AfIA

Association française
pour l'Intelligence Artificielle

CNIA

Conférence Nationale en Intelligence Artificielle

PFIA 2021

Crédit photo : Flicr/xlibber

Table des matières

Éditorial

Conférence Nationale en Intelligence Artificielle

La Conférence Nationale en Intelligence Artificielle (CNIA) est organisée au sein de la Plate-Forme Intelligence Artificielle (PFIA) et s'est déroulée du 28 juin au 30 juin 2021, à distance.

CNIA s'adresse à l'ensemble de la communauté en Intelligence Artificielle (IA). Elle est l'occasion de témoigner des dernières avancées en IA et de présenter ses résultats les plus récents dans toutes les disciplines qui la composent.

L'Intelligence Artificielle connait un essor important ces dernières années. Les recherches menées dans les différentes disciplines de l'IA produisent des résultats importants dans différents domaines. Alors que l'IA se trouve au coeur d'un nombre de plus en plus important d'applications, il est nécessaire de croiser ses différentes disciplines, de les intégrer et d'aborder les enjeux technologiques et sociétaux qui découlent du développement de ces systèmes qui ont un impact fort sur notre quotidien. La conférence invitée de Justine Cassel (Professeur à l'Université Carnegie Mellon de Pittsburgh, Etats-Unis et Directrice de recherche à Inria Paris, France) intitulée "Socially-Aware Artificial Intelligence" approfondit et ouvre de multiples questions sur ce thème.

Dans cette démarche, l'objectif de la Conférence Nationale en Intelligence Artificielle (CNIA 2021) est de faire connaitre les dernières avancées dans les différentes disciplines de l'IA, de renforcer les liens et les interactions entre ces différentes disciplines. Elle souhaite ainsi être un point de rencontre pour la communauté IA afin de rapprocher, croiser les recherches disciplinaires et établir des passerelles entre elles.

Cette année, les processus de soumission et de relecture par les membres du comité de programme ont permis de sélectionner 12 articles sur les 17 articles soumis. Ils seront présentés pendant la conférence, à distance.

La conférence accueille également les présentations d'articles acceptés à AAAI et IJCAI, écrits par des équipes françaises. Ces articles pourront être trouvés dans les actes des conférences correspondantes.

Olivier BOISSIER

Comité de programme

Président

— Olivier Boissier, MINES Saint-Étienne, LIMOS

Membres

— Isabelle Bloch, Sorbonne Université, CNRS, LIP6
— Grégory Bonnet, Université de Caen Normandie
— Elise Bonzon, Université de Paris
— Pierre-Antoine Champin, LIRIS, Université Claude Bernard Lyon1
— François Charpillet, Inria Nancy - Grand Est, LORIA
— Sylvie Coste-Marquis, CRIL - CNRS
— Benjamin Dalmas, Mines Saint-Etienne, LIMOS
— Mohamed Daoudi, IMT Lille Douai
— Yves Demazeau, LIG - CNRS
— Arnaud Doniec, IMT Lille Douai
— Helene Fargier, IRIT-CNRS
— Jean-Gabriel Ganascia, Pierre and Marie Curie University - LIP6
— Salima Hassas, Université Claude Bernard-Lyon1
— Nathalie Hernandez, IRIT
— Nicolas Lachiche, University of Strasbourg
— Florence Le Ber, icube
— Marie-Jeanne Lesot, Sorbonne Université - LIP6
— Christophe Marsala, Universite Pierre et Marie Curie - Paris 6
— Engelbert Mephu Nguifo, University Clermont Auvergne - LIMOS - CNRS
— Fabrice Muhlenbach, Laboratoire Hubert-Curien - Université de Saint-Étienne
— Odile Papini, Aix-Marseille Université
— Sylvain Pogodalla, LORIA/INRIA Lorraine
— Catherine Roussey, INRAE
— Olivier Simonin, INSA de Lyon CITI-Inria Lab.
— Catherine Tessier, Onera-DTIS
— Laurent Vercouter, LITIS lab, INSA de Rouen
— Bruno Zanuttini GREYC, Normandie Univ. UNICAEN, CNRS, ENSICAEN

Trouver un équilibre de Nash mixte algébrique dans les jeux sous forme normale et succincts

H. Fargier[1], P. Jourdan[1,2], R. Sabbadin[2]

[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 présente une approche combinatoire pour le calcul exact d'équilibres de Nash dans les jeux à N joueurs, basée sur l'utilisation de nombres algébriques. Cette nouvelle approche est une version algébrique et combinatoire de l'algorithme géométrique proposé par Wilson. Nous fournissons des preuves modernes et "constructives" des résultats de Wilson, permettant d'exploiter des outils logiciels de calcul de bases de Gröbner et de variétés algébriques, disponibles dans des librairies mathématiques efficaces. Nous montrons que notre méthode s'applique également aux jeux hypergraphiques. De plus, le gain en taille de représentation permet une limitation de la complexité au pire cas de l'algorithme.

Mots-clés

Théorie des jeux, Équilibres de Nash, Systèmes polynomiaux

Abstract

This paper presents a combinatorial approach to compute algebraic number representations of exact mixed Nash equilibria in N-person games. This approach is an algebraic, combinatorial version of Wilson's geometric algorithm. The modern and constructive proofs of Wilson's results we provide allow one to exploit algebraic tools, available in efficient mathematic librairies, for computing Groebner bases and varieties. Applying this method to hypergraphical games, we show that the decrease of the size of the representation comes along with a limitation of the worst-case complexity of the algorithm.

Keywords

Game theory, Nash equilibria, polynomial systems

1 Introduction

La méthode géométrique de parcours de chemin proposée par Wilson [20], qui étend l'algorithme de résolution de jeux à deux joueurs de Lemke-Howson [15], permet théoriquement de calculer un équilibre de Nash mixte pour un jeu à N joueurs. Cependant, bien que des implémentations de l'algorithme de Lemke-Howson existent pour les jeux bimatriciels et polymatriciels [13], l'approche de Wilson n'a pas encore mené à un algorithme implémentable pour les jeux à N joueurs. Wilson a montré que trouver un équilibre de Nash mixte revient à trouver une solution à un *Problème de Complémentarité Polynomiale (PCP)* et a suggéré une description mathématique de la résolution des PCP *non dégénérés*. La description de Wilson [20] est informelle et quelques étapes ne sont pas définies. Comme indiqué par l'auteur, dès qu'il y a plus de 2 joueurs, l'étape principale de l'algorithme (le parcours d'arc) : *"requires the solution of a set of simultaneous multi-linear equations (...) which is by no means a trivial presumption"*.

Dans cet article, nous proposons une approche combinatoire, basée sur des principes de géométrie algébrique, pour calculer un équilibre de Nash mixte exact, sous la forme d'un nombre algébrique, dans un jeu à N joueurs sous forme normale [17]. Pour ceci, nous transformons la méthode de parcours de chemin géométrique de Wilson en un algorithme algébrique et combinatoire, et nous proposons une présentation moderne et constructive des résultats de correction de la méthode de Wilson. Dans la procédure de parcours de chemin que nous proposons, les coordonnées des nœuds sont des nombres algébriques, c'est-à-dire sont représentés implicitement par des racines d'un polynôme univarié à coefficients rationnels (ou algébriques eux-mêmes). Ceci permet d'exploiter des logiciels de géométrie algébrique (SINGULAR) pour effectuer les étapes les plus difficiles de l'algorithme.

Dans la section 2, nous présentons la formulation, de Wilson [20] du problème de recherche d'équilibre de Nash en un *Problème de Complémentarité Polynomiale (PCP)*. La section 3 présente un algorithme original de parcours de chemin permettant de résoudre le PCP obtenu. La section 4 illustre cette méthode sur un jeu à trois joueurs. Finalement, la section 5, présente une extension de notre algorithme aux jeux graphiques [14] et hypergraphiques [18].

2 Jeux à N joueurs et problèmes de complémentarité polynomiale

Considérons un jeu à N joueurs $\Gamma^N = (P, \pi, a)$. $P = \{1, \ldots, N\}$ est l'ensemble des joueurs, $\pi = S_1 \times \ldots \times S_N$ est l'ensemble des stratégies jointes pures du jeu (S_n est l'ensemble des stratégies pures du joueur $n \in P$). a^n_ω est la

désutilité, strictement positive [1], reçue par le joueur n quand la stratégie jointe pure est $\omega \in \pi$. $a = (a^n_\omega)_{\omega\in\pi, n\in P}$ est la matrice de désutilités, ayant $|\pi| = \prod_{i=1..N} |S_i|$ lignes et N colonnes.
Une stratégie mixte du joueur n, $\xi^n = (\xi^n_i)_{i\in S_n}$, est une distribution de probabilité sur les stratégies pures de n. Une stratégie mixte jointe est un N-uplet $\xi = (\xi^n)_{n\in P}$ de stratégies mixtes.
Un équilibre de Nash mixte d'un jeu Γ^N est défini par :

Definition 1 (Équilibre de Nash) *Pour un jeu $\Gamma^N = (P, \pi, a)$. Une stratégie jointe mixte $\xi = (\xi^n_i)_{n\in P, i\in S_n}$ est un équilibre de Nash mixte de Γ^N si et seulement si :*

$$ED_n[\xi] \leq ED_n[(\bar{\xi^n}, \xi^{-n})], \forall \bar{\xi^n} \neq \xi^n, \forall n = 1, \ldots, N,$$

$$\text{où } ED_n[\xi] =_{def} \sum_{\omega=(\omega_1,\ldots,\omega_n)\in\pi} a^n_\omega \prod_{i=1}^{N} \xi^i_{\omega_i},$$

et $(\bar{\xi^n}, \xi^{-n})$ est la stratégie jointe mixte où la stratégie mixte ξ^n a été remplacée par la stratégie mixte $\bar{\xi^n}$.

$ED_n[\xi]$ est l'espérance mathématique de la désutilité obtenue par le joueur n lorsque la stratégie jointe mixte ξ est adoptée. En d'autres mots, un équilibre de Nash mixte est une stratégie jointe mixte pour laquelle aucun joueur ne peut diminuer sa désutilité espérée en changeant unilatéralement sa propre stratégie mixte.
Wilson [20] montre qu'il est possible de calculer un équilibre de Nash en résolvant un *Problème de Complémentarité polynomiale (PCP)*, que nous allons décrire. Dans ce PCP, une stratégie mixte du joueur n est indirectement représentée par un vecteur de réels non-négatifs, $x^n = \{x^n_i, i \in S_n\}$. $x = (x^n_i)_{n\in P, i\in S_n}$ représente indirectement une stratégie jointe mixte. x^n n'est pas une distribution de probabilité, mais nous construirons une distribution de probabilité associée en normalisant x^n.
Définissons les polynômes multilinéaires suivants :

$$A^n_i(x^{-n}) = \sum_{\substack{\omega\in\pi \\ \omega_n=i}} a^n_\omega \prod_{\nu\neq n} x^\nu_{\omega_\nu}, \forall n \in P, i \in S_n \quad (1)$$

$$A^n(x) = \sum_{i\in S_n} A^n_i(x^{-n}) x^n_i, \forall n \in P, \quad (2)$$

où, par définition, $x^{-n} = (x^\nu_i)_{\nu\in P\setminus\{n\}, i\in S_\nu}$.
Quand x^n est normalisé, c.à.d quand $x^n = \xi^n$ est une stratégie mixte, $A^n(\xi)$ est la désutilité espérée obtenue par le joueur n sous la stratégie jointe ξ et $A^n_i(\xi^{-n})$ est celle obtenue quand seul n dévie de sa stratégie mixte ξ^n en choisissant la stratégie pure $i \in S_n$.

Definition 2 (Problème de complémentarité polynomiale) *Soit $\Gamma^N = (P, \pi, a)$ un jeu sous forme normale. Le problème de complémentarité polynomiale correspondant est le système d'équations/inéquations suivant, pour les variables réelles $(x^n_i)_{n\in P, i\in S_n}$:*

$$\forall (n,i) \in I_N, \begin{cases} x^n_i & \geq & 0 \\ A^n_i(x^{-n}) & \geq & 1 \quad (\mathcal{S}^N) \\ x^n_i \cdot (A^n_i(x^{-n}) - 1) & = & 0 \end{cases}$$

où $I_N =_{def} \{(n,i), 1 \leq n \leq N, i \in S_n\}$.

$D = |I_N|$ est le nombre de variables x^n_i. Le problème $(\mathcal{S}^N)$ est appelé *problème de complémentarité polynomiale* [10] car nous cherchons une solution non négative x satisfaisant, pour tout $(n,i) \in I_N$, soit $x^n_i = 0$, soit $A^n_i(x^{-n}) = 1$ (complémentarité) et les $A^n_i(x^{-n})$ sont des polynômes multivariés de variables $(x^n_i)_{n\in P, i\in S_n}$.
Dans un jeu à deux joueurs ($N = 2$), on peut vérifier que $A^n_i(x^{-n})$ est une fonction linéaire pour tout (n,i). Le problème $(\mathcal{S}^2)$ est alors un *Problème de Complémentarité Linéaire* [15].
Une solution $x = (x^n_i)$ du PCP peut être normalisée afin d'obtenir un ensemble de distributions de probabilité $\xi = (\xi^n_i)$. Wilson [20] montre l'équivalence entre les équilibres de Nash d'un jeu et les solutions du PCP correspondant :

Proposition 1 (Équivalence NE/PCP [20]) *Soit Γ^N un jeu à N joueurs et $(\mathcal{S}^N)$ sa transformation en PCP. Les équilibres de Nash de Γ et les solutions de $(\mathcal{S}^N)$ sont en bijection :*

1. *Soit x, solution de $(\mathcal{S}^N)$. ξ définie par, $\forall (n,i) \in I_N$ $\xi^n_i = \frac{x^n_i}{\sum_{j\in S_n} x^n_j}$, est un équilibre de Nash de Γ^N.*
2. *Soit ξ, un équilibre de Nash mixte de Γ^N et*

$$x^n_i = \left(\frac{\prod_{\nu\neq n} A^\nu(\xi)}{A^n(\xi)^{N-2}}\right)^{\frac{-1}{N-1}} \xi^n_i, \forall (n,i) \in I_N.$$

$x = (x^n_i)_{n\in P, i\in S_n}$ est une solution de $(\mathcal{S}^N)$.

Une solution x d'un PCP est appelée un *nœud complémentaire*. Un point x est appelé *nœud (n,i)-presque complémentaire* s'il satisfait toutes les contraintes de $(\mathcal{S}^N)$, *à la possible exception* [2] d'une unique équation $x^n_i = 0 \cdot (A^n_i(x^{-n}) - 1) = 0$.
Chercher un équilibre de Nash mixte dans un jeu Γ^N revient donc à chercher une solution du PCP $(\mathcal{S}^N)$. Wilson [20] a proposé une approche géométrique de *parcours de chemin* pour résoudre un PCP *non dégénéré*, étendant celle de Lemke-Howson [15]. La description de Wilson laisse quelques étapes de l'algorithme non définies. De plus, comme pointé par l'auteur, la principale étape de l'algorithme (le parcours d'arc) n'était pas implémentable à l'époque. Dans la section suivante, nous proposons une réécriture originale et opérationnelle de l'approche de Wilson.

1. Nous conservons la formulation proposée par Wilson, qui considère que les joueurs minimisent leur désutilité. Tout jeu de maximisation d'utilité peut être représenté par un jeu équivalent de minimisation de désutilité.

2. Ainsi, un nœud complémentaire est (n,i)-presque complémentaire pour toute paire (n,i).

3 Un algorithme combinatoire de résolution de PCP

La version revisitée que nous proposons (incluant le parcours d'arcs, bloc manquant de l'algorithme de Wilson) est basée sur une définition des nœuds, arcs et chemins presque-complémentaires en termes d'ensembles d'équations multilinéaires. Nous détaillons ces définitions ci dessous (Section 3.1). Puis, nous montrons comment ces chemins peuvent être étendus à travers plusieurs niveaux de sous-PCP (Sections 3.2 et 3.3). La section 3.4 est dédiée au problème du parcours d'arc et l'algorithme de résolution du PCP complet est décrit dans la section 3.5.

3.1 Nœuds, arcs, chemins presque-complémentaires

Considérons un PCP $(\mathcal{S}^N)$. $\mathcal{D}^N$ est l'ensemble des points x satisfaisant l'ensemble des inéquations du PCP :

$$\mathcal{D}^N = \left\{ x = (x_i^n)_{(n,i)\in I_N}, x_i^n \geq 0,\ A_i^n(x^{-n}) \geq 1, \forall (n,i) \right\}.$$

Une solution x du PCP est un nœud *complémentaire* de $\mathcal{D}^N$. Donc, pour toute paire $(n,i) \in I_N$, nous avons soit $x_i^n = 0$, soit $A_i^n(x^{-n}) = 1$. Pour tout point $x \in \mathcal{D}^N$, nous écrivons $Z(x) = \{(n,i) \in I_N, x_i^n = 0\}$ et $W(x) = \{(n,i) \in I_N, A_i^n(x^{-n}) = 1\}$. Par définition, $x \in \mathcal{D}^N$ est une solution de $(\mathcal{S}^N)$ si et seulement si $Z(x) \cup W(x) = I_N$. Un *PCP non dégénéré au niveau* N est défini par :

Definition 3 (PCP non-dégénéré) *Le PCP $(\mathcal{S}^N)$ est non dégénéré au niveau N si et seulement si :*

1. *Aucun point de $\mathcal{D}^N$ ne satisfait plus de $|I_N|$ équations.*
2. *Aucune paire de points distincts de $\mathcal{D}^N$ ne satisfait le même ensemble de $|I_N|$ équations.*

En termes mathématiques, la condition 1 est équivalente à

$$\forall x \in \mathcal{D}^N, |Z(x)| + |W(x)| \leq D = |I_N|$$

et la condition 2 est équivalente à : $\forall x, y \in \mathcal{D}^N$,

$$\left.\begin{array}{rcl} Z(x) = Z(y) & = & Z \\ W(x) = W(y) & = & W \\ |Z| + |W| & = & |I_N| \end{array}\right\} \Rightarrow x = y. \qquad (3)$$

Intuitivement, un PCP est non-dégénéré lorsqu'aucune de ses contraintes polynomiales n'est redondante. A partir de maintenant, nous supposons la non dégénérescence du PCP. Nous définissons également la notion de *nœud presque-complémentaire* :

Definition 4 (Nœud presque-complémentaire) *$x \in \mathcal{D}^N$ est un nœud presque-complémentaire de $\left(\mathcal{S}^N\right)$, si et seulement si : $|Z(x)| + |W(x)| = |I_N|$ et $|Z(x) \cap W(x)| \leq 1$.*

En particulier, un nœud presque-complémentaire est complémentaire si $Z(x) \cup W(x) = I_N$ et (n,i)-presque complémentaire si $(n,i) \notin Z(x) \cup W(x)$. Si $(\mathcal{S}^N)$ est non dégénéré, un nœud presque-complémentaire de $\mathcal{D}^N$ est représentable par une unique paire $Z, W \subseteq I_N$ vérifiant $|Z| + |W| = |I_N|$ et $|Z \cap W| \leq 1$ et telle que x satisfait le système :

$$\left\{ \begin{array}{rl} & x \in \mathcal{D}^N, \\ x_i^n = 0, & \forall (n,i) \in Z, \\ A_i^n(x^{-n}) = 1, & \forall (n,i) \in W. \end{array} \right. \qquad (\mathcal{S}^{Z,W})$$

Pour toute paire $(Z,W) \subseteq I_N$ telle que le système $\left(\mathcal{S}^{Z,W}\right)$ admet une solution, nous notons $\rho(Z,W)$ cette solution (unique puisque le PCP est non-dégénéré). Définissons également un arc (n,i)-presque complémentaire :

Definition 5 (Arc presque complémentaire) *Considérons un PCP $(\mathcal{S}^N)$ non dégénéré. Pour toute paire $(Z,W) \subseteq I_N$ telle que $Z \cap W = \emptyset$ et $Z \cup W = I_N \setminus \{(n,i)\}$, notons $\gamma(Z,W)$, l'ensemble des points satisfaisant le système d'équation $(\mathcal{S}^{Z,W})$. Si il est non-vide, $\gamma(Z,W)$ est appelé* arc (n,i)-presque complémentaire *du PCP $(\mathcal{S}^N)$.*

Si le PCP est non-dégénéré et si $\gamma(Z,W)$ n'est pas vide, $\gamma(Z,W)$ est inclus dans l'ensemble des solutions d'un système de $D-1$ équations à D variables. La non dégénérescence implique que cet ensemble est de dimension 1 et qu'il peut être paramétré par un unique paramètre réel.
Remarque : Les points extrêmes de $\gamma(Z,W)$ appartiennent à la frontière de $\mathcal{D}^N$ et sont donc des nœuds presque-complémentaires. Un arc presque-complémentaire possède au maximum deux points extrêmes.
L'approche de [20] consiste intuitivement à suivre un chemin unidimensionnel dans $\mathcal{D}^N$, constitué d'arcs et nœuds presque-complémentaires partant d'un nœud initial (décrit plus loin) jusqu'à atteindre un nœud complémentaire, solution du PCP. Cette approche est fondée sur la proposition suivante :

Proposition 2 (Arcs voisins d'un nœud) [3] *Soit $(\mathcal{S}^N)$ un PCP non dégénéré et $\rho(Z,W)$ un nœud presque complémentaire. $\rho(Z,W)$ possède deux arcs voisins : $\gamma(Z \setminus W, W)$ et $\gamma(Z, W \setminus Z)$. Si $\rho(Z,W)$ est complémentaire, un seul de ces deux arcs est borné, alors que les deux sont bornés si il n'est pas complémentaire.*
Un arc est non borné lorsqu'il n'est voisin que d'un nœud.

La proposition 2 implique la proposition suivante :

Proposition 3 (Chemin fini) *Soit $(\mathcal{S}^N)$, un PCP non dégénéré et $\rho(Z,W)$, un nœud complémentaire de $(\mathcal{S}^N)$. Soit également $i \in S_N$. Il existe un unique chemin, constitué d'un nombre fini de nœuds et arcs (N,i)-presque complémentaires, dont une extrémité est $\rho(Z,W)$.*

La proposition 3 ne nous dit rien a propos de l'autre extrémité du chemin. Celui-ci peut aussi bien se terminer par un autre nœud complémentaire que par un arc (N,i)-presque complémentaire non borné.
Dans la suite, nous allons voir comment cet unique chemin peut être étendu à travers plusieurs "niveaux" de PCP dérivés du PCP initial, jusqu'à ce qu'il atteigne un nœud *originel*, facile à calculer.

3. Nos preuves sont disponibles (en Anglais) ici : https://figshare.com/s/230ec6a0a3c4a869db8f.

3.2 Séquence de PCP sur différents niveaux

Considérons un PCP $\left(\mathcal{S}^N\right)$ obtenu à partir d'un jeu Γ^N, une stratégie jointe arbitraire $\omega^0 = (\omega_1^0, \ldots, \omega_N^0)$ de Γ^N, deux entiers $1 \leq n \leq k \leq N$ et une stratégie pure $i \in S_n$. Nous écrivons $A_i^{n,k}\left(x^{\{1,..,k\}\setminus\{n\}}\right)$ le polynôme multivarié obtenu à partir $A_i^n(x^{-n})$ en fixant toute les valeurs x_j^ν à 0 lorsque $\nu > k$ et $j \neq \omega_\nu^0$ et à 1 lorsque $\nu > k$ et $j = \omega_\nu^0$:

$$A_i^{n,k}\left(x^{\{1,..,k\}\setminus\{n\}}\right) = \sum_{\substack{\omega\in\pi,\omega_n=i \\ \omega_m=\omega_m^0,\forall m>k}} a_\omega^n \prod_{\substack{\nu\leq k, \\ \nu\neq n}} x_{\omega_\nu}^\nu. \quad (4)$$

$A_i^{n,k}$ est un polynôme multilinéaire de degré $k-1$. Notons que si ξ est une stratégie jointe mixte de Γ^N, $A_i^{n,k}\left(\xi^{\{1,...,k\}\setminus\{n\}}\right)$ est la désutilité espérée du joueur n jouant l'action $i \in S_n$ quand les joueurs $1, \ldots, k$, à l'exception de n, jouent leur stratégie mixte dans ξ et les joueurs $j \in \{k+1, \ldots, N\}$ jouent leur stratégie pure ω_j^0.
Alors, à partir du PCP $\left(\mathcal{S}^N\right)$, nous pouvons définir la séquence suivante de sous-PCP $\left(\mathcal{S}^k\right)$, pour $k = 1, \ldots, N-1$:

Definition 6 (Sous-PCP) *Soit $\left(\mathcal{S}^N\right)$ un PCP donné. Pour tout $1 \leq k \leq N$, $I_k = \{(n,i), 1 \leq n \leq k, i \in S_n\}$ et $D_k = |I_k|$ (ainsi $D = D_N$). Nous définissons le sous-PCP $\left(\mathcal{S}^k\right)$ comme le système d'équations/inéquations polynomiales multilinéaires, de variables $(x_i^n)_{(n,i)\in I_k}$:*

$$\forall (n,i) \in I_k, \begin{cases} x_i^n & \geq & 0 \\ A_i^{n,k}\left(x^{\{1,..,k\}\setminus\{n\}}\right) & \geq & 1 \\ x_i^n \cdot \left(A_i^{n,k}\left(x^{\{1,..,k\}\setminus\{n\}}\right) - 1\right) & = & 0 \end{cases}$$

Pour $n = 1$, le PCP $(\mathcal{S}^1)$ est légèrement différent :

$$\begin{cases} x_i^1 & \geq & 0, \forall i \in S_1 \\ x_i^1 \cdot \left(\frac{a^1_{(i,\omega^0_{-1})}}{\min_{j\in S_1} a^1_{(j,\omega^0_{-1})}} - 1\right) & = & 0, \forall i \in S_1 \\ \sum_{i\in S_1} x_i^1 & = & 1 \end{cases}$$

Le sous-PCP $\left(\mathcal{S}^k\right)$ est construit à partir de $\Gamma^k(\omega^0)$, un jeu joué par les k premiers joueurs de Γ^N tandis que les autres joueurs jouent selon ω^0, de la même manière que $\left(\mathcal{S}^N\right)$ est construit à partir de Γ^N. Nous supposons que tous les sous-PCP $\left(\mathcal{S}^k\right)$ sont non-dégénérés. En particulier, supposer que $(\mathcal{S}^1)$ est non-dégénéré sous-entend que le minimum $\min_{j\in S_1} a^1_{(j,\omega^0_{-1})}$ est atteint pour un seul indice j^*. Ainsi, nous obtenons facilement le nœud complémentaire au niveau 1, décrit par $Z^1 = S_1 \setminus \{j^*\}$ et $W^1 = \{j^*\}$. Au niveau $k \geq 2$, nous pouvons définir un système polynomial $\left(\mathcal{S}_k^{Z,W}\right)$ correspondant à $Z, W \subseteq I_k$:

$$\begin{cases} & x \in \mathcal{D}^k, & \\ x_i^n = 0, & \forall (n,i) \in Z, & (\mathcal{S}_k^{Z,W}) \\ A_i^{n,k}(x^{\{1..k\}-\{n\}}) - 1 = 0, & \forall (n,i) \in W, & \end{cases}$$

où $\mathcal{D}^k$ est défini par les inéquations de la définition 6.

3.3 Nœuds complémentaires et initiaux

Nous pouvons exploiter le point de vue combinatoire de la séquence de systèmes d'équations présentée ci dessus pour concevoir un algorithme calculant une séquence de nœuds presque complémentaires jusqu'à ce qu'un nœud complémentaire soit atteint au niveau N.

Proposition 4 (Montée de niveau) *Soit $\left(\mathcal{S}^k\right)$ un sous-PCP de $\left(\mathcal{S}^N\right)$ au niveau $1 \leq k < N$ et une stratégie jointe pure arbitraire ω^0.*
On suppose que, pour $Z, W \subseteq I_k$, $\left(\mathcal{S}_k^{Z,W}\right)$ définit un nœud complémentaire de $\left(\mathcal{S}^k\right)$.
Alors, si $Z' = Z \cup \left\{(k+1, j), j \neq \omega_{k+1}^0\right\}$, $\left(\mathcal{S}_{k+1}^{Z',W}\right)$ définit un arc $(k+1, \omega_{k+1}^0)$-presque complémentaire du sous-PCP $\left(\mathcal{S}^{k+1}\right)$. De plus, cet arc est non borné et ne voisine qu'un seul nœud $(k+1, \omega_{k+1}^0)$-presque complémentaire au niveau $k+1$.

Ce nœud $(k+1, \omega_{k+1}^0)$-presque complémentaire au niveau $k+1$ peut être calculé en essayant de résoudre tous les systèmes $\left(\mathcal{S}_{k+1}^{Z'\cup\{(\nu,j)\},W}\right)$ avec $(\nu, j) \in I_{k+1} \setminus Z'$ et $\left(\mathcal{S}_{k+1}^{Z',W\cup\{(\nu,j)\}}\right)$ avec $(\nu, j) \in I_{k+1} \setminus W$ jusqu'à en trouver un possédant une solution. Un tel système possédant une solution existe (c'est une conséquence du lemme 2 de [20]). De plus, il est unique pour une séquence fixée de PCP non dégénérés. Cette solution est appelée *nœud initial* au niveau $k+1$:

Definition 7 (Nœud initial) *Un nœud initial au niveau $k+1$ est un nœud $(k+1, \omega_{k+1}^0)$-presque complémentaire solution de $\left(\mathcal{S}_{k+1}^{Z,W}\right)$, avec $Z, W \subseteq I_{k+1}$ et tel que seul l'un de ses arcs voisins est borné. Il satisfait aussi :*

$$(k+1, \omega_{k+1}^0) \notin Z \text{ et } (k+1, j) \in Z, \forall j \neq \omega_{k+1}^0.$$

Avec cette définition en tête, la proposition 3 peut être réinterprétée. Elle précise qu'à partir de tout nœud complémentaire au niveau N et pour $i \in S_N$, un chemin unique constitué de nœuds et arcs (N,i)-presque complémentaires conduit soit à un autre nœud complémentaire, soit à un *nœud initial* au niveau N. C'est évidemment également vrai pour tous les niveaux $k \in \{2, \ldots, N\}$: Ces nœuds complémentaires sont à une extrémité d'un chemin (k, ω_k^0)-presque complémentaire pour lequel l'autre extrémité est soit un nœud complémentaire, soit un nœud initial. L'ensemble des nœuds et arcs presque-complémentaires satisfaisant $\mathcal{S}^k$ constitue un ensemble de chemins disjoints dont les formes possibles sont illustrées dans la figure 1.
La procédure de *descente de niveau*, depuis un nœud initial au niveau k vers un nœud complémentaire au niveau $k-1$, peut être définie réciproquement à la procédure de montée.

Proposition 5 (Descente de niveau) *Soit $\left(\mathcal{S}_k^{Z,W}\right)$ définissant un nœud initial au niveau k. Posons $Z' = Z \cap I_{k-1}$ et $W' = W \cap I_{k-1}$.*
Alors, soit $Z' \cap W' = \{(\nu, j)\} \subseteq I_{k-1}$, soit $Z' \cap W' = \emptyset$. Dans le premier cas, l'un des deux systèmes $\left(\mathcal{S}_{k-1}^{Z'\setminus\{(\nu,j)\},W'}\right)$ ou $\left(\mathcal{S}_{k-1}^{Z',W'\setminus\{(\nu,j)\}}\right)$ définit un nœud

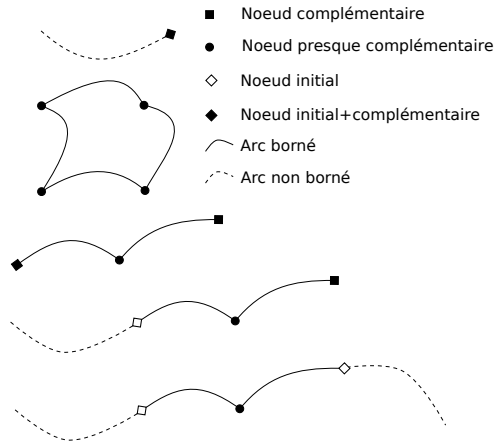


FIGURE 1 – Différentes catégories de chemin presque complémentaire à un niveau donné.

complémentaire au niveau $k-1$*. Dans le second cas,* $\left(\mathcal{S}_{k-1}^{Z',W'}\right)$ *définit un nœud complémentaire au niveau* $k-1$.

3.4 Parcours d'arc algébrique

La brique de base de notre algorithme de parcours de chemin est le *parcours* d'un arc (k,ω_k^0)-presque complémentaire [4] $\gamma^k(Z,W)$, au niveau k ($Z,W \subseteq I_k$), quittant un nœud presque-complémentaire $\rho^k(Z',W')$ avec soit (i) $Z = Z' \setminus W'$ et $W = W'$ soit (ii) $Z = Z'$ et $W = W' \setminus Z'$. Ce problème de traversée d'arc s'exprime en termes algébriques. En effet, par définition :

$$\begin{aligned}\gamma^k(Z,W) &=_{def} \mathcal{V}\left(\mathcal{S}_k^{Z,W}\right) \cap \mathcal{D}^k,\\ \rho^k(Z',W') &=_{def} \mathcal{V}\left(\mathcal{S}_k^{Z',W'}\right) \cap \mathcal{D}^k,\end{aligned}$$

où $\mathcal{V}(\mathcal{S})$ désigne l'ensemble des solutions de $(\mathcal{S})$, en ignorant la contrainte de domaine. $\mathcal{V}(\mathcal{S})$ est appelée une *variété algebrique affine* [4]. Quand le système $(\mathcal{S})$ est non-dégénéré, la variété est finie (elle est de dimension 0) pour un nœud et est de dimension 1 pour un arc.

En pratique, le calcul d'une variété $\mathcal{V}\left(\mathcal{S}_k^{Z,W}\right)$ ou de sa dimension requiert de calculer une *base de Groebner réduite* pour l'ordre lexicographique de l'idéal correspondant [4]. Dans le cas où la variété est de dimension zéro, le *Shape Lemma* [1] montre que cette base réduite consiste en (i) un polynôme en une seule variable et (ii) un polynôme pour chaque autre variable, exprimant sa valeur en fonction d'un polynôme en la première variable. Les solutions d'un tel système à coefficients rationnels sont des *nombres algébriques*, représentables finiment par les équations qui les définissent. Une approche numérique standard peut être utilisée pour résoudre une équation polynomiale à une variable à coefficients rationnels avec une précision arbitraire. Aussi, le système peut être résolu avec une précision arbitraire, en résolvant la première équation puis en injectant la valeur obtenue pour calculer toutes les autres variables.

4. L'exposant k de γ^k ou ρ^k indique que nous sommes au niveau k.

Une base de Groebner réduite pour l'ordre lexicographique peut être calculée via des opérations élémentaires sur des polynômes multivariés (additions, soustractions, multiplications). Le temps et l'espace de calcul sont généralement exponentiels en la "taille" du système [5]. Toutefois, il existe plusieurs implémentations d'algorithmes de calcul de base de Groebner, efficaces pour des problèmes avec des centaines de variables et d'équations [6]. Nous utilisons les fonctions implémentées dans la boite à outil `Singular`, accessible à partir de l'environnement `Sagemath` [6], pour calculer des bases de Groebner, leur dimension et les variétés correspondantes.

Finalement, le problème de parcours d'arc à un niveau k consiste, à partir d'un nœud (n,i)-presque complémentaire $\rho^k(Z',W')$ et d'un arc borné $\gamma^k(Z,W)$ voisin, à calculer l'autre extrémité de l'arc, $\rho^k(Z'',W'')$. L'algorithme 1 calcule ce nœud (n,i)-presque complémentaire en essayant tout les ajouts possibles d'équations au système $\left(\mathcal{S}_k^{Z,W}\right)$.

Algorithme 1 : TRAVERSEARC$((Z,W),(Z',W'),I_k)$

```
/* Calcule l'autre extrémité de l'arc
   γ^k(Z,W) d'extrémité ρ^k(Z',W').        */
/* Initialisation                          */
1  Sol ← ∅;
2  for (ν,j) ∈ I_k do
3      if (ν,j) ∈ I_k \ Z' then
4          Z_loc ← Z ∪ {(ν,j)}, W_loc ← W
5      else
6          if (ν,j) ∈ I_k \ W' then
7              Z_loc ← Z, W_loc ← W ∪ {(ν,j)}
8      if dim(V(S_k^{Z_loc,W_loc})) = 0 then
9          ρ_loc ← V(S_k^{Z_loc,W_loc}) ∩ D^k;
10         if ρ_loc ≠ ∅ then
11             Sol ← Sol ∪ {(Z_loc, W_loc, ρ_loc)}
12 return Sol
```

Lorsqu'un jeu est non-dégénéré, Sol est un singleton. Le cas des jeux dégénérés, non traité dans cet article, requiert une modification de la ligne 11.

Proposition 6 (Correction du parcours d'arc) *Pour un PCP non dégénéré, l'algorithme 1 retourne un unique triplet* $\left(Z'',W'',\mathcal{V}\left(\mathcal{S}_k^{Z'',W''}\right)\right)$ *où* $\mathcal{V}\left(\mathcal{S}_k^{Z'',W''}\right)$ *est un point.*

3.5 Procédure de parcours de chemin

Nous avons presque tous les éléments nécessaires pour construire une procédure de parcours de chemin permettant d'atteindre un nœud complémentaire d'un PCP. Il nous manque seulement l'étape d'initialisation. Cette étape initialise arbitrairement la stratégie jointe pure ω^0 puis résout

5. Voir section 5 pour une discussion sur la complexité de calcul des bases de Groebner.

6. `https://www.sagemath.org/index.html`.

$\left(\mathcal{S}^1\right)$ décrit dans la définition 6. A partir du nœud complémentaire au niveau 1, nous calculons le nœud initial correspondant au niveau 2 en utilisant la proposition 4. Puis, nous suivons le chemin au niveau 2, partant de ce nœud. Si nous atteignons un nœud complémentaire au niveau 2, nous montons au niveau 3, etc. Si, à un niveau k, nous atteignons un nœud initial, non complémentaire, alors nous calculons le nœud complémentaire correspondant au niveau $k-1$ en utilisant la procédure de *descente* (Proposition 5). Puis, nous continuons au niveau $k-1$. Dans le cas où le jeu est non-dégénéré, puisque nous partons d'un nœud complémentaire au niveau 1 et puisque que tout nœud a exactement deux arcs voisins, à l'exception du nœud initial au niveau 1 et des nœuds complémentaires au niveau N, le chemin suivi est unique pour ω^0 fixé et se termine en un nœud complémentaire au niveau N : une solution du PCP. La figure 2 illustre quelques étapes de l'algorithme.

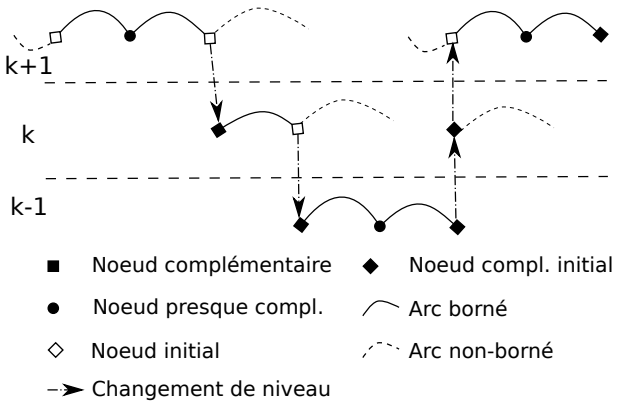


FIGURE 2 – Partie d'un chemin suivi par l'algorithme.

4 Un exemple de jeu à 3 joueurs

Considérons un exemple de jeu à trois joueurs avec deux actions pour chaque joueur, décrit par la table d'utilités/désutilités suivante (les deux représentations sont équivalentes) :

ω_1	ω_2	ω_3	u^1	u^2	u^3	a^1	a^2	a^3
0	0	0	6	0	4	2	8	4
0	0	1	7	3	7	1	5	1
0	1	0	0	7	4	8	1	3
0	1	1	5	4	0	3	4	8
1	0	0	1	6	3	7	2	5
1	0	1	4	5	2	4	3	6
1	1	0	2	1	6	6	7	2
1	1	1	3	2	1	5	6	7

(5)

Nous fixons $\omega^0 = (0,0,0)$. Alors, l'ensemble des polynômes définis par l'équation (4) est :

$$
\begin{aligned}
A_0^{1,1} = 1 \quad ; \quad A_1^{1,1} &= \frac{7}{2} \\
A_0^{1,2}\left(x^2\right) &= 2x_0^2 + 8x_1^2 \\
A_1^{1,2}\left(x^2\right) &= 7x_0^2 + 6x_1^2 \\
A_0^{2,2}\left(x^1\right) &= 8x_0^1 + 2x_1^1 \\
A_1^{2,2}\left(x^1\right) &= x_0^1 + 7x_1^1 \\
A_0^{1,3}\left(x^2,x^3\right) &= 2x_0^2x_0^3 + x_0^2x_1^3 + 8x_1^2x_0^3 + 3x_1^2x_1^3 \\
A_1^{1,3}\left(x^2,x^3\right) &= 7x_0^2x_0^3 + 4x_0^2x_1^3 + 6x_1^2x_0^3 + 5x_1^2x_1^3 \\
A_0^{2,3}\left(x^1,x^3\right) &= 8x_0^1x_0^3 + 5x_0^1x_1^3 + 2x_1^1x_0^3 + 3x_1^1x_1^3 \\
A_1^{2,3}\left(x^1,x^3\right) &= x_0^1x_0^3 + 4x_0^1x_1^3 + 7x_1^1x_0^3 + 6x_1^1x_1^3 \\
A_0^{3,3}\left(x^1,x^2\right) &= 4x_0^1x_0^2 + 3x_0^1x_1^2 + 5x_1^1x_0^2 + 2x_1^1x_1^2 \\
A_1^{3,3}\left(x^1,x^2\right) &= x_0^1x_0^2 + 8x_0^1x_1^2 + 6x_1^1x_0^2 + 7x_1^1x_1^2
\end{aligned}
$$

A partir de cet ensemble de polynômes, l'algorithme de résolution de PCP va parcourir une séquence de nœuds et arcs presque complémentaires définis par des paires $Z, W \subseteq I_k$ au niveau k.

nœud/arc	Z	W	niveau
nœud C	$\{(1,1)\}$	$\{(1,0)\}$	$k=1$
arc	$\{(1,1),(2,1)\}$	$\{(1,0)\}$	$k=2$
nœud PC	$\{(1,1),(2,1)\}$	$\{(1,0),(2,1)\}$	$k=2$
arc	$\{(1,1)\}$	$\{(1,0),(2,1)\}$	$k=2$
nœud PC	$\{(1,1)\}$	$\{(1,0),(1,1),(2,1)\}$	$k=2$
arc	$\emptyset$	$\{(1,0),(1,1),(2,1)\}$	$k=2$
nœud C	$\emptyset$	$\{(1,0),(1,1),$	$k=2$
		$(2,1),(2,2)\}$	$k=2$
arc	$\{(3,1)\}$	$\{(1,0),(1,1),$	$k=3$
		$(2,1),(2,2)\}$	$k=3$
nœud C	$\{(3,1)\}$	$\{(1,0),(1,1),$	$k=3$
		$(2,1),(2,2),(3,0)\}$	$k=3$

Finalement, une solution du PCP vérifie le système suivant, obtenu à partir de la paire (Z,W) définissant le nœud complémentaire au niveau 3 :

$$
\begin{aligned}
x_1^3 &= 0 \\
2x_0^2x_0^3 + 8x_1^2x_0^3 &= 1 \\
7x_0^2x_0^3 + 6x_1^2x_0^3 &= 1 \\
8x_0^1x_0^3 + 2x_1^1x_0^3 &= 1 \\
x_0^1x_0^3 + 7x_1^1x_0^3 &= 1 \\
4x_0^1x_0^2 + 3x_0^1x_1^2 + 5x_1^1x_0^2 + 2x_1^1x_1^2 &= 1
\end{aligned}
$$

Ce système peut être résolu en calculant une base de Groebner puis en résolvant un polynôme à une indéterminée. Ici, on peut en calculer une solution analytique :

$$
\begin{aligned}
x_0^1 = \sqrt{\frac{110}{1377}} \quad ; \quad x_1^1 &= \sqrt{\frac{1078}{6885}} \\
x_0^2 = \sqrt{\frac{18}{935}} \quad ; \quad x_1^2 &= \sqrt{\frac{45}{374}} \\
x_0^3 = \sqrt{\frac{85}{792}} \quad ; \quad x_1^3 &= 0.
\end{aligned}
$$

Après normalisation, cette solution donne l'équilibre de Nash mixte suivant :

$$\begin{aligned}
\xi_0^1 &= \frac{\sqrt{\frac{110}{1377}}}{\sqrt{\frac{110}{1377}}+\sqrt{\frac{1078}{6885}}} \sim 0.41666 \\
\xi_1^1 &= \frac{\sqrt{\frac{1078}{6885}}}{\sqrt{\frac{110}{1377}}+\sqrt{\frac{1078}{6885}}} \sim 0.58333 \\
\xi_0^2 &= \frac{\sqrt{\frac{18}{935}}}{\sqrt{\frac{18}{935}}+\sqrt{\frac{45}{374}}} \sim 0.28571 \\
\xi_1^2 &= \frac{\sqrt{\frac{45}{374}}}{\sqrt{\frac{18}{935}}+\sqrt{\frac{45}{374}}} \sim 0.71429 \\
\xi_0^3 = 1 \quad ; \quad \xi_1^3 &= 0
\end{aligned}$$

5 Jeux graphiques /hypergraphiques

L'approche présentée dans cet article s'étend naturellement aux jeux polymatriciels [21], graphiques [14] et hypergraphiques [18]. Ces jeux sont des représentations succinctes de jeux à N-joueurs où les utilités des joueurs s'expriment à partir d'utilités locales, c-à-d ne dépendant que des stratégies d'un sous ensemble de joueurs de P.
Les jeux hypergraphiques, par exemple, sont définis ainsi :

Definition 8 (Jeu hypergraphique) *Un jeu hypergraphique à N-joueurs Γ^N est défini par*

$$\Gamma^N = \{\{P_g\}_{g=1..G}, \{S_i\}_{i=1..N}, \{a^g\}_{g=1..G}\}, \text{ où}$$

- *$\forall g \in 1,\ldots,G$, $P_g \subseteq P$ est l'ensemble des joueurs du jeu local g et $\cup_{g=1..G} P_g = P$.*
- *$\{S_i\}_{i=1..N}$, est la liste des ensembles de stratégies pures des joueurs.*
- *$a^{g,n}_{\omega_{P_g}}$ est la désutilité locale (positive) d'un joueur n appartenant à P_g dans le jeu local g, lorsque la stratégie jointe des joueurs du jeu g est ω_{P_g}.*

Dans le jeu Γ^N, la désutilité du joueur $n \in N$ pour une stratégie jointe ω est obtenue à partir des désutilités locales :

$$a^n_\omega = \sum_{g, n\in P_g} a^{g,n}_{\omega_{P_g}}.$$

Cette représentation de la matrice de désutilités globale à partir de matrices de désutilités locales permet potentiellement un gain exponentiel d'espace, par rapport à une représentation en forme normale, pour la représentation d'un jeu hypergraphique.
Les jeux poylmatriciels et graphiques sont des cas particuliers de jeux hypergraphiques. Les jeux polymatriciels sont définis par le fait que leurs jeux locaux impliquent exactement deux joueurs : $|P_g| = 2, \forall g = 1,\ldots,G$. Les jeux graphiques sont des jeux dans lesquels la désutilité d'un joueur $n \in P$ ne dépend que des stratégies d'un unique sous-ensemble de joueurs $P_n \in P$. Un jeu graphique peut s'exprimer comme un jeu hypergraphique dans lequel $G = N$ (il y a un jeu local et un seul attaché à chaque joueur) et $a^{g,n}_{\omega_{P_g}} = 0, \forall \omega \in \pi, \forall n \neq g$.
Les polynômes du PCP dérivé d'un jeu hypergraphique exploitent la factorisation des fonctions de désutilité. Ils prennent la forme suivante :

$$A^n_i(x^{-n}) = \sum_{g, n\in P_g} \sum_{\substack{\omega_{P_g}\in\pi_{P_g} \\ \omega_n = i}} a^{g,n}_{\omega_{P_g}} \prod_{\substack{\omega_\nu = j \\ \nu \in P_g\setminus\{n\}}} x^\nu_j. \quad (6)$$

Dans l'expression (6), les degrés des polynômes du PCP sont tous strictement inférieurs au nombre de joueurs du plus grand jeu local. En particulier, lorsque le jeu est polymatriciel, les polynômes sont de degré 1 au plus. Dans ce cas, le problème est un *Linear Complementarity Problem (LCP)*. Howson [13] a exploité cette propriété pour étendre l'algorithme de Lemke Howson aux jeux polymatriciels.
Plus généralement, le fait de pouvoir borner les degrés des polynômes d'un PCP est utile pour un calcul efficace des bases de Groebner. Sans entrer dans les détails ici, des résultats de la littérature montrent que le problème de calcul de bases de Groebner, qui est EXPSPACE-difficile dans le cas général, n'est plus "que" PSPACE-difficile lorsque le système est homogène et l'idéal de dimension zéro [16].
Dans le cas d'un jeu hypergraphique dont le PCP est non-dégénéré, on obtient le résultat de complexité suivant :

Proposition 7 (Complexité) *La complexité de l'algorithme de parcours de chemin pour un jeu graphique/hypergraphique est simplement exponentielle en le nombre de joueurs, le nombre maximum de stratégies pures d'un joueur et doublement exponentielle en le nombre maximal de joueurs d'un jeu local.*

6 Conclusion

La littérature sur l'algorithmique de la recherche d'équilibres approchés dans les jeux à N-joueurs est abondante. Celle-ci inclut des travaux décrivant des approches numériques de parcours d'arc utilisant des *méthodes d'homotopie* [8, 9, 2, 12]. Ces travaux sont basés sur la définition d'un continuum de jeux paramétrés par un paramètre unique et joignant un jeu arbitraire (dont l'équilibre est connu) au jeu qui nous intéresse. Les méthodes d'homotopie suivent alors l'arc des équilibres du continuum des jeux paramétrés.
Par exemple, Datta [5] a proposé une méthode d'homotopie qui exploite également le concept de base de Groebner, mais uniquement pour le calcul d'un équilibre du jeu initial, choisi pour être facilement résolu. Du fait de la précision limitée des méthodes numériques de parcours d'arc, les méthodes de calcul d'équilibres basées sur le concept d'homotopie sont susceptibles aux erreurs numériques et peuvent potentiellement (souvent, en pratique !) ne pas converger.
Plus proche de notre proposition, [19] utilise les concepts d'énumération de supports de stratégies mixtes et de systèmes d'équations polynomiales pour résoudre un jeu à N

joueurs. Cependant, contrairement à notre approche, [19] ne décrit pas dans quel ordre les supports sont énumérés, ni comment les équations polynomiales doivent être résolues.

Au contraire des approches d'homotopie, notre méthode est combinatoire et exploite des principes de géométrie algébrique pour calculer un équilibre de Nash mixte "exact" sous la forme d'un nombre algébrique [3].

Le nombre d'étapes de notre algorithme peut être exponentiel en la taille de description du problème et chaque étape de parcours d'arc est également coûteuse. Cependant, pour un PCP donné, le nombre d'étapes de l'algorithme peut grandement varier selon le choix d'ordre des indices des joueurs et de la stratégie ω^0. Puisque ces choix sont arbitraires, nous pouvons exploiter des ressources de calcul parallèle pour améliorer l'efficacité de notre algorithme, en lançant un lot d'exécutions indépendantes avec différentes initialisations.

Puisque nous utilisons un logiciel tiers (Singular) pour la résolution d'équations polynomiales, nous n'avons pas de prise directe sur la complexité du calcul des bases de Groebner. Néanmoins, nous avons un peu de latitude sur le choix des systèmes lors du parcours d'un arc. Nous pouvons d'abord calculer la base de Groebner de l'idéal (de dimension 1) supportant l'arc, avant de recalculer une base pour chacun des idéaux de dimension 0 correspondant aux équations polynomiales que nous pouvons ajouter pour déterminer l'extrémité de l'arc. Cette approche peut être potentiellement plus efficace que le recalcul complet des bases de Groebner de chacun des idéaux potentiels de dimension nulle. Enfin, pour améliorer l'efficacité de notre approche, nous pouvons maintenir une liste des bases de Groebner de tous les idéaux/systèmes rencontrés et les réutiliser quand cela est approprié.

Pour finir, notre approche peut être étendue à d'autres types de jeux. Une perspective de ce travail est de l'étendre aux jeux bayesiens [11] et aux jeux stochastiques [7]. Ces extension nécessitent la représentation des équilibres de Nash de ces jeux par une solution d'un PCP. Mais puisque ces familles de jeux admettent généralement une représentation en forme normale, cette perspective est potentiellement prometteuse.

Références

[1] E. Becker, T. Mora, M.G. Marinari, and C. Traverso. The shape of the shape lemma. In *ISSAC*, pages 335–342. ACM Press, 1994.

[2] B. Blum, D. Koller, and C.R. Shelton. A continuation method for Nash equilibria in structured games. *Journal of Artificial Intelligence Research*, 25 :457–502, 2006.

[3] J. H. Conway and R. K. Guy. *The Book of Numbers*, chapter Algebraic Numbers, pages 189–190. Springer-Verlagr, New York, 1996.

[4] D.A. Cox, J.B. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms*. Springer, 4 edition, 2015.

[5] R.S. Datta. Finding all Nash equilibria of a finite game using polynomial algebra. *Econonmics Theory*, 42 :55–96, 2010.

[6] J.C. Faugère. A new efficient algorithm for computing groebner bases without reduction to zero (f_5). In *ISSAC*, pages 75–83. ACM Press, 2002.

[7] J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer-Verlag, 1997.

[8] S. Govindan and R. Wilson. A global newton method to compute Nash equilibria. *Journal of Economic Theory*, 110(1) :65–86, 2003.

[9] S. Govindan and R. Wilson. Computing Nash equilibria by iterated polymatrix approximation. *Journal of Economic Dynamics and Control*, 28(7) :1229–1241, 2004.

[10] M.S. Gowda. Polynomial complementarity problems. *Pac. J. Optim*, 3 :227–241, 2017.

[11] 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.

[12] J.J. Herings and R. Peeters. Homotopy methods to compute equilibria in game theory. *Economic Theory*, 42 :119–156, 2010.

[13] J.T. Howson. Equilibria of polymatrix games. *Management Science*, 18(5-part-1) :312–318, 1972.

[14] M. Kearns, M.L. Littman, and S. Singh. Graphical models for game theory. *UAI*, 2001.

[15] C.E. Lemke and J.T. Howson. Equilibrium points of bimatrix games. *Journal of the Society for industrial and Applied Mathematics*, 12(2) :413–423, 1964.

[16] E. W. Mayr. Some complexity results for polynomial ideals. *Journal of Complexity*, 13(3) :303–325, 1997.

[17] J.F. Nash. Equilibrium points in n-person games. *PNAS*, 36(1) :48–49, 1950.

[18] C.H. Papadimitriou and T. Roughgarden. Computing correlated equilibria in multi-player games. *Journal of the ACM (JACM)*, 55(3) :14, 2008.

[19] R. Porter, E. Nudelman, and Y. Shoham. Simple search methods for finding a Nash equilibrium. *Games and Economic Behavior*, 63(2) :664–669, 2008.

[20] R. Wilson. Computing equilibria of n-person games. *SIAM Journal on Applied Mathematics*, 21(1) :80–87, 1971.

[21] E.B. Yanovskaya. Equilibrium points in polymatrix games. *Litovskii Matematicheskii Sbornik*, 8 :381–384, 1968.

Making use of partially observed states in Markov switching autoregressive models: application to machine health diagnosis

F. Dama[1], C. Sinoquet[1]

[1] LS2N / UMR CNRS 6004, Nantes University, France

{fatoumata.dama, christine.sinoquet}@univ-nantes.fr

Résumé

Le diagnostic de bon fonctionnement des machines est une tâche fondamentale dédiée à la surveillance des systèmes dans un souci de sécurité, de prévention des accidents et de programmation des opérations de maintenance. Cette tâche est généralement accomplie grâce à des méthodes d'analyse de données (recourant par exemple à des modèles statistiques). Ces méthodes sont appliquées aux séries temporelles décrivant la dynamique du système analysé. Ces séries temporelles sont sujettes à des changements de régimes reflétant les changements d'état du système. Dans cet article, nous proposons de modéliser de telles séries temporelles par un nouveau modèle de changement de régimes Markovien appelé PHMC-LAR (Partially Hidden Markov Chain AutoRegressive model) où le processus des états décrit l'état de santé du système à chaque pas de temps. Ce modèle possède la capacité d'inclure des connaissances partielles sur le processus des états Markovien. La connaissance partielle est traduite par la présence d'états observés à certains pas de temps aléatoires. Les paramètres de notre modèle sont estimés grâce une variante de l'algorithme d'Espérance-Maximisation, que nous avons développée. La procédure d'inférence des états cachés consiste à identifier la séquence d'états la plus probable pour une série temporelle donnée ; elle est réalisée au moyen de l'algorithme de Viterbi. Les études expérimentales ont été conduites sur des données décrivant des états de machine de façon réaliste. Les résultats montrent que, pour les données utilisées, l'intégration de connaissances partielles sur le processus des états améliore considérablement les performances de l'inférence.

Mots-clés

Séries temporelles, modèle autorégressif, modèle à changement de régimes, états partiellement observés, chaîne de Markov, inférence, diagnostic de l'état d'une machine.

Abstract

Machine health diagnosis is a fundamental task dedicated to monitor systems' safety in order to prevent incidents, and to program maintenance operations. Such diagnosis is achieved through analyzing the system's features (generally recorded by sensors and depicted by time series), using data analytics methods such as statistical models. These time series are subject to regime switches reflecting changes in the system health conditions. In this paper, we propose to model such time series by a new Markov switching autoregressive model called PHMC-LAR (Partially Hidden Markov Chain AutoRegressive), where the state process depicts system health condition at each time-step. This model has the particularity to include partial knowledge about the Markovian state process. This partial knowledge is depicted by the states observed at some (random) time-steps. The parameters of our model are learnt through a variant of the Expectation-Maximization algorithm, which we developed. The inference procedure, that consists in segmenting a given time series into the most likely sequence of states, is addressed by the Viterbi algorithm. Experimental studies are performed on realistic machine condition data. The results show that, for the used datasets, the incorporation of partial knowledge substantially improves inference performance.

Keywords

Time series analysis, autoregressive model, regime-switching model, Markov chain, inference, machine health diagnosis, CMAPSS datasets

1 Introduction

Time series subject to switches in regime have been widely studied in domains such as econometry, finance or meteorology. The underlying dynamical system of such time series is associated with a state process which specifies the system behavior, in other words, its functioning mode at each time-step. Thus, two dynamics can be highlighted : (i) transitions from one state to another, which drive the global nonlinear dynamics of the system and (ii) local stationary dynamics of the time series that unfold within the regimes. The former dynamics is usually modelled through Markov models (HMMs). Linear autoregressive models are widely-used to capture the latter dynamics [5], hence the name of Markov switching autoregressive models.

Two categories of models are depicted in the literature. The first category considers observed states, and corresponds to a classical Markov mixture model with an autoregressive dynamics. The second category considers hidden states. In this case, it is usual to rely on Hidden Markov Models (HMMs) [12, 2, 8]. Models belonging to the first category are referred to as **observed regime-switching models** (OR-

SARs) [3]. The second category describes **hidden regime-switching models** (HRSARs) [9].
In this paper, we present a novel regime-switching autoregressive model which implements the intermediate case where states are partially observed, *i.e.* they are known at some random time-steps and hidden at other time-steps. This model, referred to as the Partially Hidden Markov Chain Linear AutoRegressive (PHMC-LAR), capitalizes on the observed states while the hidden states are inferred. PHMC-LAR is a unification of ORSAR and HRSAR models when the state process is a Markov Chain. The contributions reported in this paper are two-fold : (i) design of a PHMC-LAR learning algorithm, and (ii) analysis of the ability of PHMC-LAR to infer hidden states on time series generated by NASA' Commercial Modular Aero-Propulsion System Simulation (CMAPSS) model [18].
This paper is organized as follows. The PHMC-LAR model is presented in Section 2. Section 3 describes a learning algorithm that allows to estimate the model's parameters. Section 4 presents a hidden state inference algorithm. Section 5 depicts the application of the PHMC-LAR model to real-world datasets, and focuses on state inference. The last Section concludes our paper.

