs intérêts propres [2] [1]. Ils ont notamment été utilisés afin de simuler les flux de ciments Portland en Chine et au Chili [8] ou encore pour simuler les chaînes d'approvisionnement et de recyclage des granulats de béton aux Pays-Bas [11].

De plus un certain nombre de chercheurs ont couplé les modèles basés agents avec des méthodes issues de la recherche opérationnelle [3] comme des approches par méta-heuristiques, en particulier dans le cadre des problèmes d'ordonnancement [5]. Ces méthodes permettent d'obtenir un ordonnancement final de bonne qualité tout en gardant un temps de calculs raisonnable.

Dans cet article, nous présenterons dans un premier temps la problématique ainsi que son cas d'étude, puis nous proposerons les méthodes envisagées pour y répondre.

1.1 Problématique

Un premier enjeu consiste à pouvoir modéliser de façon efficace ce problème. Un deuxième enjeu est de pouvoir, à partir de la modélisation, simuler les actions de chacun des acteurs en un temps raisonnable. Enfin, un troisième enjeu réside dans l'optimisation des actions et des interactions des différents acteurs dans un objectif de diminution des coûts financiers et environnementaux.

1.2 Cas d'étude

Dans le cadre de notre étude, nous nous intéressons spécifiquement à la filière de la déconstruction sélective dans la région Haut-de-France. Un certain nombre d'acteurs du domaine de la déconstruction sélective nous apporte leurs expertises afin de nous aider à affiner nos modèles pour les rendre les plus réalistes possibles. En particulier un certain nombre de chantiers considérés comme représentatifs des

différentes configurations de chantiers par ces acteurs sont utilisés comme des chantiers tests. Ces derniers nous permettent d'obtenir des données suffisantes pour tester nos modèles.

L'objectif est de fournir, à terme, à nos partenaires un outil de modélisation permettant d'optimiser leurs propres actions au niveau du territoire, de façon à diminuer les coûts financiers et écologiques des déconstructions, tout en se mettant en conformité avec les nouvelles réglementations et législations.

2 Optimisation par simulation multi-agents

Au vu de la nature du problème, il est envisagé de réaliser une modélisation à l'aide de systèmes multi-agents. En effet, comme l'ont montrées des études précédentes [8] [11], ce type de modélisation est adapté pour représenter des systèmes complexes possédant un grand nombre d'acteurs évoluant simultanément dans un même espace. À partir de cette modélisation, il est envisagé d'optimiser l'ordonnement des actions des agents à l'aide de métaheuristiques [5].

2.1 Modèle multi-agents

La filière de la déconstruction sélective comporte un grand nombre d'acteurs aux compétences variées. Chacun de ces acteurs possède des objectifs et des contraintes qui leurs sont propres ainsi que des moyens d'agir sur le système bien déterminés. Une entreprise ou une organisation peut être décomposée en unités opératives, c'est-à-dire une unité à même de réaliser des opérations pour le compte de l'entreprise, elles-mêmes composées d'une certaine quantité de ressources (main-d'œuvre, équipement, savoir-faire ...).

Chaque unité opérative peut être modélisée par un agent qui enregistrera les capacités opératives de l'unité [11] ainsi que ses limitations. L'aide de nos partenaires du secteur nous permettra de définir aux mieux ces capacités et ces contraintes en fonction de leur expérience sur le terrain. Certains de ses agents peuvent être mobiles (unité de déconstruction, transporteurs de matériaux) tandis que d'autres seront fixes (usines de recyclage, entrepôts...).

Enfin, chacun des bâtiments impliqués dans la déconstruction sélective peut être lui aussi modélisé par un agent. Ces agents d'un type particulier enregistreront les matériaux disponibles (et leurs quantités restantes), le coût de ses matériaux, etc...

Il est envisagé de faire figurer les bâtiments qui ne sont pas directement impliqués dans la filière (c'est-à-dire ceux qui ne sont pas en cours de déconstruction ou de rénovation) dans le modèle sous la forme d'un agent passif qui modélisera la capacité de ce bâtiment à se détériorer avec le temps. L'objectif de cette représentation est de pouvoir anticiper les futurs chantiers de déconstruction, et donc les futurs stocks de matériaux possibles. Il sera ainsi possible d'avoir une simulation sur un temps plus long, ce qui nous offrira des scénarios de plus long terme.

In fine, les agents interagissent dans un graphe orienté, modélisant les différents flux de matériaux possibles [8]. Ce

graphe a vocation à être généré à partir de cartes SIG (Système d'Information Géographique) pour le rendre le plus proche possible du réel. Le choix de la plateforme de simulation n'est pas encore définitif, mais une première simulation a été réalisée à l'aide de la plateforme *GAMA* [7].

2.2 Métaheuristiques & Ordonnement

Bien que permettant de créer des simulations à la fois rapides et relativement proches de la réalité, les modélisations à base d'agents souffrent du fait que les agents qui les composent ne peuvent pas connaître l'impact final de leurs décisions sur la qualité globale de l'ordonnement, cette dernière ne s'appréciant qu'à la fin de la simulation. Ils sont donc contraints à ne prendre des décisions qu'avec les informations qu'ils ont localement. On appelle ce phénomène la myopie des agents [5]. Afin de contourner ce phénomène, différentes approches ont été proposées dans la littérature. Il est ainsi possible de combiner plusieurs niveaux de simulations à base d'agents [4], de simheuristiques [3] ou encore de méthodes dites hybrides [5] [10]. Ces dernières peuvent être séparées en deux groupes : celles incorporant des simulations à base d'agent au sein de leur méthode [1] [5] [6], et celles reposant sur une simulation agents dont certains agents prennent des décisions en fonction du résultat d'une métaheuristique [9].

Le choix de la méthode n'est pas encore fixée pour notre problème, et des études plus approfondies sont menées afin de déterminer quelles approches seraient les plus pertinentes dans notre cas. Une étude comparative entre les différents approches serait d'ailleurs intéressante à être menée. La première approche qui est envisagée est une approche de la première catégorie inspirée par les travaux de Yunfei Chu, Fengqi You et John M Wassick [5] sur le problème d'ordonnement de processus complexes par lots (scheduling of complex batch processes), qui se base sur une heuristique d'arbre de recherches (beam search [10]) qui utilise une simulation multi-agents afin d'évaluer une borne supérieure du coût de chacune des branches à chaque branchement de l'arbre, afin de choisir quelles branches de l'arbre explorer et quelles branches ignorer.

Il est prévu d'utiliser le plus possible la bibliothèque *paradisEO* (C++) dans l'implémentation de nos métaheuristiques, car cette bibliothèque est bien éprouvée et performante.

3 Conclusion

Dans ce papier, nous avons présenté les méthodes que nous souhaitons déployer pour résoudre les problèmes liés à la gestion des flux de matériaux dans la déconstruction. Tout d'abord une modélisation utilisant des systèmes multi-agents sera réalisée. Enfin à partir de cette modélisation des métaheuristiques intégrant cette dernière sont envisagées.

Remerciements

Ces travaux sont menés dans le cadre de la chaire industrielle RECONVERT avec le support de la Métropole Européenne de Lille et de l'I-Site Université Lille Nord-Europe.

Références

- [1] M. Barbati, G. Bruno, and A. Genovese. Applications of agent-based models for optimization problems : A literature review. *Expert Systems with Applications*, 39(5) :6020–6028, 2012.
- [2] David F. Batten. Fostering industrial symbiosis with agent-based simulation and participatory modeling. *Journal of Industrial Ecology*, 2009.
- [3] Manuel Chica, Angel A. Juan, Óscar Cordón, and W. David Kelton. Why simheuristics? benefits, limitations, and best practices when combining metaheuristics with simulation. *SSRN*, 2017.
- [4] Yunfei Chu, John M. Wassick, and Fengqi You. Efficient scheduling method of complex batch processes with general network structure via agent-based modeling. *AIChE Journal*, 59(8) :2884–2906, 2013.
- [5] Yunfei Chu, Fengqi You, and John M. Wassick. Hybrid method integrating agent-based modeling and heuristic tree search for scheduling of complex batch processes. *Computers Chemical Engineering*, 60 :277–296, 2014.
- [6] Yunfei Chu, Fengqi You, John M. Wassick, and Anshul Agarwal. Integrated planning and scheduling under production uncertainties : Bi-level model formulation and hybrid solution method. *Computers Chemical Engineering*, 72 :255–272, 2015. A Tribute to Ignacio E. Grossmann.
- [7] Alexis Drogoul, Edouard Amouroux, Philippe Caillou, Benoit Gaudou, Arnaud Grignard, Nicolas Marilleau, Patrick Taillandier, Maroussia Vavasseur, Duc-An Vo, and Jean-Daniel Zucker. Gama : A spatially explicit, multi-level, agent-based modeling and simulation platform. In Yves Demazeau, Toru Ishida, Juan M. Corchado, and Javier Bajo, editors, *Advances on Practical Applications of Agents and Multi-Agent Systems*, pages 271–274, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [8] Marcelo Gonzalez, Ivan Navarrete, Paz Arroyo and-Gabriel Azúa, and Jose Mena. Sustainable decision-making through stochastic simulation : Transporting vs. recycling aggregates for portland cement concrete in underground mining projects. *Journal of Cleaner Production*, (159), 2017.
- [9] Jan A. Persson, Paul Davidsson, Stefan J. Johansson, and Fredrik Wernstedt. Combining agent-based approaches and classical optimization techniques. 2005.
- [10] El-Ghazali Talbi. *Metaheuristics : from design to implementation*. Wiley, 2009.
- [11] Yifei Yu, D.M. Yazan, Silu Bhoohichoya, and Leentje Volker. Towards circular economy through industrial symbiosis in the dutch construction industry : A case of recycled concrete aggregates. *Journal of Cleaner Production*, (293), 2021.

Prédiction d'erreurs critiques pour l'aide à la maintenance industrielle

Myriam Lopez
LaBRI

Résumé

Anticiper la maintenance des équipements mécaniques est une ambition majeure de l'Industrie 4.0. Les solutions basées sur des données log sont prometteuses mais encore faiblement explorées. Nous proposons une approche par classification permettant de prédire la survenue d'erreurs susceptibles de provoquer une défaillance coûteuse. La solution, basée sur les données log émises par les machines-outils, se concentre sur la phase de préparation des données. L'efficacité de l'approche est démontrée au travers d'un cas d'usage industriel où les erreurs critiques peuvent être prédites une semaine en avance.

Mots-clés

Maintenance Prédicative, Apprentissage Supervisé, Déséquilibre de Classes, Classification, Log

Abstract

With the advent of Industry 4.0, failure anticipation is one of the key objectives in industrial research. In this context, predictive maintenance is an active research area for various applications. We present an approach to predict high importance errors using log data emitted by machine tools. To achieve reliable and accurate predictions, we focus on data preparation step. The effectiveness of the approach is demonstrated using a real industrial application.

Keywords

Predictive Maintenance, Machine Learning, Class Imbalance, Classification, Log

1 Introduction

La maintenance prédictive, d'importance capitale pour les fabricants[3] permet d'une part de réduire les coûts liés à l'immobilisation des systèmes après dysfonctionnement et d'autre part anticiper des commandes des pièces de rechange. De nos jours, la plupart des machines modernes de l'industrie sont équipées de capteurs qui mesurent diverses propriétés physiques telles que la pression de l'huile ou la température du liquide de refroidissement. Après nettoyage et traitement, le signal issu de ces capteurs permet d'identifier au fil du temps les indicateurs d'un fonctionnement anormal[8]. La richesse de ces données permet de visualiser l'état du système dans le temps sous la forme d'une estimation de la durée de vie utile restante[2]. En parallèle, les machines livrent régulièrement des journaux consignants

les différents événements qui permettent de suivre l'usage et les anomalies ([4]). Ainsi, la prédiction basée sur les événements est un sujet de recherche clé dans la maintenance prédictive ([9]).

Dans ce contexte, nous proposons une approche de préparation des données, permettant d'entraîner un modèle de prédiction d'occurrence d'erreur critique. Elle repose sur l'exploitation de données historiques de journal associées aux machines. L'objectif est de prédire suffisamment tôt l'apparition d'un dysfonctionnement critique afin de faciliter les opérations de maintenance. L'approche est appliquée dans un contexte industriel réel et les performances empiriques obtenues montrent son efficacité.

2 Méthodologie

Notre cas d'usage s'inscrit dans un contexte où nous disposons d'environ 300 machines outils, fonctionnant de manière indépendante, pour réaliser des tâches similaires.¹ Soit \mathcal{F} un journal produit par une machine \mathcal{M} où sont rapportées les erreurs émises par la machine suite à l'observation des valeurs d'un ensemble de capteurs. Les erreurs, un code de criticité, sont considérées *high* ou *low*. Notre objectif est de prédire, pour chaque type d'erreur *high* ses occurrences futures en fonction de l'occurrence de toutes les autres erreurs.

La collecte des données s'effectue avec l'élaboration d'une matrice de comptage. Chaque jour le nombre d'occurrence d'erreurs de chaque type est comptabilisé. Les *features* sont ainsi représentées par les différents types d'erreurs.

Pour entraîner un modèle de prédiction, pour une erreur cible donnée, trois paramètres, illustrés dans la figure 1, sont appliqués successivement :

- **Intervalle Prédicatif** : historique de données utilisé pour effectuer des prédictions. Sa taille en jours est définie par le paramètre PI .
- **Intervalle de Réactivité** : agit sur le modèle comme une contrainte d'anticipation. La taille de l'intervalle, exprimée par le paramètre RI , contrôle le délais d'anticipation souhaité.
- **Intervalle d'Erreur** : il permet de prédire l'occurrence d'une erreur cible au cours d'un intervalle temporel donné, plutôt qu'à une unité de temps spécifique. Sa taille est déterminée par le paramètre EI .

1. Pour des raisons de confidentialité vis-vis du fabricant, nous ne pouvons donner plus de détail sur le contexte industriel.

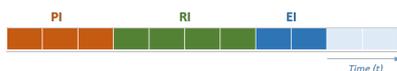


FIGURE 1 – Trois paramètres clé du modèle prédictif.

Les données de l'Intervalle Prédictif sont agrégées en une instance unique appelée la *meta-instance*. La *meta-instance* est par la suite labellisée en fonction du contenu de l'Intervalle d'Erreur. Si l'Intervalle d'Erreur contient une occurrence l'erreur cible à prédire, alors la méta-instance est labellisée POSITIF, NEGATIF sinon.

Plusieurs modèles de classifications ont été comparés, parmi lesquels un arbre de décisions, des forêts aléatoires, un réseau bayésien, un SVM et enfin un réseau de neurones profond doté de 4 couches denses. Ce dernier nous a permis d'obtenir les meilleures performances et a donc été sélectionné comme modèle prédictif.

3 Analyses et résultats

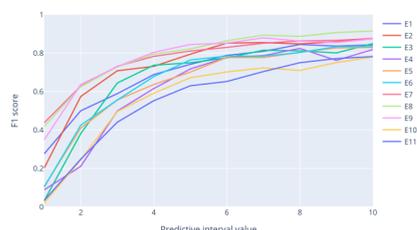
Nous nous sommes intéressés à 11 erreurs cibles corrélées à des problèmes de surchauffe. Pour chaque erreur et sachant une combinaison de valeurs de paramètres (PI ; RI ; EI), un jeu de données est obtenu suivant la méthodologie ci-dessus. Le réseau de neurones est entraîné sur 80% du jeu de données puis testé sur les 20% restants. Les jeux de données étant soumis à un problème de déséquilibre des classes, les performances ont été évaluées en termes de F1-score, données par la formule suivante :

$$F1\text{-score} = \frac{2 \times TP}{2 \times TP + FP + FN} = \frac{2(\textit{precision} \times \textit{recall})}{(\textit{precision} + \textit{recall})} \quad (1)$$

Avec TP , TN , FP et FN les vrais positifs, vrais négatifs, faux positifs et faux négatifs.

Sur les 11 erreurs critiques, les performances obtenues, s'élèvent en moyenne à 0.69 ± 0.08 .

Pour observer l'effet de la taille de l'Intervalle d'Erreur sur les performances du modèle, nous avons fait varier EI entre 1 et 10. Quelle que soit l'erreur cible, les performances s'améliorent lorsque EI augmente.

FIGURE 2 – Performances du modèle en fonction de EI .

4 Travaux relatifs

La maintenance prédictive a suscité de nombreuses recherches ces dernières années ([6]). Les approches basées sur l'apprentissage supervisé, auxquelles notre étude se rapporte, permettent de répondre à ce besoin. À notre connais-

sance, les travaux de [7] et de [5], appartenant à cette catégorie d'approche, sont les plus proches des nôtres. Les travaux de [7] s'inscrivent dans une approche d'apprentissage multi-instance (MIL)[1] dont nous nous sommes inspirés.