2 PHMC-LAR model

Let $\{X_t\}_{t\in\mathbb{Z}}$ be a time series suject to regime-switching. Let $\{S_t\}_{t\in\mathbb{N}^*}$ the state process of $\{X_t\}$ where $S_t \in \mathbf{K} = \{1, 2, \ldots, K\}$ stands for the state at time-step t and K is the number of states. We suppose that $\{S_t\}$ can be observed at some random time-steps. Let denote by $\sigma_t \subseteq \mathbf{K}$ the set of possible states at time-step t with $\sigma_t = \mathbf{K}$ when S_t is latent and $\sigma_t = \{k\}$ when S_t is observed to be state k. Thus, the σ_t's represent precise partial knowledge available about $\{S_t\}$.

The following notations will be used hereinafter :
• Symbol ':=' stands for the *definition symbol*.
• $A_{t_1}^{t_2}$ denotes the sequence $A_{t_1}, A_{t_1+1}, \cdots, A_{t_2-1}, A_{t_2}$, where t_1 and t_2 ($> t_1$) are time-steps. $A_{t_1}^{t_2}$ may be a sequence of any kind (*e.g.,* a sequence of random variables, a sequence of observed values, a sequence of annotations on a time series).
• By convention, X_{1-p}^0 denotes the p first variables of process $\{X_t\}$; that is why $\{X_t\}$ is indexed in $\mathbb{Z}$.
• Symbols in bold are used to indicate nonscalar variables or vectors of constants.

In this work, the dynamics of the time series $\{X_t\}$ is captured by a PHMC-LAR model. This model takes into account the partial knowledge about the state process $\{S_t\}$ and operates in two stages as follows.

- Modelling of state process. In the PHMC-LAR, we use available partial knowledge (σ_t's) to model the state process $\{S_t\}$ through a *Partially* Hidden Markov Chain (PHMC) [19, 17]. PHMC is an extension of the classical HMM of order 1 [11, 12] in which some states have been observed. PHMC, as classical HMC, is parametrized by a transition matrix

$$a_{i,j} = P(S_t = j | S_{t-1} = i), \quad a_{i,j} \in [0,1], \sum_{j=1}^{K} a_{i,j} = 1$$

and a stationary law

$$\pi_i = P(S_1 = i), \quad \pi_i \in [0,1], \sum_{i=1}^{K} \pi_i = 1.$$

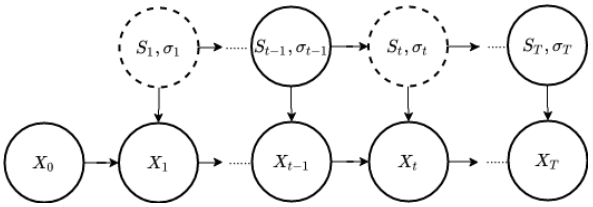


FIGURE 1 Conditional independence graph of the PHMC-LAR model when the autoregressive order $p = 1$. Observed variables (time series and observed states) are displayed in continued line and hidden states are represented in dash line. Note that if the state k has been observed at time-step $t-1$ then σ_t, the set of possible states at this time-step, is reduced to a singleton (*e.g.*, $\sigma_{t-1} = \{k\}$)

- Modelling of the dynamics given the state. Knowing S_t, the state-value at time-step t, together with past values of X_t, we model X_t relying on a linear autoregressive model (LAR) defined as follows :

$$X_t \,|\, X_{t-p}^{t-1}, S_t = k := \mu_k + \sum_{i=1}^{p} \rho_{i,k} X_{t-i} + v_k \epsilon_t, \quad (1)$$

where $\{\epsilon_t\}$ are white noises, p is the number of past values of X_t to be used in modelling, k is the state at time-step t. Within each state k, μ_k is the intercept, $(\rho_{1,k}, ..., \rho_{p,k})$ are the p autoregressive coefficients and v_k is the standard deviation. Note that Eq. 1 is not defined for the p initial values denoted by X_{1-p}^0 which are modelled by some initial law $g_0(x_{1-p}^0; \boldsymbol{\psi})$ parametrized by $\boldsymbol{\psi}$.
Thus, the PHMC-LAR model is parametrized by $(\boldsymbol{\theta}, \boldsymbol{\psi})$ where $\boldsymbol{\theta} = (\boldsymbol{\theta}^{(S)}, \boldsymbol{\theta}^{(X)})$ with $\boldsymbol{\theta}^{(S)} = ((\pi_i)_{i=1,...,K}, (a_{i,j})_{i,j=1,...,K})$, $\boldsymbol{\theta}^{(X)} = (\boldsymbol{\theta}^{(X,k)})_{k=1,...,K}$ and $\boldsymbol{\theta}^{(X,k)} = (\mu_k, \rho_{1,k}, ..., \rho_{p,k}, v_k)$. Figure 1 shows the conditional independence graph of the PHMC-LAR model.

3 Parameter learning

In this section, an approximation of the maximum likelihood estimate (MLE) of the PHMC-LAR model parameters is presented. We consider a training data set of N independent time series $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}$, with $\mathbf{x}_0^{(1)}, \ldots, \mathbf{x}_0^{(N)}$ the corresponding initial values and $\Sigma^{(1)}, \ldots, \Sigma^{(N)}$ (where $\Sigma^{(i)} = \sigma_{t=1}^{T_i}$) the partial knowledge available on the state processes. Notice that the $\mathbf{x}^{(i)}$'s have respective lengths $T_1, \cdots, T_N$, that are not necessarily equal, and that the $\mathbf{x}_0^{(i)}$ terms have length p where p is the autoregressive order.
MLE is the set of parameters that maximizes the likelihood of the training data. It is well-known that in models with latent variables as PHMC-LAR, MLE computation results in an untractable problem. In this case, it

is usual to maximize the **expectation** (*w.r.t.* **the latent variables) of the complete data likelihood** noted $\mathcal{L}^c$. $\mathcal{L}^c$ denotes the evidence/likelihood of the training data $\boldsymbol{x}_0 = (\boldsymbol{x}_0^{(1)}, \ldots, \boldsymbol{x}_0^{(N)})$ and $\boldsymbol{x} = (\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)})$ when latent/hidden variables are set at $\boldsymbol{s} = (\boldsymbol{s}^{(1)}, \ldots, \boldsymbol{s}^{(N)})$. The $\mathcal{L}^c$ writes

$$\begin{aligned}\mathcal{L}^c(\boldsymbol{\theta}, \boldsymbol{\psi}) &= P(\mathbf{X}_0 = \mathbf{x}_0, \mathbf{X} = \mathbf{x}, \mathbf{S} = \mathbf{s}; \boldsymbol{\theta}, \boldsymbol{\psi}) \\ &= P(\mathbf{X} = \mathbf{x}, \mathbf{S} = \mathbf{s} \,|\, \mathbf{X}_0 = \mathbf{x}_0; \boldsymbol{\theta}) \times P(\mathbf{X}_0 = \mathbf{x}_0; \boldsymbol{\psi}) \\ &= \mathcal{L}_c^c(\boldsymbol{\theta}) \times \prod_{i=1}^{N} g_0(\mathbf{x}_0^{(i)}; \boldsymbol{\psi}),\end{aligned} \tag{2}$$

with $\mathcal{L}_c^c$ the **conditional complete data likelihood** and g_0 the initial law of X_t.

Since $\mathbf{s}$ is unkown (it can take any value in $\mathbf{K}^{(\sum_i T_i)}$ where $\mathbf{K}$ is the set of possible states), the expectation of $\mathcal{L}^c$ with respect to the *posterior distribution* of $\boldsymbol{S}$ is considered. In Eq. 2 the second term does not depend on $\mathbf{s}$. Therefore, it can be taken out of the expectation.

$$\begin{aligned}\mathbb{E}_{P(\boldsymbol{S}=\boldsymbol{s} \,|\, \boldsymbol{X}, \boldsymbol{X}_0, \boldsymbol{\Sigma}; \boldsymbol{\theta})}[\mathcal{L}^c(\boldsymbol{\theta}, \boldsymbol{\psi})] = \mathbb{E}_{P(\boldsymbol{S}=\boldsymbol{s} \,|\, \boldsymbol{X}, \boldsymbol{X}_0, \boldsymbol{\Sigma}; \boldsymbol{\theta})}[\mathcal{L}_c^c(\boldsymbol{\theta})] \\ \times \prod_{i=1}^{N} g_0(\mathbf{x}_0^{(i)}; \boldsymbol{\psi}).\end{aligned}$$

By taking the logarithm of the previous expectation we obtain

$$\hat{\boldsymbol{\psi}} = \arg\max_{\boldsymbol{\psi}} \sum_{i=1}^{N} \ln g_0(\mathbf{x}_0^{(i)}; \boldsymbol{\psi}). \tag{3}$$

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \ln \left(\mathbb{E}_{P(\boldsymbol{S}=\boldsymbol{s} \,|\, \boldsymbol{X}, \boldsymbol{X}_0, \boldsymbol{\Sigma}; \boldsymbol{\theta})}[\mathcal{L}_c^c(\boldsymbol{\theta})] \right), \tag{4}$$

where $P(\boldsymbol{S} = \boldsymbol{s} \,|\, \boldsymbol{X}, \boldsymbol{X}_0, \boldsymbol{\Sigma}; \boldsymbol{\theta})$ is the *posterior distribution* of $\boldsymbol{S}$ and $\boldsymbol{\Sigma} = (\Sigma^{(1)}, \ldots, \Sigma^{(N)})$.

When g_0 is assumed to belong to a family of parametric distributions, Eq. 3 can be easily solved. For instance, supposing g_0 is a multivariate normal distribution $\mathcal{N}_p(\mathbf{m}, \mathbf{V})$ with mean $\mathbf{m} \in \mathbb{R}^p$, variance-covariance matrix $\mathbf{V} \in \mathbb{R}^p \times \mathbb{R}^p$, and $\boldsymbol{\psi} = (\mathbf{m}, \mathbf{V})$, we can show that

$$\hat{\mathbf{m}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_0^{(i)}, \quad \hat{\mathbf{V}} = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}_0^{(i)} - \hat{\mathbf{m}}) (\mathbf{x}_0^{(i)} - \hat{\mathbf{m}})', \tag{5}$$

where $'$ stands for matrix transposition.

In contrast, maximization in Eq. 4 is untractable. Therefore $\boldsymbol{\theta}$ is estimated by an instance of the Expectation-Maximization (EM) algorithm. The main idea behind EM is to maximize a lower bound of $\ln \left(\mathbb{E}_{P(\boldsymbol{S}=\boldsymbol{s} \,|\, \boldsymbol{X}, \boldsymbol{X}_0, \boldsymbol{\Sigma}; \boldsymbol{\theta})}[\mathcal{L}_c^c(\boldsymbol{\theta})] \right)$. This lower bound, denoted by Q and defined in Eq. 6, is obtained by applying Jensen's inequality [6]. EM is an iterative algorithm that alternates between E(xpectation) step and M(aximization) step. The E-step computes Q. Then Q is maximized in the M-step. At iteration n, we obtain

$$\text{E-step.} \qquad Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_{n-1}) = \mathbb{E}_{P(\boldsymbol{S}=\boldsymbol{s} \,|\, \boldsymbol{X}, \boldsymbol{X}_0, \boldsymbol{\Sigma}; \hat{\boldsymbol{\theta}}_{n-1})}[\ln \mathcal{L}_c^c(\boldsymbol{\theta})], \tag{6}$$

$$\text{M-step.} \qquad \hat{\boldsymbol{\theta}}_n = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_{n-1}), \tag{7}$$

with $\hat{\boldsymbol{\theta}}_{n-1}$ the estimated parameters at iteration $n-1$ and $P(\boldsymbol{S} = \boldsymbol{s} \,|\, \boldsymbol{X}, \boldsymbol{X}_0, \boldsymbol{\Sigma}; \hat{\boldsymbol{\theta}}_{n-1})$ the associated *posterior distribution* of $\mathbf{S}$.

Step E of EM. This step consists in computing the expectation $Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_{n-1})$. Following the conditional independence graph of the PHMC-LAR model (Fig. 1), the conditional complete data likelihood $\mathcal{L}_c^c$ can be written as a product of marginal and conditional probabilities. We recall that S_t only depends on S_{t-1}, and that X_t depends on S_t and X_{t-p}^{t-1}, hence the two conditional probabilities exhibited in Eq. 8.

$$\begin{aligned}\mathcal{L}_c^c(\boldsymbol{\theta}) &= \prod_{i=1}^{N} P(\mathbf{X}^{(i)} = \mathbf{x}^{(i)}, \mathbf{S}^{(i)} = \mathbf{s}^{(i)} \,|\, \mathbf{X}_0^{(i)}; \boldsymbol{\theta}) \\ &= \prod_{i=1}^{N} \Big[P(S_1^{(i)} = s_1^{(i)}; \boldsymbol{\theta}^{(S)}) \times \\ &\qquad \prod_{t=2}^{T_i} P(S_t^{(i)} = s_t^{(i)} \,|\, S_{t-1}^{(i)} = s_{t-1}^{(i)}; \boldsymbol{\theta}^{(S)}) \times \\ &\qquad \prod_{t=1}^{T_i} P(X_t^{(i)} = x_t^{(i)} \,|\, [X^{(i)}]_{t-p}^{t-1} = [x^{(i)}]_{t-p}^{t-1}, \\ &\qquad\qquad S_t^{(i)} = s_t^{(i)}; \boldsymbol{\theta}^{(X, s_t^{(i)})}) \Big],\end{aligned} \tag{8}$$

with $\boldsymbol{\theta}^{(X,k)}$ the parameters of LAR process associated with state k, and $P(X_t^{(i)} = x_t^{(i)} \,|\, [X^{(i)}]_{t-p}^{t-1} = [x^{(i)}]_{t-p}^{t-1}, S_t^{(i)} = k; \boldsymbol{\theta}^{(X,k)})$ the conditional law of $X_t^{(i)}$ within k.

When the expectation in Eq. 6 is developed and $\mathcal{L}_c^c(\boldsymbol{\theta})$ is substituted by its expression (Eq. 8), we show that $Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_{n-1})$ only depends on the following probabilities

$$\begin{aligned}\xi_t^{(i)}(k, l) = P(S_{t-1}^{(i)} = k, S_t^{(i)} = l \,|\, [X^{(i)}]_{1-p}^{T_i} = [x^{(i)}]_{1-p}^{T_i}, \\ \Sigma^{(i)}; \hat{\boldsymbol{\theta}}_{n-1}), \quad \text{for} \quad t = 2, \ldots, T_i, \quad 1 \leq k, l \leq K.\end{aligned} \tag{9}$$

$$\begin{aligned}\gamma_t^{(i)}(l) = P(S_t^{(i)} = l \,|\, [X^{(i)}]_{1-p}^{T_i} = [x^{(i)}]_{1-p}^{T_i}, \Sigma^{(i)}; \hat{\boldsymbol{\theta}}_{n-1}), \\ \text{for} \quad t = 2, \ldots, T_i, \quad 1 \leq l \leq K.\end{aligned} \tag{10}$$

The $\xi_t^{(i)}$ terms will be used to estimate the transition matrix. Intuitively, for (k, l) fixed, $\sum_t \xi_t^{(i)}(k, l)$ represents the frequency of occurrence of the two consecutive states k, l. The $\gamma_t^{(i)}$ terms are called **smoothed marginal probabilities**; they will be involved in the estimation of parameters $\theta^{(X)}$.

The previous probabilities can be computed through an extension of the *forward-backward* algorithm designed to generate the MLE estimates for HMMs [1]. This extension, called *backward-forward-backward*, adds a supplementary

step to the original algorithm in which available partial knowledge about the state process is exploited. Further details about this algorithm can be found in our previous work [4].

Step M of EM. At iteration n, this step consists in maximizing $Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_{n-1})$ with respect to parameters $\boldsymbol{\theta} = (\boldsymbol{\theta}^{(S)}, \boldsymbol{\theta}^{(X)})$. $Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_{n-1})$ can be decomposed as follows

$$Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_{n-1}) = Q_S(\boldsymbol{\theta}^{(S)}, \hat{\boldsymbol{\theta}}_{n-1}) + Q_X(\boldsymbol{\theta}^{(X)}, \hat{\boldsymbol{\theta}}_{n-1}),$$

where Q_S (respectively Q_X) only depends on parameters $\boldsymbol{\theta}_S$ (respectively $\boldsymbol{\theta}_X$). Thus, the M-step can be split in two maximization sub-steps defined as

$$\hat{\boldsymbol{\theta}}_n^{(S)} = \arg\max_{\boldsymbol{\theta}^{(S)}} \; Q_S(\boldsymbol{\theta}^{(S)}, \hat{\boldsymbol{\theta}}_{n-1}), \tag{11}$$

$$\hat{\boldsymbol{\theta}}_n^{(X)} = \arg\max_{\boldsymbol{\theta}^{(X)}} \; Q_X(\boldsymbol{\theta}^{(X)}, \hat{\boldsymbol{\theta}}_{n-1}). \tag{12}$$

On the one hand, cancelling the first derivative of $Q_S(\boldsymbol{\theta}^{(S)}, \hat{\boldsymbol{\theta}}_{n-1})$ provides the analytical expressions of $\hat{\boldsymbol{\theta}}_n^{(S)}$

$$\hat{a}_{k,l}^{(n)} = \frac{\sum_{i=1}^{N} \sum_{t=2}^{T_i} \xi_t^{(i)}(k,l)}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_t^{(i)}(k)}, \quad \hat{\pi}_l^{(n)} = \frac{\sum_{i=1}^{N} \gamma_1^{(i)}(l)}{N}, \tag{13}$$

for $1 \leq k, l \leq K$, with

$$\begin{aligned}\gamma_1^{(i)}(s) &= P(S_1^{(i)} = s \,|\, [X^{(i)}]_{1-p}^{T_i} = [x^{(i)}]_{1-p}^{T_i}, \Sigma^{(i)}; \hat{\boldsymbol{\theta}}_{n-1}) \\ &= \sum_{j=1}^{K} \xi_2^{(i)}(s,j).\end{aligned}$$

On the other hand, generally, it is difficult to derive the analytical expression of $\hat{\boldsymbol{\theta}}_n^{(X)}$. This is the reason why we are compelled to use a numerical optimization method (*e.g.*, the quasi-Newton method) to maximize $Q_X(\boldsymbol{\theta}^{(X)}, \hat{\boldsymbol{\theta}}_{n-1})$.

To set the starting point of the EM algorithm, we first run instances of EM using several vectors of p initial values chosen at random. The MLE parameters that provide the greatest likelihood across these multiple restarts constitute the starting point.

4 Inference or time series segmentation

Let $\hat{\boldsymbol{\theta}}$ a PHMC-LAR model trained on a partially labelled training dataset. Let $\mathbf{x} = x_1^T$ be an observed time series and $\mathbf{x}_0 = x_{1-p}^0$ the associated initial values. As in classical HMMs, the state process of $\mathbf{x}$ is unknown and inference consists in finding the most likely state sequence $\mathbf{s}^* = (s_1^*, \ldots, s_T^*)$ given $\mathbf{x}$ and $\mathbf{x}_0$. This is equivalent to maximizing, with respect to $\mathbf{s}$, the joint probability of $\mathbf{s} = (s_1, \ldots, s_T)$ and $\mathbf{x}$, given $\mathbf{x}_0$

$$\begin{aligned}\mathbf{s}^* &= \arg\max_{\mathbf{s}} P(\mathbf{S} = \mathbf{s}, \mathbf{X} = \mathbf{x} \,|\, \mathbf{X}_0 = \mathbf{x}_0; \hat{\boldsymbol{\theta}}) \\ &= \arg\max_{\mathbf{s}} P(\mathbf{S} = \mathbf{s} \,|\, \mathbf{X} = \mathbf{x}, \mathbf{X}_0 = \mathbf{x}_0; \hat{\boldsymbol{\theta}}).\end{aligned} \tag{14}$$

Note indeed that $P(\mathbf{S} = \mathbf{s}, \mathbf{X} = \mathbf{x} \,|\, \mathbf{X}_0 = \mathbf{x}_0; \hat{\boldsymbol{\theta}}) = P(\mathbf{S} = \mathbf{s} \,|\, \mathbf{X} = \mathbf{x}, \mathbf{X}_0 = \mathbf{x}_0; \hat{\boldsymbol{\theta}}) \times P(\mathbf{X} = \mathbf{x} \,|\, \mathbf{X}_0 = \mathbf{x}_0; \hat{\boldsymbol{\theta}})$, and that the second term in the product does not depend on $\mathbf{S}$.

Since state sequence $\mathbf{s}$ can take K^T different values where K is the number of possible states, the "greedy search" method that consists in testing all possible values is extremely costly ($\mathcal{O}(K^T)$ operations). Alternatively, the **Viterbi algorithm** [7] allows to compute the optimal state sequence in $\mathcal{O}(TK^2)$ operations.

The Viterbi algorithm operates iteratively, following a dynamic programming algorithm. Let $\delta_t(l; \hat{\boldsymbol{\theta}})$ the maximal likelihood of subsequence $(s_1, \ldots, s_t = l)$ that ends within state l

$$\begin{aligned}\delta_t(l; \hat{\boldsymbol{\theta}}) = \max_{s_1, \ldots, s_{t-1}} P(X_1^t = x_1^t, S_1^{t-1} = s_1^{t-1}, S_t = l \,| \\ \mathbf{X}_0 = \mathbf{x}_0; \hat{\boldsymbol{\theta}}), \quad \text{for } t = 1, 2, \ldots T.\end{aligned} \tag{15}$$

These probabilities can be iteratively computed as follows

$$\begin{aligned}\delta_1(l; \hat{\boldsymbol{\theta}}) = P(X_1 = x_1 \,|\, \mathbf{X}_0 = \mathbf{x}_0, S_1 = l; \boldsymbol{\theta}^{(X,l)}) \times \\ P(S_1 = l; \hat{\boldsymbol{\theta}}^{(S)}).\end{aligned} \tag{16}$$

$$\begin{aligned}\delta_t(l; \hat{\boldsymbol{\theta}}) = \max_k \left[\delta_{t-1}(k; \hat{\boldsymbol{\theta}}) \times P(S_t = l \,|\, S_{t-1} = k; \hat{\boldsymbol{\theta}}^{(S)})\right] \\ \times \; P(X_t = x_t \,|\, X_{t-p}^{t-1} = x_{t-p}^{t-1}, S_t = l; \boldsymbol{\theta}^{(X,l)}),\end{aligned} \tag{17}$$

for $t = 2, \ldots, T$.

Therefore, the maximal probability of the complete state sequence is given by $\max_l \delta_T(l; \hat{\boldsymbol{\theta}})$. Accordingly, the optimal sequence $\mathbf{s}^*$ is retrieved by backtracking as follows

$$s_t^* = \arg\max_l \begin{cases} \delta_T(l; \hat{\boldsymbol{\theta}}) & \text{for } t = T \\ \delta_t(l; \hat{\boldsymbol{\theta}}) \times \hat{a}_{l, s_{t+1}^*} & \text{for } t = T-1, \ldots, 1 \end{cases} \tag{18}$$

5 Application to machine health diagnosis

In this section, we assess the added value of using partial knowledge about the state process of Hidden Markov Chains. This evaluation is carried out on realistic machine condition data generated by the Commercial Modular Aero-Propulsion System Simulation (CMAPSS) model [18] developed at the NASA Army Research Laboratory. Data description is provided in subsection 5.1. Subsection 5.2 presents the feature extraction procedure used in our experiments. Then the generation of ground truth segmentations is explained in subsection 5.3. Subsection 5.4 describes the experimental setting. Finally, results are presented and discussed in Subsection 5.5.

5.1 Data description

The CMAPSS model allows to simulate realistic run-to-failure trajectories of aircraft turbofan engines, under dif-

ferent operational conditions and fault modes. For each simulation, the system begins in healthy state (normal functioning mode) then, at some point, the system health starts degrading and finishes by breaking down (system failure). Besides, each trajectory is described through 3 operational conditions (velocity, altitude and temperature) depicting flight conditions, and 21 time series representing as many system's features.
The NASA dataset repository provides four CMAPSS datasets, with different operational conditions and fault modes (https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/#turbofan). Each such dataset consists of a training dataset and test dataset. The training datasets are composed of run-to-failure trajectories. In contrast, trajectories within test datasets are stopped before system failure. In this work, we consider the respective training datasets of CMAPSS datasets #1 and #3 (namely *train_FD001.txt* and *train_FD003.txt* in the repository). These sets will be further referred to as #1 and #3, to simplify. Both latter sets have a single operational condition, one fault mode for dataset #1, two fault modes for dataset #3 and 100 trajectories each.

5.2 Feature extraction : machine health indicator

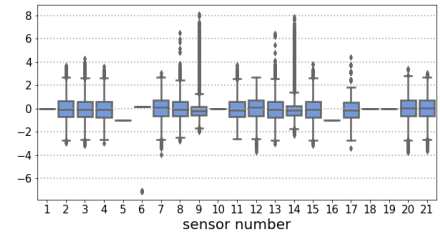
Our model cannot be directly applied to CMAPSS trajectories which are multivariate time series of dimension 21. To overcome this limitation, we reduce the dimension of our data by aggregating relevant features into a single variable called **health indicator** (HI) [13, 15]. HI is a useful indicator of the system health, computed using the following theoretical model

$$\mathrm{HI}_i(t) \equiv 1 - \exp\left(\frac{\log(0.05)}{0.95\,T_i} \times t\right), \quad t \in [\sigma_1, \sigma_2], \quad (19)$$

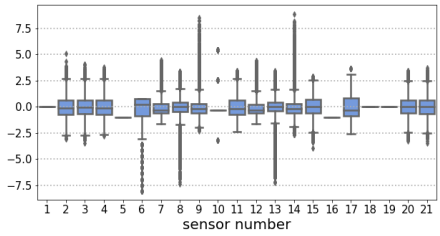
where T_i is the i^{th} trajectory length and σ_1 and σ_2 are respectively set at $T_i \times 5\%$ and $T_i \times 95\%$ as proposed in [13]. Note that the theoretical HI roughly decreases from 1 ("healthy") to 0 ("faulty") when t increases (notice that the term within the exponential is negative).
Then for each trajectory, HI is modelled by a linear regression model for which the predictors are system's features and the response variable is the theoretical HI (Eq. 19) :

$$\mathrm{HI}_i(t) = \eta_0^{(i)} + \sum_{j=1}^{q} \eta_j^{(i)}\, y_{t,j}^{(i)} + \delta_t, \quad (20)$$

where $\mathbf{y}_t^{(i)} = (y_{t,1}^{(i)}, \ldots, y_{t,q}^{(i)})$ is the feature vector of the i^{th} trajectory at time-step t and $\boldsymbol{\eta}^{(i)} = (\eta_0^{(i)}, \ldots, \eta_q^{(i)})$ are the model parameters which can be estimated by the least square method. δ_t's are independent error terms and $\mathrm{HI}_i(t)$ is defined in Eq. 19.
Once parameters $\boldsymbol{\eta}^{(i)}$ have been estimated, HI approximations, denoted by $\hat{\mathrm{HI}}_i(\mathbf{y}_t^{(i)}, \hat{\boldsymbol{\eta}}^{(i)})$ are computed following the linear model (Eq. 20). In our experiments, the subset of features $\{2, 3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20, 21\}$ that display significant variations over time (as illustrated in Fig. 2) is considered for both datasets #1 and #3. For dataset #3, feature 6 is added to this latter subset (see Fig. 2b). Figure 3 shows the estimated HI for the 100 trajectories within each of training datasets #1 and #3.

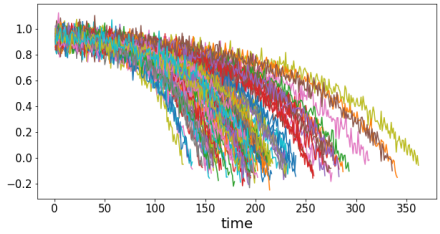


(a)

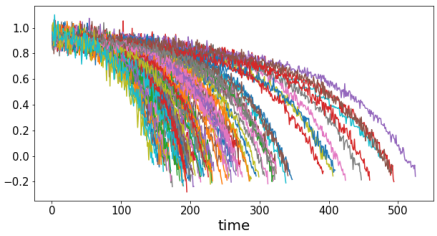


(b)

FIGURE 2 Distribution of sensor measurements for the 100 trajectories within (a) training dataset #1, (b) training dataset #3. Data have been standardized in order to show the same scale order across all 21 sensors



(a)

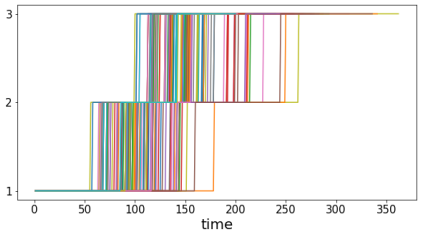


(b)

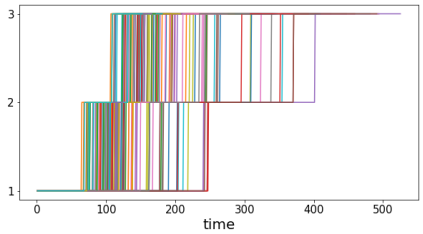
FIGURE 3 Estimated health indicator for the 100 trajectories within (a) training dataset #1, (b) training dataset #3

5.3 Ground truth segmentation

To note, in CMAPSS trajectories, system degradation levels or health states are not specified. However, we need a "ground truth" segmentation of trajectories to both feed

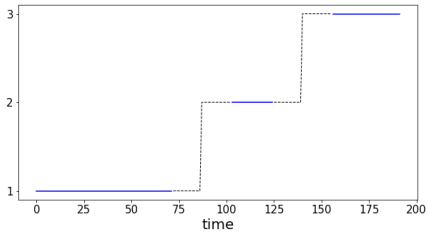


(a)

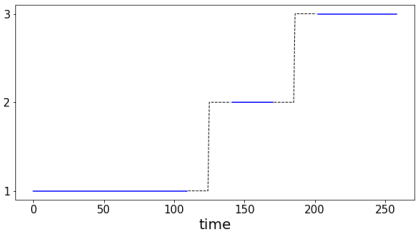


(b)

FIGURE 4 Ground truth segmentation for the 100 trajectories within (a) training dataset #1, (b) training dataset #3. Three health states or degradation levels are considered : 1 for "healthy", 2 for "intermediate" and 3 for "faulty". Each trajectory is assigned a specific color



(a)



(b)

FIGURE 5 Partial knowledge about state process of the 13^{th} trajectory within (a) training dataset #1, (b) training dataset #3. Two windows of length 31 centered around the switch from regime 1 to regime 2 and regime 2 to regime 3 respectively are considered. The observations outside these windows are labelled (solid line), whereas dash line denotes hidden states (within the windows)

PHMC-LAR with partial knowledge and validate our models. Three health states or degradation levels are considered : 1 for "healthy", 2 for "intermediate" and 3 for "faulty". For each trajectory, the "ground truth" segmentation is obtained by deterministically splitting the corresponding health indicator time series (see Fig. 3) into 3 regimes ("healthy", "intermediate" and "faulty") where a regime is a succession of time-steps having the same health state. In this work, the automatic segmentation method proposed in [14, 16] and used by [10] has been considered. This method relies on linear regression models. Figure 4 shows the "ground truth" segmentation of the 100 trajectories within training datasets #1 and #3. Note the high variability of regime duration from one trajectory to another. This suggests the difficulty for a model to infer the critical moments where system health degrades.

5.4 Experimental setting

In our experiments, trajectories within file *train_FD001.txt* and *train_FD003.txt* were split into a training dataset (60 trajectories), a validation and test datasets (20 trajectories each). Gaussian white noises were considered and the initial law g_0 was Gaussian too. The number of states K was set to 3 and a grid of values $\{1, \ldots, 6\}$ was tested for the autoregressive order p. For each candidate model, inference performance was evaluated on the validation dataset, relying on the "ground truth" segmentations (see Fig. 4), and using the mean percentage error (MPE) score. Then, the model yielding the highest inference performance was identified. Finally, the global accuracy for health state inference was assessed by computing the confusion matrix for each test dataset.

We remind that we wish to assess the added value of using partial knowledge about the state process of Hidden Markov Chains. Therefore, two modalities are considered : (i) the fully unsupervised case, in which no partial knowledge is included, referred to as Hidden Markov Chain Linear Autoregressive model (HMC-LAR) ; and (ii) the semi-supervised case denoted by PHMC-LAR. Both HMC-LAR and PHMC-LAR models are fed with health indicator data (see Fig. 3). Note that in (ii), partial knowledge is obtained from the "ground truth segmentations" (see Fig. 4) of the 60 trajectories of previous traning datasets as subsequently described. Two windows of length 31 centered around the time-steps at which the system switches from one regime to another are considered ; the observations outside these windows are labelled (see Figure 5). Note that these windows represent 22% to 48% of the training trajectories' lengths in dataset #1 against 12% to 39% in dataset #3. The readers'attention is drawn to the fact that the trajectories within the validation and test datasets are kept fully unlabelled.

5.5 Results

Table 1 displays the MPE (Mean Percentage Error) values computed on validation dataset, for different values of the autoregressive order. The results show that in the fully unsupervised tuning (HMC-LAR), the highest inference performance (reflected by lowest MPE) is obtained when $p = 3$ for dataset #1 and $p = 5$ for dataset #3. However, more

parsimonious models are selected when partial knowledge about state process is included, since for both datasets the best model has an autoregressive order equal to one ($p = 1$). Note that for both datasets, we observe that PHMC-LAR outperforms HMC-LAR with an inference accuracy five times greater.

For the best HMC-LAR models ($p = 3$ for dataset $\#1$ and $p = 5$ for dataset $\#3$), confusion matrices resulting from the inference on test trajectories are presented in Table 2. Notice that low inference accuracies are obtained for both datasets : 39% for dataset $\#1$ and 45% for dataset $\#3$. The confusion matrices show that HMC-LAR models globally fail in identifying the three regimes since too many "healthy" states are inferred as "faulty" and reversely.

Considering the best PHMC-LAR models ($p = 1$ for both datasets), confusion matrices are presented in Table 3. Unlike the fully unsupervised cases (HMC-LAR models), satisfying global accuracies are reached : 88% for dataset $\#1$ and 89% dataset $\#3$. Moreover, no confusion is made between the "healthy" and "faulty" regimes. An analysis of the transition matrices of PHMC-LAR models shows that the system health state never steps back : in other words, transitions "intermediate" $\rightarrow$ "healthy", "faulty" $\rightarrow$ "intermediate" and "faulty" $\rightarrow$ "healthy" have a null probability. Thus, the terms located above the diagonal in the confusion matrices represent anticipations of system health degradation (*i.e.*, the anticipation of the transition from a health condition to a worse one), whereas those located below the diagonal depict delays in the detection of system health degradation. Therefore, almost all inference errors are due to anticipations of system health degradation since the confusion matrices are almost upper triangular. It has to be underlined that in the machine health monitoring literature, health degradation anticipation is a desirable behavior in comparison with models that detect health degradation with some delay. In a sense, this anticipation allows to prevent serious incidents and to program maintenance operations.

p	**CMAPSS dataset** $\#1$		**CMAPSS dataset** $\#3$	
	HMC-LAR	**PHMC-LAR**	**HMC-LAR**	**PHMC-LAR**
1	68 ± 9 %	**10 ± 5.4 %**	75 ± 7.4 %	**12 ± 4.4 %**
2	69 ± 4.2 %	12 ± 8.1 %	73.9 ± 8.2 %	21 ± 13 %
3	**59 ± 10 %**	16 ± 12.5 %	73.7 ± 8.2 %	25 ± 13 %
4	71 ± 6.8 %	18 ± 11.2 %	72 ± 9.5 %	52 ± 5.1 %
5	73 ± 1.8 %	26 ± 17 %	**60 ± 18.5 %**	52 ± 5.2 %
6	64 ± 12 %	55 ± 4.4 %	73 ± 5.7 %	35 ± 12 %

TABLE 1 Mean percentage inference error (MPE) computed on validation datasets (20 trajectories of different lengths) for both unsupervised case (HMC-LAR) and semi-supervised case (PHMC-LAR). p is the autoregressive order. The minimum MPE values are displayed in bold

CMAPSS dataset $\#1$

		prediction		
		healthy	intermediate	faulty
ground truth	healthy	**1288**	1	825
	intermediate	697	**0**	437
	faulty	745	4	**436**

CMAPSS dataset $\#3$

		prediction		
		healthy	intermediate	faulty
ground truth	healthy	**1551**	271	625
	intermediate	755	**162**	242
	faulty	607	183	**445**

TABLE 2 Confusion matrices of test datasets (20 trajectories of different lengths) for the best HMC-LAR model. For dataset $\#1 : p = 3$ with a global accuracy equal to 39%. For dataset $\#3 : p = 5$ with a global accuracy equal to 45%

CMAPSS dataset $\#1$

		prediction		
		healthy	intermediate	faulty
ground truth	healthy	**1964**	190	0
	intermediate	9	**815**	310
	faulty	0	0	**1185**

CMAPSS dataset $\#3$

		prediction		
		healthy	intermediate	faulty
ground truth	healthy	**2234**	293	0
	intermediate	0	**914**	245
	faulty	0	7	**1228**

TABLE 3 Confusion matrices of test datasets (20 trajectories) for the best PHMC-LAR model : $p = 1$ for both datasets, with a global accuracy equal to 88% for dataset $\#1$ and 89% for dataset $\#3$

6 Conclusion

In this work, we have presented a new Markov switching autoregressive model which incorporates partial knowledge about the state process. This partial knowledge is represented by the states observed at some random time-steps. This model, referred to as PHMC-LAR (for Partially Hidden Markov Chain Linear AutoRegressive) model, is a generalization of the observed regime-switching models (ORSARs) and the hidden regime-switching models (HRSARs).

In the evaluation, the inference performance of PHMC-LAR model has been compared to that of the fully unsupervised HMC-LAR (Hidden Markov Chain Linear AutoRegressive) model. To this end, realistic machine condition

data available in NASA's CMAPSS datasets has been used. Three health states have been considered ("healthy", "intermediate" and "faulty") and "ground truth segmentation" has been derived. For both models, six values of the autoregressive order p (from 1 to 6) have been tested. The results show that parsimonious models (reflected by small values of p) are selected in PHMC-LAR model. Moreover, PHMC-LAR substantially outperforms the fully unsupervised HMC-LAR model. A further analysis of the confusion matrices computed on test datasets shows that PHMC-LAR is more able to anticipate the degradation of system health than HMC-LAR. Such anticipation capability is a desirable behavior in the literature of machine health diagnosis.

In future work, the multivariate extension of PHMC-LAR will be considered. Such exention will allow to directly model the relevant features of CMAPSS trajectories without any dimension reduction. On the other hand, following a case-based reasoning approach, PHMC-LAR can be adapted to failure prognostic task, which consists in predicting the remaining useful life, that is the number of time-steps remaining before system failure.

Acknowledgements

The software development and the realization of the experiments were performed at the CCIPL (Centre de Calcul Intensif des Pays de la Loire, Nantes, France).

References

[1] L.E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The annals of mathematical statistics*, 41(1) :164–171, 1970.

[2] J. Berg, T. Reckordt, C. Richter, and G. Reinhart. Action recognition in assembly for human-robot-cooperation using Hidden Markov Models. *Procedia CIRP*, 76 :205–210, 2018.

[3] J. Bessac, P. Ailliot, J. Cattiaux, and V. Monbet. Comparison of hidden and observed regime-switching autoregressive models for (u, v)-components of wind fields in the Northeast Atlantic. *Advances in Statistical Climatology, Meteorology and Oceanography*, 2(1) :1–16, 2016.

[4] F. Dama and C. Sinoquet. Partially Hidden Markov Chain Linear Autoregressive model : inference and forecasting. *arXiv preprint arXiv :2102.12584*, 2021.

[5] A.B. Degtyarev and I. Gankevich. Evaluation of hydrodynamic pressures for autoregressive model of irregular waves. In *Contemporary Ideas on Ship Stability*, pages 37–47. 2019.

[6] S.S. Dragomir. Some refinements of Jensen's inequality. *Journal of mathematical analysis and applications*, 168(2) :518–522, 1992.

[7] G.D. Forney. The Viterbi algorithm. *Proceedings of the IEEE*, 61(3) :268–278, 1973.

[8] K. Ghasvarian Jahromi, D. Gharavian, and H. Mahdiani. A novel method for day-ahead solar power prediction based on hidden Markov model and cosine similarity. *Soft Computing*, 24(7) :4991–5004, 2020.

[9] J.D. Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, pages 357–384, 1989.

[10] P. Juesas and E. Ramasso. Ascertainment-adjusted parameter estimation approach to improve robustness against misspecification of health monitoring methods. *Mechanical Systems and Signal Processing*, 81 :387–401, 2016.

[11] S. Morwal, N. Jahan, and D. Chopra. Named entity recognition using hidden Markov model (HMM). *International Journal on Natural Language Computing*, 1(4) :15–23, 2012.

[12] R. Mouhcine, A. Mustapha, and M. Zouhir. Recognition of cursive Arabic handwritten text using embedded training based on HMMs. *Journal of Electrical Systems and Information Technology*, 5(2) :245–251, 2018.

[13] E. Ramasso. Investigating computational geometry for failure prognostics. *International Journal of prognostics and health management*, 5(1) :005, 2014.

[14] E. Ramasso. Segmentation of CMAPSS health indicators into discrete states for sequence-based classification and prediction purposes. Technical report, 6839, FEMTO-ST institute, 2016.

[15] E. Ramasso. RULCLIPPER algorithm and CMAPSS health indicators. MATLAB Central File Exchange, 2021.

[16] E. Ramasso. Segmentation of CMAPSS trajectories into states. MATLAB Central File Exchange, 2021.

[17] E. Ramasso and T. Denoeux. Making use of partial knowledge about hidden states in HMMs : an approach based on belief functions. *IEEE Transactions on Fuzzy Systems*, 22(2) :395–405, 2013.

[18] A. Saxena, K. Goebel, D. Simon, and N. Eklund. Damage propagation modeling for aircraft engine run-to-failure simulation. In *2008 international conference on prognostics and health management*, pages 1–9. IEEE, 2008.

[19] T. Scheffer and S. Wrobel. Active learning of partially Hidden Markov Models. In *In Proceedings of the ECML/PKDD Workshop on Instance Selection*. Citeseer, 2001.

Éthique et IA : analyse et discussion

C. Tessier[1]

[1] ONERA/DTIS, Université de Toulouse

catherine.tessier@onera.fr

Résumé

La profusion de documents ainsi que d'instances créées pour traiter de « l'éthique de l'IA » amène à s'interroger sur les raisons pour lesquelles l'IA est devenue, depuis quelques années, un objet particulier d'attention, pourquoi cet objet est spécifiquement regardé sous un angle dit « éthique » et de quelle éthique il s'agit. L'examen des textes européens, de l'UNESCO, de l'OCDE et de la déclaration de Montréal révèle notamment une interprétation sémantique des notions qui peut prêter à confusion, et des postulats susceptibles d'affecter les réflexions. Des tensions et paradoxes peuvent être mis en évidence, que nous illustrons en particulier sur le principe du contrôle humain. Nous insistons en conclusion sur les risques de dévoiement de l'éthique et la nécessité d'une véritable réflexion éthique accompagnant les évolutions techniques et applicatives en matière d'IA.

Mots-clés

Intelligence artificielle, éthique, tensions, contrôle humain.

Abstract

The high number of documents as well as bodies created to deal with « the ethics of AI » leads us to wonder why AI has become, in recent years, a particular object of attention, why AI is specifically looked at from the « ethics » point of view and which ethics is at stake. A review of European, UNESCO, OECD documents and of the Montreal Declaration reveals a semantic interpretation of notions that can lead to confusion, and some postulates that can be misleading. Tensions and paradoxes can be highlighted, which we illustrate in particular on the principle of human control. In conclusion, we insist on the risks of misuse of ethics and the need for a true ethical reflection going with the technical and applicative evolutions in AI.

Keywords

Artificial Intelligence, Ethics, Tensions, Human control

1 Introduction

Le rapport annuel *Artificial Intelligence Index 2019* [14]-(page 273) de l'université de Stanford recensait cinquante-huit documents, toutes sources confondues (organisations officielles et gouvernementales, universités, sociétés savantes, industries, *think tanks*) associant « intelligence artificielle (IA) » et « éthique ». L'observatoire de l'intelligence artificielle créé par l'OCDE (Organisation de coopération et de développement économiques) propose sur son site une base de données interactive des documents de politiques et initiatives en matière d'IA [22], dont ceux qui traitent d'« éthique ». Quant au rapport de l'Université de Harvard [11], il analyse trente-six documents traitant de principes pour l'IA : les principes les plus souvent invoqués sont la protection de la vie privée, la répartition des responsabilités (*accountability*), la sûreté et la sécurité, la transparence et l'explicabilité, l'équité (*fairness*) et la non-discrimination, le contrôle humain, la responsabilité des professionnels, le respect des valeurs fondamentales.

La profusion de documents ainsi que d'instances créées pour traiter de « l'éthique de l'IA » amène à s'interroger sur les raisons pour lesquelles l'IA est devenue, depuis quelques années, un objet particulier d'attention, pourquoi cet objet est spécifiquement regardé sous un angle dit « éthique » et de quelle éthique il s'agit. L'examen des textes révèle notamment une interprétation sémantique des notions qui peut prêter à confusion, et des postulats susceptibles d'affecter les réflexions. Des tensions et paradoxes peuvent être mis en évidence, que nous illustrerons en particulier sur le principe du contrôle humain. Nous insisterons en conclusion sur les risques de dévoiement de l'éthique et la nécessité d'une véritable réflexion éthique accompagnant les évolutions techniques et applicatives en matière d'IA.

L'analyse qui suit est fondée principalement sur les textes européens émanant du Groupe d'experts indépendants de haut niveau sur l'intelligence artificielle, du Parlement et de la Commission ; sur le texte provisoire de l'UNESCO rédigé par le Groupe d'experts *ad hoc* ; sur la Déclaration de Montréal ; sur les textes de l'OCDE.

2 De quoi parle-t-on ?

2.1 Intelligence artificielle

Le vocable « intelligence artificielle », mal choisi [17], fait l'objet, y compris chez les scientifiques, de dérives de langage qui amènent à personnifier l'IA et à attribuer aux logiciels des caractéristiques équivalentes à celles d'un être vivant : « *une IA* fait ceci ou cela ». On observe également l'emploi de « IA » en tant que synonyme de « logiciel », « machine » ou « système », même si ceux-ci comprennent

des techniques qui ne relèvent pas de l'IA.

L' « autonomie » d'un « agent » ou d'un robot crée également des confusions, des fantasmes et des erreurs de raisonnement, y compris au sein d'instances internationales de négociation (par exemple les négociations à la Convention sur certaines armes classiques (CCAC) à Genève au sujet des systèmes d'armes létaux dits « autonomes »). De même le terme « déléguer », employé pour exprimer le fait que certaines fonctions habituellement dévolues à un être humain sont programmées dans une machine [15] sous-entend que l'être humain transfère une partie de ses responsabilités à la machine, celle-ci étant ainsi susceptible d'être dotée d'une existence morale ou juridique.

En outre, si « intelligence artificielle » est définie correctement en préambule des documents étudiés, c'est-à-dire en expliquant la diversité des approches rassemblées sous ce vocable (comme figurée sur le « diamant de l'IA » de l'AFIA [4]), le vocable est largement entendu dans le corps des textes comme un synonyme de « apprentissage machine ». Cela apporte d'autant plus de confusion que les recherches se concentrent actuellement sur l'hybridation d'approches statistiques et symboliques, ces dernières relevant de l'intelligence artificielle dans son sens premier.

Ces ambiguïtés dans le vocabulaire et les définitions contribuent à créer plusieurs écueils :

- L'objet du discours n'est pas clair : dans les textes relatifs à l'« éthique de l'IA », est-il question de l'ensemble des approches relevant de l'IA ou spécifiquement de celles qui sont fondées sur l'apprentissage machine ?
- L'anthropomorphisation de l'IA peut conduire à une surestimation des possibilités et des risques [28] : les machines pourraient ainsi « décider par elles-même » ou « prendre des initiatives » comme par exemple, pour un véhicule dit « autonome », « choisir » de renverser telle personne plutôt que telle autre ;
- Le vocabulaire employé peut laisser croire que les machines et les programmes pourraient être mis sur le même plan moral que l'être humain, voire être des « machines morales » (en particulier lorsque des connaissances ou des comportements relevant de concepts de la morale ou de l'éthique normative y sont modélisés [9]). Cela peut être renforcé par des applications qui brouillent les repères en faisant passer des machines pour des humains (imitation de l'aspect physique, de la voix, d'interactions sociales).

En ce qui concerne les deux derniers points, le texte provisoire de l'UNESCO [7]-(alinéa 126) indique que : *« Les États membres devraient instaurer des politiques visant à sensibiliser à l'anthropomorphisation des technologies d'IA, notamment en ce qui concerne les termes utilisés pour les désigner, et évaluer les manifestations, les conséquences éthiques et les possibles limites de ce phénomène [...] »*. Parmi ces conséquences figurent d'autres ambiguïtés concernant les termes qui qualifient les systèmes d'IA dont il est question dans les documents relatifs à l'« éthique de l'IA ».

2.2 Éthique

L'examen des documents produits par les groupes d'experts, comités, instances nationales ou internationales au sujet de l'« éthique de l'IA » montre que la possibilité de s'interroger sur le fait de ne pas développer ou de ne pas utiliser des systèmes visant à automatiser les processus de décision ou fondés sur l'apprentissage machine n'est évoquée que de manière très marginale. De tels systèmes existent ou vont exister et il s'agit, d'une certaine manière, de les cautionner, en rappelant que des principes doivent être respectés, en énonçant des précautions à prendre, et en suggérant une approche fondée sur une évaluation des risques. Cette démarche que Marc Hunyadi qualifie de *« petite éthique »* s'inscrit dans une logique du fait accompli, où chacun dispose d'une liberté de plus en plus limitée de choisir de ne pas posséder ou de ne pas utiliser certains objets, et qui construit petit à petit *« des modes de vie imposés par personne en particulier et auxquels tout le monde adhère »* [13]. De plus, les questionnaires d'auto-évaluation qui sont proposés de manière institutionnelle [24] ou par des organisations privées, ou les comités *ad hoc* qui sont constitués, risquent de généraliser un blanchiment éthique (*ethics washing*) en promouvant une « conformité éthique » dont la valeur et le sens peuvent être discutables. Il est question en effet d'*« IA éthique »* [8], de *« conformité éthique des systèmes d'IA »* [7], de *« certificat européen de conformité éthique »* [27], voire d'*« éthicité »* des systèmes [16]. Ceci appelle les remarques suivantes :

- Un objet, un programme ou une technique ne peut pas être « éthique » en soi et ne peut être qualifié d'« éthique ». L'adjectif « éthique » (par définition [1] : qui concerne la morale) ne peut être associé qu'à une démarche, une délibération, une réflexion, une question, un principe, une valeur, etc.
- De même une conformité ne peut être « éthique » et il ne suffit pas de dire ce qu'il convient de faire ou ne pas faire. Ce qui relève de l'éthique est instable, singulier, et a à voir avec des dilemmes qui justement vont conduire à des solutions partiellement non conformes, qui ne vérifient pas toutes les propriétés (voir 4). La conformité dont il s'agit est une conformité technique à certaines exigences, énoncées dans un cahier des charges et vérifiées, y compris les éventuels compromis, par des simulations, des expérimentations, des campagnes de vérification, des processus d'homologation.
- Le concept d'« éthique par conception » (*"ethics by design"*) [27], calqué sur celui de respect de la vie

1. TLF et Larousse

privée dès la conception (*"privacy by design"*)[2], et compris comme l'intégration de principes allant au-delà des exigences légales dans la conception de systèmes d'IA [11], se heurte aux deux premières remarques. En particulier, *« éthique et état de droit dès la conception »* signifient dans [8]-(alinéas 98 à 101) : conformité aux normes, explicabilité, essais et validation, ce qui ne relève pas *a priori* de la réflexion éthique. Il existe de plus une ambiguïté sur l'expression française « éthique par conception » où « éthique » peut être compris en tant qu'adjectif – l'objet serait « éthique » (*ethical*) par conception – ou en tant que substantif (*ethics*) – de l'éthique serait prise en compte dès la conception de l'objet. Les auteurs de [28] indiquent qu'il serait préférable de concevoir des machines qui nous aident à agir mieux d'un point de vue éthique plutôt que d'envisager des machines comme des agents moraux ou se comportant conformément à des règles morales.

D'autre part, l'expression *« IA digne de confiance »* ou *« IA de confiance »* (*"trustworthy AI"*) qui selon l'Europe est définie par les trois caractéristiques : *« IA licite »*, *« IA éthique »* et *« IA robuste »* [8] est problématique. La confiance ne se décrète pas et une machine ou un système ne peut pas porter, en soi, la confiance. C'est bien l'expérience d'une personne qui utilise un système, l'examen de la manière dont il a été conçu et les garanties démontrées de conformité technique qui sont fournies qui vont amener cette personne à avoir confiance, ou non, dans ce système pour répondre à ses besoins. Comme l'affirme J. Bryson [5], *"No one should trust IA"*.

3 Les postulats

Les textes étudiés se fondent explicitement ou implicitement sur des postulats qui peuvent être discutables et occulter des éléments de réflexion. Nous relevons trois de ces postulats.

3.1 Les systèmes d'IA sont inéluctables

Aucun des documents étudiés n'envisage que les systèmes d'IA fassent l'objet de questionnements relatifs à leur existence même, à leurs raisons d'être. C'est une approche conséquentialiste fondée sur les risques et les précautions qui est adoptée, accompagnée de la nécessité d'un *« contrôle humain »* (voir 5) des systèmes d'IA, en particulier pour ceux qui sont estimés *« à haut risque »* [10, 27]. Le document de l'UNESCO [7] envisage cependant des interrogations sur l'utilisation des systèmes d'IA, en notant que celle-ci revêt un *« caractère facultatif »* (alinéa 20) et qu'une analyse devrait être menée pour évaluer si *« l'adoption de l'IA est appropriée »* (alinéa 58).

2. En Europe, il s'agit de concevoir les systèmes qui traitent des données à caractère personnel de manière conforme au Règlement général pour la protection des données (RGPD).

3.2 Les systèmes d'IA contribuent au bien-être

Les systèmes d'IA sont conçus dans l'*« objectif d'améliorer le bien-être et la liberté des êtres humains »*, constituent *« un moyen prometteur d'accroître la prospérité humaine, en renforçant ainsi le bien-être individuel et de la société ainsi que le bien commun »*, sont *« susceptibles d'apporter des avantages considérables aux individus et à la société »* [8]. L'IA « promet d'améliorer le bien-être des individus » [21]. *« Le développement et l'utilisation des systèmes d'intelligence artificielle doivent permettre d'accroître le bien-être de tous les êtres sensibles »* [1]-(principe 1).
L'Organisation mondiale de la santé mentionne toutefois que la notion de bien-être est multidimensionnelle, comprend des éléments subjectifs, culturels, et n'a pas de définition claire [25]. Selon l'Institut national de la statistique et des études économiques (Insee), contribuent au bien-être : les conditions de vie (logement, contraintes financières), la santé physique et émotionnelle, les liens sociaux, la sécurité, les risques psychosociaux au travail, les revenus, la composition du logement, l'âge, le diplôme [3].

3.3 Les systèmes d'IA sont une solution à tout

Dans le prolongement du postulat précédent, les systèmes d'IA peuvent contribuer à *« promouvoir l'égalité entre les sexes et lutter contre le changement climatique, rationaliser notre utilisation des ressources naturelles, améliorer notre santé, notre mobilité et nos processus de production, et nous aider à surveiller nos progrès par rapport à des indicateurs de durabilité et de cohésion sociale »* [8] ; il peuvent favoriser *« le renforcement des capacités humaines et le renforcement de la créativité humaine, l'inclusion des populations sous-représentées, la réduction des inégalités économiques, sociales, entre les sexes et autres »* [21] ; ils peuvent *« amélior[er]les conditions de vie, la santé et la justice, en créant de la richesse, en renforçant la sécurité publique ou en maîtrisant l'impact des activités humaines sur l'environnement et le climat »* [1].
A contrario, les contributeurs à l'atelier Quality of AI [17] de l'ERCIM (European Research Consortium for Informatics and Mathematics) soulignent que l'IA est souvent largement surestimée, mais qu'il est difficile de décrire tout ce qu'elle permet de réaliser sans laisser à penser qu'elle constitue une solution universelle. À titre d'exemple, les motivations qui justifient le déploiement de véhicules à conduite automatisée – amélioration de la sécurité routière, fluidification du trafic, réduction de la dépense énergétique, accès à la mobilité en particulier en zones rurales – sont en réalité *« peu documentées »* et assorties de fortes incertitudes [2]-(pages 20–22).

4 Tensions et paradoxes

4.1 Tensions entre principes

Les principes et exigences énoncés dans les documents ne peuvent pas être simultanément satisfaits, des compromis sont donc nécessaires [20]. Ces compromis, bien qu'ils

constituent justement l'objet de la réflexion éthique, sont évoqués de manière très succincte, par exemple : *« Des tensions pourraient survenir entre les principes [...], pour lesquelles il n'existe pas de solution unique. [...] Il faut [...] aborder les dilemmes et arbitrages éthiques selon une réflexion raisonnée et fondée sur des éléments probants. [...] Il pourrait toutefois exister des situations dans lesquelles aucun arbitrage acceptable du point de vue éthique ne peut être déterminé »* [8]-(alinéa 54) ; *« Si toutes les valeurs et tous les principes [...] sont souhaitables en soi, dans tout contexte réel, il y a inévitablement des compromis à faire, ce qui exige de procéder à des choix complexes concernant la hiérarchisation des contextes, sans pour autant compromettre d'autres principes ou valeurs »* [7]-(alinéa 11).

Quelques exemples de tensions sont proposés ci-dessous :

- Transparence / sécurité
  La transparence, l'explicabilité et la prédictibilité des systèmes d'IA peuvent présenter l'inconvénient d'une moindre sécurité et de possibles dérives d'usages si ces propriétés sont promues par l'ouverture des algorithmes voire des codes [3]. D'autre part, la transparence doit être évaluée au regard de la préservation de la propriété industrielle.
- Précision / protection de la vie privée
  Un système d'IA fondé sur l'analyse de données est d'autant plus précis et pertinent (*accurate*) que ces données sont précises, variées, riches et peuvent discriminer des situations particulières, voire rares, ce qui peut entrer en conflit avec la protection de la vie privée et des données à caractère personnel (données de santé ou de surveillance par exemple), voire la préservation des droits fondamentaux dans le cas des systèmes de reconnaissance faciale [18].
- Précision / préservation de l'environnement
  La précision d'un système d'IA fondé sur l'analyse de données nécessite de grands ensembles de données dont la collecte, le stockage et l'exploitation sont susceptibles d'avoir un fort impact sur l'environnement.
- Performance / Autonomie humaine
  Un système d'aide à la décision ou un système « autonome », conçu pour aider l'être humain ou le remplacer dans certaines tâches, est susceptible de porter atteinte à l'autonomie humaine, en influençant la décision de la personne, voire en s'y substituant. D'autre part, l'augmentation des capacités de ces systèmes, de leur pertinence et de leur fiabilité peut conduire à une dégradation, voire à la perte, de certaines compétences ou expertises humaines.

En outre et de manière paradoxale, les systèmes d'IA doivent être conçus de manière à respecter des principes et dans le même temps constituent une menace pour ces mêmes principes, ou bien être conçus pour un objectif qu'ils contribuent également à mettre en danger. Ainsi ils doivent être conçus dans le respect des droits fondamentaux et sont susceptibles de menacer ces droits ; ils peuvent améliorer le bien-être et abaisser la qualité de vie ; réduire les inégalités et les exacerber ; renforcer les capacités humaines et contraindre les choix des individus et des groupes ; renforcer la sécurité et ouvrir de nouvelles brèches de sécurité ; contribuer à lutter contre le changement climatique et affecter les écosystèmes, l'environnement et le climat [1, 8].

4.2 Équité et biais

Le principe d'équité (*fairness*) peut faire référence aux notions d'impartialité, d'égalité, de non-discrimination et de justice et suppose un idéal d'égal traitement des individus ou des groupes [24].

« Les biais et la discrimination sont des risques inhérents à toute activité sociétale ou économique » [10], cependant il est demandé aux acteurs de l'IA de *« réduire au maximum et éviter de renforcer ou de perpétuer des biais sociotechniques inappropriés basés sur les préjugés liés à l'identité »* [7]-(alinéa 29), de corriger les biais éventuels [10], de veiller à l'absence de *« biais injustes »* [8].