5 Conclusion et perspectives

L'approche proposée permet d'exploiter les données log pour prédire la survenue d'erreurs critiques. Combinée à un réseau de neurones, notre solution s'avère suffisamment précise. Son principal atout, par rapport à d'autres techniques, est sa simplicité de mise en oeuvre. À l'avenir, nous prévoyons d'analyser plus en profondeur les erreurs pour tenter de combiner différentes prédictions et ainsi réduire le nombre de modèles. En outre, nous souhaitons automatiser le réglage des paramètres : en donnant un (ensemble de) cible(s), trouver les valeurs optimales de PI , EI et RI de telle sorte que la performance du modèle appris soit maximisée.

Références

- [1] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1) :31–71, 1997.
- [2] Liang Guo, Naipeng Li, Feng Jia, Yaguo Lei, and Jing Lin. A recurrent neural network based health indicator for remaining useful life prediction of bearings. *Neurocomputing*, 240 :98–109, 2017.
- [3] H. M. Hashemian. State-of-the-Art Predictive Maintenance Techniques. *IEEE Transactions on Instrumentation and Measurement*, 60(1) :226–236, 2011.
- [4] Shilin He, Jieming Zhu, Pinjia He, and Michael R Lyu. Experience report : System log analysis for anomaly detection. In *2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE)*, pages 207–218. IEEE, 2016.
- [5] Panagiotis Korveis, Stephane Besseau, and Michalis Vazirgiannis. Predictive maintenance in aviation : Failure prediction from post-flight reports. In *Proc. of ICDE Conf.*, 2018.
- [6] Christian Krupitzer, Tim Wagenhals, Marwin Züfle, et al. A survey on predictive maintenance for industry 4.0. *CoRR*, abs/2002.08224, 2020.
- [7] Ruben Sipos, Dmitriy Fradkin, Fabian Moerchen, and Zhuang Wang. Log-based predictive maintenance. In *SIGKDD conference*, 2014.
- [8] C. Wang, H. T. Vo, and P. Ni. An IoT Application for Fault Diagnosis and Prediction. In *IEEE International Conf. on Data Science and Data Intensive Systems*, pages 726–731, 2015.
- [9] J. Wang, C. Li, S. Han, S. Sarkar, and X. Zhou. Predictive maintenance based on event-log analysis : A case study. *IBM Journal of Research and Development*, 61(1) :121 :132, 2017.

Réseau de Neurones Siamois Multimodal pour la prédiction d'évolution de maladies neurodégénératives

C. Ostertag^{1,2}, M. Beurton-Aimar², M. Visani¹, T. Urruty³

¹ LaRochelle Université, L3i

² Université de Bordeaux, LaBRI

³ Université de Poitiers, XLIM

...

cecilia.ostertag1@univ-lr.fr

Résumé

Pour évaluer la progression de maladies neurodégénératives, l'imagerie médicale est utilisée, ainsi que des tests cognitifs et des mesures biologiques. Dans ce travail nous présentons une approche d'apprentissage profond dont le but est l'utilisation de données multimodales, pour la prédiction de l'évolution de la maladie. Notre modèle est un réseau de neurones profond, avec des sous-modules siamois dédiés à l'extraction d'informations pour chaque modalité.

Mots-clés

réseaux siamois, multimodal, apprentissage profond

Abstract

To assess the progression of neurodegenerative diseases, medical imaging is used alongside cognitive evaluations and biological measurements. In this work we present a deep learning approach aiming to use multimodal data to predict the disease evolution. Our model is a Deep Neural Network (DNN) with siamese sub-modules dedicated to extract features from each modality.

Keywords

siamese network, multimodal, deep learning

1 Introduction

Le suivi et la prédiction de l'état cognitif d'une personne affectée par une maladie neurodégénérative est cruciale, afin de fournir un traitement approprié à chaque patient, et cela le plus tôt possible. Ces patients sont donc suivis régulièrement pendant plusieurs années, dans le cadre d'études longitudinales. A chaque visite médicale, une grande quantité de données est acquise : présence de facteurs de risque, imagerie (IRM ou PET scan), résultats de tests cognitifs, prélèvements biologiques, etc. Ces différentes modalités apportent des informations sur la progression de la maladie, certaines complémentaires et d'autres redondantes.

En utilisant comme cas d'application la maladie d'Alzheimer, nous avons conçu un réseau de neurones profond ca-

pable d'utiliser deux modalités, les IRM structurales de cerveau, et les données cliniques (scores cognitifs et facteurs de risques), pour la prédiction du déclin cognitif à partir de deux visites médicales d'un patient.

2 Création du Jeu de Données

Notre jeu de données est constitué de sujets provenant de la base Alzheimer's Disease Neuroimaging Initiative (ADNI) [1]. De cette base nous avons récupéré 382 sujets ayant une visite médicale d'inclusion et une visite de suivi à 6 mois, 12 mois, et 24 mois. Notre but étant d'utiliser un réseau siamois pour évaluer l'évolution des sujets entre deux visites médicales, de façon similaire aux travaux de Bhagwat et al. [2], nous avons utilisé ces 4 visites pour créer 6 paires de visites par sujet. Enfin, nous avons utilisé l'évolution du score cognitif Mini Mental State Evaluation (MMSE) sur 72 mois pour séparer les sujets en deux groupes : stabilité ou déclin cognitif. Ce sont ces groupes qui ont servi de label pour nos paires de visites. Après pré-traitement et filtrage de nos données, nous obtenons un total de 2268 paires de visites. La figure.1 montre notre processus de création du jeu de données.

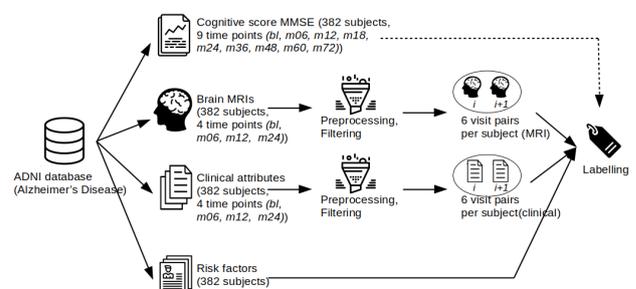


FIGURE 1 – Représentation schématique du processus de création de notre jeu de données

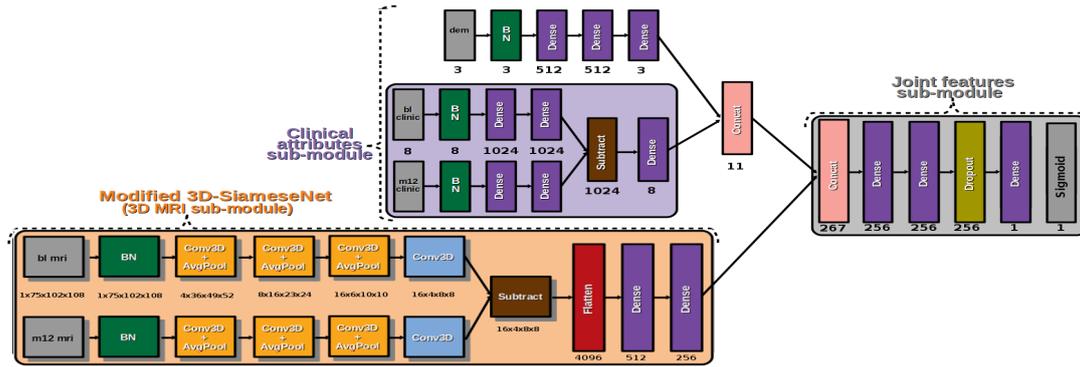


FIGURE 2 – Architecture de notre réseau siamois multimodal

3 Architecture du Modèle

Notre modèle, implémenté avec PyTorch, (voir Figure.2) est un réseau de neurones profond, conçu pour prendre en compte à la fois l’aspect multimodal et l’aspect temporel de notre tâche de classification. La gestion des modalités se fait à l’aide de sous-modules spécifiques à chaque modalité : les IRM étant des volumes il sont envoyés à un réseau convolutif 3D, et les données cliniques étant des listes de variables quantitatives et qualitatives, ils sont envoyés à un simple Multi Layer Perceptron 1D. Les modalités sont fusionnées par concaténation, puis quelques couches supplémentaires permettent un apprentissage conjoint, pour exploiter les corrélations entre modalités. Cette stratégie de fusion est basée sur les travaux de Xu et. al. [4]. La comparaison des deux visites médicales se fait grâce à l’architecture siamoise, pour chaque modalité.

4 Résultats et Discussion

Notre jeu de données a été séparé en 60%/20%/20% pour l’entraînement, la validation, et le test. Après une validation croisée, nous avons obtenu une accuracy et un score F1 de 0.91 sur le jeu de test (composé de 231 patients stables et 219 en déclin cognitif). Ce modèle multimodal est donc acceptable, et a des performances supérieures à l’utilisation des sous-modules uni-modaux séparément (F1 de 0.33 pour le module IRM, et F1 de 0.47 pour le module clinique). Nous montrons donc ainsi la valeur ajoutée de l’apprentissage multimodal.

Nous avons également comparé les valeurs de précision, rappel, et F1 obtenus en groupant les paires de visites par intervalle de temps entre les deux visites, et nous avons montré que la durée de l’intervalle n’impacte pas significativement les performances du modèle multimodal. En revanche nous avons montré que des fluctuations de l’état cognitif du patient à court terme peuvent causer le modèle à prédire une stabilité cognitive, alors que l’évolution à long terme du sujet montre un déclin cognitif (cas de faux négatifs).

Enfin, nous avons comparé notre modèle siamois multimo-

dal à un modèle récurrent (RNN) [3] utilisant des séquences de 5 visites successives, avec les modalités suivantes : facteurs de risque, scores cognitifs, et mesures de volumes calculées à partir des IRM fonctionnelles. Bien que le RNN ait de meilleures performances (accuracy et F1 de 0.97), notre modèle reste compétitif et avantageux car il ne nécessite que deux visites quelconques pour la prédiction du devenir cognitif d’un patient.

5 Conclusion

Nous avons conçu et implémenté un réseau multimodal pour la prédiction de l’évolution d’une maladie neurodégénérative, la maladie d’Alzheimer, à partir de données multimodales. Cette architecture est adaptée aux différentes modalités, et prend en entrée deux visites médicales quelconques d’un même patient, afin de prédire s’il est ou non en déclin cognitif. Notre modèle obtient une accuracy de 91%, ce qui est compétitif face à un modèle RNN (accuracy 97%), étant donné qu’il nécessite simplement une paire de visites et non une séquence de visites.

Références

- [1] ADNI | Alzheimer’s Disease Neuroimaging Initiative.
- [2] Nikhil Bhagwat, Joseph D Viviano, Aristotle N Voineskos, M Mallar Chakravarty, Alzheimer’s Disease Neuroimaging Initiative, et al. Modeling and prediction of clinical symptom trajectories in alzheimer’s disease using longitudinal data. *PLoS computational biology*, 14(9) :e1006376, 2018.
- [3] Garam Lee, Kwangsik Nho, Byungkon Kang, Kyung-Ah Sohn, and Dokyoon Kim. Predicting alzheimer’s disease progression using multi-modal deep learning approach. *Scientific reports*, 9(1) :1–12, 2019.
- [4] Tao Xu, Han Zhang, Xiaolei Huang, Shaoting Zhang, and Dimitris N Metaxas. Multimodal deep learning for cervical dysplasia diagnosis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 115–123. Springer, 2016.

Reconnaissance d'émotions à partir de la posture par LSTM

K. Réby¹, M. Beurton-Aimar¹

¹ LaBRI, équipe Images et Sons, Université de Bordeaux

Résumé

L'objectif de ce travail est de développer des méthodes d'apprentissage profond pour l'analyse et l'interprétation automatique des émotions à partir des mouvements et de la posture du corps humain. A partir du jeu de données BOLD nous avons créé un modèle basé sur les LSTM afin de détecter les émotions.

Mots-clés

Émotions, posture, informatique affective, deep learning

Abstract

The objective of this work is to develop deep learning methods for automatic analysis and interpretation of emotions from human body movements.

Keywords

Emotions, human pose, affective computing, deep learning

1 Introduction

Un des objectifs de l'« Affective Computing » tel que défini par Rosalind Picard est l'étude et le développement d'algorithmes permettant de reconnaître les émotions humaines (Picard, MIT Press 1997). L'Affective Computing a de nombreux domaines d'applications, tel que l'interaction homme-machine, la médecine, la télésurveillance etc. La reconnaissance d'émotions par ordinateur a d'abord commencé par se concentrer sur la reconnaissance faciale. Ce choix s'explique du fait que chez l'être humain le visage est le principal vecteur d'informations et d'expression des émotions. En effet, les expressions faciales sont naturellement utilisées pour communiquer des messages émotionnels entre individus. Les recherches antérieures sur l'analyse automatique des mouvements du corps humain se sont concentrées sur la reconnaissance d'activités, notamment sportives. La reconnaissance d'émotions par ordinateur a d'abord commencé par se concentrer sur la reconnaissance faciale. Ce choix s'explique du fait que chez l'être humain le visage est le principal vecteur d'informations et d'expression des émotions. Cependant l'état émotionnel d'une personne est une autre caractéristique importante qui est véhiculée par les mouvements du corps. Ainsi Kleinsmith et al. (2012) [1], Randhavane et al. (2019) [2] ont montré que les mouvements et les postures du corps contiennent des informations sur l'état émotionnel de la personne. Afin de permettre le développement de techniques d'apprentissage pour l'analyse et l'interprétation automatique des émotions à partir des mouvements du corps humain, Luo et al. [3]

ont présenté le jeu de données BOLD ("BOdy Language Dataset") dans le cadre du "First international workshop on bodily expressed emotion understanding" de la "European Conference on Computer Vision" (ECCV) 2020.

Tout comme pour l'analyse des expressions faciales, les mouvements du corps pourraient donc être encodés avec des mouvements de muscles posturaux, à l'instar des Unités d'Action (AU) dans le Système de Codage des Actions Faciales (FACS) d'Ekman et al. (1980) [4].

Notre objectif est donc d'utiliser les données du jeu de données BOLD et d'extraire certaines caractéristiques géométriques de la posture, telles que les angles entre les articulations du corps, comme données d'entrée pour un réseau de type LSTM ("Long Short Term Memory") afin de détecter les émotions.

2 Méthode

Dans [3], Luo et al. ont démontré que, bien que constituant l'état de l'art dans le domaine de la reconnaissance d'actions, les architectures à base de réseaux de neurones parallèles "2 streams networks" tel que I3D[9] sont moins efficaces que les Temporal Segments Networks (TSN). Pour les auteurs ceci est dû au fait que pour la classification d'actions, les caractéristiques spatiales sont plus importantes que les informations temporelles pour la classification, alors que dans le cas de la détection d'émotions, l'information temporelle est primordiale.

Ainsi si on veut traiter une séquence temporelle, comme une vidéo, un réseau classique ne pourra pas utiliser le traitement des événements précédents pour analyser les événements ultérieurs; en revanche les réseaux de neurones récurrents sont capables de gérer de telles dépendances temporelles. Introduits par Hochreiter Schmidhuber (1997) [6], les LSTM sont des types de réseaux de neurones récurrents conçus pour résoudre ce problème de la dépendance à long terme et donc pour traiter des séquences de données. Les LSTM ont été déjà utilisés avec succès pour l'analyse de la posture humaine [7].

2.1 Le jeu de données BOLD

Pour construire leur jeu de données, Luo et al. ont choisi des films inclus dans le jeu de données public "AVA" qui contient une liste de films de YouTube. Ces films ont été divisés en plusieurs courtes scènes, puis chaque clip a été traité par OpenPose [5] image par image pour positionner les points de repères du corps humain. Enfin, les annotations des émotions de chaque personne dans ces

clips ont été réalisées grâce à "Amazon Mechanical Turk". BOLD comporte 26 catégories d'émotions (Paix, Affection, Estime, Anticipation, Engagement, Confiance, Bonheur, Plaisir, Excitation, Surprise, Sympathie, Doute/confusion, Déconnexion, Fatigue, Embarras, Désir, Désapprobation, Aversion, Contrariété, Colère, Sensibilité, Tristesse, Inquiétude, Peur, Douleur, Souffrance) et 3 valeurs émotionnelles continues selon le modèle VAD ("Valence, Arousal, Dominance").

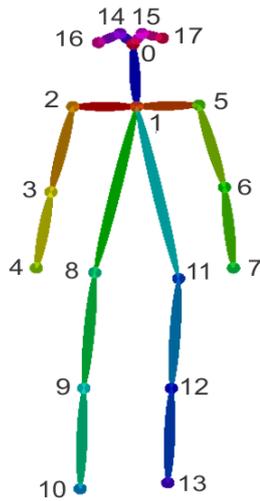


FIGURE 1 – Illustration des 18 points de repères anatomiques fournis par OpenPose [5]

2.2 Modèle

Nous proposons donc un modèle inspiré de celui utilisé par Wang et al. (2019) [8] avec une architecture de réseau de neurones LSTM détaillée dans le tableau 1. Ici les données d'entrées du réseau sont les angles locaux des articulations, chaque angle étant calculé à partir de trois points anatomiques pertinents. On obtient ainsi 10 angles caractéristiques de la posture indiqués dans le tableau 2.