Dans le même temps il est demandé de réfléchir à la définition de l'équité [24]. Il semble en particulier nécessaire de situer la définition d'une propriété d'un logiciel ou des résultats qu'il est susceptible de fournir par rapport à la notion d'équité dans le sens commun. On peut constater d'abord que la nature n'est pas équitable en soi – *"unfairness is natural"* [4], que la société véhicule de nombreux biais et que les êtres humains, consciemment ou non, ont des comportements discriminatoires. Que signifie alors de réduire les biais ou d'éviter de les renforcer dans les systèmes d'IA, sous-entendu essentiellement ceux qui sont fondés sur l'analyse de données ?

On pourrait se demander ce que serait un objet logiciel sans biais, voire « neutre » et si des résultats de calcul qui seraient moralement neutres, équitables, seraient adaptés à la société ou à la nature. D'autre part, comment formaliser sous forme mathématique, donc programmable, un raisonnement ou une décision « juste » ou « équitable » ? Il semble que ces questions ne puissent être envisagées dans l'absolu : il convient de s'interroger sur la raison d'être et les objectifs de l'utilisateur du système d'IA ainsi que les valeurs qu'il souhaite promouvoir, et comment ces objectifs et valeurs orientent la conception du système. Par exemple, un processus automatisé de sélection de CV pour une embauche pourrait être fondé sur un tirage au sort ou sur l'historique des profils des personnes qui ont « réussi » au poste concerné. La première méthode, qui ne nécessite pas de système d'IA, peut être considérée – si toute personne a la possibilité de présenter son CV – comme

3. Softbank Robotics Webinar on Responsible Robotics and AI : Concrete solutions, Feb. 2021

4. J. Bryson, Softbank Robotics Webinar on Responsible Robotics and AI : Concrete solutions, Feb. 2021

« sans biais », mais a de grandes chances d'être inadaptée. La seconde est susceptible de perpétuer l'embauche de personnes ayant toujours les mêmes caractéristiques, sauf à diversifier la notion de « réussite », qui dépend des valeurs que l'organisation qui cherche à recruter veut renforcer grâce à cette embauche.

Remarque : Fairlearn [5] n'utilise pas le terme de « biais » et définit l'équité sur la base de deux types d'impacts des systèmes d'IA sur les personnes : préjudices d'affectation et préjudices de qualité de service.

5 Exemple : le contrôle humain

Il y a un consensus international sur le principe de « contrôle humain » des systèmes d'IA, qui se traduit dans les textes par des *« garanties et des mécanismes, tels que l'attribution de la capacité de décision finale à l'homme, qui soient adaptés au contexte et à l'état de l'art »* [21], le fait de pouvoir décider de ne pas utiliser un système d'IA afin de conserver des niveaux de jugements humains, ou d'assurer la possibilité que la décision de l'humain prime sur celle calculée par le système [24]. Pour les applications dites à « haut risque » (comportant des risques d'atteinte aux individus ou à la société), l'Europe préconise une garantie de *« participation adéquate de l'être humain »* [10], de supervision humaine à tout moment, et une reprise en main humaine quand nécessaire [27], et *« qu'à tout moment, une personne humaine ait la possibilité de corriger [le système], de l'interrompre ou de [le] désactiver en cas de comportement imprévu, d'intervention accidentelle, de cyberattaques, d'ingérence de tiers dans une technologie fondée sur l'IA ou d'acquisition par des tiers d'une telle technologie »* [26].

Cette notion de « contrôle humain » reste cependant floue en particulier parce qu'elle englobe plusieurs types d'interventions humaines : en effet, elle peut concerner le fait qu'une personne ou une organisation humaine décide d'utiliser ou non le système d'IA, la nature des décisions qui restent dévolues à l'humain, la supervision, les possibilités de reprise en main, les validations humaines des résultats fournis.

Par ailleurs se pose la question de l'évaluation de la présence du contrôle humain : comment et par qui cette évaluation est-elle réalisée ? Que signifie techniquement la garantie de supervision humaine « à tout moment » ? Comment garantit-on que l'intervention humaine est pertinente ?

5.1 Un paradoxe

Les raisons pour lesquelles on souhaite automatiser des fonctions décisionnelles dans le cadre d'une application ou d'ensembles d'applications sont multiples, par exemple : les tâches à réaliser dépassent les capacités humaines (le contexte demande par exemple d'envisager une combinatoire élevée ou un espace de recherche de solutions très grand) ; elles mettent en cause la sécurité ou la santé de l'humain (le contexte est dangereux ou hostile) ; l'automatisation est plus économique ; l'automatisation est plus sûre (elle permet de pallier l'erreur humaine).

Il y a donc un paradoxe entre les raisons qui motivent l'automatisation et le fait d'exiger un contrôle humain des fonctions automatisées : se pose en effet la question de la capacité de l'humain à exercer effectivement ce contrôle. De plus, la notion de contrôle humain sous-entend que le point de vue de l'humain est pertinent et correct, et qu'il doit prévaloir sur les résultats des calculs de la machine. Enfin, le contrôle humain nécessite qu'il y ait effectivement un humain présent et disponible – par exemple il est indiqué dans [23]-(section 9) que les services publics européens doivent être largement fondés sur des systèmes numériques à base d'IA, mais que le recours à un interlocuteur humain doit toujours être possible.

5.2 Limites du contrôle humain

L'humain doit disposer d'informations et de temps, qui soient compatibles avec le contrôle à exercer. En particulier, l'humain ne peut pas être considéré comme le recours ultime dans n'importe quelle situation ou quand les fonctions automatisées « ne savent pas faire ». Par exemple, il est illusoire d'envisager le transfert du contrôle de la conduite d'un véhicule « autonome » des automatismes vers l'utilisateur si celui-ci, comme on le voit dans certaines publicités de constructeurs automobiles, est occupé à d'autres activités : une bonne conscience de situation, incluant prédiction et anticipation, est indispensable pour élaborer des décisions et des actions adaptées. Même la procédure d'arrêt en sécurité (*"stop button"*) [24] que l'être humain pourrait engager est complexe à envisager de manière opérationnelle en toutes circonstances.

En outre, les automatismes altèrent les mécanismes de contrôle classiquement utilisés par l'humain : moindre engagement dans la tâche, augmentation de la divagation attentionnelle, moindre aptitude à détecter des erreurs. En particulier s'il est novice, fatigué ou stressé, l'humain est susceptible de se reposer sur ce que préconise la machine et d'être ainsi enfermé dans des choix restreints ou erronés. Enfin, le manque d'informations ou au contraire un flot trop abondant d'informations, ou les schémas que l'humain a en tête, peuvent entraîner une mauvaise compréhension du comportement de la machine ou des résultats qu'elle propose, et entraîner des décisions humaines inadaptées.

5.3 Envisager un partage de l'autorité

Le fait d'assurer la possibilité que la décision de l'humain prime sur celle calculée par le système d'IA, qui est l'une des options de la surveillance humaine [24] suppose que l'humain est infaillible.

Il ne s'agit pas d'opposer l'humain et la machine ou les logiciels, mais de répartir les bonnes compétences aux bons

5. A Python package to assess and improve fairness of machine learning models : `https://github.com/fairlearn/fairlearn`

endroits dans le cadre d'une approche système incluant les mécanismes humains mis en jeu, et en *analysant les besoins* : la machine doit être conçue pour aider ses utilisateurs et remplir un service bien identifié en préservant l'essence même de ce qui est important pour prendre des décisions et en endosser la responsabilité. D'un point de vue technique, des critères concrets permettant de spécifier et de vérifier la façon dont la machine permet à l'humain d'exercer ses mécanismes de contrôle doivent être définis. En outre il faut s'interroger sur ce que seront les capacités cognitives de demain, certaines capacités étant susceptibles de décroître pendant que d'autres se développent ; il faudra certainement adapter les systèmes à ces capacités différentes.

5.4 La question de l'annotation

Un type particulier de « contrôle humain » est la nécessaire annotation ou transcription des données pour alimenter les systèmes d'apprentissage. Si l'Europe recommande la mise en place d'outils souverains en la matière, dans le respect des législations [23], la question de l'annotation des données et de la transcription d'échanges verbaux n'apparaît pas en tant que préoccupation éthique dans les documents étudiés. Pourtant, ce contrôle humain est largement effectué par des « micro-travailleurs » précaires, sous-rémunérés et dépourvus de couverture sociale, ou bien par des employés de sous-traitants des grandes entreprises du numérique exposés en permanence à des données personnelles sensibles ou des à propos dérangeants. Il faut certainement s'interroger sur la tension entre performances des logiciels fondés sur les données et dignité et intégrité des personnes qui contribuent à ces performances.

6 Conclusion

6.1 Des risques de dévoiement de l'éthique

On ne peut pas laisser penser qu'« une IA éthique » serait possible, et que cela consisterait à vérifier une conformité à des critères et à des normes, traduits en exigences techniques. En effet les instruments normatifs [7], les standards (IEEE P7000™ et suivants), les grilles d'évaluation (auto-évaluation « éthique » pour les projets européens), les audits éthiques [20], qui relèvent essentiellement d'une bonne gestion et de bonnes pratiques, présentent le risque d'être substitués à une véritable réflexion éthique, permanente et toujours en chantier. Par exemple, s'il est nécessaire de vérifier la conformité d'un projet d'identification biométrique [6] au Règlement général pour la protection des données (RGPD), il est tout aussi nécessaire de réfléchir aux raisons pour lesquelles ce projet est mené et, et fur et à mesure des performances constatées, de se demander s'il faut le poursuivre et dans quelles directions, et quels dilemmes se présentent.

Le risque est un dévoiement de l'éthique qui, *via* des labels ou des certificats, pourrait à la fois donner bonne conscience et servir d'argument de vente. En ce sens, *une éthique de l'éthique de l'IA* [28] est à construire.

6.2 Une indispensable réflexion éthique

« Le premier danger que présente le développement de l'intelligence artificielle consiste à donner l'illusion que l'on maîtrise l'avenir par le calcul. [...] Mais dans les affaires humaines, demain ressemble rarement à aujourd'hui, et les nombres ne disent pas ce qui a une valeur morale, ni ce qui est socialement désirable. » [1]

Outre une approche conséquentialiste fondée sur l'analyse des risques, il s'agit également de questionner l'existence même de l'objet (logiciel, robot), sa raison d'être, et se demander si cet objet est désirable et au nom de quelles valeurs. La référence étant des moyens existants ou l'être humain, l'objectif peut être de faire effectuer des tâches plus rapidement, à plus grande échelle, de manière plus précise, plus sûre, voire plus « inventive » ; de réduire des coûts ; de proposer des solutions plus simples, plus commodes, plus ludiques. On peut aussi interroger la pertinence de ces critères : pourquoi vouloir aller plus vite, etc., et éloigner toujours plus l'être l'humain, et ce de manière paradoxale car le contrôle humain est considéré comme impératif ?

Une analyse des besoins, à mettre en regard de l'évolution des capacités techniques, permettrait de distinguer usages et technologies, d'envisager les usages problématiques et les glissements insidieux vers de tels usages, d'interroger la légitimité d'utiliser certaines techniques, d'identifier des nouveaux besoins susceptibles d'être créés de toutes pièces, ou de questionner l'utilisation des technologies pour traiter des effets de dérives plutôt que de remédier aux dérives elles-mêmes (par exemple la relecture d'articles par les pairs assistée par « IA » pour faire face à l'inflation de propositions de publications [6]). Il s'agirait plus d'« embarquer » l'éthique dans la recherche et l'ingénierie relatives à l'IA plutôt que dans l'IA elle-même, c'est-à-dire la considérer pour ce qu'elle est, un processus de réflexion continu qui concerne des tensions entre principes et des questions sans « bonne solution » [19]. Il pourrait être imposé par exemple une discussion éthique dans les articles scientifiques [12]. Ce n'est pas l'« IA » qui va évoluer toute seule vers de nouvelles capacités et applications, comme peut le laisser croire la façon dont on personnifie cet ensemble de techniques, mais bien les humains qui doivent décider collectivement de ce qu'il convient ou non de faire, en analysant si ces progrès technologiques vont nécessairement vers un progrès moral.

Liens d'intérêt

L'autrice a été membre du Groupe d'experts *ad hoc* de l'UNESCO pour l'élaboration d'un avant-projet de recommandation sur l'éthique de l'intelligence artificielle [7].

6. https://www.cnil.fr/fr/biometrie

Références

[1] La Déclaration de Montréal pour un développement responsable de l'Intelligence Artificielle, 2018. https://www.declarationmontreal-iaresponsable.com/la-declaration.

[2] Développement des véhicules autonomes - Orientations stratégiques pour l'action publique. Ministère de la transition écologique, 2018. https://www.ecologie.gouv.fr/developpement-des-vehicules-autonomes-orientations-strategiques-laction-publique.

[3] M.-H. Amiel, P. Godefroy, and S. Lollivier. Qualité de vie et bien-être vont souvent de pair. Technical report, Insee, 2013. https://www.insee.fr/fr/statistiques/1281414.

[4] Association française pour l'Intelligence Artificielle. Domaines de l'IA. https://afia.asso.fr/domaines-de-lia/.

[5] J. Bryson. AI & Global Governance : No One Should Trust AI. United Nations University, Centre for Policy Research, 2018. https://cpr.unu.edu/publications/articles/ai-global-governance-no-one-should-trust-ai.html.

[6] A. Checco, L. Bracciale, P. Loreti, S. Pinfield, and G. Bianchi. AI-assisted peer review. *Humanities and Social Sciences Communications*, 8 :25, 2021. https://doi.org/10.1057/s41599-020-00703-8.

[7] Groupe d'Experts ad hoc (GEAH) de l'UNESCO. Avant-projet de recommandation sur l'éthique de l'Intelligence Artificielle, 2020. https://unesdoc.unesco.org/ark:/48223/pf0000374343_fre.

[8] Groupe d'Experts Indépendants de Haut Niveau sur l'Intelligence Artificielle. Lignes directrices en matière d'éthique pour une IA digne de confiance. Commission européenne, 2019. https://op.europa.eu/fr/publication-detail/-/publication/d3988569-0434-11ea-8c1f-01aa75ed71a1/prodSystem-cellar/language-fr/format-PDF.

[9] Projet EthicAA. Livre Blanc - Éthique et agents autonomes. Projet ANR-13-CORD-0006, 2018. https://ethicaa.greyc.fr/media/files/ethicaa.white.paper.pdf.

[10] Commission européenne. Livre Blanc : Intelligence artificielle - Une approche européenne axée sur l'excellence et la confiance, 2020. https://ec.europa.eu/info/sites/info/files/commission-white-paper-artificial-intelligence-feb2020_fr.pdf.

[11] J. Fjeld, N. Achten, H. Hilligoss, A.C. Nagy, and M. Srikumar. Principled Artificial Intelligence : Mapping Consensus in Ethical and Rights-based Approaches to Principles for AI. Technical report, The Berkman Klein Center for Internet & Society, Harvard University, 2020. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3518482.

[12] E. Gibney. The battle to embed ethics in AI research. *Nature*, 577 :609, 2020. https://media.nature.com/original/magazine-assets/d41586-020-00160-y/d41586-020-00160-y.pdf.

[13] M. Hunyadi. *La Tyrannie des modes de vie – Sur le paradoxe moral de notre temps*. Le Bord de l'eau, 2015.

[14] Stanford Human Centered Artificial Intelligence. Artificial Intelligence Index - 2019 Annual Report. Stanford University , 2019. https://hai.stanford.edu/research/ai-index-2019.

[15] Journal Officiel. Loi 2019-1428 d'orientation des mobilités, chapitre II, section 1 « Véhicules autonomes et véhicules connectés », 24 décembre 2019. https://www.legifrance.gouv.fr/jorf/id/JORFTEXT000039666574.

[16] Laboratoire national de métrologie et d'essais (LNE). Évaluer les intelligences artificielles, 2021. https://www.lne.fr/fr/on-en-parle/evaluer-intelligence-artificielle-ia.

[17] B. Levin and P. Kunz. ERCIM Workshop on Quality of AI. *ERCIM News*, 123 :4–5, 2020. https://ercim-news.ercim.eu/en123/joint-ercim-actions/ercim-workshop-on-quality-in-ai.

[18] P. Marks. Can the Biases in Facial Recognition Be Fixed ; Also, Should They ? *Communications of the ACM*, 64 :3 :20–22, 2021. https://cacm.acm.org/magazines/2021/3/250698-can-the-biases-in-facial-recognition-be-fixed-also-should-they/fulltext.

[19] B. Mittelstadt. Principles alone cannot guarantee ethical AI. *Nature Machine Intelligence*, 1 :501–507, 2019. https://doi.org/10.1038/s42256-019-0114-4.

[20] J. Mokander and L. Floridi. Ethics-Based Auditing to Develop Trustworthy AI. *Minds and Machines*, 2021. https://doi.org/10.1007/s11023-021-09557-8.

[21] OCDE. Recommandation du Conseil sur l'intelligence artificielle OECD/LEGAL/0449, 2019. https://legalinstruments.oecd.org/fr/instruments/OECD-LEGAL-0449.

[22] OECD.AI. Countries & initiatives overview, 2020. https://www.oecd.ai/countries-and-initiatives.

[23] Independent High-Level Expert Group on Artificial Intelligence. Policy and Investment Recommendations for Trustworthy AI. Commission européenne, 2019. https://ec.europa.eu/digital-single-market/en/news/policy-and-in

vestment-recommendations-trustworthy-artificial-intelligence.

[24] Independent High-Level Expert Group on Artificial Intelligence. The Assessment List for Trustworthy Artificial Intelligence (ALTAI). Commision européenne, 2020. https://ec.europa.eu/digital-single-market/en/news/assessment-list-trustworthy-artificial-intelligence-altai-self-assessment.

[25] World Health Organization. Measurement of and target-setting for well-being : an initiative by the WHO Regional Office for Europe, 2012. https://www.euro.who.int/__data/assets/pdf_file/0009/181449/e96732.pdf.

[26] Parlement européen. Intelligence artificielle : questions relatives à l'interprétation et à l'application du droit international. Résolution du 9 janvier 2021. https://www.europarl.europa.eu/doceo/document/TA-9-2021-0009_FR.html.

[27] European Parliament. Framework of ethical aspects of artificial intelligence, robotics and related technologies, 2020. https://www.europarl.europa.eu/doceo/document/TA-9-2020-0275_EN.html.

[28] T.M. Powers and J.-G. Ganascia. The Ethics of the Ethics of AI. In M.D. Dubber, F. Pasquale, and S. Das, editors, *The Oxford Handbook of Ethics of AI*. Oxford University Press, 2020.

A Machine Learning approach to improve the monitoring of Sustainable Development Goals : a case study in Senegalese artisanal fisheries

T. Bayet[1,2], T. Brochier[2,3], C. Cambier[2,3], A. Bah[3,4], C. Denis[1], N. Thiam[5] J-D. Zucker[2,6],

[1] Sorbonne Université, LIP6, 75005 Paris
[2] IRD, Sorbonne Université, UMMISCO, F-93143, Bondy, France
[3] UCAD, IRD, UMMISCO, Dakar, Sénégal
[4] Ecole Supérieure Polytechnique, UCAD, 15915 Dakar Fann, Sénégal
[5] Centre de Recherches océanographiques de Dakar-Sénégal (CRODT)
[6] Sorbonne Université, INSERM, NUTRIOMICS, F-75013, Paris, France

theophile.bayet@ird.fr

Résumé

Depuis l'adoption des Objectifs de Développement Durable (ODDs), force est de constater l'inégale répartition des efforts de la communauté internationale pour atteindre ces ODDs. Une exploration des données liées à ces objectifs montre la nécessité d'améliorer le suivi en particulier dans les pays les moins avancés. Nous proposons ici une méthode basée sur le Machine Learning pour pallier le manque de données et le besoin d'un suivi dynamique. Cette méthode se base sur trois principes : la recherche participative, la contextualisation de processus et le développement dynamique de modèles. Un exemple illustratif est présenté dans le cadre du développement de nouveaux jeux de données et de modèles de prédiction au Sénégal montrant l'intérêt de l'approche proposée.

Mots-clés

Objectif du développement Durable, Machine Learning, Science de la durabilité, approche participative, Pêcheries, Sénégal

Abstract

Since the adoption of the Sustainable Development Goals (SDGs), the international community's efforts to achieve the SDGs have been unevenly distributed. An exploration of the data related to these goals raises the urgent need to monitor them in order to better focus efforts in the least developed countries. We propose here a method based on Machine Learning to overcome the lack of data and the need for dynamic monitoring. This method is based on three principles : participatory research, context localized process and dynamic model development. An illustrative example is presented in the context of the development of new data sets and prediction models in Senegal, showing the interest of the proposed approach.

Keywords

Sustainable Development Goals, Machine Learning, Sustainable science, participatory approach, Fisheries, Senegal

1 Introduction

The effects of climate change has become increasingly visible and the scientific efforts to their study do intensify. The development of forecasting models to assess the situation is subject to extensive endeavor and these efforts are summarized in the Intergovernmental Panel on Climate Change (IPCC) reports[40]. The analysis of the simulations obtained from the forecasting models show alarming trends if the public policies do not change direction in the coming years. This trend motivated the adoption by the UN of the SDGs, that are to be achieved through international coordination and regulation. Established in 2015, these goals have been declined in 247 targets, monitored by indicators described in the Global Indicator Framework[1]. An important point that motivate our work is that climate change is expected to impact the world in an unequal way, the consequences of which are predicted to be even more drastic in countries that are already poor, were the indicators of the SDGs are more difficult to assess[6].

There are several tools to apprehend and understand our environment and thus its vulnerability to climate change ; it is these tools that Sustainable Science is interested in. Sustainable Science is concerned with applications of science in the field of sustainable development[24]. It explores ways to apply new data collection and processing methods to SDGs today, in order to better understand the related issues, propose solutions to change our behavior and promote a more sustainable development. These solutions increasingly employ the methods that are Machine Learning (ML) and Deep Learning (DL) to benefit from the increase in their predictive capabilities[21].

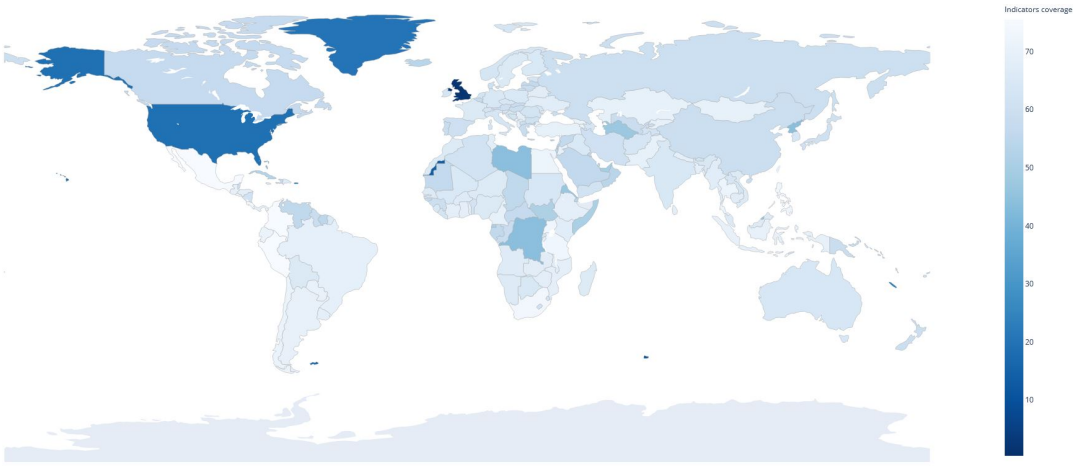


FIGURE 1 – World coverage of indicators from the Global SDG Indicators database by nation. For each of the 247 indicators of the SDGs, an indicator was considered covered by a nation if it was filled out at least once in the database, independently of the year considered, the type of data or its nature.

There is today no need to demonstrate anymore the growing impact of ML in a wide range of sectors. With the growth of datasets that become bigger and annotated in more complex ways[20], the applications of ML keep expanding to new sectors. Artificial Intelligence (AI) has been branded new general purpose technology[12], and as of one of its main component, ML is expected to enhance productivity and quality across a majority of technology sectors, just like the numerical transformation did. With such a wide area of application, ML can play a key role at different levels on the multiple facets of climate change[41], which makes it a particularly powerful tool. Many applications of ML and DL in the fight against climate change have already emerged, whether they are based on satellite data[21, 38], google street[43] or other models[47].

Despite this, there is relatively little reliable public data in African countries[21]. In order to efficiently estimate ODD indicators, data collection must follow a rigorous methodology that remains standard over the years. Failing to do so may result in incomplete datasets, non uniforms data-collection methods, and sparse data not available in practical format, and thus impact negatively the use of machine learning models which generally rely on abundant data to achieve good performance. While the low income countries are the most affected by the consequences of global warming, the assessment of the current state of their ecosystems and populations is hampered by this lack of access to exploitable data, which is either private or non-existent, and yet essential to respond to the SDGs. We argue in this paper that there is a need for an extended effort in building datasets and models especially in low income countries context, and we present a method for such efforts in Senegal.

The paper is structured as follows. Section 2 presents some previous works and explore what have been achieved so far. In Section 3 we examine the availability of data for the SDGs. Then, in Section 4 we present our approach to use machine learning to improve the monitoring of SDGs achievements. We present our future works in Section 5 and finally conclude in Section 6.

2 Previous works

There has been extensive work on dealing with the tasks of the SDGs. Vinuesa et al.[48] explored connections between AI and the SDGs, and documented whether AI would have a positive or negative impact on the SDGs based on previous studies. They showed that AI may have a positive impact on 134 targets, and a negative impact on 59 targets, as some targets may be positively or negatively impacted depending on the innovation. AI will have profound influence, both positive and negative, on the ability to achieve the 2030 Sustainable Development Agenda in countries in Africa. Unfortunately,it suffers from downsides and biases that need to be addressed in order to fight climate change. First of all, most of applications are currently biased towards SDGs issues relevant to wealthy nations. Despite being the most vulnerable to climate change[6], the less wealthy countries poorly benefit from the advances in AI so far. AI researches and applications are clustered in nations least vulnerable to climate change, and models and applications developed in such nations may not be useful in less wealthy nations, as the application context may vary drastically[3]. Such bias in development have proven to have drastic consequences, such as racist bias in IA[13]. Inequity in data is another bias that need to be addressed. Openly available datasets are often neither relevant for, nor representative of, the global south. In computer vision for example, most famous datasets do not contain images from the low income nations (MSCOCO[28], ImageNet[14], and more recently Open Images[25]). In aerial imagery, there has been a concern recently on building more global datasets[32, 16, 11] that can rival with the ones containing context images, but these lack in the same way. As for satellite imagery, which is globally available, it may be sufficient for some tasks[37], but not adequate for a

lot of other tasks, due to local context. In the domain of ship detection for example, a lot of work is dedicated to satellite[9, 2, 44, 50, 53, 55, 17] or System Aperture Radar (SAR)[49] imagery processing, to detect careers and containers. Unfortunately, satellite imagery still suffer from a lot of downsides[5], resolution being the major problem in the case of artisanal fisheries monitoring in Senegal, as fisher's canoes are too small to be efficiently detected using this data. Previous works then prove ineffective in monitoring these artisanal fisheries, with no data available and the context being too different from the previous applications.

The lack of data in developing countries is faced in two different ways in the scientific community. Researchers either try and generate their own data, or find proxies to estimate their targets. Innovative ways of using public data have emerged from this constrained situations, from using Google search data[26] to Wikipedia[45] or even nightlights as a proxy for poverty[54]. These works make use of correlations between publicly available data and their targets to evaluate the latter, and thus manage to bypass the data gap issue. On the other hand, low cost data generation profited from the emergence of new ideas and technologies. From kites[23] to Unmanned Aerial Vehicles (UAVs)[51], tremendous efforts have made generating quality aerial image data accessible even in complex environments such as deserts[31], cliffs[10], and other environments[35, 4]. The combination of DL techniques and UAV remote sensing is proving to be very promising for a lot of applications[36].

Another drawback of using AI is the lack of explainability. AI's black boxes hinder the acceptance of these techniques, especially in low-income regions where high-tech applications are rare. Distrust in AI in general has lately been exacerbated by demonstrations that it could replicates existing biases[7]. Explainability is nevertheless a key for whether policymakers will adopt machine-learning based approaches or not, as it allows to build trust in an algorithm's outputs[30]. Some works tend towards explainable ML models to address SDGs related issues[8], but they still lack the appropriate feedback of their work from policymakers or local citizens.

If AI is to become the new mainstream technology, researchers must address all of its drawbacks in order for it to be accepted in less developed countries and thus have a significant impact on the SDGs.

3 Data availability for the SDGs

Implementation of the SDGs is compulsory to achieve climate change mitigation. In order to follow these implementations in each nation, the Sustainable Development Solutions Network (SDSN) release each year the Sustainable Development Report (SDR), that frames progress on the SDGs[42]. This report is based on data from official sources as governmental institutions and non official sources like research institutions. It tracks country performances through an index with equal weight to all 17 goals. The outcomes of this report are straightforward and show that the nations with low income perform poorlier in the SDGs (Table 1) than the ones with high income, and some regions suffer from lack of data, which hampers the reporting work.

Region	Regional score
OECD	77.3
Eastern Europe and Central Asia	70.9
Latin America and the Caribbean	70.4
East and South Asia	67.2
Middle East and North Africa	66.3
Sub-Saharan Africa	53.1
Oceania	49.6
High-Income Countries	77.7
Upper-Middle-Income Countries	73.2
Lower-Middle-Income Countries	61.6
Low-Income Countries	52.5

Table 1 : Average score of SDG Index by UN sub-regions or income groups.The SDG Index tracks country performance on the 17 Sustainable Development Goals, weighted equally, as agreed by the international community in 2015. The score signifies a country's position between the worst (0) and the best or target (100) outcomes. Data source : SDR 2020[42]

Data availability remains an issue, as data are critical for turning the SDGs into practical tools for problem solving. Vinuesa et al.[48] shows that AI should have a positive impact on almost all targets of goals 1 "No poverty", 10 "Reduced inequalities" and 14 "Life below water" when they are the ones lacking the most in data availability according to SDR. Data gaps are to be addressed to efficiently feed AI methods focusing on these goals. SDR fills those data gaps with model-based estimations, which comes with some drawbacks and limitations such as uncertain accuracy, hazardous assumptions and reducing incentives to strengthen statistical capacity. Even though, model-based estimations are a compulsory trick considering the data situation. The official data source to follow the indicators of the SDGs is the Global SDG Indicators Database, and exploring through this database, we find out that some indicators are simply not available for research in the database, as 31 out of 231 unique indicators are missing, for every nation. To inquire about which country faced the worst data gaps in this database, we built an indicator coverage index. For each country, the index is built as follows :

$$Coverage_c = \frac{NCI_c}{NTI}$$

with $Coverage_c$ being the indicator coverage for a country and NCI_c the number of covered indicators for this country, a covered indicator being an indicator that is mentioned in the database for the given country, and finally NTI is the number of total indicators (247).

Figure 1 shows the indicators coverage for every nation. Results for the UN sub-regions show surprising trends since while OECD have an average coverage of 61.5 percent, East and South Asia reaches 58.4 percent coverage, Oceania 53.6 percent, Eastern Europe and Central Asia 58.4

Region	GID (%)	SDR (%)
OECD	61.5	95.0
Eastern Europe and Central Asia	58.4	76.6
Latin America and the Caribbean	62.2	76.7
East and South Asia	62.3	82.3
Middle East and North Africa	60.6	73.3
Sub-Saharan Africa	61.4	80.1
Oceania	53.6	54.5

Table 2 : Indicators coverage by region, in the Global SDG Indicators Database calculated by our method (GID, first column) and the Sustainable Development Report (SDR, second column). Disparities come from the data sources and indicators considered (all 247 indicators are taken into account for the coverage of the Global Indicators Database, while some were withdrawn in the SDR, see[42])

percent, Middle East and North Africa 60.6 percent, Sub-Saharan Africa 61.4 percent and finally Latin America and the Caribbean 62.2 percent. There then seems that no correlation exists between development of a nation and indicator coverage, although expected result would be that least developed countries suffer from lower coverage, due to lack of data and lower access to high tech solutions. Those rates also conflicts with the coverage of the indicators by official sources from the SDR, as we can see in Table 2. SDR reports a net difference between the Organisation for Economic Co-operation and Development (OECD) and the rest of the world. These contradictory results are to be put under perspective since the SDR coverage rates does not cover all the indicators while our method do, and include more sources for their indicator coverage, as well as weighting by population. This shows however that even the official database still lack available datas, and underlines the difficult problem of reliable data sources.

Senegal, a country from Sub-Saharan Africa and part of the Least Developed Countries (LDCs), is no stranger to this phenomenon. A detailed resume of its goal coverage in the Global SDG Indicators Database can be seen in Figure 2. 13 out of the 17 goals have more than 50 percent of coverage, Goal 13 "Take urgent action to combat climate change and its impacts" being the most lacking with up to 75 percent of its indicators missing and Goal 3 "Good Health and well-being" being the most covered goal with only 3.45 percent of its indicators missing. The coverage for all goals combined is 64.4 percent, meaning approximately two thirds of the indicators are provided. It could however profit from the use of AI, as it is documented as an enabler for 80% of Goal 13 indicators according to Vinuesa et al.[48]. Such AI applications should be built in localized context, in collaboration with local actors and policy deciders; and aim both to address the SDG gaps and the data gaps. Such an application using ML is presented in the next section.

4 Case study : canoe counting in Senegal

This section presents a framework to work towards the SDGs using ML in the LDCs. This framework is based on three principles : participatory research, context localized processes and dynamic model development. It has been applied to develop a canoe counting system in Senegal.

4.1 Context and Problematic

Fishery management through the use of AI has become a trend in the recent years, relying on the detection and identification of industrial vessels in harbors and high waters[55]. It has yet to be applied to developing countries, despite fishing being one of the main economic activity for some of them, Senegal included.

As the artisanal fisheries fish landings largely exceed industrial ones in Senegal (91% to 9%), previously developed models and datasets will be no help in applying this methodology in Senegal since they rely on industrial vessels identification. This is due to the large number of canoes and the use modern fishing gear (purse seine), as well as very mobile fleets that are largely deployed to the neighbouring countries following fish migrations and international agreements[46]. The monitoring of these fisheries through vessels identification and counting is very difficult due to its informal character, although institutional efforts were recently deployed to matriculate all canoes[34]. Yet, canoe counting and referencing is a task done twice a year by the Oceanographic Research Center of Dakar-Thiaroye (CRODT) during costly and logistically complex week-long surveys conducted by a few people.

From these surveys are estimated biomass landings, and related indicators used by decision makers for the management of the fisheries sector, impacting schedules and focus of the fisher communities. Our framework was applied to the task of canoe counting on the fishery of Kayar, one of the main artisanal harbour of the Senegalese coast between Dakar and Saint-Louis, where thousands of canoes lands fish for both local markets and world round exportation with a large seasonal amplitude. It aims for the automatisation of the counting process and could allow a precise understanding of the fisheries and their dynamics.

FIGURE 2 – Senegal indicators coverage by goal. For each goal, and indicator is considered covered if it was filled out at least once in the database, independently of the year considered, the type of data or its nature.

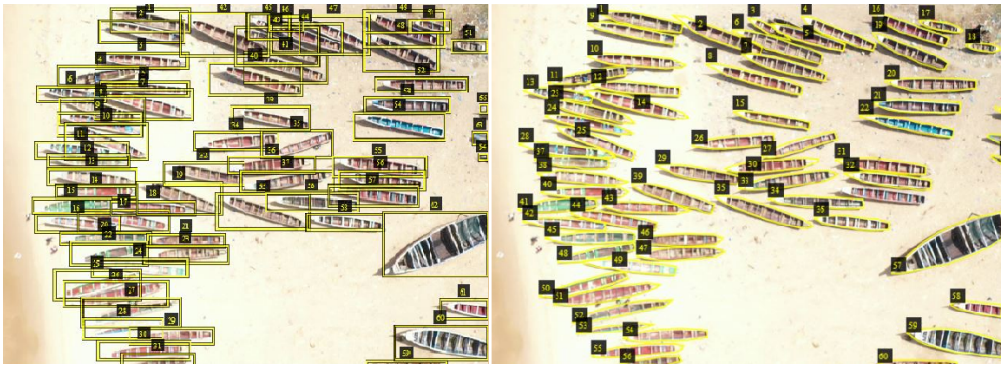


FIGURE 3 – The same image, labelled in two different ways. Left is labelled with horizontal bounding box annotations while right is labelled with region annotations. Left suffers from a high overlap between annotations, but bounding boxes annotations are way faster to generate than region annotations (3s to 15s average time per annotation).

4.2 Participatory research framework

Participatory research emphasizes the value of research partners, and stakeholders are involved in the research process definition in a collaborative and iterative approach, thus palliating one of the pitfalls of problem formulation, which is lack of live experience in the impacted field[22].

In our case, an interdisciplinary task force was set up with computer scientists, fishery scientists, governmental managers and artisanal fisheries actors in order to identify ways to improve fisheries management. Multiple aspects of fishery management were discussed and overviewed during the task force meetings, and several issues related to artisanal fishing effort variability estimates were identified. In this paper we focus on the canoe counting issue, its role in fishery management and the potential benefit of having more frequent surveys.

The process of automated counting through supervised ML methods was introduced to the task force by the authors and validated by the task force. Multiple revisions of the model development were submitted to the task force, allowing a joint development between the researchers and the actors of the fisheries, ensuring the further developments were understood and in accordance with the local needs.

4.3 Context localized dataset generation

As of now, there are a few datasets dedicated to ship detection available[2, 49], all focusing on big vessels such as tankers, carriers and containers ships, and therefore not relevant for fishery management in developing country such as Senegal, or other countries in West Africa. Senegalese canoes small size renders them barely detectable on SAR and Satellite imagery, and thus new data generation process is needed to address these features constraints. Such processes must be easily reproducible as well as considerate regarding the financial and logistical needs, and thus contextually localized.

Context localized processes refers to adapting the common development pipeline to local conditions, meaning data gathering and processing must be adapted to the local context, as well as computing processes and concrete application development. In this case study, the data generation was handled thanks to publicly available drones, as the use of UAVs have been found to be crucial to develop context localized processes. Drone remote sensing is developing fast as a cost-effective and precise tool for data collection[51], while being a small financial and logistical investment.

Publicly available drones like the Mavic Pro 2 coupled with mission planners allow for automatic high quality data collection. These data can then be processed according to the datasets needs and be annotated if necessary. In this case study, the authors flew a mavic Pro 2 above the Kayar harbour with the camera facing towards the ground on multiple occasions.

A total of two datasets were built, using two different annotation systems. As the detection techniques used to generally rely on a horizontal bounding box annotation method[29, 39], the first dataset used such annotations. However, these techniques are problematic when working with dense object disposition, moreover when the objects have a large aspect ratio, meaning they are either very long or very large[50]. Moreover, these detection techniques are built to be efficient on classic object detection (e.g. for datasets like Ms COCO[28] or ImageNet[14]). However, in aerial imagery, such techniques were shown to be less efficient due to huge variation in scale, orientation, shape, and density of objects[50]. To palliate those constraints, different annotations such as region annotations proposed in RCNN[18] and Rotation region detection[52] have been proposed, but are more time-consuming to generate compared to the previous ones. Yet, the second dataset used such region annotations, as semantic segmentation models using such annotations proved to perform best on classic benchmarks[19]. See Figure 3 for visual comparison of the two types of annotations used.

The first dataset used in this case study was built from 72 images of 4K quality (5436*3648 pixels), from which we extracted 4372 images of size 512*512, as a way to fight dense object distribution. A total of 13368 horizontal bounding box annotations were manually generated by the authors on these images, which represents a whole month of

work. As stated earlier, these annotations suffer from multiple downsides. A second dataset was then built using videos in HD quality (1920x1080 pixels) from which pictures were extracted and then annotated using region annotation, without further preprocessing. This does not ensure maximum data quality, but better reflects common usage of the drones and thus maximise reproducibility. From those videos, 718 images of size 1920*1080 were extracted, from which 11080 annotations were manually generated by the authors and one paid worker during a three month period. Each dataset was split in training, validation and test set using a 8-1-1 random distribution.

4.4 Dynamic canoe counting detection model development

A canoe counting system will rely on a canoe detection model. As stated earlier, previous works all relied on large vessels detection models, and are not relevant to our case study. Moreover, previous developments lack feedback on the effectiveness and utility of the deployed models. To palliate this, our work is based on dynamic model development, which is based on continuous feedback from the development team to parties concerned in order to adapt model development to their needs. Using a spiral process, which alternate between action and collective reflection, the models were continuously refined to adapt to the task force demands.

Following this spiral process, multiple models were built and reviewed by the fishery management task force. The first developed model was using the Single Shot Detector (SSD) method, with the Pascal-VOC pre-trained weights using the VGG16 achitecture. The model was trained during 300 epochs with a batch size of 32, with data augmentation as described in [29]. The model, presented at a gathering of the task force, was judged not efficient enough as it suffered from the use of horizontal bounding boxes, unsuitable to the dense and oriented distribution of the canoes on the harbor (Figure 3). The development team then decided to switch to semantic segmentation methods in order to avoid the previous problems. As the new annotations are more time-consuming to produce, a small batch composed of 107 images was annotated by the development team with the VIA tool[15], in order to train a model that would act as a proof-of-concept. The model was trained using the MRCNN method[19] built on FPN[27], using available MS COCO pre-trained weights. This model was trained during 100 epochs with a batch size of 64, without data augmentation.

Despite its small training dataset, a comparison between automated counting and a survey showed a 10% error in terms of number of canoes. This new method was validated in a second gathering of the task force, and one fishery scientist members asked for a further feature, the identification of the canoe type. This new feature identifies as an identification task, that will be added behind the image segmentation process on all detected canoe occurrence. The new data required to train models, which is canoe type, was generated thanks to a partnership between the development team and a fisher of the task force who is an elder chief of fishermen in Kayar. This experimented fisher, respected by the community, directly showed the different types of canoes and explained the disparities between them. Further development will include new data annotation regarding this new data and a new canoe identification model that will differentiate between the different types.

5 Discussion and future works

Setting up the task force has come with various positive outcomes, such as useful feedback on the models, connections that allowed the research team to have access to the harbors for data collecting purposes and links with experienced fishers in order to better the model. In addition, playing a role in the development of the model helped working group members feel confident in the results of the model[33], thus increasing the likelihood of acceptance of the counting system in the end.

Yet, the model already developed have multiple interest : lowering the cost of surveys and speeding them up, as well as multiplying data collection in order to supervise fishers mobility. This model will allow for better fishery management and provide useful insights on Senegalese landings. Deployment of this model in other harbors and countries in West Africa is yet to come in order to assess its robustness. The use of UAVs has proven to be of benefit on several levels as there are easy to transport and deploy, generate HQ data, are low cost and prone to automation. Even though they are today easy to control and do not require an experienced pilot nor high understanding of fly techniques, it is advised to have flying experience to prevent any damage in case of unexpected events.

To the authors' knowledge, this work presents the first datasets on low-income harbors and the first ML application on fisheries management in low-income countries. It is an important step towards future developments of models to reach the SDGs. What's more, the constrained resources used for this work ensure its possible duplication in other low-income countries, which will be the focus of future works.

6 Conclusion

In this paper we proposed a framework based on three principles as a solution to sustainably scaling ML research in low income countries. Having identified some deficiencies in the SDGs, namely the data gap, we implemented a participatory approach to build solutions to mitigate this gap. We relied on two other principles to strengthen our approach, context localized processes that allowed us to ensure that our method would be duplicable in a similar context, and dynamic model development that allowed us to integrate the feedbacks into the development loop. These three principles combined together provide a solid framework for addressing the knowledge bottleneck.

The authors highlight the fact that datasets built by and for Southern countries are more necessary than ever to better monitor and promote the achievement of the sustai-

nable development goals. Including local stakeholders in the research process is a key element to develop such datasets, using their life experience to produce valuable labels. The proposed framework needs to be adapted to the local context, but is nevertheless likely to assist the process of using ML to achieve the SDGs.

Acknowledgements

We thank Cpt Mamadou Diop (DAMCP, responsible for monitoring communal marine protected areas), Modou Thiam (CRODT, responsible for database management and canoe census organisation), Abdoulaye Diop (chief of CLPA from Kayar) for their involvement in the task force. We also thank the staff of the DAMCP, Ministry of Environment, that allowed us to gather data on the Kayar harbor and accompanied us during the collection process, and in particular Cmd Mamadou Ndiaye, the current head of the Kayar marine protected area monitoring. We also thank Nancy Diouf for her work on the annotation of the data.

Références

[1] Global indicator framework for the sdgs and targets of the 2030 agenda for sustainable development, https ://unstats.un.org/sdgs/indicators/indicators-list.

[2] Planet Team (2017). Ships in satellite imagery, https ://kaggle.com/rhammell/ships-in-satellite-imagery.

[3] Adewole S. Adamson and Avery Smith. Machine Learning and Health Care Disparities in Dermatology. *JAMA Dermatology*, 154(11) :1247–1248, 11 2018.

[4] Panagiotis Agrafiotis and Konstantinos Karantzalos et al. Correcting image refraction : Towards accurate aerial image-based bathymetry mapping in shallow waters. *Remote Sensing*, 12 :322, 01 2020.

[5] Firouz Abdullah Al-Wassai and N V Kalyankar. Major limitations of satellite images, https ://arxiv.org/abs/1307.2434.

[6] Glenn Althor, James Watson, and Richard Fuller. Global mismatch between greenhouse gas emissions and the burden of climate change, https ://www.nature.com/articles/srep20281. *Scientific Reports*.

[7] Julia Angwin, Jeff Larson, and Surya Mattu et al. Machine bias : There's a software used accross the country to predict future criminals, and it's biased against blacks. 2016.

[8] Kumar Ayush, Burak Uzkent, and Marshall Burke et al. Generating interpretable poverty maps using object detection in satellite images, https ://arxiv.org/abs/2002.01612v2. 2020.

[9] M. Bruno, K. W. Chung, and H. Salloum et al. Concurrent use of satellite imaging and passive acoustics for maritime domain awareness. In *2010 International WaterSide Security Conference*, pages 1–8, 2010.

[10] Trevor G. Carter and Andrea Begin et al. Innovative use of gis and drone photogrammetry for cliff stability modelling. *Proceedings of the Institution of Civil Engineers - Maritime Engineering*, 171(3) :89–97, 2018.

[11] Gordon Christie, Neil Fendley, and James Wilson et al. Functional map of the world, http ://arxiv.org/abs/1711.07846. 2017.

[12] Iain M. Cockburn, Rebecca Henderson, and Scott Stern. *The Impact of Artificial Intelligence on Innovation : An Exploratory Analysis*, pages 115–146. University of Chicago Press, January 2018.

[13] Kate Crawford and Trevor Paglen. Excavating ai : The politics of training sets for machine learning. 09 2019.

[14] J. Deng, W. Dong, and R. Socher et al. Imagenet : A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[15] Abhishek Dutta and Andrew Zisserman. The VGG image annotator (VIA). 2019.

[16] Adam Van Etten, Dave Lindenbaum, and Todd M. Bacastow. Spacenet : A remote sensing dataset and challenge series, http ://arxiv.org/abs/1807.01232. 2018.

[17] Fukun Bi, Bocheng Zhu, and Lining Gao et al. A visual search inspired computational model for ship detection in optical satellite images. 9(4) :749–753.

[18] Ross Girshick, Jeff Donahue, and Trevor et al. Darrell. Rich feature hierarchies for accurate object detection and semantic segmentation, http ://arxiv.org/abs/1311.2524.

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-CNN, https ://arxiv.org/abs/1703.06870.

[20] Bogdan Iancu, Valentin Soloviev, and Luca Zelioli et al. Aboships – an inshore and offshore maritime vessel detection dataset with precise annotations, https ://arxiv.org/abs/2102.05869, 2021.

[21] Neal Jean, Marshall Burke, and Michael Xie et al. Combining satellite imagery and machine learning to predict poverty. *Science*, 353(6301) :790–794, 2016.

[22] Donald Martin Jr. and Vinodkumar Prabhakaran et al. Participatory problem formulation for fairer machine learning through community based system dynamics, https ://arxiv.org/abs/2005.07572. 2020.

[23] Delord Karine and Roudaut Gildas et al. Kite aerial photography : a low-cost method for monitoring seabird colonies : Kite aerial photography. *Journal of Field Ornithology*, 86 :173, 06 2015.

[24] Robert W. Kates. What kind of a science is sustainability science ? *Proceedings of the National Academy of Sciences*, 108(49) :19449–19450, 2011.

[25] Alina Kuznetsova, Hassan Rom, and Neil Alldrin et al. The open images dataset v4 : Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020.

[26] Donghyun Lee, Suna Kang, and Jungwoo Shin. Using deep learning techniques to forecast environmental consumption level. *Sustainability*, 9.

[27] Tsung-Yi Lin, Piotr Dollár, and Ross B. Girshick et al. Feature pyramid networks for object detection, http ://arxiv.org/abs/1612.03144. 2016.

[28] Tsung-Yi Lin, Michael Maire, and Serge J. Belongie et al. Microsoft COCO : common objects in context, http ://arxiv.org/abs/1405.0312. 2014.

[29] Wei Liu, Dragomir Anguelov, and Dumitru et al. Erhan. SSD : Single shot MultiBox detector, http ://arxiv.org/abs/1512.02325.

[30] Tulio Ribeiro Marco, Singh Sameer, and Guestrin Carlos. "why should i trust you ?" : Explaining the predictions of any classifier, https ://arxiv.org/abs/1602.04938. 2016.

[31] Bryn E. Morgan and Jonathan W. Chipman et al. Spatiotemporal analysis of vegetation cover change in a large ephemeral river : Multi-sensor fusion of unmanned aerial vehicle (uav) and landsat imagery. *Remote Sensing*, 13(1), 2021.

[32] T. Nathan Mundhenk, Goran Konjevod, and Wesam A. Sakla et al. A large contextual dataset for classification, detection and counting of cars with deep learning, http ://arxiv.org/abs/1609.04453. 2016.

[33] A.V. Norström and C. Cvitanovic et al. Principles for knowledge co-production in sustainability research. *Nat Sustain*, 3 :182–190, 2020.

[34] Journal officiel de la Republique du Senegal. Arrêté ministériel n° 1718 en date du 19 mars 2007.

[35] Lucas Prado Osco and Mauro dos Santos de Arruda et al. A cnn approach to simultaneously count plants and detect plantation-rows from uav imagery, https ://arxiv.org/abs/2012.15827. 2021.

[36] Lucas Prado Osco and José Marcato Junior et al. A review on deep learning in uav remote sensing, https ://arxiv.org/abs/2101.10861. 2021.

[37] Barak Oshri, Annie Hu, and et al. Infrastructure quality assessment in africa using satellite imagery and deep learning, http ://arxiv.org/abs/1806.00894. 2018.

[38] Vikas Ramachandra. Causal inference for climate change events from satellite image time series using computer vision and deep learning, http ://arxiv.org/abs/1910.11492. 2019.

[39] Joseph Redmon, Santosh Divvala, and Ross Girshick et al. You only look once : Unified, real-time object detection, http ://arxiv.org/abs/1506.02640.

[40] Pachauri R.K. and Meyer L.A. Ipcc 2014 : Climate change 2014 : Synthesis report, https ://www.ipcc.ch/report/ar5/syr/.

[41] David Rolnick and Priya Donti et al. Tackling climate change with machine learning. 2019.

[42] J. Sachs and G. Schmidt-Traub et al. The sustainable development goals and covid-19. sustainable development report. 2020.

[43] Victor Schmidt, Alexandra Luccioni, and et al. Visualizing the consequences of climate change using cycle-consistent adversarial networks, http ://arxiv.org/abs/1905.03709. 2019.

[44] Zhenfeng Shao, Wenjing Wu, and Zhongyuan et al. Wang. SeaShips : A large-scale precisely annotated dataset for ship detection. 20(10) :2593–2604.

[45] Evan Sheehan and Chenlin Meng et al. Predicting economic development using geolocated wikipedia articles, https ://arxiv.org/abs/1905.01627v2. 2019.

[46] Modou Thiaw, Pierre-Amaël Auger, and Fambaye Ngom et al. Effect of environmental conditions on the seasonal and inter-annual variability of small pelagic fish abundance off north-west africa : The case of both senegalese sardinella. *Fisheries Oceanography*, 26(5) :583–601, 2017.

[47] Thomas Vandal, Evan Kodra, and Sangram Ganguly et al. Generating high resolution climate change projections through single image super-resolution : An abridged version. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 5389–5393, 7 2018.

[48] Ricardo Vinuesa, Hossein Azizpour, and Iolanda et al. Leite. The role of artificial intelligence in achieving the sustainable development goals. *Nature Communications*, 11 :233, 01 2020.

[49] Yuanyuan Wang, Chao Wang, and Hong et al. Zhang. A SAR dataset of ship detection for deep learning under complex backgrounds. *Remote Sensing*, 11(7) :765.

[50] Gui-Song Xia, Xiang Bai, and Jian et al. Ding. DOTA : A large-scale dataset for object detection in aerial images.

[51] Tian-Zhu Xiang, Gui-Song Xia, and Liangpei Zhang. Mini-UAV-based remote sensing : Techniques, applications and prospectives. 7(3) :29–63.

[52] Xue Yang and Qingqing Liu et al. R3det : Refined single-stage detector with feature refinement for rotating object, http ://arxiv.org/abs/1908.05612.

[53] Xue Yang, Hao Sun, and Kun et al. Fu. Automatic ship detection of remote sensing images from google earth in complex scenes based on multi-scale rotation dense feature pyramid networks, https ://arxiv.org/abs/1806.04331v1.

[54] B. Yu, K. Shi, and Y. Hu et al. Poverty evaluation using npp-viirs nighttime light composite data at the county level in china. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(3) :1217–1229, 2015.

[55] Changren Zhu, Hui Zhou, Runsheng Wang, and Jun Guo. A novel hierarchical method of ship detection from spaceborne optical image based on shape and texture features. *IEEE Transactions on Geoscience and Remote Sensing*, 48(9) :3446–3456.

Transition between cooperative and collaborative interaction modes for human-AI teaming

Adrien Metge[1,2], Nicolas Maille[1], Benoît Le Blanc[2]

[1] ONERA, BA 701, 13661 Salon-de-Provence
[2] ENSC, IMS Laboratory, 109 Avenue Roul, 33400 Talence

adrien.metge@onera.fr, nicolas.maille@onera.fr,
benoit.leblanc@ensc.fr

Résumé

Avec l'introduction de l'IA dans le pilotage des véhicules terrestres ou aériens, la répartition des rôles entre opérateur et système devra évoluer de manière dynamique. A travers une expérimentation en micro-monde sur la supervision d'un drone intelligent, nous étudions comment une telle transition entre modalités d'interaction coopératives et collaboratives peut affecter l'expérience et le choix de l'opérateur. Nous observons des variables comme le sentiment de responsabilité ou la confiance et constatons que les opérateurs ont faiblement conscience de l'influence de l'IA sur leur propre prise de décision.

Mots-clés

Equipe homme-autonomie, automatisation adaptative, interactions homme-IA, coopération, collaboration

Abstract

With the introduction of AI in the piloting of land or air vehicles, the distribution of roles between operator and system will have to evolve dynamically. Through a micro-world experiment on the supervision of an intelligent UAV, we study how such a transition between cooperative and collaborative interaction modes can affect the experience and the choice of the operator. We observe variables such as the feeling of responsibility or trust and find that operators have little awareness of the influence of AI on their own decision making.

Keywords

Human-autonomy teaming, adaptive automation, human-AI interaction, cooperation, collaboration

1 Introduction

Advances in Artificial Intelligence (AI) make it possible to envisage for the transportation industries the arrival of systems with a level of autonomy that evolves according to the needs of the user, from driver assistance to substitute driving. In October 2020, SNCF ran a BB 27000 freight locomotive for the first time in partial autonomy, under real operating conditions, with fully automated acceleration and braking functions [15]. Another first achievement in June 2020 concerned Airbus succeeding in making the taxiing, takeoff and landing of an A350 aircraft autonomous using on-board image recognition technology [1]. The increasing automation of aeronautical systems allows us to consider an improvement in safety while reducing the workload of pilots, and contributes to a progression towards cockpits that would be operable by a single pilot working as a team with AI. However, between two human operators, the division of labor and the way in which they team up can change depending on the situation. For example, during the flight for a go-around or for failure management, a change of flying pilot can be decided. An AI system replacing the second pilot would have to deploy adaptive automation to accommodate these changes in the distribution of roles that may occur, that is to say functions may have to be shared or exchanged between humans and machines in response to change in situation or human performance [7].

The evolution of the pilot towards the role of operator supervising operations requires the development of trusted AI to certify such frameworks, and will not be possible without a better understanding of the impact of adaptive automation on human behavior. Furthermore, what is true for aircraft pilots is also true for unmanned aerial vehicle (UAV) crews. It is even amplified by the physical distance between the human operator and a part of the system he or she is supervising, namely the airborne vector, because the ground station, if it includes AI, remains co-located with the operator. The central problem is decision making for choices that put the success or safety of the mission at stake, especially in the case of unexpected events or degraded situations, and that is why a human operator is left in the decision loop [4]. The aim is therefore to

define an interaction methodology that enables the human operator and the AI system to communicate their mutual points of view to achieve optimal decision-making in situations of uncertainty, and also encourages the critical judgment of the operator on the system's proposals. In this sense, we can consider the operator and the AI system as a team, a social entity composed of members who interact, synthesize and share information and expertise in order to achieve common goals.

2 Related works

Managerial and management sciences are interested, among other things, in how decisions are made within organizations. According to Roy and Bouyssou [13], "even if the ultimate responsibility for a decision rests with a clearly identified individual, it is often the result of interactions between multiple actors during a decision-making process". Thus, even if in an organization involving human operators and AI it may be agreed that the final decision remains the responsibility of human, this decision is still the result of the process of interaction between the different actors. Understanding the elements of these interactions and how they can shape the final decision making is an important issue to better develop systems based on a combination of human and AI. Turoff, White and Plotnick [16] highlight that in the literature the terms collaboration, cooperation, coordination are too often confused and they propose a scale that differentiates five increasing levels of communication in group decision-making: competitive: no trust in the information transmitted; informative: honest exchange of information about what each party is doing; coordination: mutual scheduling of what each party is doing and when; cooperation: mutual agreement on what tasks each party will perform; collaboration: mutual agreement to work together on the same tasks. In the context of decision making for important elements that could jeopardize the objectives or security of the mission, it is essentially the levels of cooperation and collaboration that can be at stake. We will therefore now detail what they cover.

Cooperative teamwork. For Dillenbourg [5], cooperation and collaboration do not differ in terms of whether the task is divided, but in the way it is divided. In cooperation, the task is hierarchically divided into independent subtasks, whereas in collaboration, cognitive processes can be heterogeneously divided into intertwined layers. Piquet [12] also defines cooperative work as a collective organization of work in which the task to be satisfied is fragmented into subtasks. Each of these subtasks is then assigned to an actor, either according to a perfectly horizontal distribution in which tasks and actors are equivalent, or according to a logic of assignment according to the skills of each one. This is a rationalized division of a task into actions that will be distributed among actors acting autonomously. Cooperative work is thus hierarchically organized and planned group work involving deadlines and task sharing according to precise coordination. Each member of the group knows what he or she must do from the beginning and communicates, exchanges or shares elements only to reach his or her individual goal [10]. At the end, everyone's work is brought together to create a single object of work. In other words, it is the progressive and coordinated succession of each person's actions that makes it possible to achieve the final objective. The responsibility of each person is committed to the sole accomplishment of the tasks that are specific to them.

Collaborative teamwork. Collaborative work, on the other hand, does not involve an a priori distribution of roles, but a merging of individual contributions in action. Interpersonal interactions are permanent to ensure overall coherence, a necessary condition for the efficiency of the action and the achievement of the final objective. Collaboration within the framework of collective work is a modality that goes beyond individual action by explicitly implicating itself in a dynamic of collective activity. It is a question for each actor of a project to feed his individual contributions with those of the others. Collaborative work implies a mutual commitment of individuals in a coordinated effort to carry out the same task and solve the same problem together [2]. It requires team members to be more interactive and more motivation and interpersonal trust than other methods of work organization [14]. Nevertheless, human-AI teams are different in nature from the human-human teams that the concepts of cooperation and collaboration generally describe. There are important differences in the ways in which human and machine acquire and process information which could make trust much more difficult to build [8]. Klein [9] explains that one of the necessary conditions for the effectiveness of a group is interpretability, i.e. the ability to predict the actions of other parties with a reasonable degree of accuracy. Each member of the group should strive to make their actions sufficiently predictable to allow for effective linkage. However, user priorities may change over time, which may require the system to manage and adapt to changing constraints and preferences. If AI requires a precise definition to solve the problem mathematically, human actions are in fact of limited rationality and often informed by heuristics that are partly spurious [17]. There may be a mismatch between the precise mathematical objectives required by the AI and the potentially fuzzy specifications that can be provided and manipulated by humans. Heterogeneity in teams can be a source of difficulties in establishing a mental model of the other, but it is also a source of opportunities because team members can complement each other by providing skills they lack [11]. Thus, when AI systems need to interact with human users, it is not so much important to reason rationally as to emulate human-type reasoning [3]. An effective decision support system must not only provide quality information, but also consider the user's constraints in terms of available cognitive resources and personal preferences.