TABLE 1 – Détails du réseau

Nombre de couches LSTM	3
Nombres de neurones	10
Taux d'apprentissage	0.003
Optimisation	Adam
Nombre d'époques	50

Une vidéo est donc représentée par une séquence d'ensembles de coordonnées des angles calculés, chacune contenant les informations de pose pour une image donnée. Le réseau LSTM reçoit ses informations et génère une sortie par pas de temps, créant ainsi un modèle de séquence à séquence. Ici, le vecteur de caractéristiques d'entrée sur un pas de temps est la liste des angles pour une image donnée, et la sortie sur un pas de temps est une suite

d'émotions identifiées dans la séquence.

Nous avons distingué :

- d'un côté un problème de type de classification pour les 26 catégories d'émotions, avec comme fonction de perte l'entropie croisée ("Cross Entropy") et comme métriques la précision moyenne ("mAP") et la moyenne de l'aire sous la courbe ROC ("mRA"),
- de l'autre un problème de régression pour prédire les valeurs des 3 émotions continues, avec comme fonction de perte l'erreur quadratique moyenne ("Mean Square Error") et comme métrique la moyenne du Coefficient de détermination ("mR²").
- A cela s'ajoute l'"Emotion Recognition Score" (ERS) développé par Luo et al. tel que :

$$ERS = 1/2(mR + 1/2(mAP + mRA))$$

3 Résultats et discussion

Les premiers résultats (voir tableau 3) montrent que les données concernant la partie inférieure du corps (hanches, genoux et chevilles) dans la base BOLD sont trop entachées de bruit. En effet nous avons constaté que de nombreuses positions de points de repères anatomiques concernant les membres inférieurs (hanches, genoux et chevilles) données par OpenPose sont erronées (coordonnées erronées ou absentes). Nous avons donc décidé de nous concentrer sur le haut du corps afin de garder des valeurs d'angles fiables et non nulle. L'avantage d'un tel modèle relativement simple est qu'il nécessite moins de temps et de puissance de calcul par rapport aux modèles utilisant les images (comme les TSN) ou le flot optique (tel que I3D). Afin d'améliorer les résultats nous allons également travailler sur des modèles utilisant des mécanismes d'attention spatiales et temporelles (LSTM, Transformer).

TABLE 2 – Points de repères anatomiques d'OpenPose utilisés pour calculer les angles d'intérêt

Angles	Premier point de référence	Deuxième point de référence	Troisième point de référence
Angle 1	nez 0	cou 1	épaule droite 2
Angle 2	nez 0	cou 1	épaule gauche 5
Angle 3	cou 1	épaule droite 2	coude droit 3
Angle 4	cou 1	épaule gauche 5	coude gauche 6
Angle 5	épaule droite 2	coude droit 3	poignet droit 4
Angle 6	épaule gauche 5	coude gauche 6	poignet gauche 7
Angle 7	cou 1	hanche droite 8	genou droit 9
Angle 8	cou 1	hanche gauche 11	genou gauche 12
Angle 9	hanche droite 8	genou droit 9	cheville droite 10
Angle 10	hanche gauche 11	genou gauche 12	cheville gauche 13

TABLE 3 – Résultats

Modèle	mAP	mRA	mR ²	ERS
Luo et al.	17.14	63.52	0.103	0.253
LSTM	6.74	51.18	0.08	0.185

Références

- [1] A. Kleinsmith and N. Bianchi-Berthouze, "Affective Body Expression Perception and Recognition : A Survey," *IEEE Transactions on Affective Computing*, vol. 4, no. 1, pp. 15-33, Jan.-March 2013, doi : 10.1109/TAFFC.2012.16.
- [2] T. Randhavane and U. Bhattacharya et al., "Identifying Emotions from Walking using Affective and Deep Features", *CoRR*, <http://arxiv.org/abs/1906.11884>, 2019.
- [3] Luo, Y., Ye, J., Adams, R.B. et al., "ARBEE : Towards Automated Recognition of Bodily Expression of Emotion in the Wild." *Int J Comput Vis* 128, 1–25 (2020). <https://doi.org/10.1007/s11263-019-01215-y>
- [4] Ekman, P., Freisen, W. V., Ancoli, S. (1980), "Facial signs of emotional experience." *Journal of Personality and Social Psychology*, 39(6), 1125–1134. <https://doi.org/10.1037/h0077722>
- [5] Z. Cao, G. Hidalgo, T. Simon, S. -E. Wei and Y. Sheikh, "OpenPose : Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172-186, 1 Jan. 2021, doi : 10.1109/TPAMI.2019.2929257.
- [6] Sepp Hochreiter and Jürgen Schmidhuber, "Long Short-Term Memory" *Neural Computation* 1997 9 :8, 1735-1780
- [7] Luo, Yue and Ren, Jimmy et al., "LSTM Pose Machines" *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2018*
- [8] C. Wang, M. Peng, T. A. Olugbade, N. D. Lane, A. C. de C. Williams and N. Bianchi-Berthouze, "Learning Temporal and Bodily Attention in Protective Movement Behavior Detection," *2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, Cambridge, UK, 2019, pp. 324-330, doi : 10.1109/ACIIW.2019.8925084.
- [9] Carreira, Joao and Zisserman, Andrew, "Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset" *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*

Extraction de sous-groupes exceptionnels de séries temporelles

Josie Signe¹

¹ Inria, Univ Rennes, CNRS, IRISA

josie.signe@irisa.fr

Résumé

La tâche de fouille de modèles exceptionnels extrait des sous-groupes intéressants, selon des attributs cibles, dans des données tabulaires. Nous étendons cette approche aux séries temporelles en les utilisant comme attribut cible pour évaluer les sous-groupes. Un sous-groupe est caractérisé à l'aide d'une description en s'appuyant sur un modèle de séries temporelles calculé à partir de toutes les séries du sous-groupe. L'évaluation de la qualité d'un sous-groupe repose sur la différence entre le modèle du sous-groupe et le modèle général, i.e., de toutes les données.

Mots-clés

Fouille de motifs, Fouille de modèles exceptionnels, Séries temporelles.

Abstract

Exceptionnal model mining task finds interesting subgroups according to several target attributes in tabular data. We extend this approach to time series data, using time series as a target attribute to evaluate subgroups. A subgroup is characterized by a description based on a model of time series computed from all series of the subgroup. Evaluation of subgroup quality is based on the difference between the subgroup model and the model of the whole dataset.

Keywords

Pattern mining, Exceptionnal model mining, Time series.

1 Introduction

L'utilisation croissante de capteurs entraîne une multiplication de données mesurées en temps réel. Ces suites de valeurs numériques, appelées *séries temporelles*, permettent de suivre l'évolution des systèmes dans de nombreux domaines (e.g. le cours de la bourse en finance, la température des animaux en agriculture). Une tâche importante concernant les séries temporelles est la détection de séries dont l'évolution est différente de la norme.

La fouille de modèles exceptionnels (EMM) [2] permet d'extraire et de caractériser des sous-groupes *exceptionnels*, i.e., qui se distinguent par rapport à l'ensemble des données sur des attributs dits *cibles*. Chaque sous-groupe est caractérisé par un ensemble d'*attributs descriptifs*, distincts des attributs cibles. EMM a donné des résultats concluants dans plusieurs domaines [2]. Par exemple, dans le domaine de la bioinformatique, EMM a permis de trouver des

sous-groupes de gènes intéressants, i.e., dont les niveaux d'expression des gènes (attributs cibles) se distinguent des autres gènes, et de les caractériser par des données cliniques de patients (attributs descriptifs). Il existe plusieurs instances d'EMM permettant de traiter des attributs numériques, nominaux ou binaires mais aucune permettant de traiter des séries temporelles comme attributs cibles.

Dans cet article nous proposons une nouvelle instance d'EMM pour pouvoir traiter des séries temporelles comme attribut cible. Nous présentons un extrait des résultats obtenus lors de nos expérimentations sur des données d'élevage.

2 Travaux antérieurs

Séries temporelles Une série temporelle est une séquence de l valeurs $t = \langle (v_1, h_1), \dots, (v_l, h_l) \rangle$, avec $v_i \in \mathbb{R}$ la valeur horodatée à h_i .

Pour comparer deux séries t_1 et t_2 , leur similarité est mesurée avec une fonction de distance $dist(t_1, t_2) \in \mathbb{R}$, calculée entre les valeurs de t_1 et t_2 . Une mesure de distance simple entre deux séries temporelles est la distance euclidienne. Elle calcule la distance entre chaque point de t_1 et t_2 qui ont le même horodatage. Une approche plus flexible appelée déformation temporelle dynamique (DTW) [1] prend en considération les décalages temporels et permet d'aligner de manière optimale chaque point des deux séries.

Fouille de modèles exceptionnels La fouille de modèles exceptionnels (EMM) [2] extrait des sous-groupes *exceptionnels* dans une base de données. Un sous-groupe est dit *exceptionnel* si les valeurs de ses attributs cibles sont très différentes des valeurs de ces mêmes attributs dans la base de données.

Soit D une base de données de n éléments $\{e_1, \dots, e_n\}$, un élément e est défini par un ensemble d'attributs descriptifs A et d'attributs cibles C tel que : $e = \{a_1, \dots, a_m, c_1, \dots, c_k\}$, $m, k \in \mathbb{N}$, $a_i \in A$, $c_i \in C$.

Un *sous-groupe* S de D est décrit par un ensemble de conditions sur les attributs descriptifs de A , appelé *description*. Une description d couvre un élément e de D si les attributs descriptifs de e respectent toutes les conditions de d . Le sous-groupe S décrit par d correspond à l'ensemble des éléments appartenant à D qui sont couverts par d .

Les attributs cibles permettent de déterminer si un sous-groupe S est intéressant. EMM modélise la relation qui existe entre les différents attributs cibles du sous-groupe S . Le modèle utilisé pour représenter cette relation est choisi

en fonction du type des attributs. Par exemple, la relation entre attributs cibles numériques pourra être représentée par une régression linéaire alors que celle entre attributs nominaux par des réseaux bayésiens.

Pour chaque sous-groupe, un modèle est calculé sur l'ensemble de ses attributs cibles. Le but d'EMM est de trouver les sous-groupes les plus exceptionnels. Pour les trouver, une mesure de qualité est utilisée pour attribuer un score à chaque sous-groupe. Cette mesure permet de quantifier la différence entre le modèle d'un sous-groupe et le modèle général. Par exemple, pour une régression linéaire, on peut calculer la différence de pente entre les deux modèles.

3 Contribution

Le but de notre approche consiste à trouver des sous-groupes dont les séries temporelles se distinguent dans leur évolution par rapport aux séries temporelles de l'ensemble des données D .

Nous proposons donc une nouvelle instance d'EMM pour laquelle l'attribut cible est une série temporelle. Soit D une base de données de n éléments $\{e_1, \dots, e_n\}$, un élément e est défini par un ensemble de n attributs descriptifs de A et d'un attribut cible t tel que : $e = \{a_1, \dots, a_m, t\}$, $m \in \mathbb{N}$, $a_i \in A$, et t une série temporelle.

Nous avons du définir un modèle représentant un sous-groupe de séries temporelles ainsi qu'une mesure permettant de comparer deux de ces modèles. Le modèle pour représenter un sous-groupe de séries temporelles est une série temporelle elle-même. Dans notre approche, elle est calculée avec la méthode DBA (DTW Barycenter Averaging) [3]. DBA calcule une série temporelle moyenne s d'un ensemble de séries temporelles $S = \{t_1, \dots, t_x\}$ qui minimise :

$$\min \sum_{i=1}^x \text{dist}_{DTW}(s, t_i)^2$$

Cette approche est appliquée à l'ensemble des données D pour générer le modèle général. Puis elle est utilisée pour chaque sous-groupe S afin de calculer leur modèle propre. Lors de l'évaluation d'un sous-groupe S , son modèle est comparé au modèle général. Nous utilisons une mesure de qualité q qui repose sur la distance DTW entre ces deux modèles pour attribuer un score d'exceptionnalité aux sous-groupes :

$$q(S) = \text{dist}_{DTW}(DBA_D, DBA_S)$$

avec DBA_D le modèle issu du DBA de l'ensemble des séries temporelles de D , et DBA_S le modèle issu du DBA des séries temporelles du sous-groupe S .

4 Expérimentations et perspectives

Nous avons expérimenté notre approche dans le cadre d'une étude sur le bien-être animal. Le but est de trouver des sous-groupe de vaches qui supportent plus ou moins bien les périodes de forte chaleur en analysant l'évolution de leur température corporelle. Dans cette expérience les éléments de

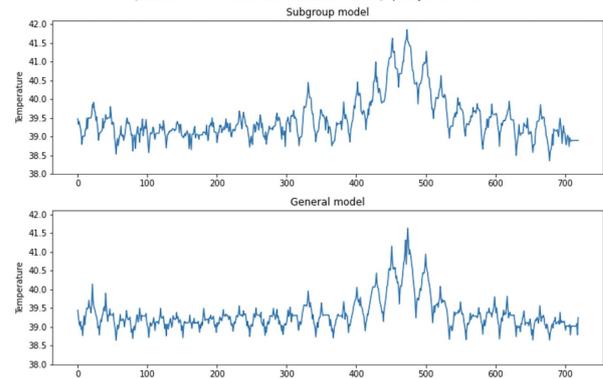


FIGURE 1 – Modèle du sous-groupe {Poids Vif ≤ 685.9 AND Ration Totale ≥ 21.48 } et modèle général.

D sont des vaches. Elles sont décrites par des attributs descriptifs numériques tel que leur poids ou leur production de lait, et leur attribut cible correspond à leur série temporelle de température corporelle mesurée en période de forte chaleur. Nous avons extrait tous les sous-groupe possible et à la figure 1, nous présentons deux modèles extraits de nos premières expérimentations. Celui du bas est le modèle général calculé sur toutes les données. Celui du haut est le modèle représentant un sous-groupe de vaches dont la description associée est “des vaches dont le poids n'est pas élevé et qui mangent beaucoup”. On voit que la température entre les heures 400 et 525 présente des pics plus élevés et qui redescendent moins par rapport au modèle général. Ce sous-groupe décrit donc des vaches sensibles aux fortes chaleurs. Dans nos expérimentations, beaucoup de sous-groupes sont extraits dont certains redondants et de petite taille. Une de nos perspectives est d'intégrer directement la taille des sous-groupes dans la mesure de qualité. De plus, nous envisageons d'explorer des approches de fouille de données s'appuyant sur la théorie de l'information pour sélectionner les sous-groupes les plus intéressants.

5 Remerciements

Ce travail a bénéficié d'une aide de l'État gérée par l'Agence Nationale de la Recherche au titre du programme d'Investissements d'avenir portant la référence ANR-16-CONV-0004

Références

- [1] Eamonn Keogh. Exact indexing of dynamic time warping. In *Proceedings of the 28th International Conference on Very Large Data Bases*, page 406–417, 2002.
- [2] Dennis Leman, Ad Feelders, and Arno Knobbe. Exceptional model mining. In *Machine Learning and Knowledge Discovery in Databases*, pages 1–16, 2008.
- [3] François Petitjean, Alain Ketterlin, and Pierre Gançarski. A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*, 44(3) :678–693, 2011.

AI's potential for Artificial Phantom Mimicking Tissue of the Human Breast Electrical Properties

A. Rangel Trejo¹, J. A. Bernardo¹, L. Konadu¹, L. Fakri-Bouchet^{1,2}

¹ Université de Lyon, CNRS, Université Claude Bernard Lyon 1, Institut des Sciences Analytiques (ISA), UMR 5280, 5 rue de la Doua, 69100 Villeurbanne, France

² INSA Lyon, France

abel.rangel@cetys.edu.mx; latifa.fakri-bouchet@insa-lyon.fr

Abstract

In this study, we present an assessment method of Early Breast Cancer Detection based on Artificial Intelligence (AI). This method includes prediction, setting and measurement of several human breast phantom mixtures for Microwave Imaging (MWI). These developed Breast Phantoms have realistic dielectric properties (important for Radar Microwave Imaging Systems & Ethical Non-Animal-Testing). We Developed Heterogeneous mimicking tissue architecture for mimicking Breast Tissues (Fat, Regular Fat, Gland, Body Fluid, Blood, Muscle-Tumor, Skin Dry, and Skin Wet) and tested using ultrawideband frequency (0.2 GHz – 4.5 GHz) by a modern Vector Network Analyzer. Our qualitative and quantitative research includes the methodology and results for using the material's dielectric parameters and AI algorithms for emulation of real Breast Tissues for Microwave Imaging Pre-clinical trials.

Keywords

Artificial-Intelligence, Breast-Cancer, Breast-Phantoms, Microwave measurement, Microwave Imaging, Radar Imaging.

1. INTRODUCTION

Today newer Breast Cancer Diagnostic systems, less expensive, more comfortable, non-ionizing with same or better efficiency are developed. Microwave Imaging (MWI) is reported as one of the best, with accuracy around 80 – 90 % compared to other systems like X-ray mammography with 70.2%, MRI with 72.9%, Ultrasound with 67.8% and Breast Clinical examinations with 63.6% of accuracy to detect malignant Tumors [1], [2].