In this study, we define teamwork as cooperative when tasks are distributed among members, and collaborative when any member can contribute to any task.

3 Study hypothesis

It appears that the way of interacting between operators can be structured in a different way according to the chosen organization, the trust between the participants, the commitment

of each one in the collective process. Other factors such as the stakes, the time pressure, the nature of the activity to be carried out can also influence this interaction process, which is not necessarily fixed in time. In the case of collective decision-making, the organization of interactions but also their temporal evolution could be elements that directly influence the outcome of the decision-making process. To investigate how the way of teaming up with a decision support system will influence an operator's decision making, we have developed an experimental micro-world. This environment allows us to simulate a UAV flight replanning task that a participant performs with a mode of interaction with the AI system that can evolve between cooperation and collaboration. The aim of the study is to better understand, in an experimental way, how the modes of interaction between a human operator and an AI can impact the decision resulting from the interactive process and how this decision is considered and accepted by the human operator. Our hypotheses are:
H1: The evolution of the mode of interaction has an impact on the operator's feeling about the decision-making process.
H2: The interaction mode has an impact on the variability of the decisions made by the operator.

4 Experimental study

4.1 Task description

A group of 20 healthy PhD students and young engineers (40% female), with an average age of 26.1 years (SD = 2.7 years), participated in this study. All subjects volunteered to participate in the study and gave their full informed consent before taking part in the experiment. They embodied a military air operator responsible for supervising a UAV to carry out missions in enemy territory (Figure 1). The objective of the missions is to fly over several targets to photograph them and then leave the enemy zone, while minimizing the risks taken and the fuel consumed. The missions take place on different territories but with a similar scenario: 1) the UAV heads towards the enemy zone with an initial flight plan, 2) enemy entities are suddenly detected, so the flight plan is no longer satisfactory, 3) the operator interacts with the system to define a new flight plan, 4) the operator validates a new flight plan, which completes the supervision task. The interaction phase occurs either in cooperative or collaborative condition, and has to be finished before entering the adverse territory.

Cooperative condition. In this operating mode, the roles between the operator and the assistance system are fixed: the operator defines the high-level tactical elements that force the modification of the plan (crossing points, objectives removed from the mission...) then the assistance system produces the optimized path taking these elements into account.

Collaborative condition. In this operation mode, the distribution of roles is not fixed and the assistance system will also propose tactical changes for the realization of the mission. The system will do this on its own initiative at the start of the replanning phase, and may then suggest other types of solutions (modification of crossing points, objectives retained or deleted....) at the operator's request. The system can also ask the operator to confirm the classification of an image for which the uncertainty would be high, and whose level of dangerousness if it were different could imply modifications to the plan (Figure 2). The tools present in cooperative mode are also present in this more advanced interaction mode, but the proposed solutions remain the same in both conditions, the constraint optimization algorithm being for its part unchanged.

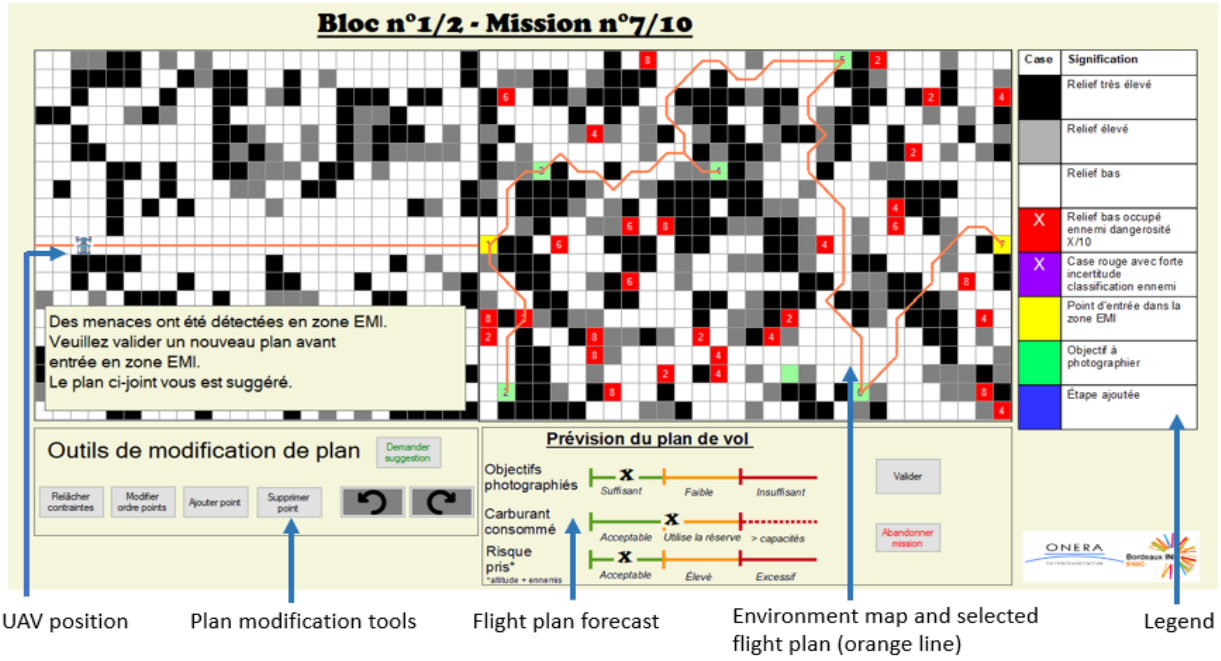


Fig. 1. HMI for replanning task in collaborative condition

Fig. 2. Tool for checking the classification of an enemy by the operator.

4.2 Metrics

All the participants completed 20 missions in the same order. Half of the participants carried out the first 10 in cooperative condition, and the last 10 in collaborative condition. The other half of the participants carried out the first 10 missions in collaborative condition, and the last 10 missions in cooperative condition. To study how the evolution of the teaming method influences the operator's final decision making, we defined two categories of metrics:

Metrics of the operator's feelings about the chosen solution. We use four metrics to evaluate the quality of teaming between the operator and the AI according to the level of teaming. After each completed mission, the participants answered three questions in the interface on 7-item Likert scales about their:

- Feeling of responsibility in the validated solution.
- Feeling of authorship of the validated solution, i.e. according to the operator who of him/her or of the system took the most part in its design.
- Confidence in the validated solution.

After completing all missions in an interaction mode, participants answered a NASA Task Load Index (NASA TLX) questionnaire to measure their perceived workload for the task [6].

Metrics for variability of decision made. To compare the dispersion of the solutions validated according to the interaction mode, we superimpose for each mission the plans validated in each of the two conditions (one plan for each participant). Then, we calculate for each mission the difference between the number of boxes that are never used in the cumulative plan of one condition and in the cumulative plan of the other condition. The sign of the dispersion index thus obtained gives us information on the condition for which the solutions are the most spread out on the map.

5 Results

Data from all 20 participants were included in the analysis. The metrics from the 10 completed missions were averaged for everyone for each condition. We set a threshold of 5% for the significance of p-values.

Impact of the evolution of the interaction mode on the operator's feeling. To test H1, we conducted repeated measures analysis of variance (ANOVA) to compare the effect of the interaction mode evolution (*cooperation then collaboration*, or *collaboration then cooperation*) on metrics of the operator's feeling about the chosen solution (feeling of responsibility, feeling of authorship, confidence and cognitive workload). There was a significant effect of the interaction evolution *collaboration then cooperation* on feeling of responsibility ($F(1,198) = 20.40$, $p < 0.001$), on feeling of authorship ($F(1,198) = 40.53$, $p < 0.001$), and on cognitive workload ($F(1,9) = 6.5$, $p = .03$), but not on confidence ($F(1,198) = 0.51$, $p = 0.47$). On the other hand, we found no significant effect of the teaming evolution *cooperation then collaboration* on feeling of responsibility ($F(1,198) = 0.03$, $p = .86$), on feeling of authorship ($F(1,198) = 3.84$, $p = .05$), on cognitive workload ($F(1,9) = 1.5$, $p = .24$) and on confidence ($F(1,198) = 0.54$, $p = 0.46$). Some of these results are in line with our hypothesis as they show as expected that the transformation of task distribution with AI influences the operator's experience. However, when the transition is from cooperative to collaborative teaming, the reverse effect is not observed. This asymmetrical change (Figure 3) does not comply with our hypothesis.

Impact of the interaction mode on the operator's decision making. To test H2, we performed a one-tailed t-test on the list of indices of dispersion of missions ($t(19)= -3.08$, $p=.003$). The average of the indices is significantly negative, which means that in collaborative teaming there are more boxes that are not covered by any validated plan than in cooperative teaming. In this sense, the variability of decisions made in collaboration is therefore lower than those made in cooperation with the AI system, which supports our hypothesis. A visual example of variability differences between teaming conditions can be visually observed for one mission in Figure 4.

6 Discussion

For the integration of Artificial Intelligence in vehicles, and especially aircrafts, cross-comprehension and trust between the decision support tools and the operator is a prerequisite to ensure the system's resilience. In this paper, we describe an experimental study where an operator performs a UAV flight replanning task assisted by a decision support system with varying levels of investment. The operator enters a process of co-constructing a flight plan in a complex environment requiring compromises between different constraints thanks to the assistance of the system. We seek to characterize the user's feeling and we hypothesize that it depends on the mode of interaction with the AI, but also on its evolution. We observe that as the operator-AI team moves from a collaborative to a cooperative teaming, where the system will progressively take less initiative in the action, the operator will feel more at the origin and responsible for the decisions taken. Withdrawing assistance to which the operator is accustomed creates a sense of loss. However, when AI will introduce plan suggestions

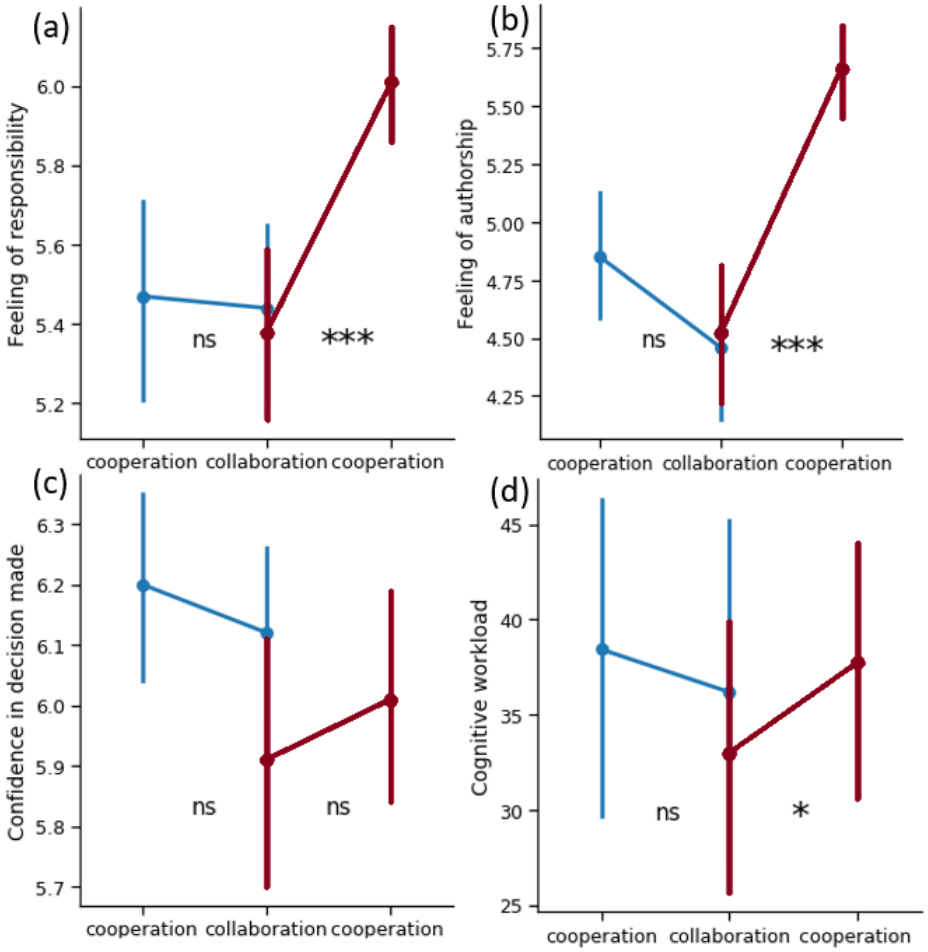


Fig. 3. Impact of the evolution of the interaction mode on the operator's feeling. (a) Sense of responsibility in the decision taken. (b) Sense of authorship in the decision taken. (c) Confidence in the decision made. (d) Felt cognitive workload for the task.

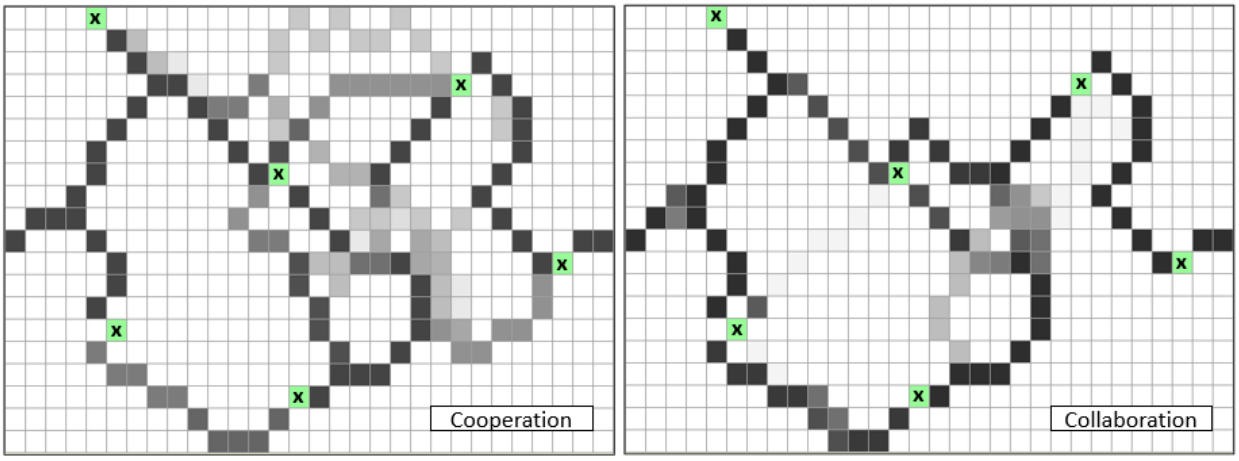


Fig. 4. Superimposition of the plans validated by the participants for one of the missions. The gray level associated to each box represents the proportion of plans validated for this level of teaming that passes through it (superimposition of 10 validated plans for each figure). The green crossed-out boxes represent the objectives to be photographed. (left) The mission is carried out in cooperative condition. (right) The mission is carried out in collaborative condition.

that were not previously present, the operator will not feel disinvested in his role. The help is accepted and does not disengage the operator. This asymmetry to change can be problematic insofar as we observe that the teaming method will nevertheless influence the decisions validated by the operator. The flight plans carried out with a cooperative organization are more varied than those carried out in a collaborative organization. When suggestions for plans are proposed by the system, the operator will tend to validate plans that are close to them, but without being conscious of this influence on his behavior. This work demonstrates that a system dedicated to assist a human operator in a decision-making task can insidiously modify the solution he or she validate. This is a critical issue when developing sophisticated support systems while the user is not aware of been influenced by the AI system. Such behavior is somewhat standardized by the AI suggestions and he or she is no longer challenging the potential solution in the same way. On the one hand, the reduction of human variability can be considered desirable because the predictability of the system can lead to a better overall reliability. On the other hand, the interest of keeping human operators lies precisely in their capacity to display a critical and subjective mind in the decision process. These first results suggest investigating if adding explanations about strengths and weaknesses of the proposed solutions, or highlighting underlying choices that lead to these solutions could reactivate the operator's alertness and avoid compliance to the AI suggestions. Moreover, the experiment shows that users combine the use of elementary and more sophisticated tools in both cooperative and collaborative modes. The next experiment should rely on a new interface, designed to enhance collaborative tools based for which human and AI initiatives could be more interdependent.

Acknowledgments. The research project of which this study is a part was granted by Agence de l'Innovation de Défense (AID) and Office National d'Etudes et Recherches Aérospatiales (ONERA).

7 References

[1] Airbus. (2020, June 29). Airbus concludes ATTOL with fully autonomous flight tests [Press release]. Retrieved from https://www.airbus.com/newsroom/press-releases/en/2020/06/airbus-concludes-attol-with-fully-autonomous-flight-tests.html

[2] Barbour, R. (2018). Collaboration versus Cooperation: Grassroots Activism in Divided Cities and Communication Networks. International Journal of Humanities and Social Sciences, 12(2), 292-295.

[3] Besold, T. R., & Uckelman, S. L. (2018). The what, the why, and the how of artificial explanations in automated decision-making. arXiv preprint arXiv:1808.07074.

[4] Board, D. I. (2019). AI Principles: Recommendations on the Ethical Use of Artificial Intelligence by the Department of Defense. Supporting document, Defense Innovation Board.

[5] Dillenbourg, P., Baker, M., Blaye, A., & O'malley, C. (1996). The evolution of research on collaborative learning In H. Spada and P. Reimann (Eds) Learning in Humans and Machines. *Elsevier*, *1*(1), 58-94.

[6] Hart, S. G. (2006, October). NASA-task load index (NASA-TLX); 20 years later. In Proceedings of the human factors and ergonomics society annual meeting (Vol. 50, No. 9, pp. 904-908). Sage CA: Los Angeles, CA: Sage publications.

[7] Inagaki, T. (2003). Adaptive automation: Sharing and trading of control. Handbook of cognitive task design, 8, 147-169.

[8] Kampik, T., Nieves, J. C., & Lindgren, H. (2019, May). Explaining sympathetic actions of rational agents. In International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems (pp. 59-76). Springer, Cham.

[9] Klein, G., Feltovich, P. J., Bradshaw, J. M., & Woods, D. D. (2005). Common ground and coordination in joint activity. Organizational simulation, 53, 139-184.

[10] Kozar, O. (2010). Towards Better Group Work: Seeing the Difference between Cooperation and Collaboration. In English Teaching Forum (Vol. 48, No. 2, pp. 16-23).

[11] Nissen, H. A., Evald, M. R., & Clarke, A. H. (2014). Knowledge sharing in heterogeneous teams through collaboration and cooperation: Exemplified through Public–Private-Innovation partnerships. *Industrial Marketing Management*, *43*(3), 473-482.

[12] Piquet, A. (2009). Guide pratique du travail collaboratif : Théories, méthodes et outils au service de la collaboration. Document destiné au «Groupe Communication» du réseau Isolement Social Brest.

[13] Roy, B., Bouyssou, D. (1993). Aide multicritère à la décision : méthodes et cas, Paris, Economica.

[14] Schöttle, A., Haghsheno, S., & Gehbauer, F. (2014, June). Defining cooperation and collaboration in the context of lean construction. In Proc. 22nd Ann. Conf. of the Int'l Group for Lean Construction (pp. 1269-1280).

[15] SNCF. (2020, December 16). SNCF et ses partenaires font circuler le premier train semi-autonome sur le réseau ferré national [Press release]. Retrieved from https://www.sncf.com/fr/groupe/newsroom/train-semi-autonome

[16] Turoff, M., White, C., & Plotnick, L. (2011). Dynamic emergency response management for large scale decision making in extreme hazardous events. In Supporting real time decision-making (pp. 181-202). Springer, Boston, MA.

[17] Tversky, Amos and Kahneman, Daniel, (1992), Advances in Prospect Theory: Cumulative Representation of Uncertainty, Journal of Risk and Uncertainty, 5, issue 4, p. 297-323.

Le coaching : un nouveau cadre pour la recommandation automatique en vue de modifications durables du comportement

J. Vandeputte[1], A. Cornuéjols[1], N. Darcel[2], F. Delaere[3], Ch. Martin[1]

[1] UMR MIA-Paris, AgroParisTech, INRAe, Université Paris-Saclay. 16, rue Claude Bernard. Paris (France)
[2] UMR PNCA, AgroParisTech, INRAe, Université Paris-Saclay. 16, rue Claude Bernard. Paris (France)
[3] Danone Nutricia Research. Palaiseau (France)

jules.vandeputte@agroparistech.fr

Résumé

Cet article introduit un nouveau scénario de recommandation : le coaching. Dans ce scénario, l'objectif est de d'aider un utilisateur à modifier durablement ses propres préférences, dans un contexte de choix répétés. À chaque fois que l'utilisateur exprime un choix, le coach peut lui suggérer une modification afin de le guider vers de meilleures habitudes. Nous montrons que la meilleure stratégie du coach dépend des caractéristiques de l'utilisateur. Six stratégies de coaching, dans lesquelles le coach apprend les caractéristiques de l'utilisateur en cours d'interactions ont été comparées.

Mots-clés

Recommandation. Apprentissage. Apprentissage par renforcement.

Abstract

This article introduces a new recommendation scenario called coaching. In this scenario, the goal is to help the user modifying his consumption habits lastingly. Each time the user U expresses a choice, the coach can suggest a modification in order to guide U towards better habits. We show that the best coaching strategy depends on the user's characteristics. Six coaching strategies, in which the coach learns the user's characteristics during interactions, have been compared on an illustrative example.

Keywords

Recommending Systems. Machine Learning. Reinforcement Learning.

1 Introduction

Les développements récents, notamment de méthodes telles que l'apprentissage profond par renforcement, ont amené ces dernières années à de nombreuses avancées dans le domaine des systèmes de recommandation. Ils sont ainsi largement utilisés afin d'aider les utilisateurs de diverses plate-formes à gérer la surcharge d'information à laquelle ils font face. S'il existe quelques études s'intéressant aux effets de tels systèmes sur les préférences de l'utilisateur, la vaste majorité des travaux se concentre sur la manière d'apprendre au mieux ces préférences, afin de fournir à l'utilisateur à chaque instant t la ou les propositions les plus susceptibles d'être acceptée(s). Ces systèmes de recommandation visent donc une maximisation du nombre de recommandations acceptées par l'utilisateur à chaque instant t sans que la notion d'histoire des préférences ne soit prise en compte. Cependant, une recommandation à un instant t peut avoir un effet sur les choix futurs de l'utilisateur.

Le but de cet article est de proposer un nouveau cadre permettant d'étudier et de concevoir des systèmes de recommandation dont l'objectif est d'accompagner un utilisateur dans un processus de modification durable de ses préférences. Ce pourrait par exemple être le cas d'un utilisateur souhaitant améliorer son régime nutritionnel en apprenant progressivement à modifier ses choix d'aliments.

Le principe est de raisonner en terme de trajectoire parcourue par l'utilisateur dans l'espace des préférences, le rôle du système de recommandation étant de faire parcourir à l'utilisateur une trajectoire l'amenant, depuis son habitude initiale de choix, à une habitude de choix meilleure, voire optimale, selon une certaine fonction de score. Une originalité de l'approche proposée ici est que le système de recommandation ne propose pas des items à l'utilisateur, mais s'appuie sur les préférences exprimées par celui-ci, son choix d'items à l'instant t, pour proposer éventuellement une modification acceptable de ce choix. En ceci, ce type de systèmes de recommandation mérite d'être appelé « système de *coaching* », car il agit comme un coach personnalisé qui analyse les choix de l'utilisateur pour suggérer des modifications bénéfiques à terme, comme le coach d'un sportif guiderait ce dernier pour améliorer ses performances.

Cet article est organisé comme suit : la section 2 définit le coaching, et la manière dont le coach et l'utilisateur interagissent. La section 3 analyse les travaux pertinents par rapport au nouveau cadre proposé. La section 4 illustre à l'aide d'un exemple simple le problème de coaching et montre en particulier que la stratégie optimale du coach dépend des caractéristiques de l'utilisateur, d'où l'intérêt d'un coaching personnalisé. Plusieurs méthodes d'apprentissage sont comparées expérimentalement dans la section 5 sur un cas simplifié mais représentatif des problèmes de coaching

possibles. La section 6 évoque des pistes pour adapter le système de coaching à des scénarios plus généraux. Finalement, la section 7 tire des leçons de cette étude novatrice dans les systèmes de recommandation.

2 Le coaching : un nouveau cadre pour la recommandation

Le coaching consiste à chercher une trajectoire acceptable par l'utilisateur pour le conduire vers une habitude de choix meilleure. Afin de formaliser ceci, il faut définir les notions de préférence de choix et de score.
Formellement, nous considérons un ensemble $\mathcal{I}$ d'items représentant des choix possibles pour l'utilisateur, et nous supposerons, dans le travail présenté ici, qu'un score peut être attribué à chaque item $i \in \mathcal{I}$ indépendamment des autres. Il s'agit donc d'un score additif. D'autres types de score seraient envisageables, comme nous l'évoquons en section 6

$$Sc : \begin{cases} \mathcal{I} \to \mathbf{R} \\ i \mapsto \text{score}(i) = Sc_i \end{cases}$$

Dans la suite, afin de simplifier l'exposé, nous supposons que l'utilisateur ne propose qu'un seul item à chaque pas de temps. Nous représentons alors une habitude de choix à un instant t par un vecteur de probabilités défini sur l'ensemble des items définis précédemment : $\Pi_t = (\pi_t(i))_{i\in\mathcal{I}}$, $\pi_t(i)$ représentant la probabilité de choix de l'item i à t. On peut alors associer à un tel vecteur Π_t une espérance de gain selon le score :

$$V[\Pi_t] = \sum_{i\in\mathcal{I}} \pi_t(i) \cdot \text{score}(i) \tag{1}$$

2.1 L'interaction entre l'utilisateur et le coach : un jeu itéré à deux joueurs

La modification durable des habitudes de choix d'un utilisateur suppose des interactions entre l'utilisateur et le coach prenant place dans le temps. Il est naturel de modéliser ce processus par un jeu itéré à deux joueurs : C le coach et U l'utilisateur. Nous proposons ici un mécanisme d'interaction en quatre temps.

1. U fait à C une proposition d'item, par exemple i, en utilisant son vecteur de préférences Π_t.
2. C analyse la proposition de U, et suggère, s'il le juge utile, une proposition de modification $i \to j$, à partir de ses connaissances de la valeur des items, et de son estimation de la capacité de U à accepter la proposition.
3. U accepte ou refuse la proposition de substitution fournie par C.
4. Si U accepte la proposition de C, il modifie, en fonction de sa capacité d'apprentissage, son vecteur de préférences Π_t de sorte à proposer de lui même plus fréquemment l'item recommandé. Sinon U ne modifie pas le vecteur de préférence. C'est ainsi que l'on rend compte du fait que U peut apprendre au fil de ses interactions avec C, l'idée étant que U, s'il accepte la modification $i \to j$ proposée par C, est davantage prêt à choisir j à l'avenir au lieu de i.

Une suggestion de substitution $i \to j$ sera plus ou moins acceptable ou réalisable selon l'utilisateur U concerné.

2.2 Modèlisation de l'utilisateur

L'utilisateur est caractérisé par trois comportements :

1. Le **choix** d'un item i selon son vecteur de probabilité instantané Π_t.
2. Son **acceptation ou refus** de la proposition de substitution par le coach. Nous supposerons que ce comportement est contrôlé par une matrice $\mathbf{M} : \mathcal{I}\times\mathcal{I} \to [0,1]$ exprimant pour chaque couple d'item $(i,j) \in \mathcal{I}^2$ la substituabilité entre i et j (ie. la probabilité que la substitution $i \to j$ soit acceptée). Dans la suite de cet article, nous supposerons cette matrice constante. Elle serait cependant susceptible de dépendre de t.
3. **Apprentissage** par mise à jour de son vecteur de probabilité Π_t quand la substitution $i \to j$ est acceptée. Afin de traduire une transmission de probabilité de l'item i à l'item j, on modélise cet apprentissage de la manière suivante :

$$\forall t \in \mathcal{T} : \begin{cases} \Pi_{t+1}(i) \;=\; (1-\lambda)\,\Pi_t(i) \\ \Pi_{t+1}(j) \;=\; \Pi_t(j) \;+\; \lambda\,\Pi_t(i) \end{cases} \tag{2}$$

Le paramètre λ représente le taux d'apprentissage de l'utilisateur. Si $\lambda = 0$, alors l'utilisateur n'apprend rien et ne modifie pas ses préférences au fil des interactions. Si $\lambda = 1$, l'utilisateur est un apprenant « parfait », et après chaque substitution $i \to j$ acceptée il transfère l'intégralité de sa probabilité de proposer i vers sa probabilité de proposer j.

2.3 Modélisation du coach

Nous définissons le coach par sa fonction de choix de substitution qui peut évoluer avec le temps $c_t : i \in \mathcal{I} \to j \in \mathcal{I}$. Cette fonction de choix dépend de :

1. La fonction de score Sc qu'il connaît.
2. L'estimation des caractéristiques de U : Π_t, $\mathbf{M}$ et λ.

On fait ici l'hypothèse que le coach représente l'utilisateur à l'aide de ces trois caractéristiques. Celles-ci étant propres à chaque utilisateur, le coach devra chercher à les estimer au fil de ses interactions avec ce dernier.

2.4 Comment évaluer un coach

Le coaching a pour but de faire suivre à l'utilisateur U une trajectoire dans l'espace des préférences, c'est-à-dire, ici, dans l'espace des vecteurs de probabilité Π_t. L'évaluation d'une stratégie de coaching, c'est-à-dire de fonction de choix c_t au cours du temps, doit donc se définir par rapport à ces trajectoires.
Notons $\Pi^\star$ les préférences optimales selon la fonction de score Sc Leur valeur associée est : $V^\star = V(\Pi^\star)$, la valeur maximale atteignable. Par ailleurs, nous noterons $V(\Pi_0)$ l'espérance de score associée au vecteur de préférence initial de l'utilisateur.

Plusieurs options sont envisageables. Nous en mentionnons trois ici :

1. On cherche à guider U vers un vecteur de préférence Π tel que : $V(\Pi) \geq \gamma V^\star$ avec $\gamma \in [0,1]$. La performance du coach est ainsi mesurée en terme de *nombre moyen d'interactions* $\overline{T}_\gamma$ pour atteindre ce niveau de performance à partir du vecteur initial Π_0.
2. Une mesure duale consiste à mesurer le *gain de performance moyen* $\overline{V}_T = \text{moyenne}\big(V(\Pi_T) - V(\Pi_0)\big)$ après T interactions.
3. Il est également possible d'envisager un critère défini sur l'ensemble de la trajectoire suivie par l'utilisateur, par exemple le *gain cumulé sur l'ensemble de la trajectoire* par rapport à l'espérance de gain initiale : $G(T) = \sum_{t=1}^{T}\big(V(\Pi_t) - V(\Pi_0)\big)$.

Dans la suite de cet article, nous nous concentrerons sur le deuxième critère. Il permet en effet des comparaisons aisées, notamment dans le cas d'un utilisateur réel, ayant avec le système un nombre d'interactions limité.

3 Travaux reliés

La recommandation en vue de modifier durablement les préférences d'un consommateur a été peu étudiée jusqu'à présent, et les travaux s'attaquant à cette question ont principalement porté sur la conception d'interfaces informatiques incitant l'utilisateur à les utiliser et à en suivre les indications. Ces travaux ressortent davantage de l'étude des caractéristiques psychologiques mises en jeu lors des interactions avec des interfaces homme-machines (voir par exemple [7]).
Nous distinguons ici les travaux relatifs à la recommandation, et ceux donnant une modélisation mathématique permettant d'étudier le coaching comme un problème d'optimisation.

Point de vue des systèmes de recommandation

Dans [3], les auteurs explorent les effets durables des systèmes de recommandation sur les habitudes de consommation, et montrent qu'ils peuvent engendrer une homogénéisation des comportements. Quelques travaux ont également été publiés sur la recommandation avec pour but de modifier le comportement. Les auteurs de [4] ont proposé en 2012 un algorithme fondé sur une approche de recommandation intra-personnelle : plutôt que de s'intéresser aux comportement des autres utilisateurs, comme dans une approche de type filtrage collaboratif, les auteurs étudient les différents comportements typiques des utilisateurs pour baser leur recommandation. L'algorithme met en lien le comportement de l'utilisateur et la mesure du but recherché, afin de proposer à l'utilisateur des modifications qu'il serait susceptible d'accepter.
Une autre approche, basée sur les modèles de Rasch a également été étudiée dans plusieurs publications [8, 9]. Dans [9] notamment, les auteurs classent les buts poursuivis en fonction de leur difficulté définie selon une échelle de Rasch. Cela permet de conduire les utilisateurs à satisfaire des buts successifs, en commençant par les plus simples. Cette approche se base sur l'hypothèse que plus une recommandation est simple à suivre, et donc fréquemment suivie, plus l'utilisateur va se l'approprier.
Dans [6] sont distingués deux cas : celui où la modification de préférences est initiée par l'utilisateur, et celui où elle est initiée par le système. Ce travail soulève l'importance de la nature progressive de l'évolution des préférences et donc de la nécessité d'en tenir compte en proposant des choix acceptables par l'utilisateur.

Point de vue apprentissage par renforcement

À chaque étape t du coaching, le système de recommandation se trouve face au problème du choix de l'item j à suggérer comme substitution pour l'item i proposé par l'utilisateur, celui-ci ayant été décidé selon le vecteur de probabilité Π_t. Ce choix vise à optimiser le critère de performance décrit en section 2.4 et on a donc :

$$\forall i \in \mathcal{I} : c_t(i) = \underset{j \in \mathcal{I}}{\text{ArgMax}}\, \textit{Perf} \tag{3}$$

Cependant, le critère *Perf*, qui peut être $\overline{V}_T$ ou $\overline{T}_\gamma$ ou $G(T)$, est difficile à optimiser. Il résulte en effet d'un processus doublement stochastique : le choix de l'item i_t à chaque instant t par U, régi par la distribution de probabilité Π_t, et l'acceptation de la substitution $i_t \rightarrow c_t(i_t)$ par U résultant de la matrice de probabilité $\mathbf{M}$, ce qui entraîne un gain instantané et une modification potentielle de Π_t.
L'utilisateur peut être considéré comme l'environnement face auquel le coach doit faire ses choix pour optimiser le critère *Perf*. Cet environnement est markovien car, dans la modélisation proposée, la matrice $\mathbf{M}$ et le coefficient λ sont constants et Π_{t+1} est seulement fonction de Π_t.
Le coaching est donc un problème de décision dans un processus markovien avec un environnement, l'utilisateur U, imparfaitement connu. Le coach est ainsi face à un dilemme exploration vs. exploitation.
Différentes techniques ont été conçues pour optimiser ce compromis. Nous citerons en particulier le problème du *bandit multi-bras* [5], dans lequel l'agent (i.e. le coach) doit choisir à chaque instant un bras (i.e. une recommandation) de manière à optimiser un gain cumulé. Chaque bras est associé à une récompense stochastique. Cependant, ce problème correspond à un environnement stationnaire. C'est pourquoi a été inventé le problème du *bandit contextuel* [1], dans lequel il peut y avoir transition entre des bandits qui soit contrôlée par le bras sélectionné à chaque instant. Le problème du *bandit turbulent* enrichit ce cadre en permettant que les propriétés des bras évoluent également avec le temps. Il est possible plus généralement de considérer la tâche du coach comme celle d'un *apprentissage par renforcement* [10] dans lequel le coach apprend par essais et erreurs les caractéristiques de son environnement et ainsi ce qui correspond ici à sa fonction de choix c_t. De fait, il faut recourir aux *processus de décision markoviens partiellement observables* (POMDP) pour rendre pleinement compte du problème du coaching puisque le coach n'a à chaque instant qu'une connaissance incomplète de l'état de son environnement, l'utilisateur, dont il n'observe que l'item choisi.

Sans développer plus avant ici les correspondances formelles avec le problème du coaching, nous utiliserons dans la suite des techniques tirées de l'apprentissage par renforcement pour aborder le problème du coaching.

4 Étude analytique d'un cas simple

On se propose ici d'étudier de manière analytique, dans un cas simple, le problème de coaching décrit en section 2. L'objectif est de montrer que la meilleure fonction de choix du coach dépend des caractéristiques de l'utilisateur et du critère de performance visé.
On considère un espace d'items à trois éléments $\mathcal{I}$ dont les scores associés sont donnés :

$$\mathcal{I} = \{i_1, i_2, i_3\}, \quad \text{avec : } \begin{cases} score(i_1) = 5 \\ score(i_2) = 20 \\ score(i_3) = 50 \end{cases}$$

et un utilisateur U dont les habitudes initiales sont définies par :

$$\Pi_0 = (1, 0, 0)^\top$$

qui indique qu'initialement U choisit toujours l'item i_1

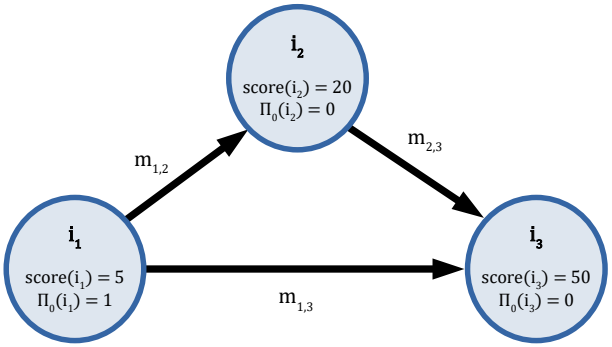


FIGURE 1 – Graphe présentant le scénario simplifié au pas de temps $t = 0$.

On suppose aussi que le coach C connaît Π_0, et maintient un modèle de U basé sur λ et $\mathbf{M}$, avec :

$$\mathbf{M} = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,1} & m_{3,2} & m_{3,3} \end{pmatrix} \quad \text{et} \quad \lambda \in [0, 1]$$

La figure 1 résume les données du problème. Comme, par définition, le coach connaît la fonction de score, il est capable d'identifier l'unique vecteur de préférence optimal $\Pi^\star = (0, 0, 1)^\top$, dont l'espérance de gain associée est : $V(\pi^\star) = 50$.
Pour les choix i_2 et i_3 de U, la fonction de recommandation du coach est évidente : $c_t(i_2) = i_3$ et $c_t(i_3) = i_3, \forall t$. Mais quelle est la meilleure recommandation quand U choisit i_1 ?

Examinons la performance associée à chacun des deux choix possibles : i_2 ou i_3.

1. Le coach choisit la substitution directe $i_1 \to i_3, \forall\, t$.
L'espérance du vecteur de préférence pour U à $t + 1$ en fonction de sa valeur en t est donnée par la formule :

$$\begin{aligned} \mathbb{E}[\Pi_{t+1}(i_1)] &= (1 - m_{1,3})\,\Pi_t(i_1) \;+\; m_{1,3}(1 - \lambda)\,\Pi_t(i_1) \\ &= \Pi_t(i_1) \;-\; m_{1,3}\,\lambda\,\Pi_t(i_1) \\ &= \Pi_t(i_1)\,(1 - m_{1,3}\,\lambda) \end{aligned}$$

d'où par récurrence : $\mathbb{E}[\Pi_t(i_1)] \;=\; \Pi_0(i_1)\,(1 - m_{1,3}\lambda)^t$.
De plus, les seuls items considérés dans ce cas étant i_1 et i_3 on a $\Pi_t(i_3) = 1 - \Pi_t(i_1)$, d'où :

$$\forall t \in \{0, \dots, T\} : \begin{cases} \mathbb{E}[\Pi_t(i_1)] = (1 - m_{1,3}\lambda)^t \\ \mathbb{E}[\Pi_t(i_2)] = 0 \\ \mathbb{E}[\Pi_t(i_3)] = 1 - (1 - m_{1,3}\lambda)^t \end{cases} \tag{4}$$

2. Le coach choisit la substitution indirecte $i_1 \to i_2$.
Par le même raisonnement que précédemment, on a :

$$\mathbb{E}[\Pi_t(i_1)] \;=\; \Pi_0(i_1)\,(1 - m_{1,2}\lambda)^t$$

Par ailleurs :

$$\begin{aligned} \mathbb{E}[\Pi_{t+1}(i_2)] &= \Pi_t(i_2) \;+\; m_{1,2}\,\lambda\,\Pi_t(i_1) \;-\; m_{2,3}\,\lambda\,\Pi_t(i_2) \\ &= \Pi_t(i_2)\,(1 - m_{2,3}\,\lambda) \;+\; m_{1,2}\,\lambda\,\Pi_t(i_1) \end{aligned}$$

qui est la version discrète de l'équation différentielle :

$$\frac{\mathrm{d}\,\Pi_t(i_2)}{\mathrm{d}t} \;=\; -\ln(1 - \lambda\, m_{1,2})\,\Pi_t(i_1) + \ln(1 - \lambda\, m_{2,3})\,\Pi_t(i_2)$$

pour laquelle la solution est connue :

$$\begin{aligned} \Pi_t(i_2) \;&= \Pi_0(i_2)\,(1 - \lambda\, m_{2,3})^t \\ &\quad + \Pi_0(i_1)\,\frac{-\ln(1 - m_{1,2}\,\lambda)\big((1 - m_{1,2}\,\lambda)^t - (1 - m_{2,3}\,\lambda)^t\big)}{-\ln(1 - m_{2,3}\,\lambda) + \ln(1 - m_{1,2}\,\lambda)} \end{aligned}$$

Ici $\Pi_0(i_2) = 0$, on a donc finalement $\forall t \in \{0, \dots, T\}$:

$$\begin{cases} \mathbb{E}[\Pi_t(i_1)] \;=\; \Pi_0(i_1)\,(1 - m_{1,2}\lambda)^t \;=\; (1 - m_{1,2}\lambda)^t \\ \mathbb{E}[\Pi_t(i_2)] \;=\; \frac{-\ln(1 - m_{1,2}\,\lambda)\left((1 - m_{1,2}\,\lambda)^t - (1 - m_{2,3}\,\lambda)^t\right)}{-\ln(1 - m_{2,3}\,\lambda) + \ln(1 - m_{1,2}\,\lambda)} \\ \mathbb{E}[\Pi_t(i_3)] = 1 - \mathbb{E}[\Pi_t(i_1)] - \mathbb{E}[\Pi_t(i_2)] \end{cases} \tag{5}$$

À chaque instant t ; le coach estime la matrice $\mathbf{M}$ à partir de l'observation des interactions avec U. Soit cette estimation valant par exemple :

$$\widehat{\mathbf{M}} = \begin{pmatrix} 1 & 0.15 & 0.1 \\ 0 & 1 & 0.2 \\ 0 & 0 & 1 \end{pmatrix}$$

On peut alors comparer les choix possibles pour le coach : $c_t(i_1) = i_3$ (Eq. (4)) et $c_t(i_1) = i_2$ (Eq. (5)), $\forall t$. Il est apparent que les expressions (4) et (5) ont leurs valeurs qui dépendent du facteur d'apprentissage λ de U et du nombre de pas d'interactions t.
Selon la valeur T du nombre maximal d'interactions, il est possible que le choix optimal $c_t(i_1)$ dépende de λ. Ici par exemple, il existe une valeur critique λ_c du paramètre λ telle que si $\lambda > \lambda_c$ (resp. $\lambda \leq \lambda_c$) il faut choisir $c_t(i_1) = i_2$ (resp. $c_t(i_1) = i_3$) pour optimiser la performance $\overline{V}_T$.

T	λ_c
10	1
20	≈ 0.6028
50	≈ 0.2615
100	≈ 0.1346
1000	≈ 0.0138

TABLE 1 – Valeurs de λ_c en fonction de T.

On observe (voir la table 1) que plus le nombre d'itérations T est important, plus la valeur de λ_c diminue, c'est -à-dire que l'apprentissage par U rend le chemin indirect $i_1 \to i_2 \to i_3$ plus intéressant que le chemin direct $i_1 \to i_3$. Dans cet exemple, pour un T donné, on voit que l'estimation des valeurs de λ et de $\mathbf{M}$, si elle est bien faite, permet au coach de déterminer le meilleur choix de recommandation $c_t(\cdot)$.
Il nous faut maintenant examiner comment le coach peut estimer ces valeurs, ou bien directement le meilleur choix $c_t(i)$ lorsque l'utilisateur propose l'item i à l'instant t.

5 Évaluation expérimentale

Dans cette section, nous comparons expérimentalement des stratégies classiques de la littérature pour répondre au dilemme exploitation vs. exploration du coach, notamment inspirées de l'apprentissage par renforcement et des problèmes de *bandit multi-bras*.

5.1 Présentation des stratégies testées

Quatre stratégies classiques et une variante ont été testées dans nos expériences et comparées également à une stratégie de choix aléatoire de substitution qui sert de référence.

1. La stratégie *gloutonne* recommande la substitution qui maximise l'espérance de gain instantanée de score :

$$c_t(i) \;=\; \underset{j\in\mathcal{I}}{\text{ArgMax}}\big\{m_{i,j}\,\big(\text{score}(j)-\text{score}(i)\big)\big\}$$

En fonction du comportement observé chez l'utilisateur (ie. refus ou acceptation), le coach met à jour ses estimations des probabilités d'acceptation $m_{i,j}$ de U selon :

$$\widehat{m}_{i,j}^{t+1} = \begin{cases} \frac{\widehat{m}_{i,j}^{t}+1}{n_{i,j}} & \text{si la propositon } i\to j \text{ est acceptée} \\ \frac{\widehat{m}_{i,j}^{t}}{n_{i,j}} & \text{sinon} \end{cases}$$

avec $\widehat{m}_{i,j}^{t}$ l'estimation de la valeur de $m_{i,j}$, et $n_{i,j}$ le nombre de fois ou la recommandation $i \to j$ a été proposée à l'utilisateur.

2. La stratégie *UCB* (Upper Confidence Bound) [2] se base sur un calcul de la récompense moyenne empirique, à laquelle est ajouté un terme dépendant du nombre de fois où une option a été testée, de sorte que moins une option a été testée, plus elle est favorisée. À chaque itération t, i étant la proposition de U, le coach choisit :

$$c_t(i) \;=\; \underset{j\in\mathcal{I}}{\text{ArgMax}}\left\{\mu_{i,j} + C\sqrt{\frac{\ln(N_i)}{n_{i,j}}}\right\}$$

où $\mu_{i,j}$ est la moyenne du gain score(j) - score(i) obtenu jusqu'à l'instant t lorsque la substitution $i \to j$ a été suggérée, C est une constante, N_i le nombre de fois où l'item i a été proposé par U, et $n_{i,j}$ le nombre de fois où la substitution $i \to j$ a été proposée par C.

3. La stratégie dite d'*échantillonnage de Thomson* cherche à estimer les paramètres $m_{i,j}$ de la matrice $\mathbf{M}$ via une distribution de probabilité. Nous avons retenu ici la loi bêta. Ainsi chaque coefficient estimé $\widehat{m}_{i,j}^{t}$ est associé à une distribution $\mathbb{P}_{i,j}^{t} = Beta(\alpha_t, \beta_t)$ qui est mise à jour à chaque fois que la substitution $i \to j$ est proposée. Le choix de C s'opère selon :

$$c_t(i) \;=\; \underset{j\in\mathcal{I}}{\text{ArgMax}}\left\{\widehat{m}_{i,j}^{t}\,\big(\text{score}(j)-\text{score}(i)\big)\right\}$$

Les paramètres α et β de $\mathbb{P}_{i,j}$ sont ensuite mis à jour en fonction du comportement utilisateur :

$$\begin{cases} \alpha_{t+1} = \alpha_t + Accept_t \\ \beta_{t+1} = \beta_t + (1 - Accept_t) \end{cases}$$

avec $Accept_t$ = 1 si la substitution est acceptée et 0 sinon. Et :

$$\widehat{m}_{i,j}^{t+1} = \mathbb{E}[Beta(\alpha_{t+1}, \beta_{t+1}]$$

Cette stratégie permet d'avoir une estimation plus performante de la valeur de $m_{i,j}$, particulièrement pour un faible nombre d'itérations.

4. L'algorithme du *Q-learning* [10] cherche directement à estimer le gain $Q[i,j]$ à attendre quand $c_t(i) = j$. À chaque fois que la substitution $i \to j$ est suggérée, $Q[i,j]$ est mise à jour selon :

$$Q[i,j] := (1-\alpha)\,Q[i,j] + \alpha\big(g_{i,j} + \gamma \underset{j'\in\mathbf{I}}{\text{ArgMax}}\, Q[i',j']\big)$$

où α est un taux d'apprentissage, $g_{i,j}$ est le gain observé (éventuellement nul si j est refusé), γ est le facteur d'atténuation de prise en compte des gains dans le futur, $i' = j$ si la substitution $i \to j$ a été acceptée par U, et $i' = i$ sinon.
Les valeurs classiques pour le taux d'apprentissage α, de 0.1, et le facteur d'actualisation $\gamma = 0.9$ sont employées dans nos expériences.
Nous avons également introduit une variante du Q-learning, appelée λ-*Q-learning*, dans laquelle le taux d'actualisation d'apprentissage γ est proportionnel au taux d'apprentissage λ (supposé connu) de l'utilisateur. Cette variante est motivée par l'hypothèse que lorsque U apprend vite, son vecteur de préférence associé Π_t varie plus rapidement, et il est alors intéressant pour le coach de regarder loin dans le futur, alors que si $\lambda = 0$, cela n'a aucune utilité.

5.2 Le scénario de test

Afin de tester les capacités des différentes stratégies à conduire à une performance maximale, nous avons défini un scénario simple dans lequel U choisit i_0 à l'étape $t = 0$ et trois recommandations différentes $c_t(i_0)$ doivent être comparées (voir figure 2).
• Le choix de l'item i_4 par U est associé au score maximal : score(i_4) = 70, mais pour atteindre i_4 il faut passer par le

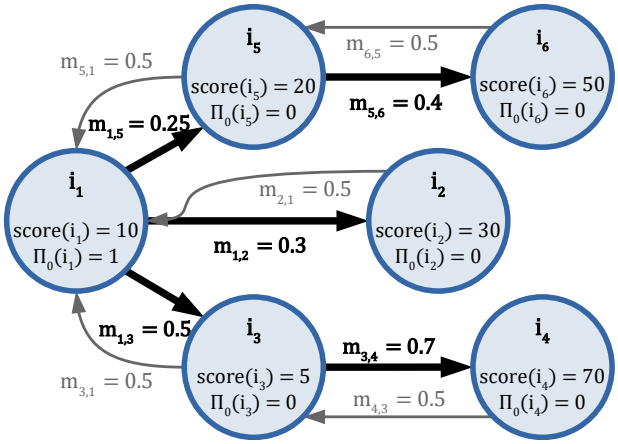


FIGURE 2 – Graphe présentant le scénario étudié dans le cadre des expériences au pas de temps initial $t = 0$.

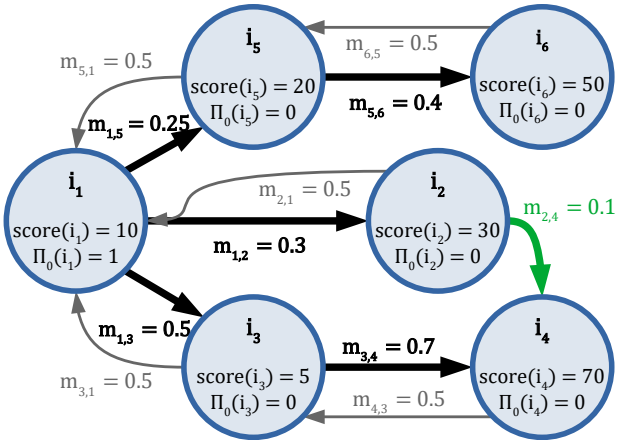


FIGURE 3 – Graphe présentant le scénario modifié au pas de temps initial $t = 0$.

chemin $i_1 \rightarrow i_3 \rightarrow i_4$ avec un score de i_3 faible : score(i_3) = 5 < score(i_1) = 10.
• Le chemin $i_1 \rightarrow i_5 \rightarrow i_6$ conduit à i_6 de score = 50 < score(i_4), en passant par i_5 de score = 20 > score(i_1).
• Finalement, le chemin le plus court $i_1 \rightarrow i_2$ est associé au gain immédiat le plus élevé : score(i_2) = 30.

5.3 Évaluation

Pour l'évaluation des résultats de simulation, nous étudions le gain de performance moyen $\overline{V}_T$ après T itérations (section 2.4). Les tests sont menés sur 200 utilisateurs simulés, et avec T = 1000 interactions avec le coach. Tous les utilisateurs simulés partagent les mêmes paramètres M, Π_0 et λ. Cependant, parce que l'acceptation ou le refus des propositions du coach est stochastique, les trajectoires dans l'espaces des vecteurs de préférence sont variables.

5.4 Résultats

5.4.1 Effet du paramètre λ

Nous testons ici l'effet du coefficient d'apprentissage λ caractérisant l'utilisateur. La table 2 donne les valeurs de $\overline{V}_T$ avec $T = 1000$ pour des valeurs de $\gamma \in \{0.005, 0.01, 0.05, 0.2, 0.4, 0.7, 1.0\}$. Les valeurs rapportées résultent de 200 simulations sur des utilisateurs U dont les caractéristiques sont données dans la figure 2.
Globalement, on observe que la stratégie du Q-learning est celle qui se comporte le mieux pour un large intervalle de valeurs de λ, approximativement dans [0.1, 1], et ce d'autant plus que l'écart-type observé est très contrôlé par rapport aux autres stratégies en compétition. En revanche, pour les valeurs de λ faibles, dans $[0, 0.1]$, les méthodes qui mettent à jour explicitement les coefficients de la matrices M l'emportent sur le Q-learning.
Deux leçons peuvent être tirées. D'une part, le choix de la meilleure stratégie dépend de la propension de U à apprendre, contrôlée par le paramètre λ. D'autre part, heureusement, cette dépendance est cependant limitée. Il suffit de savoir dans quelle grande gamme de valeurs λ s'inscrit pour savoir quelle stratégie favoriser.

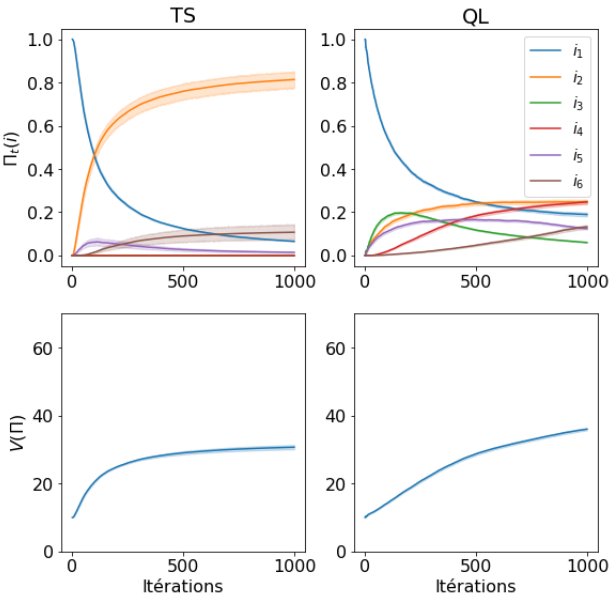


FIGURE 4 – Évolution des probabilités de choix des items et de l'espérance associée dans le cas $\lambda = 0.05$ pour les méthodes Q-Learning (à droite) et Échantillonnage de Thompson (à gauche). La courbe du haut présente la probabilité finale de choix de chaque item en fonction de T. La courbe du bas présente la valeur de $\overline{V}_T$ en fonction de T.

Pour mieux comprendre la différence de comportement entre les méthodes qui considèrent explicitement la matrice M et celles qui ne le font pas, comme le Q-learning qui est une méthode « model-free », il est intéressant de considérer la manière dont elles explorent l'espace des préférences au cours des interactions. La figure 4 comparant le Q-learning et la méthode de Thomson montre immédiatement que le Q-learning explore bien davantage l'espace des préférences, en maintenant les probabilités de choix des items à un niveau assez élevé, tandis que la méthode de Thomson converge bien plus rapidement. Cependant, le Q-

λ		Aléatoire	Gloutonne	Thomson	UCB	Q-learning	λ Q-learning
0.005	μ	13.4	19.8	**21.8**	**21.8**	10.4	18.7
	σ	0.39	2.45	0.43	0.34	0.16	0.83
0.01	μ	16.6	**26.7**	25.0	24.7	10.9	21.9
	σ	0.62	3.24	0.71	0.36	0.29	0.90
0.05	μ	28.1	30.2	29.6	29.4	21.0	**36.9**
	σ	1.88	6.34	2.98	0.68	2.53	2.29
0.2	μ	31.8	32.5	31.7	36.1	39.0	**48.6**
	σ	3.98	10.76	5.28	3.98	5.25	3.69
0.4	μ	31.1	30.1	33.9	39.7	47.9	**53.4**
	σ	4.76	14.13	6.97	6.23	6.74	4.76
0.7	μ	31.7	33.4	34.0	43.1	54.3	**55.9**
	σ	11.47	19.56	7.69	10.98	9.55	8.62
1.0	μ	34.1	35.5	34.0	42.4	**66.3**	64.7
	σ	26.74	30.54	8.02	16.72	11.53	12.44

TABLE 2 – Table des moyennes μ et écart-types σ de $\overline{V}_T$ pour $T = 1000$ calculée à partir de 200 simulations avec un utilisateur caractérisé par les paramètres fournis dans la figure 2.

λ		Aléatoire	Gloutonne	Thomson	UCB	Q-learning	λ Q-learning
0.005	μ	13.5	19.9	**24.4**	24.2	10.4	19.0
	σ	0.40	2.25	1.10	0.73	0.15	0.99
0.01	μ	16.6	26.6	**33.0**	31.8	11.0	22.8
	σ	0.66	3.30	1.61	1.40	0.33	1.18
0.05	μ	28.8	31.3	**55.8**	54.5	21.9	39.5
	σ	1.96	6.08	3.96	2.21	2.65	2.33
0.2	μ	32.1	33.0	**64.1**	63.5	41.6	52.0
	σ	4.04	9.94	5.87	2.93	5.17	3.40
0.4	μ	31.2	33.1	64.7	**65.1**	50.1	56.6
	σ	6.21	14.85	7.02	3.89	6.54	3.98
0.7	μ	32.7	31.6	64.9	**65.9**	57.2	60.7
	σ	11.44	19.22	7.88	5.02	8.79	5.49
1.0	μ	31.1	34.0	64.4	66.3	**67.9**	67.8
	σ	26.45	30.06	9.00	7.79	8.12	7.17

TABLE 3 – Table des moyennes μ et écart-types σ de $\overline{V}_T$ pour $T = 1000$ calculée à partir de 200 simulations avec un utilisateur caractérisé par les paramètres fournis dans la figure 3.

learning atteint des niveaux de performance $\overline{V}_T$ plus élevés pour un nombre d'interactions assez grand.

On notera que quand $\lambda = 1$, caractérisant un utilisateur qui adopte instantanément les suggestions du coach, la méthode du Q-learning est la seule qui permette d'approcher la performance $\overline{V}_T$ maximale qui est ici de 70.

La section suivante explore les différences entre stratégies en fonction des caractéristiques de la matrice de substituabilité qui contrôle les trajectoires possibles dans l'espace des vecteurs de préférence.

5.4.2 Effet de la matrice de substituabilité $\mathbf{M}$

Le contexte décrit par la figure 2 correspond à un cas difficile, puisque pour atteindre le choix de l'item i_4 associé au score le plus élevé, il faut d'abord passer par le choix de l'item i_3 dont le score est moins grand que le score de l'item i_1 choisi au départ par U. Le gradient de score n'informe donc pas sur la valeur potentielle de chaque item.

Ce gradient résulte à la fois des scores associés à chaque item et de la matrice $\mathbf{M}$ qui contrôle les chemins possibles dans l'espace des préférences. Dans l'exemple de la figure 3, $m_{2,4}$ a une valeur positive, ici 0.1, ce qui permet d'atteindre le choix de i_4 en passant par le choix de i_2. Le chemin $i_1 \to i_2 \to i_4$ est associé à des gradients tous positifs, et cela devrait permettre aux méthodes reposant sur cette information de suggérer les substitutions correspondantes et de guider l'utilisateur vers la préférence pour i_4.

La table 3 montre que la stratégie du Q-learning, qui ne repose pas directement sur l'évaluation du gradient de score sur les substitutions possibles, est peu sensible à cette modification de $\mathbf{M}$. En revanche, les méthodes telles que Thomson et UCB en tirent pleinement parti et permettent d'obtenir des performances $\overline{V}_{1000}$ plus élevées, notablement pour des valeurs de λ faibles, ici dans $[0, 0.2]$.

Il est présomptueux de tirer des leçons générales d'exemples particuliers. Cependant on peut conjecturer que la méthode du Q-learning est à favoriser quand la matrice $\mathbf{M}$ est clairsemée, avec de nombreuses valeurs nulles et, gé-

néralement, quand la fonction de score sur les items combinée aux substitutions possibles détermine des gradients de score négatifs sur les chemins qui conduisent aux performances maximales. À contrario, les méthodes tirant parti directement de l'information de gradient seront à favoriser quand la matrice $\mathbf{M}$ est non clairsemée et quand les gradients sont informatifs.
La meilleure stratégie à adopter pour le problème de coaching dépend bien à la fois de la fonction de score définie sur les items, de la matrice $\mathbf{M}$ et de la valeur de λ.

6 Vers un cadre plus général

Le concept de coaching décrit dans ce papier est limité car il suppose qu'à chaque instant l'utilisateur ne choisit qu'un item et que la fonction de score est définie sur chaque item indépendamment des autres. Cependant, dans de nombreux contextes, l'utilisateur doit choisir une combinaison d'items, par exemple les plats constituant un repas, et le score n'est pas additif mais prend en compte les interactions entre les items choisis, voire l'historique des choix, par exemple ce qui a été consommé sur une semaine.
Dans ce cas, il est plus difficile de définir une distribution de probabilité sur l'espace des préférences et la fonction de score n'est plus directement associée à chaque item. La définition de stratégies adaptées à ce cadre plus général est un objectif de nos travaux en cours.
Par ailleurs, il faut noter que l'estimation des caractéristiques des utilisateurs, ici discutée avec un seul utilisateur, peut bénéficier d'un apprentissage sur un ensemble d'utilisateurs, en tirant profit de la ressemblance mesurée entre eux, à l'instar de techniques de recommandation collaborative. Cela fait partie des développements futurs envisagés.

7 Conclusion

Cet article a présenté un nouveau scénario de recommandation dans lequel l'objectif est de modifier durablement les préférences de l'utilisateur à partir de l'observation répétée de ses choix. Au lieu de faire des recommandations à l'utilisateur, l'agent s'appuie sur les choix exprimés par celui-ci pour lui suggérer des modifications possibles. Nous appelons *coaching* ce processus itéré et personnalisé de recommandation. Nous avons proposé une formalisation de ce scénario comme un jeu itéré à deux joueurs. Nous avons défini plusieurs critères permettant d'évaluer la performance du coaching et montré qu'il n'y a pas de stratégie de coaching optimale pour tous les utilisateurs, mais qu'il fallait tenir compte des caractéristiques de ceux-ci pour optimiser la stratégie de coaching.
Les expériences réalisées sur des scénarios simples mais représentatifs des problèmes possibles ont permis de comparer plusieurs approches inspirées de la littérature sur l'optimisation du compromis exploration vs. exploitation et de tirer des leçons sur le meilleur choix en fonction des caractéristiques des problèmes.
Le coaching correspond à un large spectre de problèmes de recommandation pour lequel ce travail représente un premier pas.

Remerciements

Nous remercions Cécile Caumette et Hugo Vaysset pour leur contribution à l'état de l'art et à la conception de certaines expériences.

Références

[1] Robin Allesiardo. *Bandits Manchots sur Flux de Données Non Stationnaires*. PhD thesis, Université Paris-Saclay, 2016.

[2] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2) :235–256, 2002.

[3] Allison JB Chaney, Brandon M Stewart, and Barbara E Engelhardt. How algorithmic confounding in recommendation systems increases homogeneity and decreases utility. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 224–232, 2018.

[4] Robert G Farrell, Catalina M Danis, Sreeram Ramakrishnan, and Wendy A Kellogg. Intrapersonal retrospective recommendation : lifestyle change recommendations using stable patterns of personal behavior. In *Proceedings of the First International Workshop on Recommendation Technologies for Lifestyle Change (LIFESTYLE 2012), Dublin, Ireland*, page 24. Citeseer, 2012.

[5] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[6] Yu Liang. Recommender system for developing new preferences and goals. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 611–615, 2019.

[7] Dorian Peters, Rafael A Calvo, and Richard M Ryan. Designing for motivation, engagement and wellbeing in digital experience. *Frontiers in psychology*, 9 :797, 2018.

[8] Mustafa Radha, Martijn C Willemsen, Mark Boerhof, and Wijnand A IJsselsteijn. Lifestyle recommendations for hypertension through rasch-based feasibility modeling. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*, pages 239–247, 2016.

[9] Hanna Schäfer and Martijn C Willemsen. Rasch-based tailored goals for nutrition assistance systems. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, pages 18–29, 2019.

[10] Richard S Sutton and Andrew G Barto. *Reinforcement learning : An introduction*. MIT press, 2018.

Bandits-Manchots Combinatoires: du retour utilisateur à la recommandation

A. Letard[1, 2], T. Amghar[2], O. Camp[3], N. Gutowski[2]

[1] Kara Technology - Dpt R&D, F-49124 St Barthélémy dÁnjou, France
[2] Univ Angers, LERIA, SFR MATHSTIC, F-49000 Angers, France
[3] Groupe ESEO - ERIS, F-49000 Angers, France

alexandre.letard@kara.technology

Résumé

*Récemment, le problème des Bandits-Manchots COMbinatoires (*COM-MAB*) a été sujet de nombreux travaux de recherche. Au sein de systèmes en interaction avec des humains, ces techniques, basées sur un apprentissage par renforcement, exploitent une stratégie de considération du retour utilisateur en guise de fonction de récompense. Dans l'étude de ces stratégies, cet article présente les contributions suivantes : 1) Nous proposons un modèle général de stratégie en trois étapes : Feedback Identification, Feedback Retrieval et Reward Computing, chacune influant sur les performances d'un agent ; 2) Suivant ce modèle, nous proposons une nouvelle méthode de Reward Computing,* BUSBC*, améliorant significativement la précision globale des algorithmes optimistes ; 3) Nous réalisons une analyse empirique de notre approche et d'autres stratégies issues de la littérature. Nos expérimentations, réalisées sur trois jeux de données issus d'applications réelles, confirment nos propositions avec des retours utilisateurs complets ou partiels.*

Mots-clés

Bandits-Manchots Combinatoires, Systèmes de Recommandations, Apprentissage par Renforcement

Abstract

*Recently, the COMbinatorial Multi-Armed Bandits (*COM-MAB*) problem has arisen as an active research field. In systems interacting with humans, those reinforcement learning approaches use a feedback strategy as their reward function. On the study of those strategies, this paper present three contributions : 1) We model a feedback strategy as a three-step process, namely : Feedback Identification, Feedback Retrieval and Reward Computing, where each step influences the performances of an agent. 2) Based on this model, we propose a novel Reward Computing process,* BUSBC*, which significantly increases the global accuracy reached by optimistic* COM-MAB *algorithms ; 3) We conduct an empirical analysis on our approach and several feedback strategies from the literature. Our experiments, conducted on three real-world datasets, confirm our propositions with significant results whether full or partial feedback vector are used.*

Keywords

Combinatorial Multi-Armed Bandits, Recommender Systems, Reinforcement Learning

1 Introduction

Les techniques de Bandits-Manchots (*MAB*) [19] sont aujourd'hui employées dans de nombreux secteurs d'activités tels que la finance, le domaine médical ou les systèmes de recommandation [7]. Ces approches procurent de bons résultats en termes de précision globale lorsqu'un compromis entre exploitation et exploration doit être réalisé. Cependant, dans certaines applications, un agent doit être capable de recommander plusieurs éléments à chaque itération [8, 14]. Les Bandits-Manchots COMbinatoires (*COM-MAB*) [2, 8], spécialement conçus pour ces situations, constituent alors un choix judicieux. Les approches *MAB* et *COM-MAB* sont des techniques basées sur un apprentissage par renforcement : un agent réalise une action, observe une récompense et change son état [22]. Dans le cadre des systèmes de recommandations, une action correspond à une recommandation et une récompense est déterminée à partir du retour utilisateur émis suite à cette recommandation [10]. Ainsi, plusieurs stratégies de prise en compte du retour utilisateur – désignées sous le terme de "*stratégies*" dans la suite de cet article – ont été définies pour les approches *COM-MAB* [3]. Dans un cadre applicatif réel, ces stratégies peuvent être employées avec différents types de retours utilisateur : 1) Des *vecteurs complets*, lorsqu'un retour utilisateur est acquis pour chaque élément composant la recommandation [9] ; 2) Des *vecteurs partiels* [20, 15], lorsque l'utilisateur émets des retours pour une partie des éléments de la recommandation ; 3) Des vecteurs implicitement déduits des interactions entre l'usager et le système de recommandations [12, 16], tels que lors de l'emploi de modèles dits de *Bandits en Cascades*.
Pour les applications en directe interaction avec les usagers, ces sujets présentent un intérêt certain. Le jeu de donnée RSASM [1], issu d'une application réelle, en est un exemple représentatif où l'objectif est de recommander à des utilisateurs des activités à réaliser, sans connaissances initiales

1. *Recommendation System for Angers Smart City*

quant-à leurs préférences. Ce scénario correspond à un problème *MAB* ou *COM-MAB*, où les bras sont les activités pouvant être proposées et où les retours utilisateurs expriment l'opinion des usagers quant-aux activités qui leur sont recommandées. Dans de telles applications, plus le nombre d'éléments recommandés à chaque itération est important, plus il est difficile de demander aux usagers un retour pour chacun des bras de la recommandation.

En l'absence d'historique d'interactions entre le système et les utilisateurs, une stratégie adaptée est essentielle à l'apprentissage efficace d'un algorithme *COM-MAB*. Cependant, à notre connaissance, les principales approches aujourd'hui considérées sont *Bandit* [11], *Semi-Bandit* [9], ou des variantes de la stratégie *Semi-Bandit* employant des vecteurs de retours utilisateur partiels portant sur $\psi < k$ bras parmi les k éléments constituant la recommandation [20]. Des études plus approfondies sont nécessaires pour déterminer comment améliorer les approches *COM-MAB* avec ces stratégies d'apprentissage.

Ainsi, inspiré par de précédents travaux de la littérature [15, 3], nous relevons que toute stratégie peut être définie comme la succession de trois processus : a) "*Feedback Identification*" ; b) "*Feedback Retrieval*" ; c) "*Reward Computing*". Nous affirmons que chacun de ces processus influe sur les performances de l'algorithme *COM-MAB* utilisé et peut donc permettre un gain de précision globale par des ajustements adaptés. Afin de confirmer cette hypothèse, nous proposons une nouvelle méthode pour le processus "Reward Computing", "*Bandit Under Semi-Bandit Conditions*" (*BUSBC*). Notre approche est basée sur la combinaison de méthodes connues et vise à améliorer les performances des algorithmes *COM-MAB* de type *UCB*.

Nous avons réalisé des expérimentations avec plusieurs algorithmes *COM-MAB* sur trois jeux de données issus d'applications réelles. Nous avons évalué l'impact sur la précision globale de plusieurs approches pour chacun des processus identifié dans notre modèle général, avec des vecteurs de retours utilisateur complets et partiels. Nos résultats confirment nos hypothèses et montrent que *BUSBC* améliore significativement les performances des algorithmes de type *UCB*. Nous relevons également, parmi celles étudiées, des stratégies optimales pour chacun des algorithmes *COM-MAB* considérés.

Cet article est organisé comme suit. La section 2 présente les notions relatives au problème *COM-MAB* et aux stratégies de considération du retour utilisateur. La section 3 définit notre modèle général ainsi que notre approche *BUSBC*. La section 4 expose nos résultats expérimentaux. Enfin, nous concluons et ouvrons de nouvelles perspectives de recherches dans la section 5.

2 Préliminaires

2.1 Bandits-Manchots Combinatoires

Un problème *MAB* [19] implique un ensemble $\mathcal{A} = \{a_1, ..., a_m\}$ de *m* bras indépendants, où chaque bras $a \in \mathcal{A}$ est un élément à recommander. Au sein d'un système de recommandation, à chaque itération $t \in [1, T]$, *T* étant un horizon connu, un agent sélectionne un bras $a_t \in \mathcal{A}$ selon sa politique π et le recommande à l'utilisateur. Dans cet article, nous considérons le problème *COM-MAB* [2], consistant en une généralisation du problème *MAB* où l'agent doit recommander un ensemble de bras $A_k = \{a_1, ..., a_k\}$, $A_k \subseteq \mathcal{A}$, avec $1 \leq k \leq m$, $\forall t \in [1, T]$. Parmi les approches existantes, nous considérons la méthode "*Multiple Plays*" [2] qui permet l'emploi séquentiel d'un algorithm *MAB* pour définir incrémentalement un "*Super-Bras*" [8] comme suit : Tant que $|S_t| < k$, $S_t = \cup_{i=1}^{k}\{a_i\}$ où $a_i = argmax^{\pi}_{a \in \mathcal{A} \setminus S_t} \mathbb{E}[R_{t,a}]$. Le Super-Bras S_t est donc le sous-ensemble de k bras de plus haute espérance de récompense selon la politique π de l'algorithme employé. Ainsi, tout algorithme *MAB* de la littérature peut être employé dans un cadre combinatoire, qu'il soit stochastique [4], non-stochastique [6], Bayésien [1] ou avec adversaire [5].

Dans un cadre stochastique, où les récompenses sont considérées comme des variables aléatoires indépendantes et identiquement distribuées, un algorithme *COM-MAB* de politique π vise à minimiser le regret cumulé $\rho^{\pi}(T) = T\mu^* - \sum_{t=1}^{T} r_t$, où μ^* correspond à l'espérance de récompense du super-bras optimal, sans connaissances préalable quant à la distribution des espérances de récompenses $\mu_a \in [0, 1]$ parmi les bras a de $\mathcal{A}$. Dans de nombreuses applications réelles, il est préféré de considérer la maximisation de la précision globale $Acc^{\pi}(T) = \frac{\sum_{t=1}^{T} r_t}{T}$. Cette métrique est fréquemment employé pour évaluer les approches de Bandits-Manchots [10]. Dans cet article, nous considérons d'une part une "*récompense d'évaluation*" $r_t \in \{0, 1\}$ pour la recommandation S_t, inconnue de l'agent et telle que $r_t = 1$ si au moins la moitié des éléments recommandés dans S_t donnent satisfaction à l'utilisateur, et $r_t = 0$ sinon (voir sous-section 4.1.3). D'autre part, à chaque itération, l'agent *COM-MAB* observe une récompense R_t, calculée selon la stratégie appliquée. Cette récompense peut être soit un scalaire, soit un vecteur $R_t = \{R_{t,1}, R_{t,2}, ..., R_{t,\psi}\}$, avec ψ désignant le nombre de bras pour lesquels un retour utilisateur a été émis. Cette récompense est ensuite employée pour actualiser la connaissance de la distribution des espérances de récompenses des bras. Pour les algorithmes considérés, cette actualisation est réalisée au travers de la sommation des récompenses perçues jusqu'à l'itération courante, pour chacun des bras a de $\mathcal{A}$: $SR_{t,a} = SR_{t-1,a} + R_{t,a}$.

2.2 La considération du retour utilisateur

L es algorithmes *COM-MAB*, basés sur de l'apprentissage par renforcement, apprennent à l'aide des récompenses observées à chaque itération. La fonction de récompense de tels algorithmes est donc essentielle à leur fonctionnement. Une stratégie de prise en compte du retour utilisateur est une fonction de récompense particulière employée par des agents en interaction avec des humains, tels qu'au sein de systèmes de recommandations. Plusieurs stratégies ont donc été proposées pour les algorithmes *COM-MAB* [3]. Ainsi, soit $Y_t = \{Y_{t,1}, Y_{t,2}, ..., Y_{t,m}\}$, le vecteur de retours utilisateur associant un retour spécifique à chacun des bras *a* de $\mathcal{A}$ à l'itération *t*. Au sein d'applications réelles, puisqu'aucun retour de l'utilisateur ne peut être acquis pour les

bras non-recommandés, Y_t est un concept abstrait. Ainsi, soit $F_t \subseteq Y_t$ le vecteur de retours utilisateur réellement perçu par le système.
A notre connaissance, la plupart des approches traditionnelles sont des variantes des modèles suivants : a) *Full-Information* [3], où une récompense individuelle est observée pour tous les bras *a* de $\mathcal{A}$, qu'ils soient ou non [2] inclus dans S_t : $R_t^{FI} = F_t = Y_t$; b) *Semi-Bandit* [9], où une récompense individuelle, associée à chaque bras *a* de S_t, est révélée : $R_t^{SB} = F_t = \cup_a S_{t,a} Y_{t,a}$; c) *Bandit* [11], où seulement une récompense cumulée [3] associée à S_t est perçue par l'agent : $R_t^B = S_t^\top R_t^{SB}$; d) Les modèles de *Bandits en Cascade* [16], dépendants de l'application, et visant à déduire implicitement F_t en considérant un critère d'arrêt, p.ex., un clic de l'utilisateur sur le premier élément le satisfaisant dans S_t. Dans la littérature [3, 18], *Semi-Bandit* et *Bandit* désignent à la fois un niveau d'exhaustivité des retours utilisateur perçus et une méthode pour déterminer les récompenses observées par l'algorithme. Dans cet article, nous évoquerons uniquement ces approches en tant que procédés de calcul des récompenses et désignerons les contraintes liées à l'acquisition de retours utilisateur par les termes de *vecteurs complets* ou *partiels*.
Il a été démontré que, parmi ces méthodes, *Semi-Bandit* est plus efficace dans de nombreux cas [3]. Cette approche a donc été particulièrement étudiée par la littérature [21]. Cependant, dans certaines applications, S_t peut regrouper un grand nombre d'éléments [13, 14, 15]. Pour éviter à l'usager de fournir un aussi grand nombre de retours, des variantes considérant des vecteurs de retours utilisateur partiels ont été proposées [15, 17]. Dans ce cadre partiel, le vecteur de retours utilisateur n'est défini que pour un sous-ensemble $P_t \subseteq S_t$. Ainsi, P_t est un vecteur de bras pour lesquels un retour est demandé à l'utilisateur, tandis que F_t est le vecteur de retours fournit en réponse par l'utilisateur u_t, les deux vecteurs étant de même dimension ψ. Dans cet article, l'objectif d'un agent est de recommander les k meilleurs bras à chaque itération. Lors de l'emploi de vecteurs complets, cet objectif est suivi en percevant à chaque itération k retours utilisateur tandis que seulement $\psi < k$ retours sont prodigués lors de l'emploi de vecteurs partiels.

3 Modélisation

3.1 Stratégie de considération du retour utilisateur : un modèle général

Soit $\mathcal{U} = \{u_1, ..., u_n\}$ un ensemble de *n* utilisateurs. À chaque itération $t \in [1, T]$, un utilisateur u_t sollicite une recommandation S_t de *k* éléments extraits de $\mathcal{A} = \{a_1, ..., a_m\}$, où $\mathcal{A}$ est l'ensemble des éléments connus par un système de recommandation employant un algorithme *COM-MAB* de politique π. Nous affirmons que chacune des stratégies précédemment exposées s'ancre dans un modèle générique composé de trois processus successifs :

Feedback Identification **:** Un ensemble $P_t \subseteq \mathcal{A}$, pour lequel des retours utilisateurs seront attendus, est déterminé. Pour les modèles de *Bandits en Cascade*, cette étape définit le critère d'arrêt à relever dans l'interaction de l'utilisateur avec le système, p.ex., la sélection d'un élément préféré associé à un bras a_s de S_t. Lors de la considération d'un vecteur partiel de retours utilisateur, cette étape définit la méthode pour construire P_t à partir de S_t. Les autres approches considèrent soit $P_t = \mathcal{A}$ ou $P_t = S_t$.

Feedback Retrieval **:** Un vecteur de retours utilisateur initial F_t est construit. Pour les *Bandits en Cascades*, cette étape est employée pour déduire implicitement F_t en considérant les distances relatives de chaque bras a de S_t par rapport au bras a_s ayant déclenché le critère d'arrêt précédemment défini. Les autres méthodes présentées à la sous-section 2.2 sollicitent explicitement un retour de l'utilisateur pour chacun des bras constituant P_t.

Reward Computing **:** Une récompense finale R_t, observée par l'agent, est calculée à partir de F_t. Ainsi, R_t peut correspondre au vecteur de retours utilisateur F_t lui-même, ou au résultat de n'importe quel traitement appliqué sur F_t, p.ex., une pondération ou un produit scalaire entre F_t et P_t.

De nombreux travaux ont montré l'intérêt du processus *Feedback Retrieval* avec des variantes de *Bandits en Cascades* [16, 12]. Dans cet article, notre objectif est d'étudier l'impact sur la précision globale des processus de *Feedback Identification* et *Reward Computing*.

3.2 Bandit under Semi-Bandit Conditions : BUSBC

Suivant ce modèle général, nous avons implémenté un nouveau processus de *Reward Computing*, *BUSBC*. Cette méthode est basée sur la combinaison du principe de récompense cumulée de l'approche *Bandit* et des conditions d'octroi de récompenses individuelles de l'approche *Semi-Bandit*. L'objectif de cette démarche est d'améliorer les performances des algorithmes *COM-MAB* de type *UCB* en canalisant l'usage de la récompense cumulée.
Comme pour toute stratégie, les processus *Feedback Identification* et *Feedback Retrieval* doivent être appliqués avant d'employer notre méthode. Dans cet article, nous considérons que le processus *Feedback Retrieval* est réalisé par des demandes explicites de retours utilisateur à l'usager u_t. Ainsi, lors de l'emploi de vecteurs complets de retours utilisateur, $\forall a_i \in S_t$, un retour est sollicité auprès de l'utilisateur, c.-à-d., $P_t = S_t$. Avec des vecteurs partiels, des retours ne sont prodigués que pour $\psi < k$ bras. Le vecteur de retours utilisateur est donc seulement défini pour un sous-ensemble $P_t \subseteq S_t$, construit incrémentalement tel que : $P_t = P_{t,\psi}$ avec $P_{t,0} = \varnothing$ et $\forall j \in [1, \psi]; P_{t,j} = P_{t,j-1} \cup \{a_i\}$ où $a_i \in S_t$ est sélectionné selon le processus de *Feedback Identification* (lignes 1 à 4 de l'algorithme 1). Concernant ce processus de *Feedback Identification*, nous considérons et approfondissons l'étude des approches suivantes, extraites de la littérature [14, 15] :

Reinforce - RE : Cette méthode décrit le procédé le plus répandu pour le processus de *Feedback Identi-*

2. *Full-Information feedback* n'est possible que pour les applications n'étant pas en directe interaction avec les usagers.
3. En simulation ou pour le calcul objectif de cette récompense, un vecteur de retours utilisateur, tel qu'avec *Semi-Bandit* est nécessaire.

fication. Elle sélectionne les ψ bras de S_t présentant les plus hautes espérances de récompenses $\mathbb{E}[R_{t,a}]$:

$$a_i = argmax^{\pi}_{a \in S_t \setminus P_{t,j-1}} \mathbb{E}[R_{t,a}] \quad (1)$$

Optimal-Exploration - OE : Cette méthode vise à maximiser la connaissance de l'agent sur la distribution des espérances de récompenses $\{\mu_1, ..., \mu_k\}$ des bras composant S_t sans considérer son état courant en regard de la politique π de l'agent. Elle sélectionne les ψ bras de S_t pour lesquels le moins de retours utilisateur ont été fournis à l'itération *t* :

$$a_i = argmin_{a \in S_t \setminus P_{t,j-1}} obs_{a,t} \quad (2)$$

Où $obs_{a,t}$ est le nombre de retours utilisateur émis pour le bras a jusqu'à l'itération t.

Lors de l'exécution du processus *Feedback Retrieval*, nous construisons le vecteur de retours utilisateur F_t à partir de Y_t en sollicitant l'utilisateur u_t pour un retour sur chacun des bras *a* inclus dans P_t (ligne 5 de l'algorithme 1) :

$$F_t = \{Y_{t,a} \mid a \in P_t\} \quad (3)$$

En tant que processus de *Reward Computing*, *BUSBC* détermine d'abord une récompense cumulée R^B_t (ligne 6 de l'algorithme 1), telle que :

$$R^B_t = P_t^{\top} F_t \quad (4)$$

L'algorithme *COM-MAB* observe ensuite cette récompense et l'utilise pour mettre à jour sa politique uniquement pour les bras de P_t jugés pertinents par l'utilisateur u_t (lignes 7 à 11 de l'algorithme 1) :

$\forall a \in P_t$, si $F_{t,a} > 0$:

$$SR_{t,a} = SR_{t-1,a} + R^B_t \quad (5)$$

Où $SR_{t,a}$ est la somme des récompenses observées pour le bras a jusqu'à l'itération t.

L'algorithme 1 décrit l'utilisation de *BUSBC* dans une stratégie considérant des vecteurs partiels de retours utilisateur avec *OE* en tant que processus de *Feedback Identification*. Pour changer le processus de *Feedback Identification* pour *RE*, la ligne 3 devrait être remplacée par le mécanisme de sélection correspondant, défini par l'équation 1. De la même façon, pour employer *BUSBC* au sein d'une stratégie considérant des vecteurs complets de retours utilisateur les lignes 1 à 4 devraient être remplacées par $P_t = S_t$.
La stratégie *Bandit* peut être inefficace dans certains cas, notamment à cause de récompenses cumulées trop importantes et de l'octroi de ces récompenses à tous les bras de S_t, incluant ceux qui n'ont pas été satisfaisants en réalité. Le second problème peut être traité en n'attribuant les récompenses cumulées qu'aux bras de S_t dont le succès a été effectivement observé, tel que dans une approche *Semi-Bandit* avec des récompenses de Bernoulli. Concernant le premier problème, notre théorie est qu'une récompense cumulée pourrait en réalité être avantageuse pour les approches optimistes. Ainsi, *BUSBC* est principalement conçu pour les algorithmes *COM-MAB* de type UCB.

Algorithme 1 : *P-BUSBC*-OE

Entrées : S_t : Super-bras recommandé.
Y_t : Retours associés à $\mathcal{A}$ ou utilisateur u_t.
π : Politique de l'agent.
ψ : Nombre de retours utilisateur attendus.

1 $P_t \leftarrow \emptyset$
2 **tant que** $|P_t| < \psi$ **faire**
3 Construire P_t avec : $P_t = P_t \cup \{argmin_{a \in S_t \setminus P_t} obs_{a,t}\}$
4 **fin**
5 Acquérir les retours utilisateur : $F_t = \{Y_{t,a} \mid a \in P_t\}$
6 Calculer la récompense cumulée : $R^B_t = P_t^{\top} F_t$
7 **pour** $a \in P_t$ **faire**
8 **si** $F_{t,a} > 0$ **alors**
9 Mettre à jour la politique π avec : $SR_{t,a} = SR_{t-1,a} + R^B_t$
10 **fin**
11 **fin**

4 Expérimentations

4.1 Cadre Expérimental

4.1.1 Jeux de Données

Nos expériences sont exécutées sur plusieurs jeux de données issus d'applications réelles : **RSASM**, **Jester** et **MovieLens** (voir Tableau 1). RSASM est un jeu de données dédié à la recommandation de services, où les retours utilisateur sont des variables de Bernoulli, c.-à-d., pour chaque bras a, $F_{t,a} = 1$ si l'utilisateur considère l'élément associé au bras a comme pertinent ou $F_{t,a} = 0$ sinon. Jester traite de la recommandation de blagues et MovieLens est un jeu de données pour la recommandation de films très employé dans la littérature. Pour ces deux jeux de données, les retours utilisateur correspondent à des évaluations comprises entre 0 et 5. Nous considérons que $F_{t,a} = 1$ si l'évaluation est supérieure ou égale à 4, et $F_{t,a} = 0$, sinon.

Jeu de données	Utilisateurs	Bras	Interactions	Source
RSASM	2 152	18	$> 38K$	Kaggle
Jester	59 132	150	$> 1.7M$	Kaggle
MovieLens	942	1682	$> 100K$	Groupelens.org

Tableau 1 – Jeux de données

4.1.2 Algorithmes & Approches Concurrentes

Notre objectif est d'observer l'impact de stratégies de prise en compte du retour utilisateur sur la précision globale d'algorithmes *COM-MAB* en regard de leur politique π. Ainsi, lors de nos expériences, nous avons appliqué l'algorithme *Multiple Plays* [2] aux algorithmes *MAB* suivants, extraits de la littérature :

- *ε-greedy* [22], avec $\varepsilon = 0.0009$
- *Thompson Sampling (TS)* [1]
- *Upper Confidence Bounds 1 (UCB1)* [4]
- *Upper Confidence Bounds 2 (UCB2)* [4]

Ces algorithmes ont inspiré de nombreuses variantes au sein de la littérature. Puisqu'ils ne tirent avantage d'aucune optimisation dépendante du cadre applicatif (p.ex. disponibilité d'informations contextuelles), nous pensons que nos observations quant-à leurs performances seront également valables dans leurs déclinaisons plus récentes et spécifiques.
Néanmoins, cette étude porte sur les stratégies de prise en compte du retour utilisateur ; les algorithmes *COM-MAB* fournissent un cadre d'évaluation et ne sont pas en compétition. Nous comparons donc notre méthode *BUSBC* avec les approches suivantes, extraites de la littérature :

- *Bandit (B)* [3, 11]
- *Semi-Bandit (SB)* [3, 9]
- *Bandit and Semi-Bandit (BSB)* [15]

À notre connaissance, cet article est également le premier a considérer des stratégies *Bandit (B)* avec des vecteurs de retours utilisateur partiels.

4.1.3 Métrique

Précision Globale. Nous considérons la précision globale [10], définie par l'équation suivante :

$$Acc(T) = \frac{\sum_{t=1}^{T} r_t}{T} \tag{6}$$

Comme expliqué à la sous-section 2.1, $r_t \in \{0, 1\}$, est une *récompense d'évaluation* modélisant l'opinion globale de l'utilisateur u_t concernant la recommandation S_t. Elle n'est employée que dans le calcul de la précision globale et demeure inconnue de l'agent. Cette récompense est calculée en considérant les retours utilisateur associés à chacun des bras a de S_t comme suit :

$$r_t = \begin{cases} 1 & \text{si } \sum_{a=0}^{k} F_{t,a} \geq \frac{k}{2} \text{ , avec } F_{t,a} \in \{0, 1\} \\ 0 & \text{sinon} \end{cases} \tag{7}$$

Les récompenses observées par l'algorithme *COM-MAB* (R_t) sont déterminées selon la stratégie employée à partir des retours utilisateur acquis. Une variable d'évaluation (r_t), opérant indépendamment de l'algorithme et de la stratégie est donc nécessaire à l'évaluation de ces approches. Nous pouvons ainsi observer objectivement les impacts de la stratégie employée et de la quantité de retours utilisateur perçus sur les performances de l'algorithme.

4.1.4 Protocole Expérimental

Les stratégies évaluées dans cet article impliquent les processus de *Reward Computing* suivants : *Bandit*, *Semi-Bandit*, *Bandit and Semi-Bandit*, *Bandit under Semi-Bandit Conditions*. Nous les évaluons pour chaque algorithme *COM-MAB* sur chaque jeu de données, avec des vecteurs de retours utilisateur complets et partiels. Lors de l'emploi de vecteurs partiels, nous comparons pour le processus de *Feedback Identification* les méthodes *Optimal-Exploration* et *Reinforce*. Les stratégies étudiées dans cet article sont identifiées par leurs paramètres et nommées comme suit :

S-C-I, avec :

S : Le niveau d'exhaustivité des retours utilisateur, des vecteurs complets (FV) ou des vecteurs partiels (P).

C : Le processus de *Reward Computing*, *Bandit* (*B*), *Semi-Bandit* (*SB*), *Bandit and Semi-Bandit* (*BSB*) ou *Bandit under Semi-Bandit Conditions* (*BUSBC*).

I : Le processus de *Feedback Identification*, *Optimal-Exploration* (*OE*), *Reinforce* (*RE*) ou "/" lors de l'emploi de vecteurs complets.

Le nombre k d'éléments recommandés à chaque itération à été choisi selon les jeux de données : RSASM dispose du plus faible nombre de bras (18) et seulement un tiers des retours exprimés sont positifs. Puisque 6 bras, en moyenne, peuvent satisfaire un utilisateur, nous considérons des recommandations de $k = 6$ éléments. Avec des retours utilisateur partiels, les algorithmes *COM-MAB* présentent leurs pires performances pour $\psi = 2$ [20]. Afin d'observer les pires performances des algorithmes selon la stratégie employée, nous considérons donc $\psi = 2$ dans ces expériences. Pour chaque expérience, nous exécutons 10 simulations d'horizon $T = 10\ 000$ itérations. Ainsi, avec des vecteurs complets, 60 000 retours auront été émis au terme de l'horizon T, contre 20 000 avec des vecteurs partiels. Pour simuler l'arrivée séquentielle d'utilisateurs, nous les sélectionnons aléatoirement un par un depuis le jeu de données.
Les Tableaux 2 et 3 présentent la précision globale obtenue par chaque algorithme en fonction de la stratégie employée. Le Tableau 4 expose les stratégies offrant les meilleures performances en moyenne, pour chaque algorithme, lors de l'emploi de vecteurs complets et partiels. La Figure 1 présente l'évolution de la précision globale de *UCB1* et *UCB2* sur MovieLens lors de la considération de vecteurs complets, en fonction du processus de *Rewards Computing* appliqué. La Figure 2 montre l'évolution de la précision globale de *ϵ-Greedy* et *UCB1* sur RSASM avec des vecteurs de retours partiels, selon les processus de *Rewards Computing* et *Feedback Identification* employés. Enfin, nous analysons nos résultats dans la sous-section 4.2. Nous vérifions que les résultats de précision globale obtenus avec les approches étudiées sont significativement différents en réalisant des tests de Kruskal-Wallis et de rangs signés de Wilcoxon.

4.2 Analyse des Résultats

4.2.1 Tests Statistiques

Nous réalisons des tests de *Kruskal-Wallis* afin de mettre en évidence les inégalités entre les résultats de chaque stratégies (pour chaque algorithme), c.-à-d., nous testons l'hypothèse nulle H_0 : "*Il n'y a pas de différences significatives entre les résultats des différentes stratégies (médianes) lorsqu'appliquées à un même algorithme*". Lorsque les tests de *Kruskal-Wallis* indiquent qu'il existe des différences significatives entre les résultats, nous réalisons des tests de rangs

signés de *Wilcoxon* deux à deux sur la précision globale, c.-à-d., nous testons l'hypothèse nulle H_0 : " *Il n'y a pas de différences significatives entre chaque paire de stratégies appliquées à un même algorithme*".

4.2.2 Synthèse des Résultats

Dans cette partie, nous présentons et analysons les résultats obtenus par les algorithmes avec différentes stratégies. Pour des raisons de clarté, nous nous limitons, dans cet article, à une synthèse de nos principales observations.

		RSASM	Jester	MovieLens
Algorithme	**Stratégie**	$Acc(T)$	$Acc(T)$	$Acc(T)$
ε-greedy	FV-B-/	0,490 $\pm 0,044$	0,402 $\pm 0,003$	0,879 $\pm 0,006$
	FV-BSB-/	0,551 $\pm 0,008$	0,416 $\pm 0,007$	0,907 $\pm 0,007$
	FV-BUSBC-/	**0,557** $\pm 0,008$	*0,436* $\pm 0,005$	*0,919* $\pm 0,004$
	FV-SB-/	*0,555* $\pm 0,008$	**0,453** $\pm 0,006$	**0,928** $\pm 0,003$
TS	FV-B-/	0,351 $\pm 0,071$	0,193 $\pm 0,017$	0,853 $\pm 0,005$
	FV-BSB-/	0,390 $\pm 0,051$	0,395 $\pm 0,018$	0,855 $\pm 0,005$
	FV-BUSBC-/	*0,528* $\pm 0,027$	*0,415* $\pm 0,009$	*0,883* $\pm 0,010$
	FV-SB-/	**0,564** $\pm 0,004$	**0,446** $\pm 0,005$	**0,923** $\pm 0,003$
UCB1	FV-B-/	0,488 $\pm 0,004$	0,402 $\pm 0,004$	0,853 $\pm 0,004$
	FV-BSB-/	*0,557* $\pm 0,005$	*0,407* $\pm 0,004$	*0,873* $\pm 0,004$
	FV-BUSBC-/	**0,563** $\pm 0,004$	**0,431** $\pm 0,005$	**0,921** $\pm 0,004$
	FV-SB-/	0,555 $\pm 0,004$	0,404 $\pm 0,006$	0,759 $\pm 0,004$
UCB2	FV-B-/	0,488 $\pm 0,004$	0,173 $\pm 0,005$	0,871 $\pm 0,012$
	FV-BSB-/	*0,545* $\pm 0,011$	0,177 $\pm 0,004$	*0,874* $\pm 0,005$
	FV-BUSBC-/	**0,559** $\pm 0,005$	**0,183** $\pm 0,003$	**0,915** $\pm 0,005$
	FV-SB-/	*0,545* $\pm 0,005$	*0,178* $\pm 0,002$	0,837 $\pm 0,002$

Tableau 2 – Résultats avec vecteurs complets de retours utilisateur, meilleurs résultats en gras, seconds en italique

Vecteurs complets : Le Tableau 2 présente les résultats de précision globale obtenus par chaque algorithme sur chaque jeu de données, en fonction du processus de *Reward Computing* employé. Ces résultats sont presque tous significativement différents (p-value < 0.05), la majorité des différences non significatives étant observées entre les résultats obtenus par les stratégies *FV-BSB-/* et *FV-B-/*, extraites de la littérature. Par ailleurs, nous observons que parmi les méthodes étudiées, notre approche *FV-BUSBC-/* est celle présentant le plus de différences significatives. Nos résultats indiquent que *UCB1* et *UCB2* obtiennent leur meilleure précision globale, sur chaque jeu de données, lors de son utilisation. De plus, la Figure 1 confirme que *BUSBC* permet rapidement et pour tout l'horizon de bien meilleurs résultats que les méthodes concurrentes. Concernant *ϵ-greedy* et *Thompson Sampling*, leurs meilleurs résultats sont obtenus avec le processus de *Reward Computing SB* (stratégie *FV-SB-/*). Il est néanmoins intéressant de noter que pour ces algorithmes, sur chaque jeu de données, *BUSBC* correspond au 2nd meilleur choix malgré le fait que cette approche n'est pas conçue pour améliorer leurs performances.

Vecteurs partiels : Le Tableau 3 présente les résultats de précision globale obtenus par chaque algorithme sur chaque jeu de données, lors de l'acquisition de $\psi = 2$ retours par itération, en fonction des processus de *Reward Computing* et de *Feedback Identification* appliqués. Dans ce cadre partiel, les différences significatives entre les processus de *Reward Computing* étudiés sont confirmés (p-value < 0.05). Cependant, ces différences sont moins importantes du fait

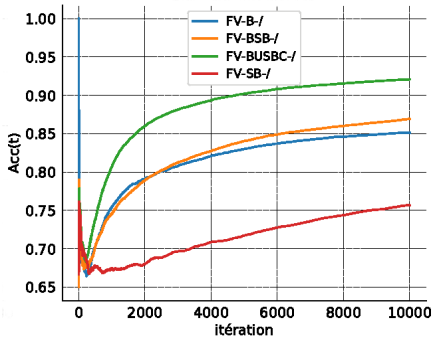
		RSASM	Jester	MovieLens
Algorithme	**Stratégie**	$Acc(T)$	$Acc(T)$	$Acc(T)$
ε-greedy	P-B-OE	0,504 $\pm 0,037$	0,418 $\pm 0,016$	0,875 $\pm 0,006$
	P-B-RE	0,448 $\pm 0,042$	0,397 $\pm 0,014$	0,860 $\pm 0,007$
	P-BSB-OE	0,480 $\pm 0,042$	**0,429** $\pm 0,009$	0,871 $\pm 0,008$
	P-BSB-RE	0,483 $\pm 0,036$	0,411 $\pm 0,009$	0,877 $\pm 0,005$
	P-BUSBC-OE	0,511 $\pm 0,025$	0,423 $\pm 0,008$	0,878 $\pm 0,009$
	P-BUSBC-RE	*0,527* $\pm 0,020$	0,421 $\pm 0,007$	*0,882* $\pm 0,006$
	P-SB-OE	0,523 $\pm 0,012$	0,424 $\pm 0,008$	0,881 $\pm 0,004$
	P-SB-RE	**0,529** $\pm 0,008$	*0,425* $\pm 0,005$	**0,893** $\pm 0,003$
TS	P-B-OE	0,469 $\pm 0,053$	0,407 $\pm 0,016$	0,843 $\pm 0,008$
	P-B-RE	0,361 $\pm 0,042$	0,424 $\pm 0,009$	0,847 $\pm 0,018$
	P-BSB-OE	0,520 $\pm 0,030$	*0,427* $\pm 0,012$	*0,882* $\pm 0,006$
	P-BSB-RE	0,498 $\pm 0,025$	0,422 $\pm 0,006$	0,872 $\pm 0,005$
	P-BUSBC-OE	*0,533* $\pm 0,023$	*0,427* $\pm 0,011$	0,865 $\pm 0,008$
	P-BUSBC-RE	0,507 $\pm 0,030$	**0,443** $\pm 0,005$	0,878 $\pm 0,011$
	P-SB-OE	0,531 $\pm 0,007$	0,420 $\pm 0,005$	0,843 $\pm 0,006$
	P-SB-RE	**0,548** $\pm 0,007$	0,424 $\pm 0,002$	**0,888** $\pm 0,003$
UCB1	P-B-OE	**0,547** $\pm 0,007$	**0,433** $\pm 0,007$	**0,858** $\pm 0,008$
	P-B-RE	0,491 $\pm 0,021$	0,395 $\pm 0,005$	0,777 $\pm 0,005$
	P-BSB-OE	0,534 $\pm 0,018$	*0,422* $\pm 0,005$	0,821 $\pm 0,007$
	P-BSB-RE	0,500 $\pm 0,015$	0,398 $\pm 0,003$	0,784 $\pm 0,003$
	P-BUSBC-OE	*0,545* $\pm 0,008$	0,421 $\pm 0,006$	*0,846* $\pm 0,006$
	P-BUSBC-RE	0,505 $\pm 0,018$	0,404 $\pm 0,005$	0,826 $\pm 0,005$
	P-SB-OE	0,510 $\pm 0,008$	0,401 $\pm 0,006$	0,754 $\pm 0,004$
	P-SB-RE	0,503 $\pm 0,005$	0,384 $\pm 0,004$	0,714 $\pm 0,003$
UCB2	P-B-OE	**0,517** $\pm 0,033$	0,174 $\pm 0,007$	*0,867* $\pm 0,006$
	P-B-RE	0,344 $\pm 0,054$	0,173 $\pm 0,007$	0,849 $\pm 0,010$
	P-BSB-OE	*0,512* $\pm 0,014$	0,170 $\pm 0,005$	0,842 $\pm 0,011$
	P-BSB-RE	0,431 $\pm 0,048$	**0,178** $\pm 0,004$	0,852 $\pm 0,006$
	P-BUSBC-OE	0,508 $\pm 0,023$	0,175 $\pm 0,003$	0,863 $\pm 0,014$
	P-BUSBC-RE	0,466 $\pm 0,031$	*0,176* $\pm 0,006$	**0,875** $\pm 0,003$
	P-SB-OE	0,467 $\pm 0,034$	0,174 $\pm 0,002$	0,805 $\pm 0,016$
	P-SB-RE	0,473 $\pm 0,005$	**0,178** $\pm 0,004$	0,857 $\pm 0,005$

Tableau 3 – Résultats avec vecteurs partiels de retours utilisateur, meilleurs résultats en gras, seconds en italique

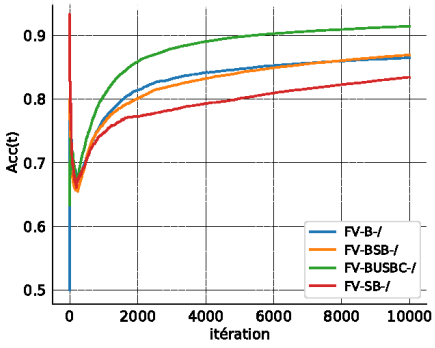
	Stratégie Optimale	
Algorithme	**FV**	**P**
ε-greedy	FV-SB-/	P-SB-RE
TS	FV-SB-/	P-SB-RE
UCB1	FV-BUSBC-/	P-B-OE
UCB2	FV-BUSBC-/	P-B-OE

Tableau 4 – Identification des combinaisons optimales avec vecteurs de retours utilisateur complets et partiels

de la réduction du nombre de retours utilisateur, menant à davantage de similarités dans les comportements des approches étudiées. Concernant les processus de *Feedback Identification*, nous observons également un impact sur la précision globale des algorithmes *COM-MAB*. Dans nos expériences, la plupart des différences observées sont statistiquement significatives, avec *OE* offrant de meilleures performances que *RE* dans la plupart des cas. Nous pouvons noter que le processus de *Rewards Computing* Bandit (stratégies *P-B-OE et P-B-RE*) présente des résultats plus compétitifs lors de l'emploi de vecteurs partiels, devenant le meilleur choix pour l'algorithme *UCB1* lorsque employé avec *OE* dans ce cadre. Ce constat s'explique notamment par la réduction du nombre de bras insatisfaisants percevant une récompense. Cependant, nous notons que sur Jester et RSASM, la précision globale obtenue par *UCB1* avec

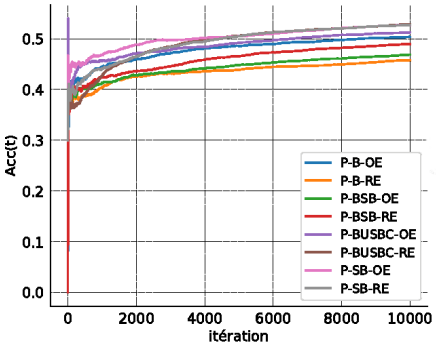


(a) UCB1

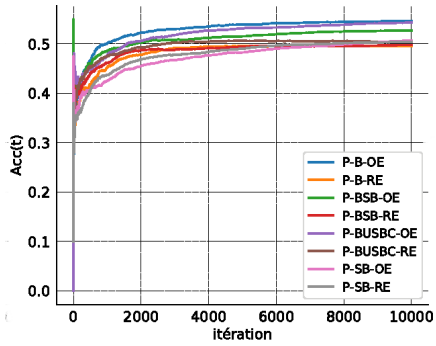


(b) UCB2

FIGURE 1 – Évolution de la précision globale sur MovieLens avec des vecteurs de retours utilisateur complets



(a) ϵ-Greedy



(b) UCB1

FIGURE 2 – Évolution de la précision globale sur RSASM avec des vecteurs de retours utilisateur partiels

P-BUSBC-OE devient équivalente à celle obtenue avec *P-B-OE* (cf. Figure 2b). Un point important à noter est que dans ce cadre, les performances des processus de *Rewards Computing* sont 1) restreintes par le faible nombre de retours perçus ; 2) influencées par le processus de *Feedback Identification* employé. Ainsi, pour un plus grand horizon ou lorsque davantage de retours utilisateur sont perçus, *P-BUSBC-OE* pourrait être un meilleur choix. Par ailleurs, en observant les résultats indépendamment du processus de *Feedback Identification* employé, nous pouvons constater que *BUSBC* reste en réalité un meilleur choix pour *UCB1* sur chaque jeu de données.

Discussion. Le Tableau 4 expose les stratégies optimales pour chaque algorithme *COM-MAB* selon l'exhaustivité des retours utilisateurs. Nous observons que, généralement, la stratégie optimale reste la même sur l'horizon. Cependant, des changements de rangs entre stratégies concurrentes peuvent être observés (voir figure 2a : *P-BUSBC-RE*). Globalement, nous notons que lorsque la stratégie optimale pour un algorithme *COM-MAB* est inconnue, les approches *BUSBC* et *OE* constituent de meilleurs choix.

Bien que la plupart des différences entre les résultats de précision globale obtenus soient significatives, certaines ne le sont pas. Ces situations résultent notamment de deux faits : a) les processus de *Reward Computing* étudiés ont des comportements proches ; b) les jeux de données présentent des biais. De plus, l'impact de ces faits augmente en proportion de la réduction du nombre de retours utilisateur exploités à chaque itération. Enfin, avec des vecteurs de retours utilisateur partiels, nous observons un certain nombre de différences non significatives entre les résultats obtenus par l'application de stratégies de composition totalement différentes (voir Figure 2b, *P-SB-OE* et *P-BUSBC-RE*). Ces observations illustrent que l'association des processus composant la stratégie impacte autant la précision globale que leur ajustement indépendant.

5 Conclusions et Perspectives

Dans cet article, nous avons établi que, pour les algorithmes *COM-MAB*, toute stratégie de prise en compte du retour utilisateur pouvait être définie au travers d'un modèle générique composé de trois processus. Nous avons également montré que la modification d'au moins un de ces processus impacte significativement la précision globale d'un algorithme *COM-MAB*. Ainsi il est possible d'améliorer les performances d'un algorithme par l'ajustement de sa stratégie. Nous avons proposé *Bandit under Semi-Bandit Conditions* (*BUSBC*), un nouveau processus de *Reward Computing*, qui accroît la précision globale des algorithmes de type *UCB*. De plus, notre analyse empirique illustre que, concernant le processus *Feedback Identification*, l'approche *OE* surpasse généralement *RE*, bien que cette seconde méthode décrive le procédé le plus répandu dans la littérature.

La collecte de retours utilisateur et leur usage reste un

défi dans le domaine de l'apprentissage automatisé. Dans la mesure où la stratégie optimale peut varier au cours du temps pour un même algorithme *COM-MAB*, une perspective particulièrement intéressante serait l'implémentation d'une approche portfolio pour sélectionner dynamiquement différentes variantes de chacun des processus durant l'exploitation. Sur le plan industriel, il semble également pertinent d'employer simultanément plusieurs stratégies. Par exemple, un algorithme *COM-MAB* pourrait employer une stratégie de type "Bandits en Cascades" pour acquérir un premier niveau de connaissances avec un minimum de contraintes pour l'utilisateur ainsi qu'une stratégie considérant un vecteur partiel de retours utilisateur pour enrichir cette connaissance lorsque l'utilisateur est disposé à fournir davantage d'informations.
Nous sommes convaincus que des approches de ce type ouvriront la voie à une nouvelle génération de systèmes de recommandation, plus adaptés aux attentes des utilisateurs et dotés d'un apprentissage plus efficace, capables de mieux répondre à de nombreuses problématiques rencontrées dans les applications réelles.

Remerciements

Ces travaux ont été réalisés avec le soutien de l'Association Nationale de la Recherche et de la Technologie (ANRT). Les auteurs tiennent également à remercier les relecteurs anonymes pour leurs pertinents commentaires.

Références

[1] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *COLT*, 2012.

[2] Venkatachalam Anantharam, Pravin Varaiya, and Jean Walrand. Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays-part i : I.i.d. rewards. *IEEE Transactions on Automatic Control*, 1987.

[3] Jean-Yves Audibert, Sebastien Bubeck, and Gabor Lugosi. Minimax policies for combinatorial prediction games. *COLT*, 2011.

[4] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *JMLR*, 3, 2002.

[5] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. Gambling in a rigged casino : The adversarial multi-armed bandit problem. *IEEE 36th Annual Foundations of Computer Science*, 1995.

[6] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing - SICOMP*, 32(1), 2002.

[7] Djallel Bouneffouf and Irina Rish. A survey on practical applications of multi-armed and contextual bandits. *ARXIV*, 2019.

[8] Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit : General framework and applications. In *International Conference on Machine Learning - ICML*, 2013.

[9] Richard Combes, Mohammad Sadegh Talebi Mazraeh Shahi, Alexandre Proutiere, and Marc Lelarge. Combinatorial bandits revisited. In *NIPS*. Curran Associates, Inc., 2015.

[10] Nicolas Gutowski. *Context-aware recommendation systems for cultural events recommendation in Smart Cities*. PhD thesis, Université d'Angers, Angers, France, 2019.

[11] Shinji Ito, Daisuke Hatano, Hanna Sumita, Kei Takemura, Takuro Fukunaga, Naonori Kakimura, and Ken-Ichi Kawarabayashi. Improved regret bounds for bandit combinatorial optimization. In *NIPS*. Curran Associates, Inc., 2019.

[12] Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvari. Combinatorial cascading bandits. In *NIPS*. Curran Associates, Inc., 2015.

[13] Paul Lagrée, Claire Vernade, and Olivier Cappé. Multiple-play bandits in the position-based model. In *NIPS*, 2016.

[14] Alexandre Letard, Tassadit Amghar, Olivier Camp, and Nicolas Gutowski. Bandit et semi-bandit avec retour partiel : Une stratégie d'optimisation du retour utilisateur. In *APIA*, 2020.

[15] Alexandre Letard, Tassadit Amghar, Olivier Camp, and Nicolas Gutowski. Partial bandit and semi-bandit : Making the most out of scarce users' feedback. In *ICTAI*, 2020.

[16] Shuai Li, Baoxiang Wang, Shengyu Zhang, and Wei Chen. Contextual combinatorial cascading bandits. In *ICML*, 2016.

[17] Alexander Luedtke, Emilie Kaufmann, and Antoine Chambaz. Asymptotically optimal algorithms for multiple play bandits with partial feedback. *ARXIV*, 2016.

[18] Gergely Neu. First-order regret bounds for combinatorial semi-bandits. In *COLT*, Proceedings of Machine Learning Research. PMLR, 2015.

[19] Herbert Robbins. Some aspects of the sequential design of experiments. *Bull. of the AMS*, 1952.

[20] Aadirupa Saha and Aditya Gopalan. Combinatorial bandits with relative feedback. In *NIPS*, pages 985–995. Curran Associates, Inc., 2019.

[21] Karthik Abinav Sankararaman. Semi-bandit feedback : A survey of results. In *CoRR*, 2016.

[22] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning : An introduction*, volume 1. MIT press Cambridge, 1998.

Dynamical system approach to explainability in recurrent neural networks

Alexis Dubreuil

Institut de la Vision, Sorbonne Universités, INSERM, CNRS, F-75012 Paris, France

alexis.dubreuil@gmail.com

Résumé

Les technologies basées sur l'IA, notamment les machines neuronales, sont souvent qualifiées de boîtes noires, limitant leur déploiement dans toute une gamme d'applications. Ici nous présentons des méthodologies qui permettent d'ouvrir ces boîtes noires. Elles se basent sur le formalisme de la théorie des systèmes dynamiques qui a été initialement mis à profit pour comprendre les calculs dans les réseaux de neurones biologiques. Nous décrivons des travaux qui appliquent ces méthodes pour la modélisation en neurosciences et pour le traitement automatique du langage. Ceci nous permet, à partir d'exemples concrets, d'illustrer comment l'explicabilité peut contribuer aux développements de l'IA.

Mots-clés

explicabilité, apprentissage profond, systèmes dynamiques, neurosciences, Traitement Automatique du Langage

Abstract

AI based technologies and neural machines in particular are often referred to as black-boxes, which limits their deployment in various fields. Here we present methodologies to open these black-boxes. They rely on the formalism of dynamical system theory which has initially been leveraged to understand neural computations in brain circuits. We describe works that applied these methodologies for modeling in neuroscience and for natural language processing, allowing us to discuss concrete examples demonstrating the potential of explainability in fostering further developments in AI.

Keywords

explainability, deep-learning, dynamical systems, neuroscience, Natural Language Processing

1 Introduction

AI based technologies are developing at a fast pace, in particular thanks to recent progress in harnessing neural network based machines. Further expanding the scope of applications requires the functioning of AI systems to be explainable in a human understandable format so that they can be trusted, for instance in safety-critical applications such as self-driving cars. While many levels of explainability can be defined based on the deployment context of a technology [4], a necessary step for neural network based machines is to get access to their "inner workings" [26]. However, the functioning of neural networks is particularly difficult to grasp as they typically are high-dimensional non-linear systems with thousands or millions of parameters tuned by a learning algorithm. Here we review efforts that have been undertaken over the last ten years to reverse-engineer these neural machines. The reviewed works focused on recurrent neural networks (RNN), a generic neural architecture, known to be particularly well suited to deal with time varying inputs and outputs such as those in natural language processing tasks (see e.g. [35]). In sections 2 and 3, we introduce the theoretical concepts, based on dynamical system theory [1], that have been developed since the 80's to understand neural computations. In section 4 we present methodological tools that leverage this conceptual framework to reverse-engineer RNN. In section 5 we illustrate how explainability enables AI approaches to foster interactions between theoretical and experimental neuroscience. In section 6 we show how these methodologies have been used to understand how RNN solve natural language processing tasks. Finally in discussion we outline the potential of these methodologies for scientific applications of AI, practical applications of AI and theoretical investigations of neural computations.

2 RNN as dynamical systems

Recurrent neural networks are fully connected networks, i.e. each neuron a priori receives inputs from all the other neurons in the network (Fig. 1a). Given that they typically receive inputs that extend over time, they are often referred to as deep-in-time neural networks as illustrated in Fig. 1b. A typical way of formalizing the time evolution of the state of a RNN composed of N neurons, $\vec{x}^t \in \mathbb{R}^N$, is through

$$\vec{x}^{t+1} = W_{rec}\Phi\left(\vec{x}^t\right) + W_{in}\vec{u}^t \tag{1}$$

where $W_{rec} \in \mathbb{R}^{N\times N}$ is the recurrent connectivity matrix, $W_{in} \in \mathbb{R}^{N\times N_{in}}$ connects input neurons to recurrent neurons, $\vec{u}^t \in \mathbb{R}^{N_{in}}$ models the activity of input neurons at time t, and $\Phi(.)$ is a non-linear function, typically the hyperbolic tangent or the rectified-linear function. The results

1. see [31] for a friendly introduction to dynamical system theory

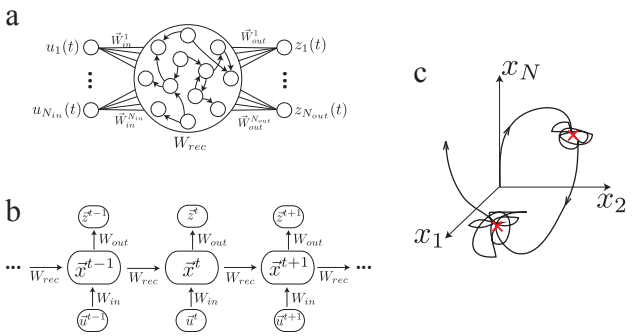


Figure 1. a. Schematic of a recurrent neural network (RNN). b.Typical representation of a RNN seen as a deep-in-time neural network. c. The activity of a RNN while it solves a task and computes on its inputs can be represented as a trajectory in the state-space defined by the activity of each individual units. Red crosses represent stable activity states that could be used to store computationally relevant variables. Transitions between such states would then be triggered by inputs.

of computations in the RNN are read out by readout neurons with activity $\vec{z}^t \in \mathbb{R}^{N_{out}}$:

$$\vec{z}^t = W_{out}\Phi(\vec{x}^t) \tag{2}$$

where $W_{out} \in \mathbb{R}^{N_{out}\times N}$ connects recurrent neurons to the readout neurons.
These equations describe so called Vanilla RNN or Elman networks. Note that sometimes the time evolution is defined as $\vec{x}^{t+1} = \Phi\left(W_{rec}\vec{x}^t + W_{in}\vec{u}^t\right)$. These two formulations can be shown to be equivalent up to a change of coordinates [7]. With the addition of a leak term to eq. (1), the time evolution of the network can be expressed in continuous time as

$$\tau\dot{\vec{x}}(t) = -\vec{x}(t) + W^{rec}\Phi(\vec{x}(t)) + W^{in}\vec{u}(t) \tag{3}$$
$$\vec{z}(t) = W^{out}\Phi(\vec{x}(t)) \tag{4}$$

This is the standard equation for the dynamics of a rate model used for the modeling of biological neural networks [10] and whose behavior has been studied extensively as we illustrate below. RNN are thus high-dimensional ($N \gg 1$) non-linear dynamical systems whose activity $\vec{x}(t)$ can be conveniently thought of as a trajectory in a state-space of dimension N as illustrated in Fig. 1c. When the network is performing a task, these trajectories will reflect the computations performed by the RNN.