To make pre-clinical trials of these MWI device systems, our team developed Breast Phantoms with the help of AI. The used AI tools include Machine learning (ML) and Deep Learning (DL) algorithms in non-linear regression problems that involve multi-prediction. Our research work shows that using kitchen cost-effective ingredients and AI (Artificial Neuronal Networks: ANN), is possible to match the real breast tissue dielectric properties with high precision up to 1.71 of mean absolute error (MSE) and 5.69% of mean absolute percentage error (MAPE) as seen in (Fig. 2).

2. STATE-OF-THE-ART

In the literature, Breast Phantoms mimicking tissues for this kind of systems had been fabricated with complex laboratory chemicals without a successful complete matching all the Breast Tissues (with both Dielectric Constant and Conductivity properties for Radar Microwave Systems along a complete Ultra-Wide-Band (UWB) frequency range) [3].

1.1 AI for Breast Phantom Development

To our knowledge, there are no studies reported about Breast Phantom developments using AI's ANN or other DL techniques. This confirms a new field with big opportunities to push the development of a new generation of mimicking tissues Phantoms for several testing and validation proposes in the medical or industrial field, with parameters & conditions closer to reality.

1.2 Electromagnetic Properties of Breast Phantom

These Artificial breast phantoms can provide the following important Electrical Properties that interact with the Electromagnetic Waves.

- Relative Permittivity or Dielectric constant (ϵ_r)
- Dielectric Loss (ϵ_r'')
- Loss Tangent (δ) [Formula: $\delta = \epsilon_r''/\epsilon_r'$]
- Conductivity ($\sigma(\omega)$) [Formula: $\sigma(\omega) = \epsilon_r'' \times (2\pi\omega) \times \epsilon_0$]

However, for this case of MWI radar principle (detection of only scattering microwave signals), the only interesting properties for our phantoms to match are 2 of these 4 parameters, ϵ_r and $\sigma(\omega)$. This is related to the energy that travels across the bulk behaving differently according to the sample's conductivity (S/m), before it encounters another interphase.

Tissues	Dielectric Constant (ϵ_r)	Conductivity ($\sigma(\omega)$)
Skin Dry	~ 40.8	0.90 S/m
Skin Wet	~ 45.6	0.89 S/m
Muscle & Tumor	~ 54.76	0.98 S/m
Blood	~ 61	1.59 S/m
Fat	~ 5.4	0.05 S/m
Breast Fat	~ 5.4	0.05 S/m

Body Fluid	~ 69	1.67 S/m
Gland	~ 59.4	1.08 S/m

Table 1 - Real dielectric properties of tissues at 1GHz.

2 MATERIALS & METHODS

2.1 AI Predictions & Materials

To make our predictions we used MATLAB software with Deep learning Toolbox for the use of Artificial Neuronal Networks (ANN). For training, we used datasets of different dielectric properties from (0.2 – 4.5 GHz) corresponding to ingredients combinations (*Corn Flour, Water, Salt, and Sunflower Oil*) using the respective quantities in grams.

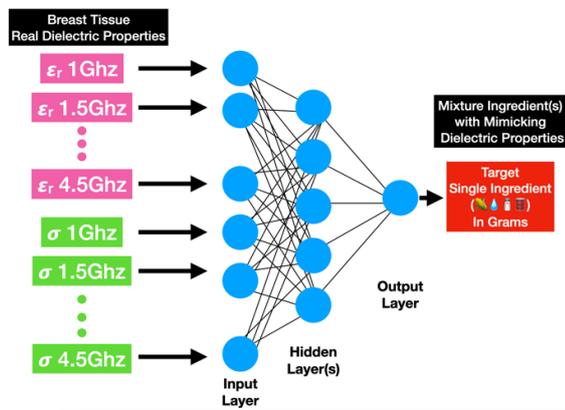


Figure 1 - ANN structure for Breast Phantom Ingredient Mixture Prediction

Pre-training treatment was defined by average data for multiple measures of the same mixtures, Outlier removal and Split Dataset for 70% train, 15% validation, and 15% test. Also, we did Jittering for data amplification, because the data set was not too big (~ 66 observations), in comparison to the Degrees of Freedom (DOF) theory [4].

For training we used a custom-made script to automatically test different architectures of ANN (with *Levenberg-Marquardt backpropagation*, and *Bayesian Regularization* algorithms) keeping only one single Target at the time and selecting the best hyperparameters at the end. Comparisons were done in terms of performance MSE of the Test Set.

2.2 Dielectric Measurements for Validation

The predicted sample mixtures were constructed and then measured in UWB range (200 MHz – 4.5 GHz). A dielectric probe kit, with Keysight Materials Measurement Suite Software, and Vector Network Analyzer (VNA) were used to measure (dielectric parameters) and compare the original ANN predictions.

3 RESULTS & DISCUSSIONS

The graphs in (Fig. 2) show the Breast Phantom Mixture predictions done by our “AI_Algorithm”, using ANN with our aimed MSE.

The optimal number of neurons (in a single layer) found by our algorithm, confirmed that shallow ANN are good candidates for our multi-response non-linear problem. The MSE & Root mean square error (RMSE) performance indicators confirmed good prediction generality for unseen data (measured from our unseen test-Dataset).

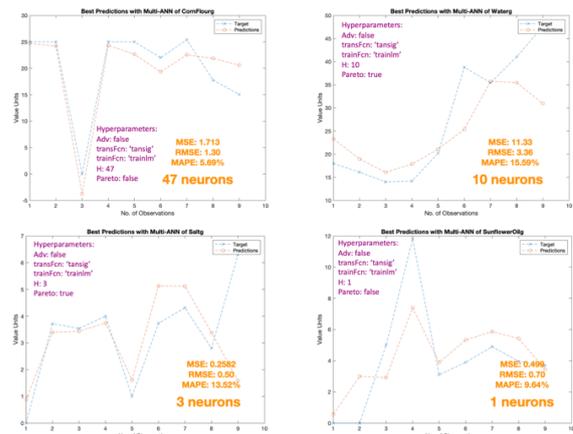


Figure 2 - Breast Phantom Mixture Predictions with Performance values

4 CONCLUSION

Our AI algorithms were validated for 88.89 % prediction accuracy (using MAPE) to predict the Phantom ingredients quantities for this non-linear Multi-response and Multi-frequency prediction problem (with very few data ~ 66 observations related to DOF theory). Results confirm that these breast phantoms are perfect candidates for a complex heterogeneous model for MWI system tests. The found RMSE ranges below 5 validated our data predictions, as they show a close scale to the one of our original datasets (in grams). In conclusion the results errors were close to ±1.3 grams for corn flour, ±3.36 grams for water, ±0.5 grams for salt and ±0.7 grams for sunflower oil.

Acknowledgments

Thanks to CONACYT Mexico for the sponsorship. Thanks to the ISA’s teams of biosensors & interphases, the technical team with Catherine Jose, Stephan Hupont and Christophe Pages for their time and experience.

References

[1] L. Wang, “Microwave Sensors for Breast Cancer Detection,” *Sensors*, vol. 18, no. 2, p. 655, Feb. 2018, doi: 10.3390/s18020655.

- [2] A. Rahman, M. T. Islam, M. J. Singh, S. Kibria, and Md. Akhtaruzzaman, “Electromagnetic Performances Analysis of an Ultra-wideband and Flexible Material Antenna in Microwave Breast Imaging: To Implement A Wearable Medical Bra,” *Scientific Reports*, vol. 6, no. 1, pp. 1–11, Dec. 2016, doi: 10.1038/srep38906.
- [3] M. Islam, M. Samsuzzaman, M. Islam, and S. Kibria, “Experimental Breast Phantom Imaging with Metamaterial-Inspired Nine-Antenna Sensor Array,” *Sensors*, vol. 18, no. 12, p. 4427, Dec. 2018, doi: 10.3390/s18124427.
- [4] J. Brownlee, “A Gentle Introduction to Degrees of Freedom in Machine Learning,” *Machine Learning Mastery*, Apr. 24, 2020. <https://machinelearningmastery.com/degrees-of-freedom-in-machine-learning/>.

DeepIndices : Une nouvelle approche des indices de télédétection basée sur l'optimisation et l'approximation de fonctions par DeepLearning. Application aux indices de végétation sur des données non calibrées

Jehan-Antoine Vayssade ¹, Jean-Noël Paoli ¹, Christelle Gée ¹, Gawain Jones ¹

¹ Agroécologie, AgroSup Dijon, INRA, Univ. Bourgogne-Franche-Comté, F-21000 Dijon, France

jehan-antoine.vayssade@inrae.fr

Résumé

L'une des avancées les plus importantes dans le domaine de l'observation de la terre est la découverte des indices spectraux, ils ont notamment prouvé leur efficacité dans la caractérisation des surfaces agricoles, mais ils sont généralement définis de manière empirique. Cette étude basée sur l'intelligence artificielle et le traitement du signal, propose une méthode pour trouver un indice optimal. Et porte sur l'analyse d'images issues d'une caméra multi-spectrale, utilisée dans un contexte agricole pour l'acquisition en champ proche de végétation. À partir de six bandes spectrales, cinq modèles ont été testés et déployés dans un framework d'apprentissage profond. Les performances des indices standards et des indices profonds ont été évaluées avec le score mIoU (moyenne de l'intersection sur l'union), démontrant ici la force des DeepIndices pour séparer la végétation du sol.

Mots-clés

Deep-learning, télédétection, indices spectral, images multi-spectrale, agriculture de précision, proxidtection

Abstract

One of the most important advances in the field of earth observation is the discovery of spectral indices, they have proved their effectiveness in the characterization of agricultural plots, but they are generally defined empirically. This study based on artificial intelligence and signal processing, proposes a method to find an optimal index. It focuses on the analysis of images from a multi-spectral camera, used in an agricultural context for near-field acquisition of vegetation. From six wavebands images, five equations have been tested. The performance of standard indices and deep indices were evaluated with the mIoU score (mean intersection over union), demonstrating here the power of DeepIndices to separate vegetation from the soil.

Keywords

Deep-learning, remote sensing, indices spectral, multi-spectral images, precision agriculture, proximal detection

1 Introduction

Les indices spectraux ont prouvé leur efficacité dans la caractérisation des surfaces agricoles. Cependant, ils sont généralement définis de manière empirique. Ainsi de meilleurs modèles permettant de caractériser une surface peuvent exister, pour des paramètres d'acquisitions spécifiques. De plus, ces indices fonctionnent avec des données calibrées en réflectance, ce qui peut être contraignant pour des applications temps réel en extérieur. Ils sont donc peu résistants aux variations d'éclairage [1] et inadéquats sur des images partiellement ombragées. Bien que quelques études existent pour la recherche des meilleurs bandes spectrales et l'optimisation des coefficients (généralement 2 ou 3 bandes) [2], aucune étude ne semble rechercher la meilleure forme par approximation de fonction, utilisant toutes les données, qui permettrait de s'abstraire de ces deux problématiques.

2 Matériel et données

Les images ont été acquises avec la caméra multi-spectrale Airphen (Hyphen, Avignon, France) et configurée en utilisant six bandes spectrales centrées sur 450, 570, 675, 710, 730 et 850 nm. Le choix de ces bandes spectrales a été défini par une étude antérieure [3]. Ainsi 91 images ont été acquises et annotées entre 2019 et 2020 (figure 1) dans des conditions climatiques différentes (soleil, matin, soir, pluvieux, nuageux et partiellement ombragées) et sur deux sites : INRAe (Montoldre, Allier, France) et à AgroSup (Dijon, Bourgogne, France). Ces données sont utilisées pour l'apprentissage (80%) et le test (20%).

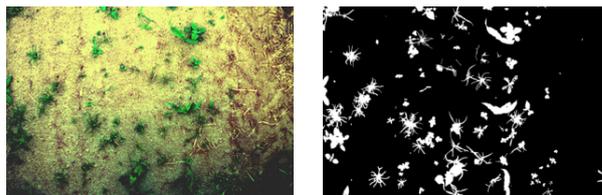


FIGURE 1 – Exemple en fausse couleur d'une image acquise avec la caméra multi-spectrale (gauche) et sont annotation manuelle, utilisée pour l'apprentissage (droite)

3 Indices standards de végétation

À partir de la base de données d'indices IDB [2] (indexdatabase.de), 89 indices de végétation ont été identifiés comme étant compatibles avec nos longueurs d'onde. Ces indices ont été évalués afin de comparer les performances.

4 DeepIndices

Les deux formes les plus récurrentes d'indice de végétation correspondent à une **combinaison linéaire** et à un **rapport de combinaison linéaire** et peuvent être généralisées pour prendre en compte toutes les bandes spectrales. Trois autres modèles sont intéressants à optimiser car ils peuvent approximer toutes fonctions, présentés ci-dessous :

Fonction polynomiale Selon le théorème de Stone-Weierstrass, toute fonction continue définie sur un segment peut être uniformément approchée par une fonction polynomiale. Et permet aussi de résoudre les équations partiellement différentiable [4].

Approximateur de fonction universelle Le modèle gaussien de l'espace colorimétrique [5] montre que la distribution spatio-spectrale de l'énergie de la lumière incidente E est l'intégration pondérée du spectre ρ_i . $E(\rho_i)$ peut être décrit comme une série de Taylor. Utilisable pour décomposer les spectres vers un indice colorimétrique.

Opérateur morphologique dense Comme pour les séries de Taylor, les fonctions continues par morceaux peuvent être approximées par des opérateurs morphologiques tels que la dilatation et l'érosion [6]. Plusieurs érosions et dilatations sont définies pour chaque bande spectrale i avant d'être combinées linéairement.

5 Résultats & Discussion

indices standard	train	test
MTVI 1	81.46	73.71
Enhanced Vegetation Index 2	87.06	67.94
Enhanced Vegetation Index 3	71.52	65.05
NDVI	90.64	63.98

TABLE 1 – Top 4 des scores des test des indices standards

deep indices	train	test
Combinaison linéaire	95.37	80.36
Rapport de combinaison linéaire	95.84	81.35
Fonction polynomiale	95.46	81.21
Approximateur de fonction universelle	95.92	81.49
Opérateur morphologique dense	96.24	82.19

TABLE 2 – Meilleurs scores des DeepIndices

Ces résultats démontrent que les Deep indices sont plus performants dans la segmentation sol/végétation que les indices standards. De plus, ceux-ci, appris sur une base de

données non calibrées et issues de traitement du signal, ne nécessitent pas de calibration préalable à leurs utilisations.

6 Conclusion

Dans ce travail, différents indices standards de végétation ont été évalués ainsi que de nouveaux indices profonds à travers différents modèles permettent de reconstruire la fonction de transfert. Parmi les 89 indices de végétation standard testés, le MTVI (Modified Triangular Vegetation Index 1) donne les meilleurs résultats. Les résultats nous permettent de conclure que toute combinaison linéaire simple est tout simplement plus efficace que l'ensemble des indices standards. Ces modèles avancés suggèrent que les données non calibrées peuvent être utilisées efficacement. Les opérateurs morphologiques denses plus performant, montre également que la solution est une fonction continue par morceaux. Nous convenons donc qu'il est important d'optimiser à la fois la structure arithmétique de l'équation et les coefficients des bandes spectrales. Ils permettent également de prendre en compte la variation de l'éclairage dans l'équation et donc de s'abstraire d'un problème difficile qui est le calibrage radiométrique des données. Ainsi, les images partiellement ombrées sont correctement évaluées, ce qui n'est pas possible avec les indices standards.

Références

- [1] Lifu Zhang, Xuejian Sun, Taixia Wu, and Hongming Zhang. An analysis of shadow effects on spectral vegetation indexes using a ground-based imaging spectrometer. *IEEE Geoscience and Remote Sensing Letters*, 12 :1–5, 11 2015.
- [2] V Henrich, E Götze, A Jung, C Sandow, D Thürkow, and C Gläßer. Development of an online indices database : Motivation, concept and implementation. In *Proceedings of the 6th EARSeL Imaging Spectroscopy SIG Workshop Innovative Tool for Scientific and Commercial Environment Applications, Tel Aviv, Israel*, pages 16–18, 2009.
- [3] Marine Louargant, S. Villette, Gawain Jones, N. Vigneau, J.N. Paoli, and Christelle Gée. Weed detection by uav : simulation of the impact of spectral mixing in multispectral images. *Precision Agriculture*, pages 1–20, 06 2017.
- [4] Hongli Sun, Muzhou Hou, Yunlei Yang, Tianle Zhang, Futian Weng, and Feng Han. Solving partial differential equation based on bernstein neural network and extreme learning machine algorithm. *Neural Processing Letters*, pages 1–20, 2018.
- [5] Jan-Mark Geusebroek, Rein van den Boomgaard, Arnold Smeulders, and Anuj Dev. Color and scale : The spatial structure of color images. pages 331–341, 06 2000.
- [6] Ranjan Mondal, Sanchayan Santra, and Bhabatosh Chanda. Dense morphological network : An universal function approximator. *CoRR*, abs/1901.00109, 2019.

DeepIndices : Une nouvelle approche des indices de télédétection basée sur l'optimisation et l'approximation de fonctions par DeepLearning. Application aux indices de végétation sur des données non calibrées