3 Recurrent computations on inputs

Neural trajectories are the results of two factors, the input drive $W^{in}\vec{u}(t)$ and the recurrent activity $W^{rec}\Phi(\vec{x}(t))$. In biological and artificial neural networks, neural trajectories are typically attracted towards low-dimensional sub-spaces of dimension $D \ll N$ [9; 2], such that recurrent activity can typically be described by D coordinates or latent variables representing the overlap between the neural activity $\vec{x}(t)$ and particular directions in state-space. The connectivity structure W_{rec} determines the shape of the sub-spaces on which these latent variables evolve. In the next two subsections we present classical examples illustrating these features and explicit how latent variables can be related to computations. In the last sub-section we present a class of RNN for which the high-dimensional dynamics eq. (3) can be reduced to a D-dimensional dynamical system governing the time evolution of latent variables, making this class of networks particularly amenable to reverse-engineering.

3.1 Point attractors in Hopfield networks

As a first example, we consider Hopfield networks that have been proposed as models of associative memory [19; 20]. A set of P memories $\vec{\xi}^\mu \in \{-1,+1\}^N$, $\mu = 1,...,P$ are stored in a RNN by choosing the connectivity matrix

$$W_{rec} = \vec{\xi}^1\vec{\xi}^{1T} + ... + \vec{\xi}^P\vec{\xi}^{PT} \tag{5}$$

where T denotes the transpose operation. If the number of memories P is not too large, neural trajectories evolve towards one of P fixed-points (the dynamical landscape associated to eq. (3) is said to contain P attractors). The dynamics of the network during the retrieval of a memory μ can be effectively reduced to a one-dimensional system over a latent variable representing the overlap between the network state and the memory pattern $\kappa_\mu = \langle\Phi(\vec{x}(t)), \vec{\xi}^\mu\rangle$ [11]

$$\dot{\kappa}_\mu = -\kappa_\mu + F\left(\kappa_\mu\right) \tag{6}$$

where $F(.)$ is a sigmoid-shaped function. When the RNN is cued with an input that triggers neural activity in the direction $\vec{\xi}^\mu$, neural activity sets up in a fixed point characterized by $\kappa_\mu \simeq 1$, $\kappa_{\nu\neq\mu} \simeq 0$ and the memory μ is said to be retrieved.

3.2 Continuous attractors

In the previous example, the recurrent dynamics is attracted towards a set of P discrete points randomly spread throughout state-space. Activity of biological and artificial neural networks can also evolve on continuous sub-spaces such as line attractors (D=1) [29; 21] or plane attractors (D=2) [36; 2]. In [29], a line attractor (illustrated in Fig. 2b) is imprinted in the network's dynamics by choosing a rank one recurrent matrix with a carefully chosen singular value

$$W_{rec} = \vec{m}\vec{n}^T \tag{7}$$

that generates activity $W_{rec}\Phi(\vec{x}(t))$ along the direction $\vec{m}$. The N-dimensional dynamics of such a network can thus be reduced to the one-dimensional dynamics of a latent variable κ. Given the particular structure of the connectivity matrix, inputs that trigger neural activity along the direction $\vec{n}$ will be kept in network activity for a time that can be much longer than the single neuron time constant τ, while inputs triggering neural activity along directions orthogonal to $\vec{n}$ will be forgotten from network activity on a time-scale

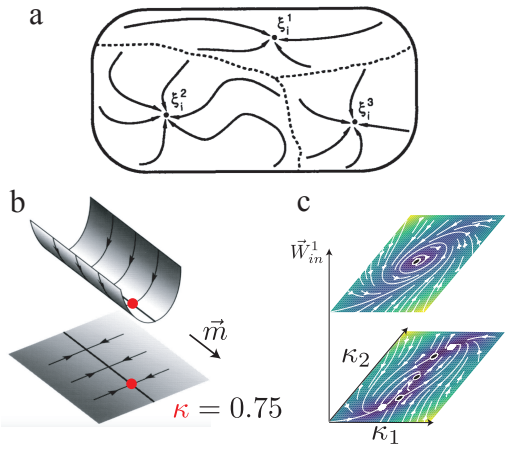


Figure 2. a. Schematic projection of the state space for a Hopfield network storing three memories (taken from [18]). b. Schematic representation of a line attractor. The activity of the network can be thought of as a ball evolving on the half-pipe representing the energy function of the system. The location of this ball on the line composed of marginally stable fixed-points encodes a scalar value (taken from [29]). c. Dynamical landscapes of a 2-D reduced system (see eq. (10)) obtained from a trained rank-2 RNN composed of $N = 1024$ neurons. Filled white circles denote stable fixed-points, empty circles denote unstable fixed-points, color coded is the absolute speed at which latent variable trajectories evolve and the white arrows depict the directionality of the latent variables flow. Inputs to the network can modify the effective couplings between latent variables and thus reconfigure their dynamical landscape. (taken from [13])

τ (cf Fig. 2B). Such latent variables that encode analog values and maintain them on long-time scales can store various computationally relevant information such as the position of an animal [36] or the sentiment associated with a text in a language processing network [21] (see section 6).

3.3 Reduced dynamics over latent variables

The above examples illustrate how latent variables can be used to relate patterns of neural activity to their role in computations. As shown by recent theoretical works [24; 12; 5; 13], the dynamics of low-rank RNN, whose connectivity matrix can be written as

$$W_{rec} = \sum_{k=1}^{D} \vec{m}_k \vec{n}_k^T \tag{8}$$

can be analytically reduced to a dynamics over D latent variables ($D \ll N$). Using tools from statistical physics, the neural activity can be expressed as [24]

$$\vec{x}(t) = \sum_{k=1}^{D} \kappa_k(t)\vec{m}_k + W_{in}\vec{v(t)} \tag{9}$$

where the latent variables are projections of neural activity onto the connectivity vectors $\vec{m}_k$ ($\kappa_k(t) = \langle \vec{x}(t), \vec{m}_k \rangle$) and $\vec{v(t)}$ corresponds to inputs $\vec{u}(t)$ low-pass filtered with a time constant τ. The full N-dimensional dynamics eq. (3) is then fully characterized by a D-dimensional dynamical system [12], that can be concisely expressed as

$$\tau \dot{\kappa_j}(t) = -\kappa_j(t) + \sum_{k=1}^{D} \tilde{\sigma}_{jk}^{rec} \kappa_k(t) + \sum_{k=1}^{N_{in}} \tilde{\sigma}_{jk}^{in} v_k(t) \tag{10}$$

where the effective couplings $\tilde{\sigma}$ are non-linear functions of the κ_k's and inputs $\vec{u}$ (see [12; 13] for explicit expressions of the $\tilde{\sigma}$'s). In particular an input u_k can have two qualitatively different effects, it can either be integrated by a latent variable κ_j through $\tilde{\sigma}_{jk}^{in} v_k(t)$ or it can modulate the effective couplings between latent variables and inputs by controlling the value of one of the $\tilde{\sigma}$. In Fig. 2c we show two dynamical landscapes associated with a single system of two-latent variables, for different applied inputs. At the bottom, the dynamical landscape is composed of two stable fixed-points (white dots) towards which latent variables are attracted. As an input is applied onto the RNN, the effective dynamics of the latent variables is reconfigured and they will oscillate on a limit cycle. Trained RNN make use of this principle to solve tasks [12; 13].
As we will see below, low-rank networks, whose dynamics can be reduced in a principled manner, are particularly well suited to provide access to their inner workings.

4 Reverse-engineering methodologies for RNN

4.1 Characterizing the dynamical landscape by linearization around fixed-points

According to the principles exposed in section 2, understanding computations in a RNN requires to characterize the dynamical landscape associated with the set of trained parameters $\{W_{rec}, W_{in}, W_{out}\}$. Towards such a characterization, Sussillo and Barak [32] developed a numerical procedure to find the fixed-points (or slow points, $\dot{\vec{x}}(t) \simeq 0$) of a trained RNN (Fig. 3a). Then by studying the linearized dynamics of trained RNNs around these slow points they could reveal important computational properties of these systems. For instance by doing so on a RNN trained on a 3-bit flip-flop task (Fig. 3b), they could identify multiple stable fixed-points that enable the system to remain in various states important for solving the task (black crosses in Fig. 3b). They also found saddle-points (fixed-points with some unstable directions, green crosses) that allows inputs to the network to induce transitions between stable fixed-points. By visualizing the locations of these fixed-points and the neural trajectories (blue and red thin lines) in a relevant three dimensional sub-space (identified by principal component analysis on network trajectories concatenated across multiple task trials), they could provide a concise description of how the network solves the task. Various authors have used this approach to reverse-engineer RNN [8; 21] and a tensorflow toolbox is available to find fixed-points and characterize the linear

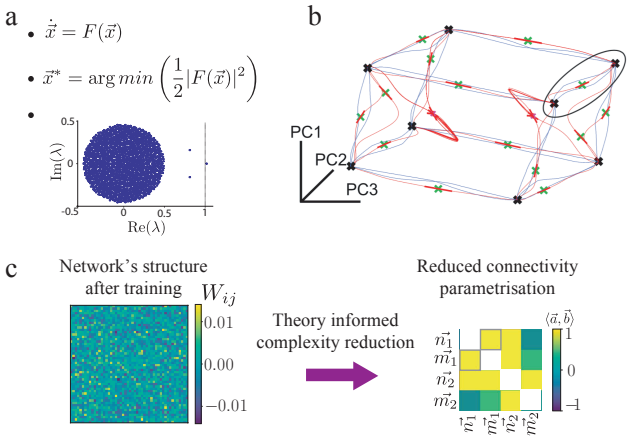


Figure 3. a. Numerical procedure to characterize the dynamical landscape of RNN. For a generic dynamical system defined by a function F(.), the first step is to locate the fixed-points (or slow points) of the trained RNN, then to characterize the linearized dynamics around these fixed points, which can be done by analyzing the eigenspectrum of the stability matrix associated to each fixed point (points beyond the dashed lines are associated with state-space directions that are unstable). b. Visualization of neural trajectories (blue and red thin lines) for a network that solves a 3-bit flip-flop task : a network is excited by three input neurons whose activity takes non-zero values at random time points, three output neurons are trained to represent the sign of the last activation of the inputs neurons (taken from [32]). Black crosses denote the location of stable fixed-points in which the network settles in between inputs. Possible input-triggered transitions between network states are allowed by the presence of saddle points (green crosses) whose direction of instability connects (red thick lines) stable fixed-points. c. Illustration of the analytical procedure to reduce the dimensionality of trained RNN. After training one obtains a $N \times N$ connectivity matrix, the complexity of this matrix is reduced by computing averaged quantities that determine RNN dynamics, e.g. the overlaps between the recurrent connectivity vectors $\vec{m}$ and $\vec{n}$.

dynamics around these fixed-points [15].

4.2 Reducing the dimensionality of trained RNN

Here we present two methodologies that have been proposed to reduce the dimensionality of trained RNN and to describe their computations in a simple language.

In [27], the authors first dissect a trained RNN using an approach similar to what has been described in the above section and noticed that recurrent activity mainly lies in a two-dimensional sub-space. Then, inspired by the knowledge distillation approach [6], they trained a two-units RNN to reproduce the projected two-dimensional activity of the original network and obtained a simpler two-dimensional dynamical system that performs the task equally well than the original network. This allowed them to synthesize the behavior of the network by describing the dynamics of only two variables.

In [12], the authors trained RNN with a connectivity matrix of fixed rank D as in eq. (8). To do so, they reparametrized learning so that a loss function is minimized over the parameters $\Theta = \{\vec{m}_k, \vec{n}_k, W_{in}, W_{out}\}_{k=1,\dots,D}$ instead of $\{W_{rec}, W_{in}, W_{out}\}$ as is usually done. In the training procedure, the rank D was treated as an hyper-parameter, and for each task they identified the minimal rank that allows the task to be solved. They then used the theoretical insights presented in section 3.3 to concisely summarize the dynamics of their RNN with a dynamical system of minimal dimensionality. This analytical characterization of the reduced system allows to relate properties of neural trajectories to the connectivity structure through the mathematical expression of the functional couplings $\tilde{\sigma}$, which depends on the statistical structure of the connectivity vectors Θ (see Fig. 3c). This detailed understanding has allowed, from RNN trained on simple tasks, to engineer similarly functional networks without appealing to a learning algorithm [12; 13]. Thus this methodology, in which solutions to a task are looked for in the restricted space of minimal rank RNN, appears to provide a solution to the reverse-engineering problem that is satisfactory according to the criterion set by Richard Feynman's quote : "What I can not create, I do not understand". An application of this method is described in section 5.1.

5 RNN implementing elementary cognitive processes and brain circuits modeling

Here we review recent works that leveraged these reverse-engineering methodologies in the context of neuroscience modeling (see e.g. [23; 34; 8; 37; 12; 13; 17; 14]). We show how this has allowed to understand the network implementation of elementary cognitive processes. In the first subsection we review recent works that extracted network mechanisms for motor control and context-dependent computations. These works have been performed in parallel to experimental neuroscience studies that involve designing minimal behavioral tasks to isolate core cognitive processes such as working memory, decision making, motor-control or context-dependent computations (see Fig. 4a for an example). In the second subsection we outline how accessing explainable RNN allows to make interpretations and predictions regarding experimental measurements in neuroscience.

5.1 Network implementation of elementary cognitive processes

Motor control. Here we describe the reverse-engineering of a RNN trained on a task inspired by a study of motor control where monkeys slide their fingers on a screen to navigate from a starting point to a target point, with various configurations of obstacles in between [34]. Applying the

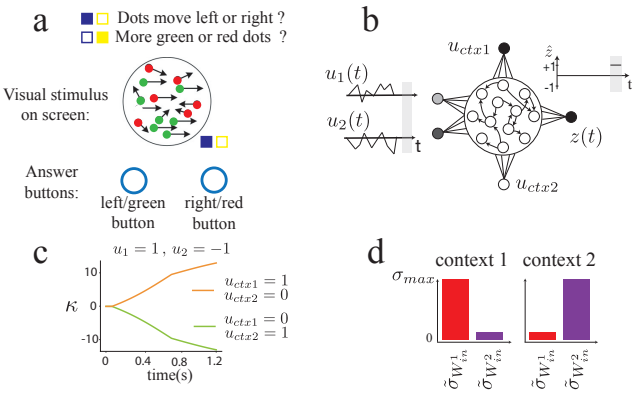


Figure 4. a. Schematic of a behavioral task used in neuroscience experiments. On a screen, a subject watches dots that are colored either green or red and that move erratically (black arrows represent velocity vectors of the dots, which are redrawn randomly at each time step). Based on some contextual cues, the subject's task is to report, by pressing one of two buttons, whether the averaged movement of the dots is to the left or to the right (context 1), or whether there are more green or red dots (context 2). b. Minimal modeling of this behavioral task for a RNN : two sensory inputs $u_1(t)$ and $u_2(t)$ fluctuate over time, the readout neuron should respond $+1$ (resp.-1) if the relevant sensory input has a positive (resp. negative) average. The relevant sensory input is determined by the activities of the two contextual neurons. c. Dynamics of the latent variable characterizing recurrent activity for two task trials. The same constant sensory inputs are shown for the two trials, but the contextual input changes. The latent variable integrates the relevant sensory input. d. In the reduced description of the RNN, the effective interactions coupling the latent variable with the two sensory inputs are modulated by context.

methodology presented in section 4.1 they performed the linear-stability analysis of the single fixed point found in the RNN. While most of the eigenvalues of the stability matrix were associated to decaying modes of activity (eigenvalues with negative real parts), three pairs of eigenvalues (complex conjugate pairs of purely imaginary numbers) were associated with three typical oscillatory modes observed in the neural trajectories for both trained RNN and experimental data.

Context-dependent computations. In [12] the authors trained low-rank RNN to implement the context-dependent task described in Fig. 4a,b whose implementation relies on a form of if... then... else... statement. They performed reverse-engineering using the methodology described in section 4.2. They found that a rank one network was sufficient to perform this task and could summarize the recurrent computation by the one dimensional non-linear dynamical system

$$\dot{\kappa} = -\kappa + \tilde{\sigma}_\kappa \kappa + \tilde{\sigma}_{W^1_{in}} u_1(t) + \tilde{\sigma}_{W^2_{in}} u_2(t) \qquad (11)$$

Within this formulation, the behavior of the network could be explained in simple terms : the contextual inputs u_{ctx1} and u_{ctx2} control the effective couplings between the latent variable and the sensory inputs $u_1(t)$ and $u_2(t)$ (Fig. 4d), allowing the latent variable to compute the temporal average $\langle u_1(t)\rangle_t$ if $u_{ctx1} = 1 \,\&\, u_{ctx2} = 0$ and $\langle u_2(t)\rangle_t$ if $u_{ctx1} = 0 \,\&\, u_{ctx2} = 1$ (Fig. 4c). This explanation is equivalent, although more concise, to the one we could give by running the methodology of section 4.1 : recurrent activity evolves on a line attractor to which is associated an input selection vector whose direction is tuned by contextual inputs to integrate the relevant sensory inputs [23; 22]. Other tasks involving context-dependent computations have been shown to rely on such context-dependent dynamical landscape for the latent variables [13].

5.2 Explainability for neuroscience experiments

The insights obtained by reverse-engineering these RNN have turned out to be quite useful for biological network modeling. A typical approach goes as follows : i) the activity of part of a brain circuit has been characterized in experiments, ii) a modeler explores various training settings (e.g. plays with regularization techniques, [34]) to obtain RNN whose neural activity matches the biological one, iii) the RNN is reverse-engineered. This leads to two types of contributions. First, by synthesizing the available data in a mechanistic framework, it allows to propose a functional role to observed properties of brain activity. For instance in [34], by using various learning procedures they obtained different RNN implementations of the task and could show that the simplest RNN (in terms of their dynamical properties) better matched the data. They could moreover show that the simplest RNN were more robust to perturbations such as neuron erasure, suggesting that features of neural activity observed in motor cortex correspond to a robust implementation of motor control. Second, it allows to make predictions for parts of the circuit, or properties of the circuit, that have not been characterized experimentally. For instance in [13] the authors propose predictions regarding the structure of neural selectivity (how the activity of neurons relate to the task variables) to be observed in animals performing specific tasks.

6 Reverse-engineering RNN performing language processing tasks

These methods have been used to reverse-engineer networks trained on language processing tasks. In [21] the authors trained various RNN architectures (Vanilla RNN, GRU, LSTM) to perform a binary sentiment analysis tasks on the Yelp review, the IMDB movie review and the Stanford Sentiment Treebank datasets. Each trial consists of a sentence where each word is mediated through an input vector to the RNN, at the end of each sentence a readout neuron is asked to provide a binary value, $+1$ if the review is positive, -1 if the review is negative. Analyzing trained networks they found that the learning algorithm builds

networks whose activity is low-dimensional (a PCA analysis shows that 90% of the RNN hidden state variance is captured with only two principal components). By visualizing neural trajectories produced throughout the presentation of a sentence in a two-dimensional space, they noticed that neural activity mainly evolves along a single dimension aligned with the network's readout vector W_{out}, with sentences associated with positive reviews driving activity in one direction and negative reviews driving activity in the other direction (see Fig. 5a). The amplitude of an activity shift along this direction was dependent on the current word being presented, with neutral words (e.g. "the", "car") eliciting no movement along this dimension and highly negative (e.g. "bad", "horrible") or positive (e.g. "amazing", "great") words eliciting large movements. By finding the fixed points of this system and analysing their linear stability (section 4.1), they could show that these trajectories are supported by a line attractor (section 3.2). They realized that a RNN implementing only a line attractor would perform as well as a bag-of-words model, although their trained RNN showed better performance. In a subsequent study [22], still using the same reverse-engineering techniques, they could reveal that trained RNN performed better than bag-of-words because they are able to process words in a context-dependent manner. They focused on the effect of modifier words (e.g. "not") that reverse the meaning of a subsequently appearing word such as in "not bad" versus "bad". They found that modifier words drive activity in a direction orthogonal to the line attractor towards points in state-space where the response to subsequent inputs is reversed (Fig. 5b), revealing a similar mechanism than the one described in section 5.1. Remarkably, using the insights gained by their reverse-engineering studies they could extend bag-of-words models to include the effect of such modifier words and reach performance similar to the one of the more complex RNN initially used.
In another recent study [2] the authors performed the same type of reverse-engineering on other NLP tasks (document classification, review star prediction, emotion tagging). With learning leading to low-dimensional activity in the RNN, they could similarly describe how the properties of the networks' dynamical landscapes govern the RNN computations.

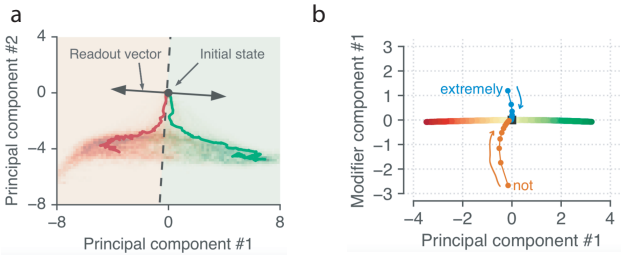


Figure 5. a. Two neural trajectories in a 2-D cut of the state-space of RNN performing a sentiment analysis task. The green trajectory corresponds to the activity of the RNN while it is presented with a positive review : positive words push neural activity towards the right. The red trajectory corresponds to a negative review. (taken from [21]). b. Modifier words push neural activity in directions that are orthogonal to the line attractor direction. Neural activity then reaches a point in state-space where the dynamical landscape is such that subsequent inputs generates a shift in state-space in a direction opposite to the one that would occur in the absence of a modifier word (taken from [22]).

7 Discussion

We have mainly reviewed two methodologies that have been developed to characterize the dynamical landscape of RNN. We have first introduced a numerical procedure that allows to find the fixed-points of trained RNN and to characterize the linearized dynamics of networks around these fixed-points (section 4.1). We have shown on examples how stable fixed-points are associated with internal network states that are important for the storage of information. We have also shown how training algorithms can build other types of dynamical structures to support other aspects of computation, such as saddle-points organizing the set of possible input-dependent transitions between internal states (Fig 3b).
We have then introduced an analytical approach that allows to reduce RNN to low-dimensional dynamical systems of latent variables corresponding to collective modes of activity of the full RNN (section 4.2). These reduced dynamical systems are parametrized by averaged quantities, or summary statistics, of the trained connectivity matrices (Fig. 3c). This allows to describe the relationship between properties of neural trajectories and the learnt structure of RNN, and thus to provide satisfactory solutions to reverse-engineering problems. This approach relies on reparametrizing the connectivity matrix by a rank D connectivity matrix and could appear rather limited. Nevertheless, it has been shown that training full rank RNN on simple tasks leads to low-dimensional implementation of the tasks that can be captured by low-rank RNNs [28], and that rank-D RNN can in principle implement any D-dimensional dynamical system [5]. In section 5.1 we have presented how this methodology has been leveraged to show how RNN implement context-dependent computations, with contexts re-configuring the effective dynamical system that governs the time evolution of latent variables.

These methodologies grant access to the inner workings of trained networks and this level of explainability appears well suited for i) applications of AI to the sciences, ii) practical applications of AI as well as for iii) theoretical investigations of neural computations.
These methodologies have been developed within the context of biological neural network modeling. As such they have allowed to tighten the link between experimental neuroscience and neural network theory by ascribing a functional role to various correlative experimental observations, and by making predictions for future experiments (section 5.2). Given the expanding use of deep-networks,

and RNN in particular, in various fields of science, like population genetics [1] or linguistic [25], this approach is expected to be useful to reveal the network mechanisms at stake in various natural phenomena.
Reverse-engineering of RNN trained on practical tasks have also been performed, confirming that the theoretical concepts presented in section 3 are also appropriate within this context. We have presented works focused on natural language processing tasks (section 6). By leveraging the insights provided by reverse-engineering RNN, researchers could a posteriori build minimal extensions of bag-of-words models that perform as well as more complicated RNN on a sentiment analysis task. This is reminiscent of the knowledge distillation approach [6] that has been proposed to reduce the complexity of trained machine learning models and make their commercial implementations more efficient. The insights gained from reverse-engineering studies can also be used to design better training algorithms by proposing new forms of regularization [17].
Finally the description of neural computations in terms of collective or latent variables, that can be thought of as encoding symbols (see e.g. section 3.1), might allow to formulate neural computations in a language closer to standard notions of computations. For instance, from the dynamical system point of view, the RNN trained on the 3-bit flip-flop task can be interpreted as a finite-state automaton, with attractors corresponding to machine states and saddle-points programming possible input-triggered transitions between states (see Fig. 3b). It would be interesting to describe what other dynamical features underlie the computing power of neural networks [30] and their ability to implement pushdown automata or Turing machines. Moreover, based on the comparison between latent variables and symbols, the dynamical system approach to neural computations might turn out relevant, as an exploratory tool, for the development of hybrid AI architectures, which combine the respective strengths of symbolic AI and artificial neural networks, and that have been identified as promising for the development of explainable IA architectures [4].

Acknowledgment

I would like to thank Michele Sebag, Thomas Bonald and Jean-Louis Dessalles for useful discussions about explainability.

Références

[1] Adrion, J. R., Galloway, J. G. Kern, A. D. Predicting the Landscape of Recombination Using Deep Learning. Molecular Biology and Evolution 37, 1790–1808 (2020).

[2] Aitken, K. et al. The geometry of integration in text classification RNNs. Proceedings of the International Conference on Learning Representations (2021).

[3] Barak, O. Recurrent neural networks as versatile tools of neuroscience research. Current Opinion in Neurobiology 46, 1–6 (2017).

[4] Beaudouin, V., Bloch, I., Bounie, D., Clémençon, S., D'Alché-Buc, F., Eagan, J., Maxwell, W., Mozharovskyi, P., Parekh, J. Flexible and Context-Specific AI Explainability : A Multidisciplinary Approach (2020)

[5] Beiran, M., Dubreuil, A., Valente, A., Mastrogiuseppe, F. Ostojic, S. Shaping dynamics with multiple populations in low-rank recurrent networks. arXiv :2007.02062 [q-bio] (2020).

[6] Bucila, C., Caruana, R., Niculescu-Mizil, A. Model Compression. Proceeding of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (2006).

[7] Ceni, A., Ashwin, P. Livi, L. Interpreting Recurrent Neural Networks Behaviour via Excitable Network Attractors. Cogn Comput 12, 330–356 (2020).

[8] Chaisangmongkon, W., Swaminathan, S. K., Freedman, D. J. Wang, X.-J. Computing by Robust Transience : How the Fronto-Parietal Network Performs Sequential, Category-Based Decisions. Neuron 93, 1504-1517.e4 (2017).

[9] Cunningham, J. P. Yu, B. M. Dimensionality reduction for large-scale neural recordings. Nature Neuroscience 17, 1500–1509 (2014).

[10] Dayan, P., Abbott, L.F. Theoretical neuroscience : computational and mathematical modeling of neural systems. Computational Neuroscience Series (2001).

[11] Derrida, B., Gardner, E. Zippelius, A. An exactly solvable asymmetric neural network model. EPL (Europhysics Letters) 4, 167 (1987).

[12] Dubreuil, A., Valente, A., Mastrogiuseppe, F. and Ostojic, S. Disentangling the roles of dimensionality and cell classes in neural computations. NeurIPS Neuro-AI Workshop (2019).

[13] Dubreuil, A., Valente, A., Beiran, M., Mastrogiuseppe, F. and Ostojic, S. Complementary roles of dimensionality and population structure in neural computations. bioRxiv (2020).

[14] Fanthomme, A., Monasson, R. Low-Dimensional manifolds support multiplexed integrations in recurrent neural networks. Neural Computation (2021).

[15] Golub, M. and Sussillo, D. FixedPointFinder : A Tensorflow toolbox for identifying and characterizing fixed points in recurrent neural networks. Journal of Open Source Software, 3(31), 1003, https ://doi.org/10.21105/joss.01003 (2018).

[16] Graves, A. et al. Hybrid computing using a neural network with dynamic external memory. Nature 538, 471–476 (2016).

[17] Haviv, D., Rivkind, A. Barak, O. Understanding and controlling memory in recurrent neural networks. Proceedings of the 36th International Conference on Machine Learning (2019).

[18] Hertz, J., Krogh, A., Palmer, R. G. Introduction to the theory of neural computation. CRC Press (1991).

[19] Hopfield, J. J. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Proceedings of the National Academy of Sciences 79, 2554–2558 (1982).

[20] Hopfield, J. J. Neurons with graded response have collective computational properties like those of two-state neurons. Proceedings of the National Academy of Sciences 81, 3088–3092 (1984).

[21] Maheswaranathan, N., Williams, A., Golub, M., Ganguli, S. and Sussillo, D. Reverse engineering recurrent networks for sentiment classification reveals line attractor dynamics. Advances in neural information processing systems (2020).

[22] Maheswaranathan, N. Sussillo, D. How recurrent networks implement contextual processing in sentiment analysis. Proceedings of the 37th International Conference on Machine Learning, PMLR 119 :6608-6619, (2020).

[23] Mante, V., Sussillo, D., Shenoy, K. V. and Newsome, W. T. Context-dependent computation by recurrent dynamics in prefrontal cortex. Nature 503, 78–84 (2013).

[24] Mastrogiuseppe, F. Ostojic, S. Linking Connectivity, Dynamics, and Computations in Low-Rank Recurrent Neural Networks. Neuron, doi :10.1016/j.neuron.2018.07.003, (2018).

[25] Lakretz, Y. et al. The emergence of number and syntax units in LSTM language models. NAACL (2019).

[26] Peterson, G. E. Foundation for neural network verification and validation. In Science of Artificial Neural Networks II, volume 1966, pages 196–207. International Society for Optics and Photonics (1993).

[27] Schaeffer, R., Khona, M., Meshulam, L., Fiete Rani, I. Reverse-engineering recurrent neural network solutions to a hierarchical inference task for mice. NeurIPS (2020).

[28] Schuessler, F., Mastrogiuseppe, F., Dubreuil, A., Ostojic, S. Barak, O. The interplay between randomness and structure during learning in RNNs. Advances in neural information processing system (2020).

[29] Seung, H. S. How the brain keeps the eyes still. Proceedings of the National Academy of Sciences 93, 13339–13344 (1996).

[30] Siegelmann, H.T. and Sontag, E.D. On the computational power of neural nets. J. Comput. Syst. Sci., 50(1) :132–150 (1995).

[31] Strogatz, S.H. Nonlinear dynamics and chaos. Westview Press (1994).

[32] Sussillo, D. Barak, O. Opening the Black Box : Low-Dimensional Dynamics in High-Dimensional Recurrent Neural Networks. Neural Computation 25, 626–649 (2013).

[33] Sussillo, D. Neural circuits as computational dynamical systems. Current Opinion in Neurobiology 25, 156–163 (2014).

[34] Sussillo, D., Churchland, M. M., Kaufman, M. T. Shenoy, K. V. A neural network that finds a naturalistic solution for the production of muscle activity. Nature Neuroscience 18, 1025–1033 (2015).

[35] Sutskever, I., Vinyals, O. Le, Q. V. Sequence to Sequence Learning with Neural Networks. Advances in neural information processing system (2014).

[36] Tsodyks, M. and Sejnowski, T. Proceedings of the Third Workshop on Neural Networks : from Biology to High Energy Physics. Int. J. Neural Syst. (6) Suppl., 81 (1994).

[37] Wang, J., Narain, D., Hosseini, E. A. Jazayeri, M. Flexible timing by temporal scaling of cortical responses. Nature Neuroscience 21, 102–110 (2018).

Fast and memory efficient AUC-ROC approximation for Stream Learning

Subhy. Albakour[1,2], Alain-Pierre. Manine[1], and Erick. Alphonse[1]

[1]IDAaaS S.A.S.
[2]Institut Polytechnique de Paris, IP Paris.

Abstract

Machine Learning applied to never-ending data streams has unique resource-consumption constraints that require processing one data-point at a time without storing. Much research has been devoted to develop algorithms that learn from data streams, and most algorithms are available in libraries such as MOA[1] and River[2]. However, less is done on the evaluation of the models generated by these algorithms. Area under the ROC Curve (AUC-ROC) has more discrimination power, as an evaluation metric, than accuracy and all other confusion-matrix based metrics. Nonetheless, computing the AUC-ROC of a model is expensive, and violates streaming constraints in practical applications, as it requires the entire history of the model's predictions when computed. In this paper, we show how we can extend sketching algorithms to summarize the prediction history of a model, and then produce a high-quality approximation of its AUC-ROC in bounded memory and constant update time. This renders AUC-ROC practical for evaluating and selecting Stream Learning models.

Keywords

Stream Learning, Data Stream Mining, Model Evaluation, AUC-ROC, Data Sketching.

1 Introduction

Data streams are generated in a wide variety of contexts, such as application logs, IoT sensor readings, network traffic, financial data, marketing data, click streams, epidemics and computer security to name a few [4, 17, 11].
Real-time analytics, done over data streams, have led to a totally different machine learning paradigm concerning training, evaluating and deploying models. This paradigm is called *Machine Learning for Data Streams* or *Stream Learning* for short, as opposed to the main Machine Learning paradigm based on *Batch Learning*. In Batch Learning, datasets are typically loaded from disk and models are built off-line, usually with multiple passes over the data before being evaluated and deployed. However, in Stream Learning, models are in constant progress; they are built, evaluated and deployed online, i.e., one data-point at a time. Since data streams could be infinite, only one pass over the data is allowed. Finally, space and time complexity must be constant to be able to process the infinite data stream, and to make sure that Stream Learning models are ready to make predictions at any time, without lagging behind the data stream [4].

Most of the research in Stream Learning is dedicated to develop learning algorithms [13, 4, 5, 18], but less is done on the evaluation metrics. These metrics should be adapted to meet the constraints of Stream Learning as well. Developing a streaming version of a metric is trivial for the ones that are confusion-matrix based, such as accuracy and precision, where running counters can capture the full state of the matrix at any given time. However, this is not the case for AUC-ROC since it exhibits no memory bound and requires the entire history of the model predictions, i.e., the pairs (predicted score, label)[21].

Bringing AUC-ROC to Stream Learning requires using approximations. Brzezinski [7] proposed to tackle this issue with a window-based approximation which only considers the most recent examples. However, this approach is not practical, as the window size depends on the data stream at hand. Actually, research on confidence intervals of AUC-ROC shows that window size should be defined as a function of the class ratio, i.e., the smaller the ratio of the minority class is, the wider the window should be in order to report good-quality estimations [2]. Thus a significant window size can be very large (especially for imbalanced data) [2], and one can end up allocating more space for the evaluation metric than the learning models themselves, and violating the memory constraints of Stream Learning. A window-based AUC-ROC computation also requires, either storing the window twice (one for the forgetting mechanism, and another to keep a sorted structure), or sorting the entire window for each computation. This makes one choice bad for memory consumption, and the other bad for time complexity ($\mathcal{O}(N \log N)$ with N is the window size).

In this paper, we propose a new AUC-ROC approximation that can make use of the entire prediction history, thanks to an extension of some frequency-based sketching

[1] https://github.com/Waikato/moa
[2] https://github.com/online-ml/river

algorithms. It exhibits the same properties as the metrics based on the confusion matrix, i.e., (i) independent of data distribution, (ii) independent of the learner, (iii) memory-bounded, (iv) can be updated in a constant time.

Our experiments show that the method is more accurate than a window-based approximation, and reports at least 2 significant digits after the decimal point, which is enough for model selection in practice.

The rest of the paper is organized as follows. Section 2 presents the relevant work from the literature. Section 3 defines AUC-ROC and shows how it is computed. We present the approximation method in section 4. Section 5 how the approximation is adapted to streaming contexts. In section 6, we report some experimental results. Section 7 provides theoretical and experimental error analysis before concluding in section 8.

2 Related Work

A Receiver Operating Characteristics (ROC) graph is a technique first used in the context of signal processing to depict the trade-off between hit rates (true positives) and false alarms (false positives), which helps in selecting high-quality classifiers [21]. It is also widely used in decision making for medical diagnoses, and extensive work has been done in this area [16, 9].

Since the early works of Provost and Flach [19, 10], AUC-ROC has been extensively used in evaluation Machine Learning models. This is motivated by the attractive properties of AUC-ROC, such as comparing classifiers across all possible class distributions, and its discrimination power in imbalanced data settings[23]. However, the streaming version of AUC-ROC is relatively recent, and to the best of our knowledge, no native online method, i.e., one data-point at a time, has been developed. Actually, the state-of-the-art method to compute AUC-ROC in streaming is window-based, this method is not easily usable in practice as it consumes too much resources (section 1). This is true especially for imbalanced data streams, a use case for which AUC-ROC is most meaningful.

On the other hand, AUC-ROC approximation is an older problem. However, it is not always considered in order to save resources as needed for data streams. Continuous approximations of AUC were developed in order to solve the problem of indifferentiability, so they can be used as objective functions in gradient descent optimizations [14, 24]. Nonetheless, these approximations are not online, nor do they reduce the resource consumption, as they still need the entire history of predictions when computed.

A more related line of work in approximations considers online settings where AUC-ROC learning curves should be updated upon the arrival of each data-point [20]. However, this approach's main concern is the computation time, not the memory, and it still needs to store the entire prediction history. It uses an efficient way to iterate over data that skips less important information.

The most related work is the approximation proposed by Bouckaert [6], where scores are distributed into bins over which the computation is done. The bins uniformly divide the interval $[0, 1]$, and counters for positive and negative scores are kept up-to-date and are used to compute the AUC-ROC. This method has two drawbacks that we overcome in this paper. First, it uses static bins, and this is problematic as the range of the scores produced by a learner may differ from one learner to another, this makes the approximation error *learner-dependent*. Second, the method considers just *one specific distribution* of scores (the one that if transformed with a log-log transformation, yields a uniform distribution). Our method makes no assumptions about the data distribution.

Frequency-based sketching algorithms produce summaries that are independent of the underlying distribution [1, 12], and can dynamically split the range of scores based on their frequencies. Building on this property, we show in this paper how to extend frequency-based sketches, and use them to approximate AUC-ROC. We also analyse the approximation error and its sources.

3 AUC-ROC

We start by introducing some notation. Let the data space be $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}, \mathcal{Y}$ are feature space and label space respectively. For binary classification, the labels are $\mathcal{Y} = \{-1, +1\}$. Suppose the data points are drawn from an unknown distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, and $\mathcal{D}_+$, $\mathcal{D}_-$ are class conditional distributions $\mathcal{D}_+ = \mathcal{D}_{X|Y=+1}$, $\mathcal{D}_- = \mathcal{D}_{X|Y=-1}$. We denote by $\mathcal{S} = \{(x_1, y_1), ..., (x_N, y_N)\} \in (\mathcal{X} \times \mathcal{Y})^N$ the data stream. n is number of negative data points in the stream $n = |\{i | y_i \in \mathcal{S}_Y, y_i = -1\}|$ and p is the number of positive data points $p = N - n$. Finally, $\mathcal{R}$ is the function used to score the data points, i.e., the ranker itself.

Geometrically, AUC-ROC represents the area under the ROC curve (Figure 1), which is the curve of true positive rate as a function of the false positive rate of a classifier for all possible decision thresholds. From a probabilistic point of view, AUC-ROC is the probability that a positive example drawn at random is assigned a higher score than a negative example drawn at random[2, 21, 22]. Formally, suppose two random variables X, X' that follow the distributions $X \sim \mathcal{D}_+, X' \sim \mathcal{D}_-$, then AUC-ROC of a ranker $\mathcal{R}$ is,

$$AUC(\mathcal{R}) = \mathbb{P}(X > X') + \frac{1}{2}\mathbb{P}(X = X')$$

For a sample $\mathcal{S}$, this is estimated by the following un-biased estimator,

$$\widehat{A}(\mathcal{R}, \mathcal{S}) = \frac{1}{pn} \sum_{i|y_i=+1} \sum_{j|y_j=-1} \mathbb{1}_{\mathcal{R}(x_i)>\mathcal{R}(x_j)} + \frac{1}{2}\mathbb{1}_{\mathcal{R}(x_i)=\mathcal{R}(x_j)}$$

which is also known as Wilcoxon-Mann-Whitney statistic [2]. This defines an algorithm to estimate AUC-ROC, and it is the one used in libraries. For a sample $\mathcal{S}$, computing AUC-ROC is done by counting for each positive point x_i the number of negative points x_j that are given a lower score. This can be done in a single pass once the points are sorted [22].

4 AUC-ROC Approximation

In this section, we describe how to compute AUC-ROC (or simply AUC) while respecting streaming constraints, using a high-quality approximation that only needs a memory-bounded data structure (namely, a histogram). In the next section, we will show how to build such a structure in a streaming way.

Figure 1 shows the ROC curve in an un-normalized space (the red curve) for a ranker $\mathcal{R} : \mathcal{X}^{\mathcal{N}} \to I \subseteq \mathbb{R}$. We partition the N axis into b intervals, and apply the trapezoid method to approximate the area as follows (the area under the blue curve):

$$\widehat{AUC} = \frac{1}{P_b N_b} \sum_{i=1}^{b} \left(\frac{1}{2} n_i p_i + n_i P_{i-1} \right) \tag{1}$$

where $n_i = N_i - N_{i-1}$, $p_i = P_i - P_{i-1}$ and $P_0 = N_0 = 0$. Notice that the term $\frac{1}{2} n_i p_i$ represents the area of a triangle in figure 1, and the term $n_i P_{i-1}$ is the area of the rectangle underneath.

As it is easier to deal with the score space, the partition in ROC space can be seen as a result of a partition in the score space I, represented by b disjoints and *decreasingly* ordered sub-intervals $I_1, \ldots, I_b$. The values n_i, p_i are respectively the number of negative and positive points whose scores are inside the interval I_i.

Notice that this approximation is memory-bounded, as it only requires storing $2b$ values, i.e., $p_i, n_i, i \in \{1 \ldots b\}$, and the complete ranker history is not needed anymore. The number of intervals b represents a trade-off between memory and approximation error.

To perform the computation in Equation 1, it suffices to build a structure where each interval I_i is mapped to a bin B_i that keeps the counts p_i, n_i. This structure is known as "histogram". This means that streaming AUC computation is reduced to streaming computation of a histogram.

5 AUC-ROC for Stream Learning

In this section, we show how to adapt *data sketching algorithms*, so they build histograms that serve in approximating AUC in streaming contexts.

5.1 Histograms and Data Sketching

Data sketching is a technique that summarizes huge datasets into relatively small data structures called

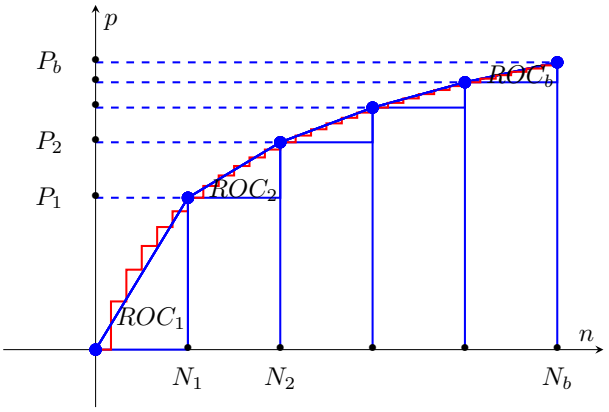


Figure 1: ROC curve + area approximation

sketches. Sketches are used to approximate query answers over the original datasets. Histograms are sketches from a family called *frequency-based sketches*, they are usually used to answer queries, such as quantile queries [12].

Given a sample S, drawn from an interval $I \subset \mathbb{R}$, and an interval partition $\mathcal{P}(I)$ of I, i.e., $\mathcal{P}(I)$ is a set of intervals that satisfy $\cup_{J \in \mathcal{P}(I)} = I$, and $\forall J, J' \in \mathcal{P}(I), J \neq J' \implies J \cap J' = \varnothing$, then a histogram $\mathcal{H}_S$ over the sample S is a mapping

$$\mathcal{H}_S : \mathcal{P}(I) \to \mathbb{R}, \mathcal{H}_S(J) = |J \cap S| \tag{2}$$

The structure that counts the elements in $J \cap S$ is called *bin*, and $J \cap S$ is called *bin set*. Each interval J corresponds to a bin B_J. Usually, a uniform partition is chosen.
As data is not known beforehand, dynamically building a histogram from a stream is not a trivial task. However, several streaming algorithms are available in the literature [1, 17, 3, 8].

5.2 Two-Dimensional Histograms

While data sketching algorithms only summarize one dimensional data into histograms, computing AUC requires two-dimensional data to be summarized, i.e., the pairs (predicted score, label). For this purpose, we introduced the concept of a two-dimensional (2D) histogram, an extended version of the histogram that stores the information relevant to the task of AUC approximation. Data sketching algorithms have to be adapted so they can build 2D histograms.

The key operation in a sketching algorithm is $Insert$ that updates the sketch with a new data-point. To adapt an algorithm that builds 1D histograms to the 2D case, we need to adapt $Insert$ operation so that a bin can (i) take two-dimensional data as input, i.e., (score,label) in our case, (ii) store more information about its bin set. In the context of AUC approximation, bins need to store two counters p_i, n_i (positives and negatives), instead of just one counter. However, the algorithm-specific logic that defines

the partition $\mathcal{P}(I)$ stays unchanged.

Algorithm 1 exemplifies an adapted version of $Insert$ operation in the case of DDSketch[17]. Following the same principles, we also adapted other sketching algorithms (BenHaim [3] and TDigest [8]). But for the sake of brevity, no details about them are provided in this paper.

Algorithm 1: Insert(s,l) [DDS2]
An example of a 2D histogram update method

input : a (score, label) pair $(s, l) \in \mathbb{R}^+ \times \{-1, +1\}$.
output: a set of bins $B_1, \ldots, B_b$.

$i \leftarrow \lceil log_\gamma(s) \rceil$;
if $l = +1$ **then**
  $B_i.p \leftarrow B_i.p + 1$;
else
  $B_i.n \leftarrow B_i.n + 1$;
if $|\{j : B_j.p + B_j.n > 0\}| > m$ **then**
  $i_0 \leftarrow min(\{j : B_j.p + B_j.n > 0\})$;
  $i_1 \leftarrow min(\{j : B_j.p + B_j.n > 0, j > i_0\})$;
  $B_{i_1}.n \leftarrow B_{i_1}.n + B_{i_0}.n$;
  $B_{i_1}.p \leftarrow B_{i_1}.p + B_{i_0}.p$;
  $B_{i_0}.n \leftarrow 0$; $B_{i_0}.p \leftarrow 0$;
end

Compared to the original algorithm 2, the adapted one 1 only has different input, and keeps more information in the bin B_i (positives and negatives counts $B_i.p$, $B_i.n$) instead of just one. A histogram built by the algorithm 1, i.e., the bins $B_1 \ldots B_b$, can be used anytime during the lifetime of the stream to compute AUC using equation 1. For details about DDSketch and its parameters γ, m , we refer to the original paper [17].

Algorithm 2: Insert(s) [DDS, original version [17]]

input : $s \in \mathbb{R}^+$.
output: a set of bins $B_1, \ldots, B_b$.

$i \leftarrow \lceil log_\gamma(s) \rceil$;
$B_i \leftarrow B_i + 1$;
if $|\{j : B_j\}| > m$ **then**
  $i_0 \leftarrow min(\{j : B_j\})$;
  $i_1 \leftarrow min(\{j : B_j > 0, j > i_0\})$;
  $B_{i_1} \leftarrow B_{i_1} + B_{i_0}$;
  $B_{i_0} \leftarrow 0$;
end

5.3 Selected Algorithms: Histograms

Many frequency-based sketching algorithms (histograms for short) can be found in the literature due to their importance in real-time analytics [1, 17, 3, 8]. They are a key to high-performance data analysis, and they are already incorporated in data processing engines such as Druid, Spark and MacroBase [11].

No relevant criterion is available in the literature to select histograms in the context of AUC approximation, as it is a new application. Taking into account (i) popularity and (ii) ease of manipulation (to adapt a 2D version), we selected 3 fairly different algorithms, BenHaim "BH" (Apache Druid and Hive) [3], TDigest "TD"(Microsoft and Netflix) [8] , DDSketch "DDS"(DataDog) [17], we refer to the original papers for more details [3, 8, 17]. We extend the original version of DDSketch[3]. For TDigest we use the version implemented in stream-lib[4], we also adapt Hive's version of BenHaim[5]. We call theses 2D versions of the histograms DDS2, TD2 and BH2 respectively.

6 Experiments and Results

For the experiments, we present the evaluation setting in section 6.1, then we present the datasets and learners in section 6.2. We validate our claims by showing that approximation is accurate in section 6.3 and memory-bounded in section 6.4. Finally we show the improvement over the state of the art window-based (our baseline) in section 6.5.

6.1 Evaluation Method

The prequential method (test then train one example at a time) is a general methodology to evaluate models in streaming scenarios[15]. The performance of a model is reported as a learning curve, i.e., the evaluation metric value at times $t \in T = \{rW_s : r \in \mathbb{N}, r \leq \frac{N}{W_s}\}$, where W_s is a sampling period (window), and N is the total number of data points. For example, the learning curve of AUC, is the time series of AUC values $AUC_t, t \in T$. To test the approximation method, we use the distance between the learning curve of exact AUC, and the learning curve of the approximation method. The distance is taken to be the mean absolute difference between the curves, i.e., the mean absolute error (MAE). The MAE of the approximation method that uses a histogram $\mathcal{H}$ is calculated as follows

$$MAE(\mathcal{H}) = \frac{1}{|T|} \sum_{t \in T} |\widehat{AUC}_t(\mathcal{H}) - AUC_t| \qquad (3)$$

where AUC_t is the exact AUC value at time t, $\widehat{AUC}_t(\mathcal{H})$ is the approximation using a histogram $\mathcal{H}$ at time t, and T is the set of sampling timestamps.

6.2 Datasets and Learners

We use three sets of datasets, synthetic, public standard real datasets and private real datasets. Synthetic datasets (generators) provide greater flexibility in controlling size and class ratio. For a description of the generators we refer to [5, 13]. We use the default parameters of the generators provided by MOA.2019.05.0 [5]. We use the *Imbalanced Stream* provided by MOA to control the class ratio, it randomly samples data points according to the desired distribution.

[3] https://github.com/DataDog/sketches-java
[4] https://github.com/addthis/stream-lib
[5] https://github.com/apache/hive

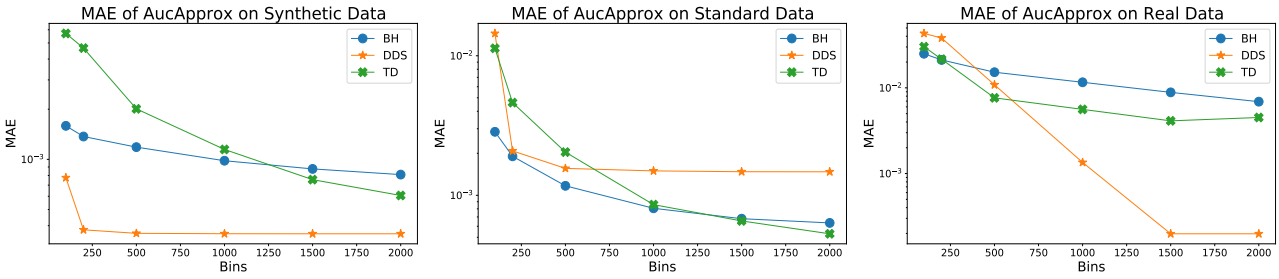


Figure 2: total MAE of AucApprox per histogram w.r.t number of bins

Parameter	Used Values
Generator	[RTG , RRBF , SEA , STAG , AGR , SINE , HYPR]
Class Ratio	[0.5 , 0.2 , 0.1 , 0.05 , 0.02 , 0.01 , 0.005]
Instances	[100K]

Table 1: Parameters of Synthetic Datasets. Generated by the cross product (Generator × Class Ratio × Instances).

Dataset	# Instances	# Features	Class Ratio
Electricity	45312	8	0.575
COVTYPE1	581012	54	0.365
PAKDD09	50000	28	0.197
Airlines	539383	7	0.445
KDDCup99	494021	41	0.197
GMSC	150000	10	0.067

Table 2: Standard Datasets

We use 7 class ratios for each generator, which adds up to $7 \times 7 = 49$ synthetic datasets (check table 1). We use 6 standard real datasets (check table 2). We refer to [13, 7] for descriptions. Note that COVTYPE1 is a binary version of COVTYPE, where the first class is considered positive.

We use 14 private datastets that represent "click-streams" on ads. They are highly imbalanced with class ratios of {0.94, 0.53, 0.34, 0.59, 0.18, 1.83, 0.49, 0.24, 0.72, 1.00, 1.02, 0.55, 1.62, 0.70} $\times 10^{-3}$, each dataset has 100K data points of 13 features.

We use different learners to cover different score distributions. We use Naive Bayes "NB", Logistic Regression "SGD", Hoeffding Tree "HT", and a bagged model "OzaBag" of 10 HT base learners. We use the implementations provided by MOA(2019.05.0) with the default parameters.

Table 3 contains the values used in the experiments. When a parameter value does not appear in the specifications of an experiment, the reported metric is averaged over all possible parameters included in the table. We fix the sampling window $W_s = 1000$. We also fix $\alpha = 0.01$ for DDSketch. We call the proposed approximation method $AucApprox$, and the window based version (baseline) $AucWin$.

Parameter	Used Values
Learner	[NB , SGD , HT , OzaBag]
Bins	[100 , 200 , 500 , 1000 , 1500 , 2000]
Histogram	[BH2 , TD2 , DDS2]

Table 3: Parameters used in experiments. Generated by the cross product (Learner × Bins × Histogram)

6.3 Mean Absolute Error

In this section we show that $AucApprox$ has a small Mean Absolute Error (MAE) when compared to the exact method "AUC". We test all 69 datasets (49 synthetic, 6 standard and 14 real), using all 4 learners, which adds up to 276 different settings for each pair (Histogram, Bins "b").

Figure 2 shows how small the approximation error is, how it differs for different histograms, and how it decreases when the number of bins increases. Notice that if DDSketch is selected, error always goes below 10^{-2}, thus $AucApprox$ can always compute AUC value with 2 significant digits after the decimal point.

Figure 2 also shows that the error converges quickly, and in case of DDSketch, the error nearly does not change after 1500 bins. This suggests that 1500 is a versatile value for the parameter b. It also means that the method can be considered as parameter-free, since we can always set $b = 1500$ as a task-independent parameter.

6.4 Memory Consumption

Here we show that $AucApprox$ exhibits a big memory gain over the exact method, while being sufficiently accurate (check previous section 6.3). To compute exact AUC, we use MOA's implementation. We use $AucApprox$ with 2000 bins in order to maximize the memory consumption. We use all datasets, except those that do not have enough data points (less than $100K$), so we exclude PAKDD99 and Electricity. We report the relative memory gain, computed as follows

$$G(\mathcal{H}) = \frac{1}{|D|.|T|} \sum_{d \in D} \sum_{t \in T} \frac{M_t}{M_t(\mathcal{H})} \tag{4}$$

Where $G(\mathcal{H})$ is the relative memory gain, D is the set of datasets, $M_t(\mathcal{H})$ the memory allocated by the $AucApprox$

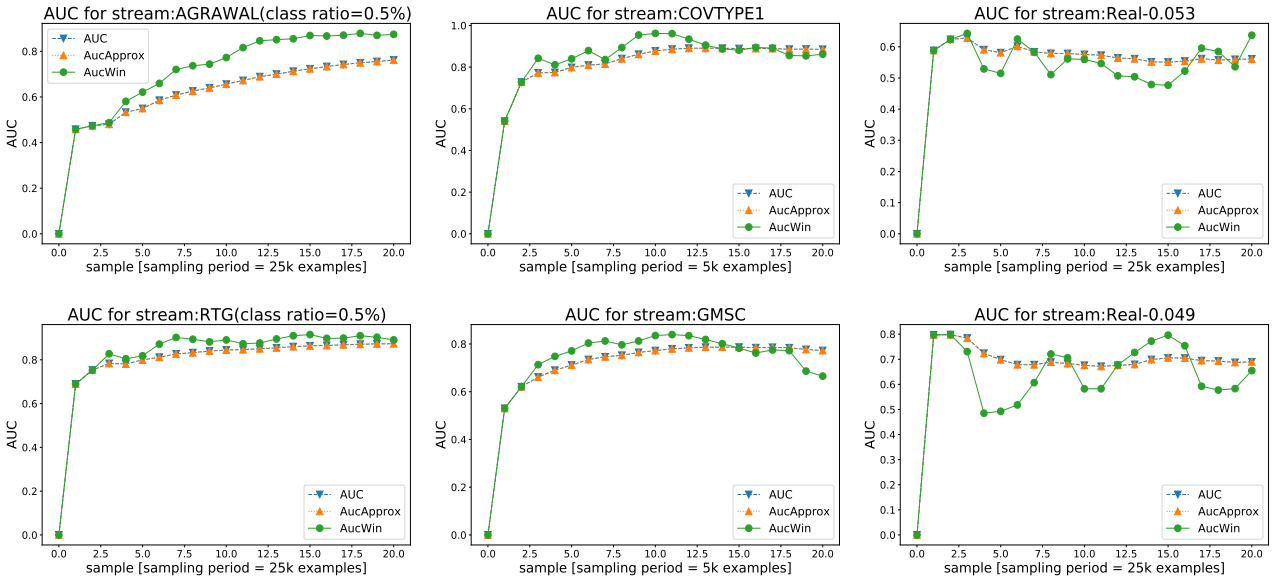


Figure 3: Some learning curves of AucApprox(DDS), AucWin against exact AUC (all evaluate OzaBag). AucArppox and AUC are nearly identical.

Data	BH2	DDS2	TD2
Synthetic	197.6	**493.4**	392.6
Standard	140.2	**271.1**	102.4
Real	134.6	**416.3**	203.5

Table 4: Relative memory gain when using AucApprox

Data	Window	AucApprox	AucWin	Mean Ratio
Real	10K	**0.012**	10.191	1431
	50K	**0.008**	4.354	1126
Synthetic	10K	**0.092**	5.667	129
	50K	**0.091**	3.465	252

Table 5: MAE(%) of AucApprox vs AucWin

at time t, M_t the memory allocated by the exact method at time t, and T is the set of sampling time stamps.

Table 4 reports the relative memory gain. It shows that $AucApprox$ consumes at least 271 times less memory than the exact version. It should be noted that the size of histograms changes with time, and with dataset because the bins are allocated dynamically.

6.5 Performance Against the Baseline

Here we only test highly imbalanced data, because this is where the approximation is needed in practice. This means that the standard datasets cannot be used. We just use the real datasets, and we use smaller class ratios for synthetic data $\{0.005, 0.002, 0.001\}$. We test two window sizes (10K, 50K) for 100K, 500K data points respectively. For $AucApprox$ we use DDSketch ($b = 1500$).

Figure 4 shows the MAE of both methods w.r.t class ratio. We can see the trend that the error of window method increases when class ratio decreases. This supports the claim that window size should be set as a function of class ratio. We can also see that AucWin performs badly (error can reach 21% for real data, and 11% for synthetic data) despite the fact that we use huge window sizes of 10K and 50K. Usually it is set to far smaller values, just 1K in the original paper [7].

Table 5 reports mean values of the curves in Figure 4 and the mean ratio (not ratio of means). The approximation method is at least 1000 times more precise than the window version for real data, and 100 times more precise for synthetic data. This is because real data is more imbalanced.

Figure 3 shows some learning curves for $AucApprox$, $AucWin$, and how they compare to the exact AUC. It can be seen how the curves of $AucApprox$ and exact AUC are nearly identical, contrary to WinAUC, where there is noticable difference. Please note that we use smaller window for standard data (second column), and smaller sampling period too, because they have less data points.

7 Error Analysis

In this section, we inspect the possible sources of error (MEA as in equation 3). For simpler analysis, we provide a formula to compute the *exact* AUC similarly to equation 1. In figure 1, notice that the partitioning defines sub ROC curves ROC_i, each curve corresponds to the same ranker $\mathcal{R}$ on the sub-domain I_i, say $\mathcal{R}_{I_i}$. The sub ranker $\mathcal{R}_{I_i}$ has an AUC value of auc_i. This gives the following formula of exact AUC (using the notation of equation 1)

$$AUC = \frac{1}{P_b N_b} \sum_{i=1}^{b} (auc_i n_i p_i + n_i P_{i-1}) \qquad (5)$$

When comparing equations 1 and 5, one can see that $AucApprox$ supposes that $auc_i = \frac{1}{2}, \forall i \in \{1 \dots b\}$.

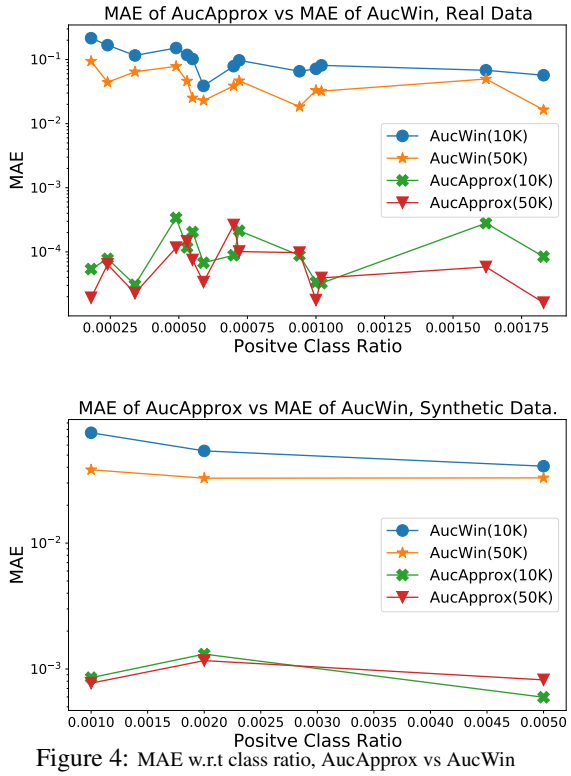


Figure 4: MAE w.r.t class ratio, AucApprox vs AucWin

7.1 Error Components

In a streaming context, two sources of error can be identified. First is the loss of information when summarizing data into a histogram. Second is the imperfections when building the histogram itself. These imperfections result when some points are assigned to the wrong bins. We call this phenomenon "miss-placement".

Formally, for a partition $\mathcal{P}(I) = J_1, ..., J_b$ (sorted increasingly w.r.t their centers) and their corresponding set of bins $B_1, ..., B_b$. Having miss-placed points means having a non empty set M.

$$M = \{(x, x') \in \mathbb{R}^2 : x \in B_i, x' \in B_j, i < j, x > x'\}$$

To compute miss-placement error, the error resulted from the loss of information should be left out. For that, we need a histogram that do not loose information, i.e. each bin stores its entire bin set inside each bin. We call this a "complete histogram" $\mathcal{H}_c$. This way we can compute auc_i, i.e., the exact value of AUC for the bin set. Then, for such histogram, and using the equation 5, we get an approximation using exact values of auc_i By construction, the only source of error when using a complete histogram is the miss-placement. We define the miss-placement error E_m as follows,

$$E_m(\mathcal{H}) = AUC(\mathcal{H}_c) - AUC \tag{6}$$

where $AUC(\mathcal{H}_c)$ is computed using equation 5, and auc_i is computed exactly for the bin set of B_i.
How the points are miss-placed is specific to each sketching algorithm. This makes miss-placement error a good way to compare sketching algorithms in the context of AUC approximation. We inspect that in the next section 7.2.

Data	Error	BH2	DDS2	TD2
Synhtetic	MAE	0.116	**0.045**	0.137
	Miss	0.014	**0.000**	0.128
	Summ	0.102	0.045	**0.016**
Standard	MAE	0.068	0.148	**0.066**
	Miss	0.008	**0.000**	0.056
	Summ	0.060	0.148	**0.017**
Real	MAE	0.886	**0.020**	0.412
	Miss	0.047	**0.000**	0.360
	Summ	0.840	**0.020**	0.082

Table 6: MAE(%) Components

When leaving out E_m, what is left of the error is the error resulted from the loss of information. Because it is a result of data summarization, we call it the summarization error E_{sm}, and it is computed as follows,

$$E_{sm}(\mathcal{H}) = \widehat{AUC}(\mathcal{H}) - AUC(\mathcal{H}_c) \tag{7}$$

and the total error is the sum of both types of errors,

$$E(\mathcal{H}) = \widehat{AUC}(\mathcal{H}) - AUC = E_{sm}(\mathcal{H}) + E_m(\mathcal{H}) \tag{8}$$

7.2 Experimental Error Analysis

In this section we report error components in order to compare sketching algorithms in the context of AUC approximation $AucApprox$ ($b = 1500$). We use all datasets in section 6.2, we average error over time and over datasets.

Table 6 shows the average error components. Note that we report the mean absolute value of the error components, not the signed value, that is why components do not add to the total MAE. Notice that DDSketch has zero miss-placement error, which is theoretically expected given that the bins are always fully ordered. For TDigest, the miss-placement error is always dominant, which is also theoretically expected given the frequent changes in the bin-to-interval mapping. In view of the results, we can't make any clear assertion regarding BenHaim.

Finally, it seems that DDSketch is the most suitable histogram for AUC approximation since it provides the best approximation error, and it does not have miss-placement error, which makes it more predictable and easier to analyse. Futhermore, DDSketch is fully-mergeable, which is an important property that can be leveraged to bring the approximation some nice properties.

8 Conclusion

This paper introduces a new approach to approximate the AUC-ROC evaluation metric for Stream Learning. We have shown how we can adapt frequency-based sketching algorithms to summarize the entire model's prediction history and derive an accurate approximation of the AUC.

This approach exhibits the properties required in stream, namely, (i) independent of data distribution, (ii) independent of the learner, (iii) memory-bounded, (iv) can be updated in a constant time.

Experiments have shown that our approximation reports 2 significant digits after the decimal point. This is a achieved with 270 times smaller memory footprint on average. When compared to the window-based AUC approximation in a highly imbalanced data scenario, our method reports up to 1000 times more accurate results.

We analyzed the error of several sketching algorithms and showed that DDSketch with 1500 bins seems to be a robust combination.

In Future work, we plan to leverage the idea of mergeable sketches to introduce forgetting mechanisms to our method, and make it usable for drift detection.

References

[1] Pankaj K. Agarwal, Graham Cormode, Zengfeng Huang, Jeff M. Phillips, Zhewei Wei, and Ke Yi. Mergeable summaries. *ACM Transactions on Database Systems*, pages 38–26, 2013.

[2] Shivani Agarwal and Thore Graepel. Generalization bounds for the area under the roc curve. *Journal of Machine Learning Research*, 6:393–425, 2005.

[3] Yael Ben-Haim and Elad Tom-Tov. A streaming parallel decision tree algorithm. *Journal of Machine Learning Research*, 11:849–872, 2010.

[4] Albert Bifet, Ricard Gavaldà, Geoff Holmes, and Bernhard Pfahringer. *Machine Learning for Data Streams: with Practical Examples in MOA*. MIT Press, 2018.

[5] Albert Bifet, Geoff Holmes, Richard Kirkby, and Bernhard Pfahringer. Moa: Massive online analysis. *The Journal of Machine Learning Research*, 11:1601–1604, 2010.

[6] Remco Bouckaert. Efficient auc learning curve calculation. page 181–191. Australian conference on artificial intelligence., 2006.

[7] Dariusz Brzezinski and Jerzy Stefanowski. Prequential auc: properties of the area under the roc curve for data streams with concept drift. *Knowledge and Information Systems*, 52:531–562, 2017.

[8] Ted Dunning and Otmar Ertl. Computing extremely accurate quantiles using t-digests. *arXiv preprint*, 2019. arXiv:1902.04023.

[9] Metz Charles E. Basic principles of roc analysis. In *Seminars in nuclear medicine*, volume 8, pages 283–298. Elsevier, 1978.

[10] Peter A. Flach. The geometry of roc space: understanding machine learning metrics through roc isometrics. pages 194–201. International Conference on Machine Learning (ICML), 2003.

[11] Edward Gan, Jialin Ding, Kai Sheng Tai, Vatsal Sharan, and Peter Bailis. Moment-based quantile sketches for efficient high cardinality aggregation queries. page 1647–1660. PVLDB, 2018.

[12] Cormode Graham. Sketch techniques for approximate query processing. *Foundations and Trends in Databases. NOW publishers*, 2011.

[13] Gomes Heitor, Bifet Albert, Read Jesse, Jean Paul Barddal, Enembreck Fabrício, Pfharinger Bernhard, Holmes Geoff, and Abdessalem Talel. Adaptive random forests for evolving data stream classification. *Machine Learning*, 106(9):1469–1495, 2017.

[14] Alan Herschtal and Bhavani Raskutti. Optimising area under the ROC curve using gradient descent. In *International Conference on Machine Learning (ICML)*, volume 69 of *ACM International Conference Proceeding Series*. ACM, 2004.

[15] Gama Joao, Sebastiao Raquel, and Rodrigues Pedro Pereira. On evaluating stream learning algorithms. *Machine learning*, 90(3):317–346, 2013.

[16] Beck JR and Shultz EK. The use of relative operating characteristic (roc) curves in test performance evaluation. *Archives of pathology laboratory medicine*, 100(1):13–20, 1986.

[17] Charles Masson, Jee Rim, and Homin Lee. Ddsketch: A fast and fully-mergeable quantile sketch with relative-error guarantees. *PVLDB*, 12(12):2195–2205, Aug 2019.

[18] Jacob Montiel, Jesse Read, Albert Bifet, and Talel Abdessalem. Scikit-multiflow: A multi-output streaming framework. *Journal of Machine Learning Research*, 19(72):1–5, 2018.

[19] Foster Provost and Tom Fawcett. Analysis and visualization of classifier performance with nonuniform class and cost distributions. In *Proceedings of AAAI-97 Workshop on AI Approaches to Fraud Detection & Risk Management*, pages 57–63, 1997.

[20] Nikolaj Tatti. Efficient estimation of auc in a sliding window. pages 671–686. ECML PKDD: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2018.

[21] Fawcett Tom. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[22] Shaomin Wu, Peter Flach, and Cesar Ferri. An improved model selection heuristic for auc. In *European Conference on Machine Learning*, pages 478–489. Springer, 2007.

[23] Charles X.Ling, Jin Huang, and Harry Zhang. Auc: a statistically consistent and more discriminating measure than accuracy. In *Ijcai*, volume 3, pages 519–524. Ijcai, 2003.

[24] Lian Yan, Robert Dodier, Michael C. Mozer, and Richard Wolniewicz. Optimizing classifier performance via an approximation to the wilcoxon-mann-whitney statistic. pages 848–855. International Conference on Machine learning (ICML), 2003.

Génération aléatoire d'un graphe spatio-temporel localement cohérent

Aurélie Leborgne, Marija Kirandjiska, Florence Le Ber

Université de Strasbourg, CNRS, ENGEES, ICube UMR 7357, F-67000 Strasbourg
aurelie.leborgne@unistra.fr, florence.leber@engees.unistra.fr

Résumé

Dans cet article, nous présentons une approche pour générer des graphes spatio-temporels localement cohérents et incluant des motifs fréquents inexacts. Cette approche s'appuie sur un travail précédent, qui consistait à développer un générateur paramétrable de graphes spatio-temporels. Dans ces graphes, les arêtes spatiales et spatio-temporelles sont étiquetées avec les relations spatiales de la théorie RCC8. Pour vérifier la cohérence des relations au cours de la génération du graphe, nous utilisons la méthode de la chemin-cohérence, qui s'appuie sur la faible composition des relations. L'approche est décrite et des expérimentations sont présentées. L'objectif final de notre travail est de disposer d'un générateur pour tester les méthodes de fouille de graphes.

Mots-clés

Graphe spatio-temporel, RCC8, cohérence, génération de graphes.

Abstract

This paper presents an original approach for the generation of locally coherent spatio-temporal graphs embedding frequent inexact patterns. This approach relies on a previous work, that have implemented a configurable generator for spatio-temporal graphs. These graphs contains spatial an spatio-temporal edges that are labelled with the RCC8 topological relations. In order to check the consistency of these relations when building the graph, the path-consistency method, based on relation weak composition, was implemented within the graph generator. The approach is described and some experiments are detailed. Our final aim is to build a test generator for graph mining methods.

Keywords

Spatio-temporal graph, RCC8, consistency, graph generation.

1 Introduction

L'amélioration régulière des outils et techniques de recueil des données amènent de plus en plus souvent à modéliser et analyser des données qui ont une dimension spatiale mais également temporelle. Une manière naturelle de modéliser de telles données est d'utiliser les graphes spatio-temporels (graphes ST), qui permettent de représenter différents phénomènes naturels (évolutions des occupations du sol dans un territoire [11], le déplacement de dunes [7], *etc.*) ou biologiques (évolution de la connectivité cérébrale [10], *etc.*). Une manière d'analyser les données recueillies est de s'intéresser aux phénomènes récurrents dans le temps et/ou l'espace (ensembles d'entités ayant des relations spatiales et évoluant les unes par rapport aux autres dans le temps). Il faut alors mettre en évidence les régularités dans les phénomènes spatio-temporels étudiés pour faciliter la compréhension des experts des différents domaines d'application. À titre d'exemple, en urbanisme, un même schéma d'artificialisation se répète selon des temporalités variées : à court terme, une zone végétalisée (forêt, parcelle agricole, etc.) est rasée progressivement, remplacée par du sol nu, puis des infrastructures apparaissent (routes, ponts, etc. ainsi que des maisons individuelles. La figure 1 repésente un exemple d'une telle situation. À plus long terme, cette zone se restructure : les maisons devenues anciennes disparaissent au profit d'immeubles collectifs. Cette évolution peut être alors modélisée par un motif spatio-temporel caractéristique d'une urbanisation.

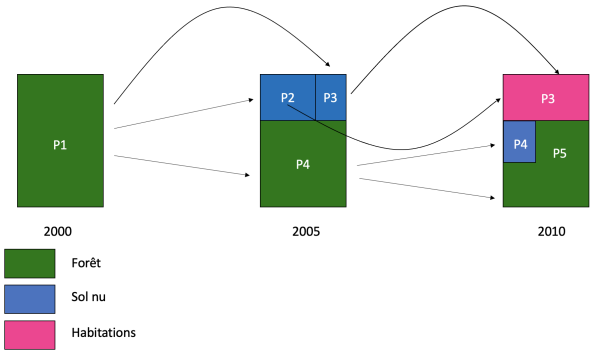


FIGURE 1 – Evolution des occupations du sol d'un territoire

Pour réaliser une telle extraction, il est nécessaire de mettre au point des algorithmes de recherche de motifs/sous-graphes fréquents dans des graphes ST. Cependant, pour développer de tels algorithmes, il est indispensable d'avoir une base de test de graphes ST annotés. Malheureusement, l'annotation de telles données est un travail très fastidieux et même impossible à réaliser précisément au vu de la quantité de données que nous manipulons. La seule solution per-

mettant d'obtenir une base de test est donc de développer un générateur de graphes spatio-temporels dans lesquels nous maitrisons les motifs fréquents présents dans ces graphes ST.
La génération de tels graphes a été décrite dans [12]. Dans cet article, nous nous intéressons à la manière de gérer la cohérence des graphes pour l'ensemble des relations considérées, à savoir, les relations spatiales qualitatives de la théorie RCC8 [14]. Ces relations permettent notamment de modéliser les évolutions des occupations du sol sur un territoire, comme dans l'exemple présenté ci-dessus.
L'article est organisé comme suit. La section 2 décrit les éléments théoriques sur lesquels notre approche est fondée, à savoir le modèle de graphes spatio-temporels, les relations RCC8 puis les réseaux de contraintes qualitatives. Les sections 3 et 4 présentent l'algorithme de génération puis les expérimentations menées. La section conclusive dresse quelques perspectives de ce travail.

2 Préliminaires

2.1 Simulation de graphes

La génération de graphes est une question importante dans de nombreux domaines, pour simuler des graphes réels, tester des algorithmes ou des applications permettant d'analyser, visualiser ou transformer des données [2]. Dans la plupart des cas, le but est de générer des graphes réalistes. De nombreux modèles ont été présentés en ce sens pour la génération de graphes complexes adaptés à la représentation de systèmes naturels ou humains (web sémantique, réseaux sociaux). Le modèle Barabási-Albert [8, 3] est un des plus connus. Ces approches sont fondées sur des distributions statistiques de propriétés des graphes (nombre de sommets, arêtes, degré des sommets, *etc.*). La question de la cohérence n'est pas traitée, dans la mesure où la sémantique des relations est peu prise en compte dans ces approches.
Dans [12], nous avons développé une approche permettant de générer, de manière aléatoire, des graphes sémantico-temporels, où les arêtes sont dotées d'une sémantique. Nous introduisons ci-dessous le modèle d'un tel graphe, inspiré de [7]. Nous utilisons pour exemple le graphe (figure 2) associé au schéma d'évolution de la figure 1. Il s'agit ici d'un graphe spatio-temporel, défini comme l'union de trois sous-graphes :

— **Le graphe des relations spatiales**, qui caractérise spatialement les interactions entre entités (dans l'exemple de la figure 1, les occupations du sol d'un territoire) à un moment donné. Il est composé des nœuds (en noir), et des arêtes pleines verticales (en vert) sur la figure 2.
— **Le graphe des relations spatio-temporelles**, qui se base sur les mêmes caractéristiques que le graphe des relations spatiales, mais en considérant des entités à des temps différents. Il est composé des nœuds (en noir), et des arêtes pleines (en rouge) sur la figure 2.
— **Le graphe des relations de filiation**, défini sur le concept d'identité. Il permet de caractériser la transmission de l'identité des entités à travers le temps. Il est composé des nœuds (en noir), et des arêtes pointillées (en bleu) sur la figure 2.

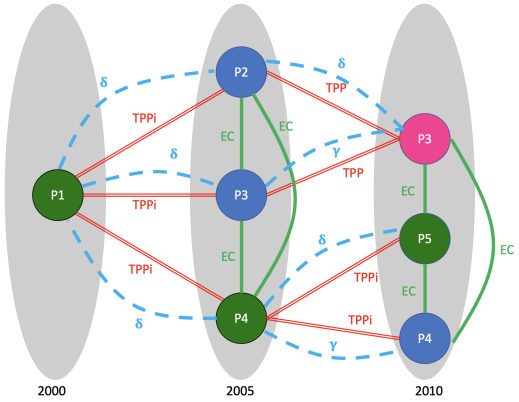


FIGURE 2 – Graphe ST modélisant l'évolution du territoire de la figure 1

De manière formelle, un graphe ST est défini ainsi. Soit $\mathcal{T} = \{t_1, t_2, ...t_n\}$, un domaine temporel, où t_i représente une instance de temps d'une granularité donnée et $t_i < t_{i+1}$ pour tout $i \in [1, n]$. Soit Δ un ensemble d'entités, $\{e_1, e_2, ..., e_m\}$. Soit également Σ, un ensemble de relations spatiales, et Φ, un ensemble de relations de filiation. Un graphe spatio-temporel $\mathcal{G}$ est un tuple (U, E_Σ, E_Φ), où U est l'ensemble des sommets $(e_l, t_i) \in \Delta \times \mathcal{T}$, E_Σ est l'ensemble des tuples $((e_i, t_i)T(e_j, t_j))$ où $(e_i, t_i), (e_j, t_j) \in U$, $t_i \leq t_j \leq t_{i+1}$, et $T \in \Sigma$, et E_Φ est l'ensemble des tuples $((e_i, t_i)\rho(e_j, t_{i+1}))$ où $(e_i, t_i), (e_j, t_{i+1}) \in U$, et $\rho \in \Phi$.
A l'origine, ce modèle de graphe a été introduit pour représenter l'évolution d'entités géographiques [7]. D'autres relations de filiation sont étudiées dans [6] ; parallèlement un algorithme a été proposé pour vérifier la cohérence de cette représentation avec les caractéristiques des données issues d'une base de données spatio-temporelles.

2.2 Relations RCC8

Les relations spatiales et spatio-temporelles que nous utilisons sont les relations de base $\mathbf{B}$ de la théorie RCC8 [14] sur le domaine spatial Δ. Ces relations définissent la position de deux régions : $DC(x, y)$ les régions x et y sont déconnectées ; $EC(x, y)$ elles sont connectées exterieurement ; $PO(x, y)$ elles se recouvrent partiellement ; $TPP(x, y)$ x est partie propre tangentielle de y ; $TPP_i(x, y)$ y est partie propre tangentielle de x ; $NTPP(x, y)$ x est partie propre non tangentielle de y ; $NTPP_i(x, y)$ y est partie propre non tangentielle de x ; $EQ(x, y)$ égalité de x et y (voir figure 3).
L'ensemble $2^{\mathbf{B}}$ représente l'ensemble de relations construit à partir des relations de base. Il est muni des opérations ensemblistes usuelles, l'union et l'intersection, de l'opération inverse et de la composition faible. Une relation de $2^{\mathbf{B}}$ s'écrit donc comme une union de relations de base, par exemple $R = \{DC, EC\}$ et s'interprète comme une

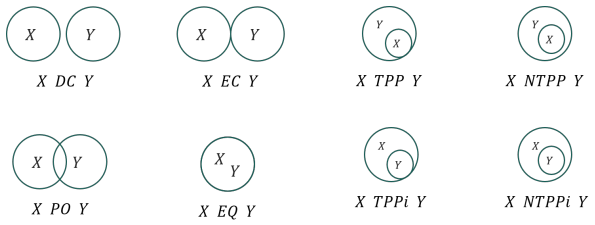


FIGURE 3 – Les relations de base de la théorie RCC8

disjonction. L'inverse (noté $\smile$) d'une relation est l'union des inverses de ses relations de base. La composition faible est notée $\diamond$ et définie ainsi : soient R et S deux relations de $2^{\mathbf{B}}$, $R \diamond S = \{b \in \mathbf{B} | b \cap (R \circ S) \neq \emptyset\}$ où $R \circ S = \{(x,z) \in \Delta^2 | \exists y \in \Delta, (x,y) \in R \,\mathtt{et}\, (y,z) \in S\}$. La composition faible des relations de base est représentée dans une table de composition [13], comme le montre la figure 7. Par exemple, supposons que trois régions x, y, z ont les relations $TPP(x,y)$ et $EC(y,z)$, on vérifie alors que $\{DC, EC\}(x,z)$ (voir figure 4).

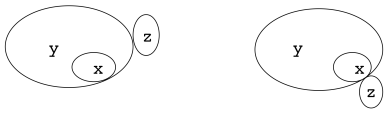


FIGURE 4 – Deux configurations possibles pour x et z connaissant $TPP(x,y)$ et $EC(y,z)$

2.3 Cohérence

La notion de cohérence d'un ensemble de relations spatiales ou temporelles reliant des régions a été étudiée dans le cadre des réseaux de contraintes qualitatives [4, 5]. Un réseau de contraintes qualitatives est un couple $N = (V, C)$ où V est un ensemble de variables sur un domaine continu $\mathcal{D}$ et C une application qui associe à chaque couple de variables (V_i, V_j) un ensemble C_{ij} de relations de base $\{r_1, ..., r_l\}$ prises dans une algèbre de relations. Cet ensemble représente la disjonction des relations possibles entre les deux entités représentées par les variables V_i et V_j. Une instanciation cohérente de N est une instanciation où chaque variable V_i prend une valeur $e_i \in \mathcal{D}$ telle que pour tout couple (V_i, V_j) la relation atomique vérifiée par les variables V_i et V_j appartienne à C_{ij}.

La vérification de la cohérence d'un réseau est un problème NP-complet dans le cas général [17]. Des méthodes locales ont été proposées pour vérifier des formes plus faibles de cohérence, dont la chemin-cohérence : un réseau de contraintes qualitatives N est dit chemin cohérent si pour toutes les variables $V_i, V_j, V_k \in V$, $C_{ij} \subseteq C_{ik} \circ C_{kj}$ [5].

La méthode de la chemin-cohérence consiste à réaliser l'opération de triangulation : $C_{ij} = C_{ij} \cap (C_{ik} \circ C_{kj})$ pour tout triplet jusqu'à obtention d'un point fixe. Le réseau final est chemin-cohérent et équivalent au réseau initial. Dans le cadre de RCC8, où on ne peut pas utiliser la composition, mais la composition faible $\diamond$, on parle de fermeture algébrique [15].

Différents algorithmes ont été proposés pour vérifier la chemin-cohérence. Leur temps de calcul est lié au nombre d'accès à la table de composition et donc à la taille de l'ensemble de relations $2^{\mathbf{B}}$ [1]. Dans [16] est présenté un algorithme permettant de vérifier de manière incrémentale la cohérence d'un réseau de contraintes qualitatives, qui croît par ajout d'entités spatiales ou temporelles ; l'algorithme exploite un graphe triangulé : quand une entité est ajoutée à l'étape t, elle est reliée à toutes les entités présentes à l'étape $t-1$ (triangulation). La méthode de la chemin-cohérence est ensuite appliquée sur le graphe. Notre approche est différente dans la mesure où nous cherchons à générer un graphe où chaque relation est atomique, et où il s'agit d'affecter à l'arête courante une relation qui soit cohérente avec l'existant, comme nous le détaillons ci-après.

3 Génération de graphes ST localement cohérents

Nous présentons ici une approche pour générer des graphes spatio-temporels localement cohérents et incluant des motifs (sous-graphes) fréquents inexacts. Nous nous appuyons pour cela sur la méthode présentée dans [12], qui permet de générer de tels graphes, incluant des motifs, mais sans traiter la question de la cohérence.

Nous décrivons ici succinctement l'algorithme de génération des graphes spatio-temporels et des motifs qui y sont intégrés, puis détaillons et justifions l'algorithme permettant de vérifier la cohérence locale des relations spatiales et spatio-temporelles générées. Notons que ces deux types de relations sont issues de l'ensemble $2^{\mathbf{B}}$ (voir en section 2.2).

L'algorithme décrit dans [12] permet de simuler des graphes spatio-temporels composés en partie de motifs *noyés* dans une génération stochastique uniforme de nœuds et d'arêtes. Il s'agit d'un algorithme entièrement paramétrable, exploitant la loi de Poisson, comme proposé dans [9], et dans lequel il est possible de choisir la taille du graphe généré, le nombre de relations spatiales, temporelles et filiation par nœud, ainsi que la taille des motifs-source à incruster et le nombre de leurs transformations. Dans cet article, nous considèrons un algorithme modifié pour contrôler la part des motifs, en nombre de nœuds par rapport au graphe total. Les autres paramètres sont recensés dans le tableau 1.

Cet algorithme se déroule en trois étapes principales. Il calcule au départ le nombre de nœuds total (paramètre λ_n, tableau 1) du graphe à générer.

Étape 1 : génération et transformation de motifs-sources, selon des paramètres propres (voir ci-dessous) ; chaque motif est affecté à une temporalité du graphe, cette information est stockée dans le paramètre $patterns$.

Étape 2 : génération aléatoire des nœuds pour chaque temporalité dans le graphe (paramètres λ_r et $labels_n$). Le nombre de nœuds est ajusté en fonction du nombre de nœuds des motifs affectés à la temporalité courante.

Étape 3 : génération aléatoire des relations entre les nœuds (paramètres Λ_e et $labels_e$). Les nœuds de la temporalité courante sont reliés entre eux et avec les nœuds de la tem-

Paramètre	Description
λ_n	Espérance de la loi de Poisson zéro-tronquée pour le nombre total de nœuds dans le graphe
λ_r	Espérance de la loi de Poisson zéro-tronquée pour le nombre de nœuds par temporalité
Λ_e	Triplet des espérances des lois de Poisson pour le nombre de relations spatiales / spatio-temporelles / de filiation par nœud
$labels_n$	Liste des étiquettes disponibles pour les nœuds
$labels_e$	Tableau de taille 3 de listes d'étiquettes disponibles pour chaque type de relation
$patterns$	Une liste de tuples dont chaque tuple est composé d'un numéro de temporalité et du motif à insérer à cette temporalité

TABLE 1 – Paramètres pour la génération de graphes spatio-temporels

poralité précédente. Le nombre d'arêtes est ajusté pour les nœuds des motifs (s'ils en possèdent déjà). Chaque arête est étiquetée avec une relation atomique.
Lors de la première étape, la génération de motifs obéit aux paramètres suivants (voir tableau 2) : le nombre de motifs insérés dépend d'une proportion p (en pourcentage du nombre de nœuds) que ces motifs doivent représenter dans le graphe total. Le nombre de nœuds dans un motif-source est tiré au hasard dans un intervalle ($pnodes$). Les paramètres λ_r et λ_e ont le même rôle que pour le graphe complet. Chaque motif-source est répété selon une valeur (support) tirée aléatoirement dans l'intervalle $support$. Finalement chaque répétition donne lieu à des transformations (paramètre λ_t) de sorte à introduire une variation autour de chaque motif-source.
La complexité théorique de cet algorithme est en moyenne $O(\lambda_n \times \lambda_r)$, chaque nœud du graphe étant potentiellement relié à tous les nœuds de sa temporalité et de la temporalité précédente. Au pire, quand le nombre de temporalités diminue, la complexité tend vers $O(\lambda_n^2)$ [12].
L'objectif du travail présenté ici est de générer un graphe localement cohérent fondé sur le modèle de cohérence présenté dans la section 2.3. Plus précisément, cela consiste à générer des relations spatiales et spatio-temporelles tout en s'assurant qu'elles forment des triangles spatiaux ou temporels cohérents avec les arêtes préexistantes dans le graphe.

Définition 1 *Un **triangle cohérent** est une clique de 3 sommets dans laquelle les trois relations modélisées par des arêtes sont cohérentes entre elles, soit, pour e_i, e_j, e_k les sommets d'un tel triangle, $R_{ik} \subseteq R_{ij} \diamond R_{jk}$ et $R_{ij} \subseteq R_{ik} \diamond R_{kj}$.*

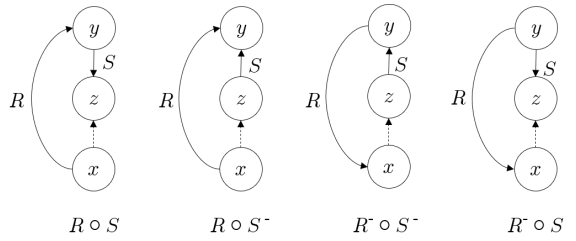
Définition 2 *Un sous-graphe constitué de trois nœuds x, y, z qui se trouvent dans la même temporalité est appelé un **triangle spatial**. Quatre configurations sont à considérer pour la composition, selon le sens des arêtes xy d'une part et yz d'autre part (figure 5a).*

Paramètre	Description
p	Proportion de nœuds dans les motifs / nombre dans le graphe
$pnodes$	Intervalle pour le nombre de nœuds dans un motif-source
λ_r	Espérance de la loi de Poisson zéro-tronquée pour le nombre de nœuds par temporalité du motif-source
Λ_e	Triplet des espérances des lois de Poisson pour le nombre de relations spatiales / spatio-temporelles / de filiation par nœud du motif-source
$support$	Intervalle pour le nombre de répétitions d'un motif-source
λ_t	Espérance de la loi de Poisson pour le nombre de transformations à effectuer sur un motif-source

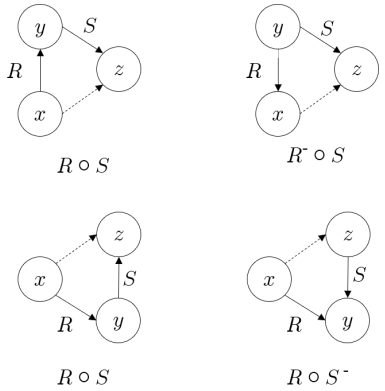
TABLE 2 – Paramètres pour la génération de motifs

Définition 3 *Un sous-graphe constitué de trois nœuds x, y, z qui se trouvent dans deux temporalités voisines est appelé **triangle spatio-temporel**. Quatre configurations sont à considérer pour la composition, selon le sens des arêtes xy d'une part et yz d'autre part (figure 5b).*

Le choix de se limiter à une cohérence locale (3-cohérence) est liée à des aspects pratiques : d'une part, les motifs inscrustés sont de petite taille (3-4 nœuds par temporalité, 2 ou 3 temporalités au plus) ; d'autre part, le nombre de relations par nœud est généralement faible (même si nous faisons des expérimentations avec des densités élevées, voir section 4). Finalement, nous cherchons à limiter la complexité de génération des graphes ST.
L'algorithme 1 décrit cette méthode. Une phase d'initialisation (l. 1) permet de créer une liste L avec l'ensemble des relations de **B**. Pour déterminer une relation entre les nœuds x et z, on recherche tous les nœuds y, qui ont une relation à la fois avec x et avec z (l. 2). Les différentes configurations (figure 5) sont étudiées. Pour chaque nœud, la liste est mise à jour en ne conservant que les relations possibles de L (l. 4, 6, 8, 10). Finalement, la relation entre x et z est assignée au hasard parmi les relations de L (l. 14). Si la liste est vide, aucune relation n'est assignée.
La complexité théorique de l'algorithme 1 est $O(\lambda_r)$ puisque, étant donnée la paire (x, z), il parcourt au pire tous les nœuds des temporalités courante et précédente. En revanche, comme les relations sont atomiques, un seul accès à la table de composition est nécessaire pour traiter chaque triangle.
Un exemple de graphe obtenu est présenté en figure 6 : les nœuds des motifs sont désignés par la lettre P, tandis que les nœuds génériques ont seulement un numéro. Des nœuds portant le même numéro portent la même étiquette. Sur cet exemple, on voit à la temporalité t_4 que P2 est relié à P1



(a) Cas spatial



(b) Cas spatio-temporel

FIGURE 5 – Différentes configurations pour la composition des arêtes reliant les nœuds x et y d'une part, et y et z d'autre part, dans un triangle spatial ou spatio-temporel.

par une arête spatiale $NTPP$, P2 est relié à P3 (t_5) par une arête spatio-temporelle DC, P1 et P3 sont reliés à P6 (t_5) respectivement par des arêtes portant les étiquettes PO et EC. Pour relier P1 et P3, l'algorithme doit donc examiner successivement les triangles (P1,P2,P3) et (P1,P6,P3) :

— Pour (P1,P2,P3), on a $NTPP^{\smile} \diamond DC = NTTP_i \diamond DC = \{DC, EC, PO, TPP_i, NTPP_i\}$, comme le montre la case bleue de la table de composition sur la figure 7.

— Pour (P1,P6,P3) on a $PO \diamond EC^{\smile} = PO \diamond EC = \{DC, EC, PO, TPP_i, NTPP_i\}$, comme le montre la case jaune de la table de composition sur la figure 7.

Finalement la relation entre P1 et P3 doit être choisie dans l'ensemble $\{DC, EC, PO, TPP_i, NTPP_i\}$, ici c'est TPP_i qui a été sélectionnée.

4 Expérimentations

Dans cette phase expérimentale nous avons étudié le comportement, en terme de temps de calcul, de l'algorithme de génération en faisant varier les différents paramètres (voir tableaux 1 et 2). Afin d'observer l'influence des différents paramètres sur la complexité de la génération des graphes spatio-temporels cohérents, nous avons fait varier ces paramètres un par un. Les valeurs par défaut des paramètres sont présentées au tableau 3.

Cette batterie de tests a été effectuée sur une machine

Algorithm 1 Génération d'une relation entre deux nœuds avec vérification de la cohérence locale

Input: sommets x, z
Output: relation entre x et z

$L = \mathbf{B}$
for each sommet y tel que (x,y) et (y,z) $\in E_\Sigma$ **do**
  if $R(x, y)$ et $S(y, z)$ **then**
    $L \leftarrow L \cap R \diamond S$
  else if $R(x, y)$ et $S(z, y)$ **then**
    $L \leftarrow L \cap R \diamond S^{\smile}$
  else if $R(y, x)$ et $S(z, y)$ **then**
    $L \leftarrow L \cap R^{\smile} \diamond S^{\smile}$
  else
    $L \leftarrow L \cap R^{\smile} \diamond S$
  end if
end for
if $L \neq \emptyset$ **then**
  return relation au hasard dans L
else
  return pas de relation
end if

Paramètre	Valeur
Génération du graphe	
λ_n	10000
λ_r	100
Λ_e	[5,5,2]
Génération de motifs	
p	30
$pnodes$	[5,15]
λ_r	2
Λ_e	[5,5,2]
$support$	[10,20]
λ_t	taille moyenne des motifs / 2

TABLE 3 – Valeurs par défaut des paramètres lors des expérimentations

Ubuntu 18.04.4 LTS, 32 Go de RAM et 32 cœurs. Cependant l'algorithme n'a utilisé qu'un de ces cœurs.

4.1 Variation du nombre de nœuds

Dans cette première expérience, seul varie le paramètre λ_n (de 10000 à 10^6), qui règle le nombre total de nœuds dans le graphe. Le paramètre λ_r est fixé de sorte que le nombre de temporalités $\mathcal{T}$ ne change pas (le nombre de nœuds par temporalité varie proportionnellement au nombre de nœuds total, $\lambda_r = \lambda_n/100$) La figure 8 montre que le temps de génération des graphes croît exponentiellement avec le nombre total de nœuds et donc avec le nombre de nœuds par temporalité. De fait, pour chaque nœud créé, l'algorithme doit parcourir les nœuds de sa temporalité et de la précédente pour établir les relations : comme λ_r est le nombre moyen de nœuds par temporalité il y a donc en moyenne au plus $2\lambda_r \times \lambda_n \approx \lambda_n^2$ opérations, puisque $\mathcal{T}$ est fixé.

Dans un deuxième temps, afin d'examiner l'influence du nombre de nœuds par temporalité, nous avons fixé le nombre de nœuds total et avons fait varier le nombre de nœuds par temporalité (de 100 à 2000). Cette expérience

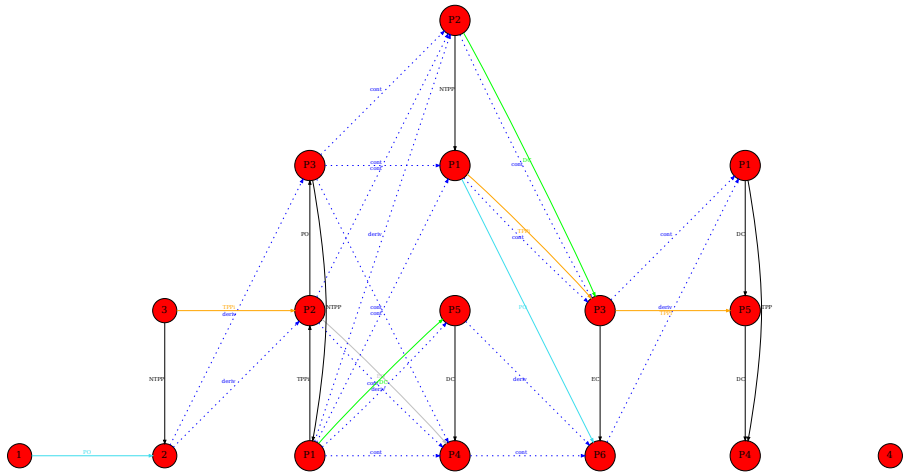


FIGURE 6 – Un graphe spatio-temporel cohérent incluant des motifs : les arêtes spatiales sont représententées en noir, les arêtes spatio-temporelles en couleur, les arêtes de filiation en traits tiretés

montre qu'il y a une relation linéaire entre le temps de génération des graphes et le nombre de nœuds par temporalité (voir figure 9), pour un nombre total de nœuds fixé. Tout étant fixé par ailleurs, seul varie le nombre de nœuds à parcourir pour établir les relations dans une temporalité et avec la temporalité précédente. Comme ci-dessus, le nombre d'opérations est en moyenne $2\lambda_n \times \lambda_r$. Le paramètre λ_n étant fixé, le temps de calcul est donc proportionnel à λ_r.

4.2 Variation du nombre d'arêtes

Nous nous intéressons ici à l'influence de la génération des arêtes sur le temps de génération des graphes. Pour ce faire, nous avons fait varier le nombre de relations par nœud : Λ_e varie de [0,0,0] à [200,200,200]. Dans cette expérience, le nombre total de nœuds et de nœuds par temporalité sont fixés. La figure 10 illustre cette expérience. Il existe une relation linéaire entre le nombre de relations par nœud et le temps de génération du graphe complet jusqu'à arriver à un plateau aux alentours de 100 relations par nœud. La partie linéaire s'explique de cette manière : pour chaque nœud nous avons en moyenne $\lambda_e = \Lambda_e[1] + \Lambda_e[2] + \Lambda_e[3]$ créations d'arêtes et $\Lambda_e[1] + \Lambda_e[2]$ vérifications de contraintes à réaliser. Pour la totalité du graphe, nous avons donc $\lambda_n \times \lambda_e$ opérations, nombre qui croît linéairement avec λ_e, λ_n étant fixé. La partie constante est due à la saturation du graphe, c'est-à-dire que chaque nœud a atteint son nombre maximal d'arêtes. En effet, dans cette expérience, une temporalité est composée en moyenne de 100 nœuds, qui ont chacun au plus une seule relation de chaque type avec un nœud de même temporalité ou de la temporalité précédente. Notons que le temps de calcul associé à la vérification de la cohérence des relations n'est liée qu'au nombre de triplets à examiner, puisque chaque arête porte une relation atomique.

4.3 Variation du nombre de motifs

Dans cette dernière expérience, nous faisons varier le nombre de motifs insérés dans les graphes (ou plus exactement la proportion de nœuds provenant de motifs, réglée par le paramètre p (qui varie de 0 à 100%), à taille de graphe constante. La figure 11 montre qu'en augmentant cette proportion, le temps de génération des graphes augmente linéairement. Ceci s'explique ainsi : le temps de génération et de transformation des motifs-sources est constant (les paramètres $pnodes$, λ_r, Λ_e, $support$ et λ_t sont fixés), la taille des motifs est constante (paramètre $pnodes$), seul varie donc le nombre de motifs-sources à générer pour atteindre une proportion donnée de nœuds par rapport au nombre total de nœuds dans le graphe.

Finalement, la complexité expérimentale observée est du même ordre que la complexité théorique. Elle est liée à l'algorithme principal de génération des nœuds et des relations entre les nœuds. Le temps nécessaire à la vérification de la cohérence est négligeable au regard de l'algorithme principal.

5 Conclusion

Cet article présente une méthode permettant de générer des graphes spatio-temporels dont les arêtes spatiales et spatio-temporelles sont localement cohérentes. Pour ce faire, nous nous sommes appuyées sur un algorithme de génération de graphes spatio-temporels aléatoires existant [12]. Cet algorithme a été modifié dans le sens où à chaque ajout d'une arête (spatiale ou spatio-temporelle) entre deux nœuds, la cohérence de cette arête avec les arêtes préexistantes reliant ces deux nœuds à un même nœud est vérifiée. La méthode de la chemin-cohérence est utilisée pour cela, dans le cadre de la théorie RCC8.

La particularité de l'algorithme est d'insérer dans le graphe généré des motifs fréquents inexacts. Ces motifs sont gé-

$\diamond$	DC	EC	PO	TPP	$NTPP$	TPP_i	$NTPP_i$	EQ
DC	$DC, EC, PO, TPP, NTPP, TPP_i, NTPP_i, EQ$	$DC, EC, PO, TPP, NTPP$	$DC, EC, PO, TPP, NTPP$	$DC, EC, PO, TPP, NTPP$	$DC, EC, PO, TPP, NTPP$	DC	DC	DC
EC	$DC, EC, PO, TPP_i, NTPP_i$	$DC, EC, PO, TPP, TPP_i, EQ$	$DC, EC, PO, TPP, NTPP$	$EC, PO, TPP, NTPP$	$PO, TPP, NTPP$	DC, EC	DC	EC
PO	$DC, EC, PO, TPP_i, NTPP_i$	$DC, EC, PO, TPP_i, NTPP_i$	$DC, EC, PO, TPP, NTPP, TPP_i, NTPP_i, EQ$	$PO, TPP, NTPP$	$PO, TPP, NTPP$	$DC, EC, PO, TPP_i, NTPP_i$	$DC, EC, PO, TPP_i, NTPP_i$	PO
TPP	DC	DC, EC	$DC, EC, PO, TPP, NTPP$	$TPP, NTPP$	$NTPP$	$DC, EC, PO, TPP, TPP_i, EQ$	$DC, EC, PO, TPP_i, NTPP_i$	TPP
$NTPP$	DC	DC	$DC, EC, PO, TPP, NTPP$	$NTPP$	$NTPP$	$DC, EC, PO, TPP, NTPP$	$DC, EC, PO, TPP, NTPP, TPP_i, NTPP_i, EQ$	$NTPP$
TPP_i	$DC, EC, PO, TPP_i, NTPP_i$	$EC, PO, TPP_i, NTPP_i$	$PO, TPP_i, NTPP_i$	EQ, PO, TPP, TPP_i	$PO, TPP, NTPP$	$TPP_i, NTPP_i$	$NTPP_i$	TPP_i
$NTPP_i$	$DC, EC, PO, TPP_i, NTPP_i$	$PO, TPP_i, NTPP_i$	$PO, TPP_i, NTPP_i$	$PO, TPP_i, NTPP_i$	$PO, TPP, NTPP, EQ, TPP_i, NTPP_i$	$NTPP_i$	$NTPP_i$	$NTPP_i$
EQ	DC	EC	PO	PO	$NTPP$	TPP_i	$NTPP_i$	EQ

FIGURE 7 – Table de composition des relations de base de la théorie RCC8

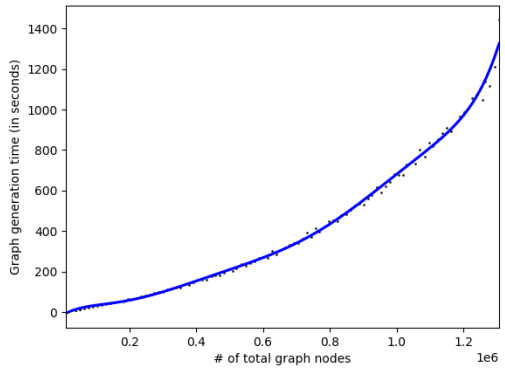


FIGURE 8 – Temps de génération des graphes en fonction du nombre de nœuds total

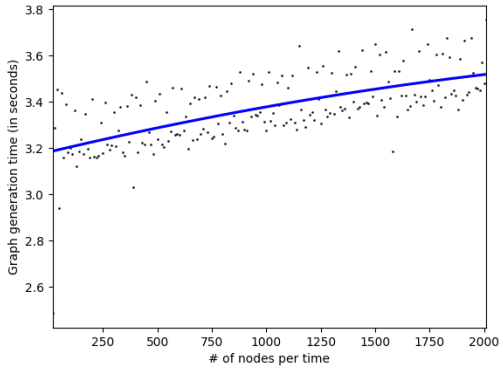


FIGURE 9 – Temps de génération d'un graphe en fonction du nombre de nœuds par temporalité

nérés et transformés en vérifiant également leur cohérence locale.

De plus, des tests de complexité ont été réalisés afin de mettre en évidence les étapes les plus coûteuses de la génération des graphes spatio-temporels. Les résultats de ces expérimentations se sont montrés cohérents avec la complexité théorique de l'algorithme. Une comparaison avec d'autres approches n'a pu être menée, car, à notre connaissance, il n'existe pas de méthode permettant de générer des graphes spatio-temprels comme nous le proposons.

Dans la suite du travail, une chemin-cohérence plus large pourra être développée, en s'inspirant des travaux de [16]. Au delà, nous voulons utiliser les graphes générés par notre approche pour tester diverses méthodes de recherche de motifs fréquents dans un graphe spatio-temporel. Pour comparer les résultats à ceux qui seraient obtenus sur une base de graphes réels, nous nous attacherons à paramétrer le générateur de sorte à produire des graphes similaires à des graphes réels, pour les applications que nous traitons, dans les domaines agricole et médical.

Remerciements

Ces travaux ont été réalisés dans le cadre du projet METEC-Graphe, financé par l'Idex de l'Université de Strasbourg.

Références

[1] C. Bessière. A simple way to improve path consistency processing in interval algebra networks. In *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI-96)*, pages 375–380, 1996.

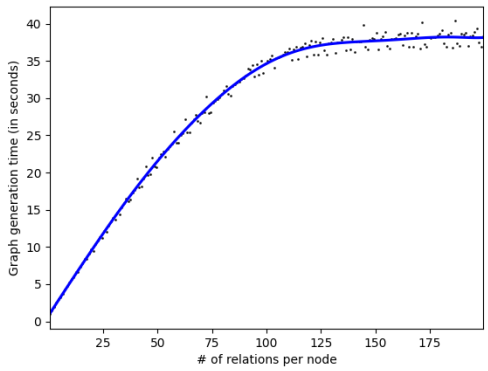


FIGURE 10 – Temps de génération des graphes en fonction du nombre de relations par nœud

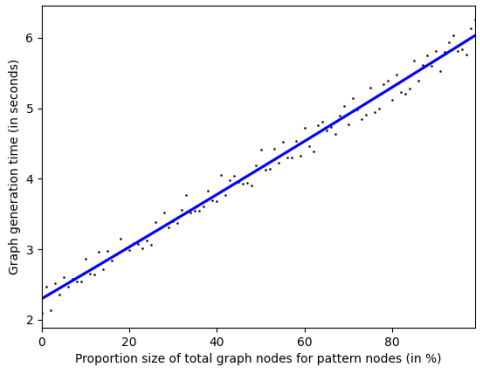


FIGURE 11 – Temps de génération des graphes en fonction de la proportion de motifs insérés (a) Paramètres (b) Courbe obtenue

[2] A. Bonifati, I. Holubová, A. Prat-Pérez, and S. Sakr. Graph generators : State of the art and open challenges. *ACM Comput. Surv.*, 53(2), April 2020.

[3] C. Campbell, K. Shea, and R. Albert. Comment on control profiles of complex networks. *Science*, 346(6209) :561–561, 2014.

[4] J.-F. Condotta. Problèmes de satisfaction de contraintes : algorithmes et complexité. Thèse de l'Université Toulouse 3, 2000.

[5] J-F. Condotta and Würbel E. Réseaux de contraintes temporelles et spatiales. In F. Le Ber, G. Ligozat, and O. Papini, editors, *Raisonnements sur l'espace et le temps*, chapter 7, pages 181–223. Hermès, 2007.

[6] G. Del Mondo, M. A. Rodríguez, C. Claramunt, L. Bravo, and R. Thibaud. Modeling consistency of spatio-temporal graphs. *Data & Knowledge Engineering*, 84 :59–80, 2013.

[7] G. Del Mondo, J. G. Stell, C. Claramunt, and R. Thibaud. A graph model for spatio-temporal evolution. *Journal of Universal Computer Science*, 16 :1452–1477, 2010.

[8] R. Ferrer i Cancho and R.V. Solé. Optimization in complex networks. In *Statistical mechanics of complex networks*, pages 114–126. 2003.

[9] M. Kuramochi and G. Karypis. Frequent subgraph discovery. In *Proceedings 2001 IEEE Int Conference on Data Mining*, pages 313–320, 2001.

[10] A. Leborgne, F. Le Ber, D. Niezgoda, C. Meillier, and S. Marc-Zwecker. Utilisation des graphes pour la représentation spatio-temporelle lors d'un examen d'irm fonctionnelle cérébrale. In *Journée Santé & IA 2020 (dans le cadre PFIA)*, Jun 2020.

[11] A. Leborgne, A. Meyer, H. Giraud, F. Le Ber, and S. Marc-Zwecker. Un graphe spatio-temporel pour modéliser l'évolution de parcelles agricoles. In *SAGEO*, Clermont-Ferrand, France, November 2019.

[12] A. Leborgne, J. Nuss, F. Le Ber, and S. Marc-Zwecker. An approach for generating random temporal semantic graphs with embedded patterns. In *Graph Embedding and Mining, ECML-PKDD 2020 Workshop Proc.*, 2020.

[13] D. A. Randell, A. G. Cohn, and Z. Cui. Computing transitivity tables : A challenge for automated theorem provers. In D. Kapur, editor, *Automated Deduction—CADE-11*, pages 786–790. Springer Berlin Heidelberg, 1992.

[14] D. A. Randell, Z. Cui, and A. G. Cohn. A Spatial Logic based on Regions and Connection. In *Proceedings 3rd International Conference on Knowledge Representation and Reasoning*, pages 165–176. Morgan Kaufmann Publishers, 1992.

[15] J. Renz and G. Ligozat. Weak composition for qualitative spatial and temporal reasoning. In Peter van Beek, editor, *Principles and Practice of Constraint Programming - CP 2005*, pages 534–548. Springer Berlin Heidelberg, 2005.

[16] M. Sioutis and J.-F. Condotta. Incrementally building partially path consistent qualitative constraint networks. In *AIMSA 2014 : Artificial Intelligence : Methodology, Systems, and Application*, pages 104–116, 09 2014.

[17] M. Vilain, H. Kautz, and P. V. Beek. Constraint propagation algorithms for temporal reasoning : A revised report. In D. S. Weld and J. De Kleer, editors, *Readings on Qualitative Reasoning about Physical Systems*, pages 373–381. Morgan Kaufmann, 1989.

Réseaux de Neurones Convolutifs pour la Caractérisation d'Anomalies Magnétiques

J. Cárdenas Chapellín[1], C. Denis[2], H. Mousannif[3], C. Camerlynck[4], N. Florsch[1]

[1] Sorbonne Université, UMMISCO
[2] Sorbonne Université, LIP6
[3] Université Cadi Ayyad, LISI
[4] Sorbonne Université, METIS

julio.cardenas_chapellin@sorbonne-universite.fr

Résumé

Cette contribution présente l'utilisation des réseaux de neurones convolutifs pour la détection d'anomalies magnétiques. L'approche développée permet la localisation de dipôles magnétiques, avec le comptage du nombre de dipôles, leur position géographique et la prédiction de leurs paramètres (moment magnétique, profondeur et déclinaison). Elle sera ensuite testée sur des données réelles, dans le cadre par exemple, d'une détection pyrotechnique pour la prospection de munitions non explosées, avant d'envisager une application vers d'autres méthodes géophysiques.

Mots-clés

Apprentissage profond, réseaux de neurones convolutifs, géophysique, méthodes magnétiques

Abstract

This contribution introduces the use of convolutional neural networks for the characterization of magnetic anomalies. The developed approach allows one the localization of magnetic dipoles, including counting the number of dipoles, their geographical position, and the prediction of their parameters (magnetic moment, depth, and declination). Subsequently, it will be tested on real data, for example, in the framework of pyrotechnic detection for unexploded ordnance prospection. Finally, an application to other geophysical methods will be considered.

Keywords

Deep learning, convolutional neural networks, geophysics, magnetic methods

1 Introduction

Les premiers réseaux de neurones ont été développés dans les années 1950. Cependant, jusque dans les années 1980, il n'existait pas à la fois une puissance de calcul et des algorithmes efficaces pour prendre en compte des topologies de réseaux de neurones plus complexes, permettant d'améliorer leurs capacités de prédiction. L'algorithme de rétropropagation du gradient couplé à une plus forte puissance de calcul a permis d'obtenir des résultats spectaculaires dans le domaine de la reconnaissance des formes et de la perception.

De nombreuses disciplines scientifiques orientent également leurs activités de recherche vers l'apprentissage machine profond. C'est aussi le cas de la géophysique pour faciliter le traitement automatisé des données géophysiques et la résolution des problèmes d'inversion : par exemple pour l'identification des trains d'ondes ou le filtrage du bruit sismique [12], ou pour obtenir une estimation de l'épaisseur globale de la croûte terrestre [11].

Pendant la dernière décennie, les méthodes d'apprentissage profond se sont révélées très prometteuses, notamment dans le domaine de l'interprétation sismique, ou en mettant en évidence des failles dans les sections sismiques par la génération d'un attribut de probabilité [10], ou encore avec la prédiction par un réseau neuronal convolutif d'un modèle élastique du sous-sol directement à partir des données sismiques enregistrées [3].

L'inversion des données magnétiques par apprentissage machine profond est un sujet qui n'a été abordé que très récemment. Des modèles d'apprentissage profond ont récemment été utilisés pour réaliser l'inversion de structures magnétiques 3D [6]. Pour cela, un jeu de données contenant des millions de modèles géologiques annotés pour chaque structure géologique a été généré pour alimenter les réseaux de neurones convolutifs. Leurs modèles de réseaux de neurones permettent de classifier et de prédire les paramètres d'une structure géologique présente dans les cartes magnétiques. Cependant, cette approche est limitée à l'analyse d'une seule configuration et, le fait d'analyser l'ensemble des anomalies plutôt que chacune d'entre elles séparément, multiplie le nombre de cas à traiter et donc le temps de prédiction.

Le projet du présent article vise à caractériser des données magnétiques en comptant le nombre d'anomalies magnétiques dipolaires puis de prédire leurs positions (x,y) respectives ainsi que les paramètres associés (moment magnétique, profondeur et déclinaison).

1.1 Dynamique traditionnelle dans les méthodes magnétiques en géophysique et perspectives sur l'application de l'apprentissage profond

Dans les méthodes magnétiques en géophysique, les propriétés physiques d'intérêt sont la susceptibilité magnétique (aimantation induite) et la densité d'aimantation (cas rémanent). On s'intéresse ici au cas induit, le plus fréquent. L'aimantation induite permet de caractériser la quantité de matériau magnétisé dans un champ magnétique primaire. Le matériau magnétisé crée un champ magnétique secondaire (souvent appelé champ induit –mais en toute rigueur il faut parler d'induction magnétique) et les données mesurées lors d'une prospection sont la superposition du champ terrestre primaire et des champs induits secondaires. Ces données sont parfois interprétées en termes d'unités ou structures géologiques réelles (comme les failles ou les intrusions magmatiques), ou inversées pour obtenir la distribution de moments magnétiques induits sous la surface (figure 1).

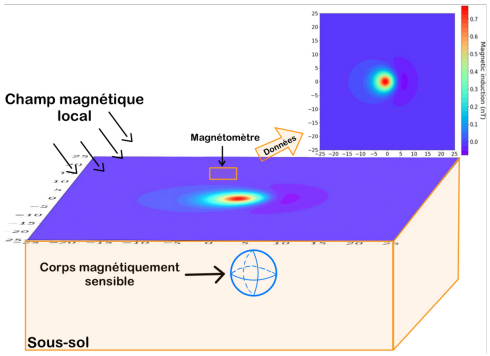


FIGURE 1 – Éléments d'une prospection magnétique : un champ magnétique local, un corps magnétisable enfoui dans le sous-sol, un instrument de mesure (magnétomètre) et des cartes générées après traitement de données.

Afin d'interpréter les données magnétiques, les géophysiciens effectuent des corrections, fournissant les données traitées comme produit final. Ces corrections, en partie réalisées par des logiciels d'inversion classiques, permettent d'isoler le champ anomal causé par des éléments d'intérêt enfouis dans le sous-sol afin d'interpréter les données magnétiques en termes de caractéristiques et de structures en profondeur (Figure 2). Parfois la situation se complique pour les cartes magnétiques où les corps responsables ont des formes compliquées et des dimensions variables. Il est classique que ce processus implique un certain degré de subjectivité et dépendra de l'expertise du géophysicien.
Notre approche de l'application de l'apprentissage profond (Figure 2) devrait d'une part pallier les faiblesses des algorithmes d'inversion, pour lesquels dans un problème multiparamétré à grand nombre d'inconnues, la fonction objectif est souvent très aplatie autour de son minimum global. D'autre part, les différents bruits affectant les données réclament souvent un ajustement fin des paramètres d'amortissement en privilégiant un compromis entre la variance de la solution et la variance des erreurs d'ajustements.

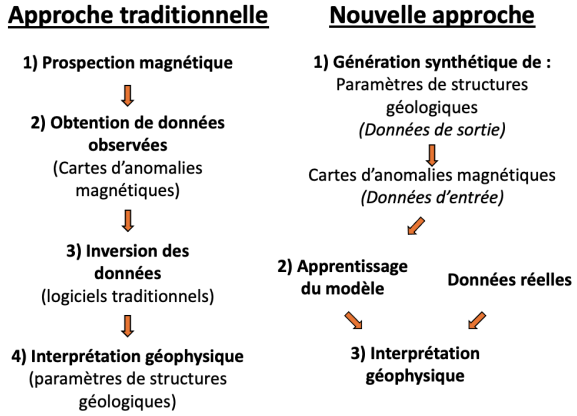


FIGURE 2 – Schéma de la dynamique traditionnelle suivie en géophysique (à gauche), et celui de la nouvelle approche proposée dans cette étude (à droite).

2 Méthodologie

Comme nous l'avons présenté précédemment, l'obtention d'une carte avec la distribution de la susceptibilité magnétique résulte d'une opération de prospection sur le terrain. Les cartes obtenues sont souvent imparfaites en raison de multiples facteurs, comme par exemple : le bruit instrumental, les conditions de régularité du terrain, les difficultés d'accès, etc. En conséquence, qualité et quantité des données géophysiques sont limitées.
Dans l'apprentissage profond, la précision des prédictions réalisées grâces aux réseaux de neurones dépend fortement de la complexité de l'architecture et de la quantité de données disponibles pour l'apprentissage. Contrairement à d'autres domaines dans lesquels les réseaux de neurones ont connu un énorme succès, comme pour la classification des images où le nombre de données disponibles est souvent supérieur à 1 million, la disponibilité de données géophysiques est un facteur limitant, car le nombre de cartes disponibles est faible (maximum d'un millier par exemple). Pour surmonter cette difficulté, on utilise des données simulées pour entraîner et valider le réseau neuronal convolutif. Par exemple, la figure 3 montre l'anomalie magnétique générée par une sphère de moment magnétique 52 $A.m^2$, qui se trouve à 1.4 m de profondeur et en présence d'un champ primaire horizontal.

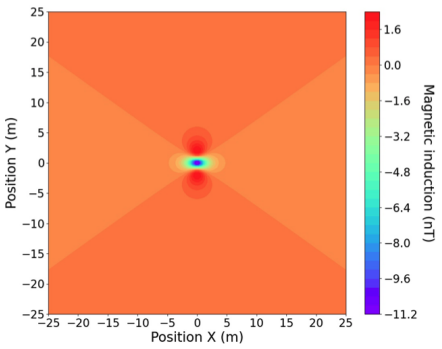


FIGURE 3 – Exemple d'une carte d'anomalie magnétique générée synthétiquement.

2.1 Génération des modèles synthétiques

En utilisant l'équation 8.16 proposée par [15] (cf. équation 1), nous avons pu générer des cartes d'anomalies magnétiques pour différentes configurations de corps magnétiques, qui sont ici assimilés à des dipôles magnétiques. Premièrement nous avons choisi une gamme de valeurs pour les paramètres de la profondeur et de l'amplitude magnétique. Ensuite, les dipôles ont été générés pour chaque combinaison de paramètres en faisant varier aléatoirement leurs positions (x,y). Pour finir et afin d'augmenter le nombre de dipôles, nous avons sommé les cartes générées avec la même distribution de paramètres, en respectant une distance minimum de 2 mètres entre les dipôles pour éviter toute coalescence des anomalies (figure 4a et 4b). Le choix d'utiliser les mêmes distributions pour sommer les cartes permet d'éviter l'occultation d'anomalies dont les amplitudes seraient trop différentes. Cette difficulté, qui est souvent observée dans les cas réels, fera l'objet d'une étude ultérieure.

$$\vec{B_a} = \frac{\mu_o}{4\pi}[\frac{3\hat{r}(\hat{m}\cdot\hat{r}) - \vec{m}}{r^3}] \cdot 10^9 \qquad (1)$$

Avec :
$\vec{B_a}$ = anomalie magnétique (en nT) ;
$\vec{m}$ = moment magnétique du dipôle induit (en $A.m^2$) ;
r = distance entre le dipôle magnétique et le point d'observation (en m) ;
$\hat{r}$ = vecteur unitaire en direction du dipôle magnétique ;
μ_o = perméabilité magnétique du vide (en H/m)

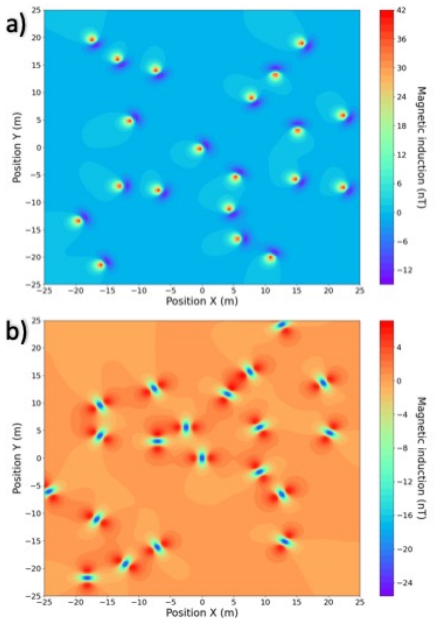


FIGURE 4 – Cartes d'anomalies magnétiques induites incluses dans le jeu de données synthétiques utilisés pour l'apprentissage. Les anomalies de type dipolaire correspondent à des valeurs de déclinaison variable et à une inclinaison de 30^o(a) et 0^o(b).

Les paramètres de chaque dipôle magnétique ont été utilisés comme données de labellisation de notre algorithme. Notre jeu de données, comportant environ 10^4 exemples, a été divisé comme suit : 50% apprentissage, 25% validation croisée et 25% test. Cette répartition est considérée comme optimale par Chollet F. [2], lorsque le jeu de données est limité (nombre d'exemples inférieur à 10^6). Pour choisir la taille de la base d'apprentissage, nous avons testé différents tailles de celle-ci. La meilleure stabilité du modèle a été observée à partir de 10^4 exemples. Les caractéristiques de ce jeu de données sont décrites dans le tableau 1.

Caractéristiques du jeu de données		
Cas magnétique	**Aimantation rémanente**	
Dipôles	Non	
Grille		
Taille	**Espacement**	
100 m	0.5 m	
Caractéristiques physiques		
Profondeur	**Rayon**	**Hauteur du capteur**
[1.0 – 1.4] m	[1.0 – 1.4] m	0.5 m
Caractéristiques magnétiques		
Champ local	**Inclinaison**	**Déclinaison**
47.000 nT	[0, 30, 60, 90] dégrées	[0-180] dégrées

TABLE 1 – Caractéristiques utilisées pour générer le jeu de données simulées.

2.2 Sélection d'architecture

Au début de notre recherche, en l'absence de bibliographie spécifique sur notre application magnétique, nous avons décidé d'utiliser les réseaux de neurones convolutifs (« CNNs ») en raison de la similitude entre la représentation classique de données magnétiques (sous forme de carte coloriée) et les images couleur ou noir/blanc. Nous considérons que les images contiennent dans chaque pixel une valeur d'induction magnétique similaire à l'information de l'un des canaux RVB d'une image couleur.
Cette hypothèse nous a permis de considérer et d'évaluer plusieurs combinaisons de modèles CNNs, souvent appliqués aux images, avant de trouver celle qui s'adapte le mieux à nos objectifs. Le tableau 2 montre un résumé des architectures utilisées par ordre avec des commentaires sur les avantages et les limitations observées.

Modèle (prédictions)	Retour d'expérience
1) VGG16 (Paramètres)	- Prédiction limitée à un objet. - Pas d'information sur localisation.
2) VGG16 multi-objets (paramètres)	- Ralentissement de l'apprentissage lié à l'augmentation du nombre de neurones en sortie. - Pas d'information sur localisation.
3) VGG16 multi-objets (paramètres + position)	- Ralentissement de l'apprentissage lié à l'augmentation du nombre de neurones en sortie. - Diminution de la précision dû à la complexité des prédictions.
4) VGG16 multi-objets (paramètres) + U-Net	- Information précise des positions (x, y) mais imprécise de la largeur et de la longueur des objets.
5) YOLO + DenseNet multi-objets (paramètres)	- Information précise sur la localisation et la classification des objets. - Amélioration du temps d'apprentissage et de la prédictions des paramètres.

TABLE 2 – Ordre de sélection des architectures et les retours d'expériences correspondants.

Après plusieurs expérimentations, nous avons observé que la combinaison de deux modèles, «YOLO» [13] et «DenseNet» [7] (figure 5), est la plus performante pour atteindre nos objectifs de classification et de régression. Nous avons premièrement utilisé l'architecture "YOLO" pour localiser et classer chaque dipôle. Pour effectuer la régression sur les paramètres, nous avons ensuite mis en œuvre l'architecture "DenseNet" afin de prédire individuellement les paramètres de chaque dipôle détecté par le modèle YOLO.

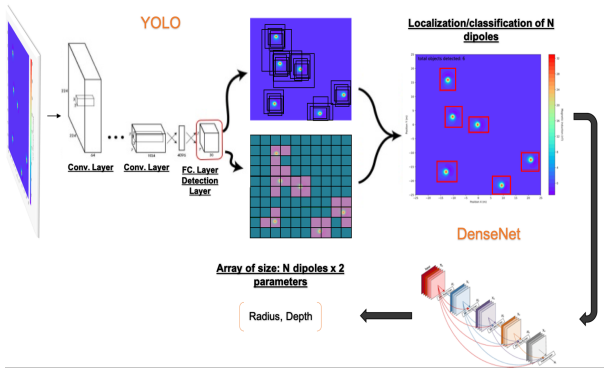


FIGURE 5 – Schéma de notre architecture CNN combinant deux méthodes distinctes (YOLO et DenseNet).

"YOLO" est une architecture efficace de reconnaissance d'objets capable d'identifier la présence d'objets dans les images. Elle divise l'image en régions et prédit les rectangles et les probabilités pour chacune. D'autre part, "DenseNet" est un type d'architecture CNN où chaque couche est connectée à toute les couches suivantes. Cette idée permet d'atténuer le problème de l'évanescence de gradient et d'encourager la réutilisation des caractéristiques lors de l'apprentissage.

Pour améliorer la performance de notre modèle de régression et ainsi déterminer un bon ensemble d'hyperparamètres, nous avons utilisé l'algorithme "hyperband" [9]. Il consiste à accélérer la recherche aléatoire d'hyperparamètres grâce à une allocation adaptative des ressources et à un "early-stopping", ce qui offre une accélération d'un ordre de grandeur supérieur à celle d'autres algorithmes d'exploration. Parmi les hyperparamètres, nous avons utilisé «dropout» pour la régularisation. De multiples fonctions d'optimisation ont été comparées et "AdaGrad" a été sélectionné pour sa stabilité et sa convergence rapide. La fonction d'activation "tanh" a été choisie en raison de valeurs négatives et positives dans les quantités d'entrée. Il est important de préciser que cette fonction a été appliquée à toutes les couches sauf à la couche de sortie car, pour un modèle de régression, nous avons besoin de valeurs continues à la fin. Finalement, nous avons utilisé le coefficient de détermination R^2 pour calculer la précision et la fonction d'erreur quadratique moyenne (MSE) en guise de fonction coût. Le tableau 3 montre le nom du modèle de régression, les valeurs des hyperparamètres et celles de sa performance.

Model	**Dropout**	**Learning rate**
DenseNet	0.1	0.0003
Fonction d'activation	**Précision**	**Perte**
tanh	0.9998	0.0005

TABLE 3 – Combinaison d'hyperparamètres.

2.3 Étiquetage des données.

Nous avons procédé à un processus d'étiquetage suite à la génération des modèles géologiques. Vu que deux modèles d'apprentissage profond seront utilisés, les formats adoptés pour les données de sortie sont : le format d'étiquetage "YOLO" pour le modèle de détection et classification, et les valeurs réelles magnétiques pour le modèle de régression.
Le format d'étiquetage "YOLO" consiste en la création d'un fichier.txt qui contient les annotations pour l'image correspondante, c'est-à-dire la classe et les valeurs du rectangle (« Bounding Box ») autour de chaque objet. "Bounding box" est un type d'annotation couramment utilisé en vision par ordinateur et il comporte les coordonnées, la hauteur et la largeur de l'objet.
Les CNNs sont particulièrement adaptés à l'analyse des données d'images. En plus de les utiliser pour la classification, on peut aussi les utiliser pour prédire des données continues. Dans notre recherche, ces données sont les paramètres des dipôles magnétiques (la déclinaison, la profondeur et l'amplitude du champ magnétique).

3 Explicabilité des algorithmes

L'utilisation de l'apprentissage machine engendre des interrogations de différentes nature concernant son fonctionnement. Dans le cadre de notre application en géophysique, il existe une pression forte sur l'explication des résultats produits par l'apprentissage machine. L'explication peut être destinée par exemple à des développeurs ou des ingénieurs en R&D qui utilisent le plus souvent l'apprentissage machine en mode "boite noire". L'explication peut être aussi destinée à des chercheurs académiques pour améliorer la connaissance scientifique en raison du caractère limité de la prédiction sans explication.
Dans le cadre de cette contribution, le besoin d'explication a été motivée pour sélectionner un modèle parmi d'autres ayant quasiment la même performance statistique. Nous avons utilisé les outils informatiques permettant d'obtenir des informations visuelles :

- La zone discriminatoire de notre réseau a été visualisé en utilisant l'outil Grad-Cam [14].
- L'outil t-SNE [8] a été utilisé pour réduire la dimensionnalité des données.

"Grad-Cam" est une technique permettant de produire des "explications visuelles" pour les décisions d'une grande classe de modèles basés sur CNN. Elle utilise les informations de gradient qui circulent dans la dernière couche convolutionnelle de la CNN pour comprendre chaque neurone en vue d'une décision d'intérêt. De l'autre côté, t-SNE est une autre technique de réduction de la dimensionnalité et est particulièrement bien adaptée à la visualisation d'un ensemble de données à haute dimension.

4 Résultats et Discussion

Nous avons mis en place plusieurs expériences pour tester la robustesse de notre modèle et évaluer sa capacité de généralisation prenant en compte la variation des caractéristiques physiques et magnétiques des dipôles (tableau 1), le niveau de bruit et le nombre de dipôles présents dans le modèle. Les résultats montrent que la méthode YOLO obtient de très bonnes performances. Bien que notre modèle ait été entraîné avec un nombre limité de dipôles (entre 1 et 8), sa capacité de généralisation lui permet d'identifier jusqu'à 20 dipôles (figure 6a et 6b) dipôles avec une confiance moyenne supérieure à 90%.

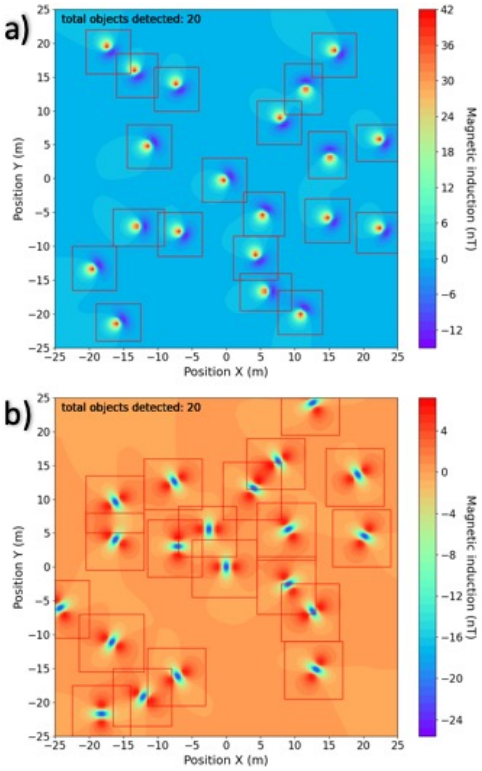


FIGURE 6 – Prédictions du modèle YOLO pour 2 inclinaisons magnétiques de 30° (a) et 0° (b). L'algorithme localise chaque anomalie en l'identifiant dans un rectangle. Le total des dipôles détectés est de 20 dans les deux cas ; ces valeurs sont inscrites en haut à gauche de chaque figure.

A propos de l'évaluation de la performance du réseau YOLO développé, chaque score de confiance reflète la probabilité qu'un rectangle contienne un objet *(Pr(objet))*, ainsi que la précision de ce rectangle en évaluant son chevauchement avec celui de la vérité de terrain mesurée par le score IoU (« Interception over Union »). Par conséquent, le score de confiance devient *Pr(objet) * IoU*. Dans cette étude, nous avons calculé la moyenne du score de confiance pour mesurer l'impact de l'augmentation des dipôles sur la précision de toutes les anomalies détectées. Concernant le réseau DenseNet, on remarque une performance élevée (score de R^2 supérieur à 95%) (Figure 7). Ce résultat s'explique par le fait que le réseau n'analyse que les zones contenant une anomalie d'intérêt, localisée précédemment par le réseau YOLO. Cependant, on observe que la performance des deux réseaux diminue à partir de la présence de 10 dipôles (Figure 7). Cette diminution se produit parce qu'un plus grand nombre de dipôles augmente la coalescence d'anomalies, ainsi que la probabilité d'avoir une anomalie proche des bords de la zone modélisée. Le réentraînement du modèle avec ces cas dans le jeu de données doit surmonter ces limitations.

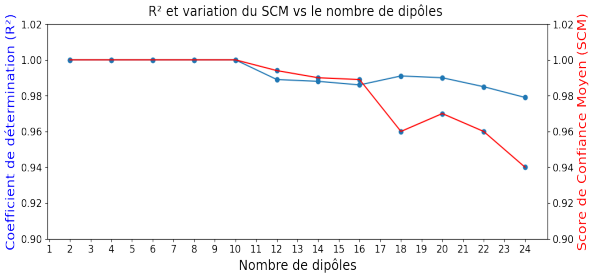


FIGURE 7 – Variation du coefficient de détermination R2 (DenseNet) à gauche et du score de confiance moyen (YOLO) en fonction du nombre de dipôles présents dans le modèle, à droite.

D'autre part, il est important de mentionner que les résultats obtenus correspondent à un jeu de données d'une certaine complexité et sans bruit. La prise en compte de la détectabilité d'une anomalie en présence d'un bruit fond magnétique est beaucoup plus complexe que le simple dépassement d'un seuil constant [4]. Par conséquent, nous avons ajouté un bruit gaussien à notre jeu de données dans une première tentative pour simuler un bruit magnétique naturel. Bien que ce bruit soit basique en comparaison, son étude est importante pour évaluer la capacité de nos modèles à s'adapter correctement à de nouvelles données, tirées de la même distribution que celle utilisée pour leur création.

Pour évaluer les nouveaux modèles magnétiques bruités, nous avons utilisé notre modèle préalablement entraîné. Nous observons une altération de la forme des dipôles lorsque le niveau de bruit augmente et une variation considérable des valeurs d'induction magnétique. Ces changements pourraient baisser la performance de notre modèle parce qu'il dépend initialement de sa capacité à détecter chaque objet. Cependant, pour la détection, le modèle arrive à prédire le nombre exact et les rectangles correspondants à chaque dipôle (figure 8) avec une confiance supérieure à 92%. D'autre part, le modèle prédit les paramètres de chaque dipôle avec un coefficient de détermination supérieur à 96%. Ces résultats nous permettent de déduire que notre modèle est robuste jusqu'à ce niveau de bruit.

Comme indiqué dans la section 4, en plus d'utiliser les mesures statiques pour évaluer la performance de notre modèle, l'application d'outils visuels est important pour comprendre son fonctionnement et expliquer son principe de fonctionnement, indispensable pour son acceptabilité opérationnelle.

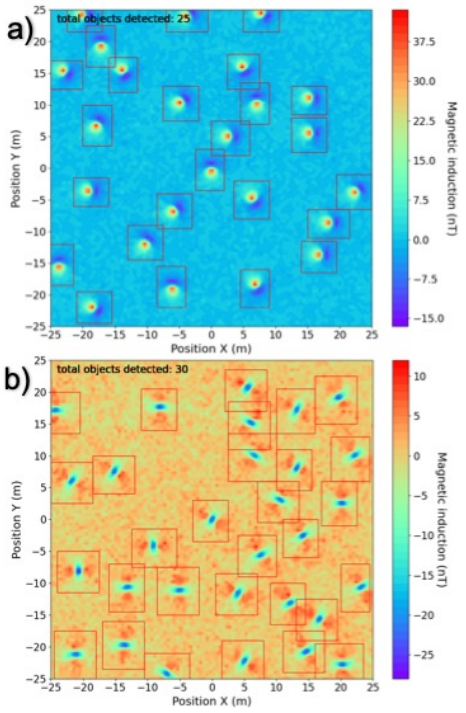


FIGURE 8 – Prédictions du modèle YOLO pour 2 inclinaisons magnétiques de 30°(a) et 0°(b) avec du bruit gaussien. Le total des dipôles détectés est a) 25 et b) 30.

Au début, l'application de l'outil Grad-Cam à la première version de notre modèle actuel nous a permis d'observer, dans la dernière couche de neurones, une carte de chaleur floue qui ne montrait pas la zone discriminatoire ciblée (Figure 9a). Cette carte nous a suggéré de supprimer plusieurs couches qui n'avaient probablement pas d'influence sur les prédictions. En suivant cette stratégie, nous avons trouvé une couche avec une zone discriminatoire cohérente (Figure 9b).

Ce changement a amélioré la stabilité de la courbe d'apprentissage, la performance des prédictions et a diminué considérablement le nombre de paramètres à entraîner du modèle, diminuant ainsi le temps de calcul de la phase d'apprentissage.

En plus de la capacité à discriminer, une autre caractéristique importante à évaluer est la capacité du modèle à différencier les paramètres utilisés pour créer le jeu de données. Pour cette évaluation, celui-ci était constitué par 16 combinaisons possibles entre les valeurs de profondeur et de rayon (tableau 1) et nous avons appliqué l'outil TSNE au jeu de données d'apprentissage (figure 10a), validation croisée (figure 10b) et test (figure 10c). Suite à cette application, nous observons que les prédictions de ces paramètres n'étaient pas aléatoires. Les graphiques générés confirment que le modèle parvient à différencier les 16 combinaisons. Bien que nous observons des petits clusters que le modèle n'arrive pas à rattacher à un groupe plus important (figure 10a) et des classifications ponctuelles erronées dans les données test (figure 10c), les résultats confirment la bonne capacité de notre modèle à différencier parmi les différentes combinaisons de paramètres.

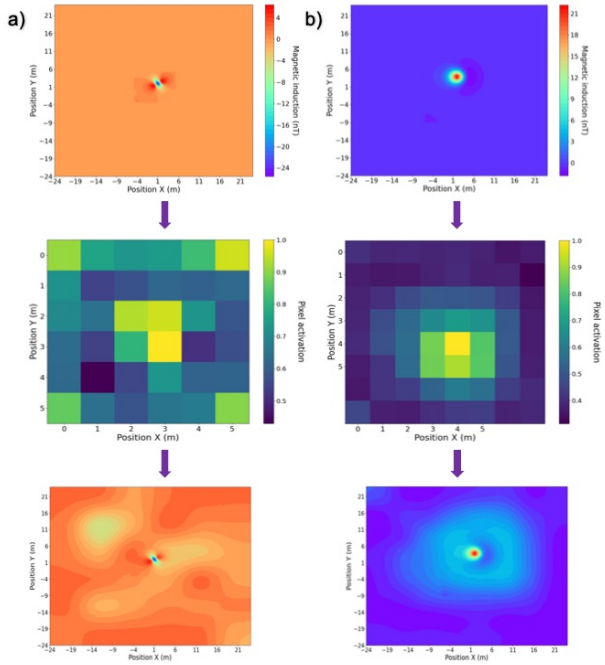


FIGURE 9 – Identification des zones discriminatoires pour a) une ancienne version et b) une actuelle version du même modèle. Du haut vers le bas, on trouve premièrement le modèle géophysique analysé, deuxièmement les pixels d'activations identifiés par Grad-Cam, et troisièmement la zone discriminatoire.

Finalement, le temps de prédiction de notre modèle est d'une vingtaine de secondes, ce qui le rendrait compatible avec une utilisation temps réel pendant une prospection.

5 Conclusions et Perspectives

Nous avons présenté dans cette contribution une stratégie pour détecter, à l'aide de l'apprentissage machine, l'ensemble des dipôles présents dans une carte magnétique. Nous prédisons ensuite les caractéristiques physiques et magnétiques de chaque dipôle détecté. Il a été généré par simulation un jeu de données synthétique, ne disposant pas de mesures géophysiques labellisés en quantité suffisante. Nos expérimentations ont montré que la combinaison des modèles, YOLO et DenseNet, fournit les meilleures performances : le modèle YOLO permet de détecter chaque dipôle tandis que le réseau DenseNet estime les paramètres de chaque dipôle identifié avec une précision supérieure à 90%.

Au-delà de la performance statistique, nous avons optimisé le modèle pour que la zone discriminatoire, visualisée par l'outil Grad-Cam, soit conforme aux résultats escomptés pour éviter la création d'artefacts statistiques. Il est à noter que cela permet de faciliter l'explication des résultats obtenus. En particulier, nous avons optimisé le nombre de paramètres du modèle pour augmenter sa stabilité. Il s'agit donc d'une première étape de vérification fonctionnelle de la boite noire de notre modèle.

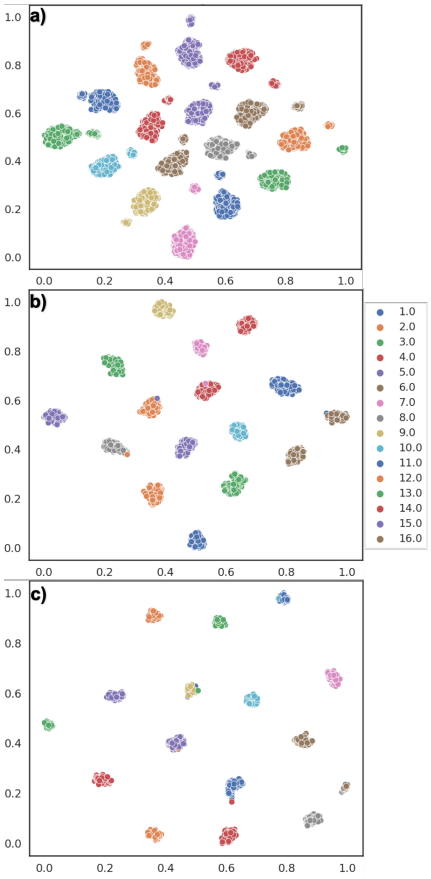


FIGURE 10 – Exemple de graphiques obtenus grâce à l'utilisation de l'outil TSNE sur le jeu de données de a) apprentissage, b) validation croisée et c) test. Les axes n'ont pas d'unité spécifique et les couleurs des clusters représentent les 16 combinaisons possibles parmi les paramètres utilisés pour créer le jeu de données.

Enfin, l'utilisation de l'outil TSNE a permis d'évaluer la capacité du modèle à différencier parmi toutes les combinaisons possibles de paramètres présentes dans le jeu de données. Ces résultats montrent que ces outils de visualisation sont des alliés incontournables pour étudier la robustesse et la stabilité d'un modèle d'apprentissage profond en allant au-delà de la mesure de la précision statistique.

Les perspectives de ce travail consistent à évaluer la robustesse de l'approche proposée en utilisant des données réelles. Un possible cas d'étude consiste à détecter des munitions non explosées enfouies dans le sous-sol (UXO) en présence un bruit de fond magnétique naturel, e.g. des sites peu ou intensément bruités ou présentant des anomalies géologiques importantes. Il sera nécessaire d'expliquer les prédictions prises par nos modèles en utilisant des techniques d'analyse post-hoc. Ces explications seront évaluées qualitativement par un panel d'expert en géophysique. Il sera aussi important de mettre en place des techniques d'adaptation de domaine, par exemple en utilisant une méthode d'apprentissage par transfert bien que les distributions statistiques des données synthétiques peuvent différer par rapport à celles des données réelles.

Références

[1] BERKLEY D. et al., Overcoming Algorithm Aversion : People Will Use Imperfect Algorithms If They Can (Even Slightly) Modify Them, *Management Science*, pp. 1155-1170, 2018.

[2] CHOLLET F., DEEP LEARNING with PYTHON *Manning Publications Co*, pp. 1941–1945, 2018.

[3] DAS V. et al., Convolutional neural network for seismic impedance inversion, *SEG Technical Program Expanded Abstracts*, pp. 2071–2075, 2018.

[4] DWAIN K. Butler, Potential fields methods for location of unexploded ordnance, *The Leading Edge*, pp. 890-895, 2001.

[5] FLORSCH N., et al., Géophysique appliquée pour tous - Volume 2, Méthodes magnétiques et Slingram, *ISTE Editions*, 2019.

[6] GUO J. et al., 3D geological structure inversion from Noddy-generated magnetic data using deep learning methods, *Computers and Geosciences*, pp. 104701, 2021.

[7] HUANG G. et al., Densely Connected Convolutional Networks, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261-2269, 2017.

[8] LAURENS V. et GEOFREY H., Viualizing data using t-SNE. Journal of Machine Learning Research, *Journal of Machine Learning Research*, pp. 2579-2605, 2008.

[9] LISHA L. et al., Hyperband : A Novel Bandit-Based Approach to Hyperparameter Optimization, *Journal of Machine Learning Research*, pp. 1-52, 2018.

[10] MA Y. et al., A deep-learning method for automatic fault detection, *SEG Technical Program Expanded Abstracts*, pp. 1941–1945, 2018.

[11] MEIER U. et al., Fully nonlinear inversion of fundamental mode surface waves for a global crustal model : GLOBAL CRUSTAL MODEL, *Geophysical Research Letters*, pp. 706–722, 2007.

[12] MIRKO V. et Christian J., Neural networks in geophysical applications, *GEOPHYSICS*, pp. 1032-1047, 2000.

[13] REDMON J. et al., You Only Look Once : Unified, Real-Time Object Detection, *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, 2016.

[14] SALVARAJU R. et al., Grad-CAM : Visual Explanations from Deep Networks via Gradient-based Localization, *Springer Science and Business Media LLC : International Journal of Computer Vision*, pp. 336–359, 2019.

[15] SCOLLAR I., Archaeological prospecting and remote sensing, *Cambridge University Press*, Cambridge University Press, 1990.

Vers l'application de l'apprentissage par renforcement inverse aux réseaux naturels d'attention

Bertille Somon[1,2], Aurélien Fermo[1,3], Frédéric Dehais[2,1], Caroline P. C. Chanel[2,1]

[1] ANITI, Artificial and Natural Intelligence Toulouse Institute, France
[2] ISAE-SUPAERO, Université de Toulouse, France
[3] ENS-PSL, Département d'Études Cognitives, Paris, France

Résumé

Le cerveau humain, pour allouer de manière optimale les ressources attentionnelles limitées dont il dispose, supprime ou renforce l'activation de circuits neuronaux : il implémente des heuristiques. Dans une approche novatrice, nous proposons d'utiliser l'apprentissage par renforcement inverse pour caractériser la dynamique d'activation de ces réseaux. Un protocole expérimental est proposé, et les données collectées devraient permettre, à terme, de vérifier cette démarche.

Mots-clés

Processus Décisionnels de Markov, Apprentissage par renforcement inverse, Electroencephalographie, Connectivité dirigée

Abstract

The human brain possesses limited attentional resources and requires heuristics in order to optimise their allocation through neural networks reinforcement. We propose here that using inverse reinforcement learning is an interesting approach to characterize the dynamic activation of these networks. We present an experimental setting aiming at this characterization, and we propose that data collection will comfort this position.

Keywords

Markov Decision Process, Inverse Reinforcement Learning, Electroencephalography, Directed Connectivity

1 Introduction

La répartition optimale de l'attention est une question essentielle dans nos activités multi-tâches de la vie quotidienne. Elle s'appuie sur un compromis entre des politiques d'exploration et d'exploitation des flux d'information pertinents, c'est-à-dire qu'elle consiste à focaliser et maintenir l'attention tout en la laissant permissive aux changements inattendus [15, 9]. À la lumière de certaines études, les corrélats neuronaux de cette dynamique attentionnelle ont pu, en partie, être identifiés. Notamment, les mécanismes attentionnels descendants et ascendants (dits *top-down* et *bottom-up*) sont respectivement délimités par des réseaux cérébraux dorsaux et ventraux. Ces réseaux sont eux-mêmes en étroite interaction avec le cortex cingulaire antérieur responsable de l'allocation des ressources [15, 9]. En condition normale, l'attention est dite "divisée" et permet de traiter efficacement les informations de l'environnement dans différentes modalités (e.g. visuelle, auditive, tactile). Aussi des mécanismes neuronaux oscillatoires sont-ils mis en œuvre pour synchroniser et augmenter l'activité des ces réseaux qui traitent les informations les plus importantes. Enfin, des mécanismes de phasage permettent d'alterner et de rythmer le traitement des informations liées à des tâches secondaires ou inattendues [13].

Toutefois, il a été avancé que la fatigue, un stress intense ou une importante charge de travail pouvaient entraîner un déficit de l'homéostasie entre ces réseaux attentionnels et conduire à une attention dite focalisée, voire, dans les cas extrêmes, "tunnélisée" (pour un revue voir [10]). Concrètement, ces situations dégradées entraînent surtout la suppression de l'activité liée aux tâches secondaires et *bottom-up*, laquelle joue pourtant le rôle primordial d'alerter le cerveau en cas d'imprévus [14, 33, 10]. Bien que ce mécanisme permette, à la manière d'un fusible, de prévenir la surcharge mentale et d'éviter la distraction de l'attention dans des situations complexes, l'omission d'informations essentielles peut avoir des conséquences dévastatrices dans des scénarios de la vie réelle. Ce phénomène, appelé cécité ou surdité attentionnelle, est reconnu pour être à l'origine d'accidents de la route et dans l'aéronautique lorsque des stimuli visuels (e.g. un autre véhicule) ou auditifs (e.g. une alarme) ont pu être négligés [11].

La compréhension et la caractérisation de ces dynamiques cérébrales représentent donc un enjeu de recherche important. En pratique, le suivi en ligne des connectivités cérébrales pourrait permettre le développement d'interfaces cerveau-machine capables de détecter, en situation opérationnelle, des états attentionnels dégradés [12]. Ensuite, l'étude des mécanismes cérébraux et des heuristiques mises en œuvre par la biologie de l'évolution pourraient en retour inspirer de nouveaux algorithmes d'intelligence artificielle [24]. Dans cette perspective, nous pensons qu'il serait pertinent d'appliquer le cadre formel de l'apprentissage par renforcement inverse (IRL [27]) à l'identification des corrélats neuronaux de la dynamique attentionnelle. En particulier, nous suggérons de nous appuyer sur la notion de *fonction de récompense* pour modéliser les stratégies d'activa-

tion ou de suppression de réseaux attentionnels que le cerveau est susceptible de mettre en oeuvre. Dès lors serions-nous en mesure de mieux caractériser les *politiques* attentionnelles d'un ensemble d'agents et de distinguer formellement celles qui sont efficaces (au sein d'une population dite experte) de celles qui ne le sont pas (population dite novice).

1.1 Travaux antérieurs

De nombreuses études se sont attachées à comprendre les liens de causalité ou de corrélation qui peuvent exister entre les différentes aires cérébrales, améliorant ou perturbant les changements d'attention sélective inter-modale [1] [5]. De plus en plus d'évidences montrent que les influences intermodales sur les cortex sensoriels primaires, responsables de la détection de stimuli (par exemple auditifs ou visuels), sont modulées par la synchronisation d'oscillations neuronales grâce à une ré-initialisation des phases ou bien un entraînement neuronal (i.e. le processus au travers duquel deux ou plusieurs oscillateurs auto-entretenus sont couplés et se synchronisent), ou encore une combinaison de ces deux mécanismes [5]. Les mesures de connectivité cérébrale permettent de mettre en évidence ces communications (dirigées ou non) entre plusieurs aires.

Dans le domaine des neurosciences, la connectivité dite *effective* permet d'identifier les réseaux fonctionnels qui varient dans le temps par des techniques basées entre autres sur la causalité de Granger. Plusieurs études ont utilisé ces métriques de connectivité effective comme entrée d'algorithmes d'apprentissage (voir par exemple [19] pour une revue) permettant de transférer et tester ces résultats en ligne dans différentes conditions. Ces algorithmes permettent d'obtenir des résultats de classification de différents états cognitifs avec des performances très élevées [12].

Par ailleurs, les mesures de connectivité permettent d'établir des graphes de connectivité dirigés et dynamiques. Les graphes sont alors définis par un ensemble de noeuds et de connections qui permettent d'obtenir une représentation abstraite des éléments d'un système et de leurs interactions [7]. Appliquée à l'électroencéphalographie (EEG), la théorie des graphes permet de définir des liens entre l'activité cérébrale de différentes aires, par seuillage ou validation statistique des mesures de connectivité. Il est à noter que des couples de connectivité seuillés peuvent définir une matrice d'adjacence de graphe. Il est alors possible d'obtenir des noeuds et des arrêtes représentant les aires cérébrales significativement actives et les métriques de connectivité qui les relient de manière significative. Cette approche permet d'obtenir une modélisation dynamique en considérant que chaque réseau fonctionnel représente un état d'un système dynamique [18, 7]. Plus récemment, cette approche à été étendue à l'utilisation de Chaînes de Markov pour modéliser la dynamique de ce système [34].

Nous posons que la dynamique d'états (réseaux fonctionnels d'attention) est régie par une politique optimisant l'allocation des ressources. Approximer cette dynamique par des chaînes de Markov dites contrôlées serait une approche envisageable. Toutefois, comme le soulignent Ng et Russell [27], l'apprentissage par renforcement (RL) reste peu applicable à des cas concrets de simulation du comportement humain puisque la fonction de récompense y est supposée connue. C'est pourquoi l'apprentissage par renforcement *inverse* (IRL) nous paraît être le cadre le plus approprié si l'on veut expliciter la dynamique et la qualification d'action (fonction de récompense et politique associée).

1.2 Problématique

Dans ce contexte, l'objectif de l'étude que nous souhaitons mener est de caractériser l'efficacité de la dynamique cérébrale de l'attention sélective par le biais de mesures d'activité EEG lors d'une tâche expérimentale contrôlée de changement attentionnel inter-modal. Plus précisément, nous supposons la dynamique attentionnelle guidée par une politique cérébrale intrinsèque et nous pensons pouvoir la décrire formellement en nous fondant sur des données EEG et comportementales collectées. L'apprentissage de la fonction de récompense (générant cette politique) par IRL permettra de comprendre la sélection des actions. Étant capable, grâce à cette fonction de récompense, de représenter le degré d'efficacité d'une politique attentionnelle menée par un agent, nous pourrons ainsi mieux prédire les performances cognitives et aider au développement d'algorithmes d'intelligence artificielle qui soient bio-inspirés [24].

2 IRL et dynamique cérébrale

L'IRL appliqué à un Processus Décisionnel de Markov (MDP) a été caractérisé de manière informelle par Russel [27] comme la définition de la fonction de récompense qu'il est nécessaire d'optimiser connaissant : i) des mesures du comportement d'un agent au cours du temps et dans différentes circonstances, ii) si nécessaire, des mesures des entrées sensorielles de cet agent, et iii) si possible, un modèle de l'environnement [27].

Rappelons d'abord la définition formelle d'un MDP. Un MDP à horizon infini est un n-uplet $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma\}$, où $\mathcal{S}$ est l'ensemble d'états ; $\mathcal{A}$ est l'ensemble d'actions ; $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ est la fonction de transition d'état qui spécifie la probabilité de transiter vers l'état $s' \in \mathcal{S}$ depuis l'état $s \in \mathcal{S}$ quand l'action $a \in \mathcal{A}$ est réalisée, telle que $\mathcal{T}(s', a, s) = p(s'|s, a)$; $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ est la fonction de récompense qui définit une récompense $r(s, a)$ quand l'action $a \in \mathcal{A}$ est prise dans l'état $s \in \mathcal{S}$; enfin, $\gamma \to [0, 1[$ est le facteur d'oubli. Pour résoudre un MDP à horizon infini, il suffit de rechercher une politique markovienne déterministe stationnaire $\pi : \mathcal{S} \to \mathcal{A}$ qui maximise la fonction de valeur généralement définie comme :

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, \pi(s_t)) \mid s_0 = s\right] \qquad (1)$$

1. L'attention sélective inter-modale correspond à l'attention portée de manière sélective et alternée sur différentes modalités sensorielles lorsqu'elles sont présentées simultanément.

Cette équation peut être développée et ré-écrite en tant que :

$$\begin{aligned} V^{\pi}(s_0) &= \mathbb{E}\left[\gamma^0 r(s_0, \pi(s_0))\right] + \\ &\quad \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t r(s_t, \pi(s_t)) \mid s_t = s_1, s_0, \pi(s_0)\right] \\ V^{\pi}(s_0) &= r(s_0, \pi(s_0)) + \gamma \sum_{s \in S} p(s|s_0, \pi(s_0)) V^{\pi}(s) \end{aligned}$$

Ceci indique que la politique optimale (stationnaire déterministe markovienne) π^* peut être calculée, ainsi que la fonction de valeur optimale $V^* = V^{\pi^*}$, sur la base de l'équation de Bellman :

$$V^*(s) = \max_{a \in \mathcal{A}} \left[r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V^*(s') \right] \quad (2)$$

$$\pi^*(s) = \arg\max_{a \in \mathcal{A}} \left[r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V^*(s') \right] \quad (3)$$

Il est à noter que la valeur Q de l'état s et de l'action a sous la fonction de valeur V^{π^*}, dénotée $Q^{\pi^*}(s,a)$, telle que :

$$Q^{\pi^*}(s,a) = r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V^{\pi^*}(s') \quad (4)$$

est la valeur anticipée d'une étape de l'action a dans l'état s en suivant la politique optimale π^* et en obtenant V^*, la vraie valeur optimale attendue. Nous précisions que la valeur d'un état peut alors être calculée par :

$$V^*(s) = \max_{a \in \mathcal{A}} Q^{\pi^*}(s,a)$$

De nombreux algorithmes de solution exacte ou approchée de MDP ont été proposés dans la littérature (voir [21]). Les algorithmes les plus récents explorent des heuristiques pour guider la recherche dans l'espace d'état vers des régions qui sont plus susceptibles d'être visitées par la politique optimale. Ainsi la valeur d'un état, ou la Q-valeur d'une action dans un état, peut être approchée de façon efficace.

L'apprentissage par renforcement (RL) est un des moyens de résolution des MDP où le modèle descriptif n'est pas connu. À la place, il est admis qu'un modèle génératif ou un simulateur de l'environnement est disponible. Le but de l'agent MDP est alors d'approcher la fonction de valeur (et par conséquent la politique) en interagissant avec l'environnement par exploration ou exploitation (ou les deux) des séquences d'actions, tout en évaluant les récompenses obtenues en moyenne. La valeur d'une action (Q-valeur) est alors approchée par diverses méthodes (voir [31]) fondées sur des modèles estimés ou non.

L'IRL, quant à lui, s'intéresse à l'apprentissage de la fonction de récompense [27]. Cela est généralement réalisé sur la base des trajectoires (état-action) effectuées par un agent (en MDP), l'objectif étant de comprendre quelle fonction de récompense a guidé la politique de cet agent dont on observe le comportement.

Initialement, le développement de l'IRL a été motivé par le fait que les recherches sur le RL étaient généralement peu applicables à des cas concrets, et d'autant moins au comportement humain, puisqu'elles avaient tendance à supposer que les fonctions de récompense étaient fixes et connues [27]. Or, dans la majorité des cas, les fonctions de récompense permettant à un agent d'agir avec succès dans son environnement ne sont pas prédéfinies. Par ailleurs, elles sont aussi plus complexes que les fonctions de récompense généralement utilisées en RL et varient d'un agent à un autre. C'est pourquoi il est nécessaire de les inférer selon l'environnement d'application mais aussi selon le type d'agent visé. L'un des intérêts majeurs est que les fonctions de récompense sont notamment plus robustes, rapides et transférables que la politique de l'agent telle que définie dans le contexte du RL [27, 2].

Ainsi l'IRL est-il proposé comme une façon de résoudre un problème d'apprentissage lorsque la fonction de récompense n'est pas supposée connue. L'une des premières motivations en faveur de l'IRL vient de l'intérêt pour l'inférence des objectifs, stratégies ou intentions qui président au comportement animal [27]. Dans le cadre du RL classique, lorsque les intentions d'un agent ne sont pas connues, modéliser son comportement nécessite, après avoir entraîné le modèle, de re-paramétrer la fonction de récompense et d'entraîner à nouveau le modèle jusqu'à obtenir le comportement désiré. Au contraire, l'IRL entend inférer automatiquement cette fonction de récompense en prenant en entrée un ensemble d'observations émanant d'un expert (ou simplement de l'agent dont on veut connaître les objectifs).

Le cadre de résolution de l'IRL est un MDP $\mathcal{M}\backslash_{\mathcal{R}}$: un modèle MDP classique mais dont on a retiré la fonction de récompense. L'ensemble des observations qui proviennent de l'agent dont on veut connaître la fonction de récompense est dénoté $\mathcal{D} = \{\tau_0, \tau_1, \tau_2, ...\tau_n\}$, $n \in \mathbb{N}$ avec $\tau_i = \{(s_0, a_0), (s_1, a_1), ..., (s_k, a_k)\}_i$, $s_k \in \mathcal{S}$, $a_k \in \mathcal{A}$, $i \in \mathbb{N}$ une trajectoire. L'IRL prend donc en entrée le couple $\{\mathcal{M}\backslash_{\mathcal{R}}, \mathcal{D}\}$ et donne en sortie $\hat{R}^*$, la fonction de récompense estimée de l'agent. Bien que retrouver la fonction de récompense ne soit pas nécessairement l'objectif principal de tous les algorithmes d'IRL, il apparaît que la grande majorité des algorithmes suppose que $R^*(s, \pi(s)) \in \mathcal{H}_\phi(s, \pi(s))$, où $\mathcal{H}_\phi(s, \pi(s)) = \{\boldsymbol{\theta}^T \boldsymbol{\phi}(s, \pi(s)), \boldsymbol{\theta} \in \mathbb{R}^n\}$ est l'espace d'hypothèses de la fonction de récompense. En d'autres termes la plupart des algorithmes d'IRL supposent que chaque fonction de récompense que l'on veut tester est représentée par une combinaison linéaire des n propriétés (*features*) d'une paire état-action :

$$\begin{aligned} r(s, \pi(s)) &= \theta_0 \phi_0(s, \pi(s)) + \ldots + \theta_n \phi_n(s, \pi(s)) \\ &= \boldsymbol{\theta}^T \boldsymbol{\phi}(s, \pi(s)) \end{aligned}$$

Ainsi, nous cherchons à représenter la performance d'une politique comme suit :

$$\begin{aligned} V^{\pi}(s) &= \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t \boldsymbol{\theta}^T \boldsymbol{\phi}(s_t,\pi(s_t)) \mid s_0 = s\right] \\ &= \boldsymbol{\theta}^T \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t \boldsymbol{\phi}(s_t,\pi(s_t)) \mid s_0 = s\right] \\ &= \boldsymbol{\theta}^T \boldsymbol{\mu}^{\pi}(s) \end{aligned}$$

Pour une fonction de récompense donnée, $\boldsymbol{\mu}^{\pi}(s)$ est l'espérance de propriétés (*feature expectations*) obtenue en suivant la politique π associée. Ainsi l'objectif commun aux algorithmes d'IRL est de trouver le vecteur de paramètres optimal $\boldsymbol{\theta}^*$ tel que : $\boldsymbol{\theta}^{*T}\boldsymbol{\mu}^{\pi^*}(s) \geq \boldsymbol{\theta}^{*T}\boldsymbol{\mu}^{\pi}(s), \forall \pi \in \Pi$. Il faut trouver $\boldsymbol{\theta}^*$ tel que la performance (définie ci-dessus) de la politique qui lui est associée soit supérieure à celle de n'importe quelle autre politique.
D'où si deux politiques partagent les mêmes espérances de propriétés alors leurs valeurs sont identiques [1] : $\boldsymbol{\mu}_1^{\pi}(s) = \boldsymbol{\mu}_2^{\pi}(s) \Rightarrow \boldsymbol{\theta}^T\boldsymbol{\mu}_1^{\pi} = \boldsymbol{\theta}^T\boldsymbol{\mu}_2^{\pi} \Rightarrow V^{\pi_1}(s) = V^{\pi_2}(s)$. C'est pourquoi la plupart des algorithmes d'IRL cherchent à minimiser une fonction de perte $\mathcal{L}$ tel que :

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\mu}^{\pi^*}, \boldsymbol{\mu}^{\boldsymbol{\pi_\theta}}) \tag{5}$$

L'objectif est donc de réduire, par itérations successives, la distance entre les espérances de propriétés de l'agent (appelé *expert*) dont on veut inférer la fonction de récompense et celles qu'on trouve à chaque itération de l'algorithme.
Soulignons que la minimisation de la fonction $\mathcal{L}$ peut se faire de différentes manières : (i) le *feature matching* dont l'objectif est précisément de faire correspondre les espérances de propriétés de l'algorithme avec celles de l'expert en réactualisant les valeurs de $\boldsymbol{\theta}$ à chaque itération [1, 32, 35] ; (ii) le *max-margin* où l'on optimise sous contrainte l'espérance des récompenses des trajectoires de l'expert observées tel qu'elle soit supérieure à celle qui est obtenue en suivant n'importe quelle autre politique [29, 22, 20] ; (iii) en représentant la fonction de récompense comme le simple paramètre conditionnant en probabilité une classe de politiques et en maximisant cette probabilité par inférence bayésienne, méthode de gradient, etc. [28, 26, 4]
D'une manière générale, les algorithmes d'IRL suivent les étapes suivantes afin d'inférer la fonction de récompense :

1. définition de l'ensemble des observations $\mathcal{D}$ et du modèle $\mathcal{M}_{\mathcal{R}}$;
2. initialisation des paramètres de la fonction de récompense ;
3. résolution de $\mathcal{M}$ par RL classique sous l'hypothèse de la fonction de récompense courante ;
4. calcul de la fonction de perte $\mathcal{L}(\boldsymbol{\mu}^{\pi^*}, \boldsymbol{\mu}^{\boldsymbol{\pi_\theta}})$ puis optimisation des paramètres de la fonction de récompense pour minimiser la fonction de perte ;
5. répéter 3. et 4. jusqu'à réduire la divergence en dessous d'un seuil fixé [2].

Notre but est, en nous appuyant sur une expérimentation assez fondamentale (présentée dans la section suivante), d'approcher au mieux les fonctions de récompenses des participants, jugés performants ou non selon leurs réponses, pour mieux expliquer la politique intrinsèque de changement (inter-modal auditif/visuel) attentionnel mise en œuvre par leur cerveau.

3 Protocole expérimental

L'objectif de cette étude est de caractériser la dynamique cérébrale associée à l'attention sélective grâce à des mesures d'activité EEG lors d'une tâche de changement attentionnel inter-modal. Nous proposons d'enregistrer l'activité EEG de participants en les soumettant à des stimulations audiovisuelles durant lesquelles ils devront porter leur attention sur la modalité visuelle, la modalité auditive ou bien changer (condition « *switch* ») entre les deux modalités. Une tâche de mémoire de travail (*N-back*[2] [30]) sera incluse dans chaque modalité et permettra de définir (i) le moment où le participant doit changer de modalité sensorielle dans la condition *switch*, et (ii) les performances du participant dans toutes les conditions en termes de précision et de temps de réaction. Les performances des participants dans la condition auditive et dans la condition visuelle permettront de définir deux groupes : un groupe expert – aux performances élevées – et un groupe novice. Des métriques de l'activité cérébrale associées aux transitions inter-modales permettront de caractériser les corrélats neuronaux des changements d'orientation attentionnelle pour chaque modalité (visuelle et auditive) mais aussi de modéliser les transitions efficaces et inefficaces à la fois pour la population experte et pour les novices.
Les métriques les plus pertinentes seront alors utilisées pour la définition des graphes, lesquels constitueront des états. Les actions, ainsi que la fonction de transition d'un modèle MDP, seront alors définies et approchées respectivement en fonction des trajectoires d'état observées. La Figure 1 illustre la modélisation envisagée. La fonction de récompense sera alors estimée par une méthode d'IRL permettant non seulement de caractériser la politique des changements attentionnels visuels et auditifs, mais aussi de discriminer les deux populations. Le protocole expérimental détaillé dans la suite a reçu l'avis favorable du Comité d'Éthique de la Recherche (CER) de l'Université Fédérale Toulouse Midi-Pyrénées (CER $2020-322$).

3.1 Matériel et Méthode

3.1.1 Participants

La littérature portant sur l'attention sélective uni- et inter-modale fait référence à des calculs de puissance statistique [23] qui permettent d'établir un minimum de 20 participants pour observer l'effet de la modalité (visuelle, auditive ou audiovisuelle) ainsi que l'effet de l'expertise (experts vs.

2. La tâche de N-back est une tâche de mémoire de travail où l'on présente des stimuli successifs au participant qui doit répondre quand un stimulus a déjà été présenté *N* positions auparavant. Par exemple dans une tâche de 2-back, avec la séquence *M V A V B*, le participant doit répondre au deuxième V qui est présenté deux essais après le premier.

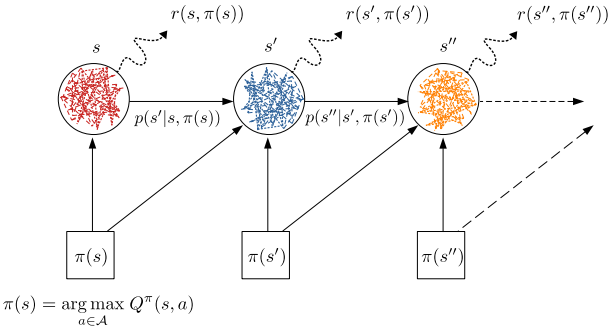


FIGURE 1 – Modèle MDP envisagé représentant la dynamique attentionnelle des participants. Chaque graphe de connectivité, obtenu sur la base des données EEG, constitue un état s. $\pi(s)$ est l'action prise par le participant (la réponse aux stimuli) qui, dans un état s, fait transiter vers un autre état attentionnel s' avec probabilité $p(s'|s,a)$, engendrant la récompense r.

novices). Pour notre propre expérience nous en recruterons 30 au minimum en prévision d'éventuelles pertes de données lors des enregistrements.

3.1.2 Stimuli

Les stimuli visuels consistent en un damier noir et blanc, modulé par une onde sinusoïdale à $48Hz$ (voir fig. 2a), et en une série de points rouges qui apparaissent au centre du damier, à raison de deux, trois ou quatre occurrences successives. Les stimuli auditifs consistent en un son sinusoïdal à $500Hz$, modulé par un son à $40Hz$ (voir fig. 2c), et en de brèves augmentations de 200% de l'intensité de cette modulation, à raison de deux, trois ou quatre occurrences successives. Chaque stimulus (visuel ou auditif) est présenté sur une durée de 2 cycles et permet d'effectuer la tâche de N-back.

3.1.3 Tâche et conditions expérimentales

Dans le but d'obtenir des performances convenables et d'associer l'absence de réponse à une absence de détection des stimuli de changement de modalité (*switch*), une tâche de 0-back est effectuée. Les participants doivent donc détecter les trains de deux stimulations consécutives (i.e. les cibles) et appuyer sur un bouton le cas échéant (une touche spécifique étant associée à chaque type de stimulus).
Les stimuli visuels et auditifs sont présentés de manière simultanée et continue par bloc de 3 minutes. Chaque bloc contient donc environ 40 trains visuels et 40 trains auditifs. Trois conditions expérimentales sont alors déterminées par la présentation de trois types de bloc :

- des blocs durant lesquels le participant doit focaliser son attention sur la tâche visuelle sans tenir compte des stimulations auditives (5 blocs au total) ;
- des blocs auditifs, où la consigne est inversée par rapport aux blocs visuels (5 blocs au total) ;
- des blocs inter-modaux visuo-auditifs, où les stimuli cibles entraînent un changement de focus attentionnel entre le visuel et l'auditif de manière alternée (10 blocs au total).

Pour chaque condition, 30% des trains sont des cibles ($\sim$ 240 cibles au total dont 120 représentent des changements inter-modaux). La figure 2e est une représentation schématique des enchaînements de trains lors des blocs de changement attentionnel.

3.1.4 Recueil et analyse des données

Les données comportementales et EEG sont recueillies de manière continue tout au long des 20 blocs d'expérimentation, et seront prétraitées selon les standards en vigueur.

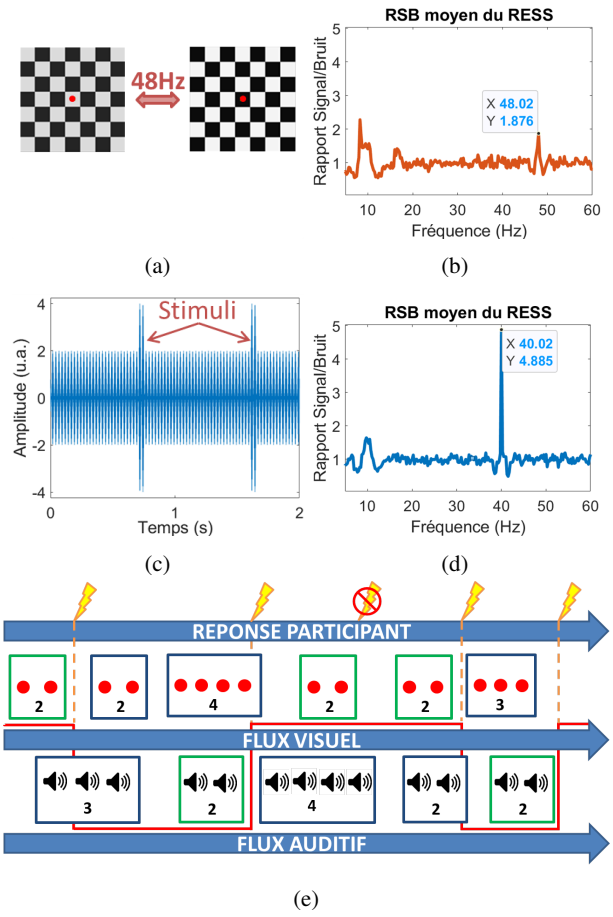


FIGURE 2 – Stimulations (a) présentées à $48Hz$ dans le domaine visuel et (c) modulées à $40Hz$ dans le domaine auditif afin de déclencher des activités cérébrales de *steady-states* (b) visuels à $48Hz$ et (d) auditifs à $40Hz$. (e) Description de la tâche de changement attentionnel inter-modal durant laquelle le flux auditif (dernière ligne) et le flux visuel (ligne intermédiaire) sont présentés simultanément. Lors des blocs *switch* le participant doit appuyer sur une touche du clavier (réponse participant en haut) et changer son attention pour l'autre modalité lorsqu'une cible (deux points ou deux sons – en vert sur le schéma) est présentée. Il doit ensuite rester focalisé sur la même modalité jusqu'à la prochaine cible dans cette modalité. Le trait rouge indique le "chemin attentionnel" du participant en fonction des stimuli et de ses réponses ou absences de réponse (e.g. à la troisième cible).

Comportementales : Les taux d'erreurs et les temps de réaction moyens seront analysés après le déroulement de l'expérience à l'aide d'analyses de variance (ANOVA) à mesures répétées. Les taux d'erreurs permettront de définir les deux groupes (experts et novices).

Electrophysiologiques : Deux types de mesures EEG seront extraites : des mesures de puissance fréquentielle et des mesures de connectivité. Concernant la puissance fréquentielle, la méthode RESS (*Rythmic Entrainment Source Separation* [8]) sera appliquée afin d'identifier les sources cérébrale émettrices des activités de *steady-states* [3] visuels et auditifs habituellement observés dans ce type de tâche, de manière guidée. Les résultats d'une telle mesure observés lors de pré-tests à l'expérimentation sont présentés pour la modalité visuelle (fig. 2b) et pour la modalité auditive (fig. 2d) à titre d'exemples. La quantification dynamique sera obtenue en extrayant la magnitude de la transformée de Hilbert sur la fréquence du *steady-state* au cours du temps. Ces deux mesures seront comparées dans les différentes conditions expérimentales grâce à une ANOVA à mesures répétées.

4 Connectivité et graphes

Dans chaque condition, des métriques de connectivité dirigée seront extraites sur des fenêtres temporelles de 5 secondes, centrées sur la fin de la stimulation cible, afin de rendre compte des activations ou suppressions des réseaux attentionnels lors des passages de la modalité visuelle à l'auditive, et inversement. Pour cela, les modèles d'autorégrésseurs multi-variées (ou Multivariate Autoregressive (MVAR) models) seront utilisés.

Soit $X(t) = \sum_{d=1}^{p} A_{ij}(d)X(t-d) + e(t)$ le modèle MVAR, avec $X(t)$ une série temporelle représentant l'activité cérébrale aux différentes k électrodes, $A_{ij}(d)$ la matrice de taille $k \times k$ des coefficients d'estimation du modèle d'ordre p à chaque instant d associée au couple d'électrodes ij, et $e(t)$ l'erreur de prédiction. Il est possible de calculer la fonction de transfert $\hat{H}(f)$ du modèle MVAR dans le domaine fréquentiel telle que [17] :

$$\hat{H}(f) = \left(\sum_{d=0}^{p} A(d) e^{-2i\pi f \Delta t} \right)^{-1}$$

où l'élément $H_{ij}(f)$ de la matrice $\hat{H}(f)$ décrit la connexion entre la $j^{\text{ème}}$ entrée (électrode) et la $i^{\text{ème}}$ sortie (électrode) du système, et Δt est un intervalle de temps.

La fonction de transfert spectrale ainsi que la matrice de corrélation permettent d'estimer différentes métriques de connectivité : la fonction de transfert dirigée (*Directed Transfer Function* ou DTF) et la Cohérence Partielle Dirigée (*Partial Directed Coherence* ou PDC) ainsi que leurs métriques affiliées (e.g. la *full frequency* DTF). Ces mesures représentent respectivement le flux d'entrée ou le flux de sortie causal de l'électrode j vers i.

3. Les *steady-states* sont définis comme des activités oscillatoires résultant de stimulations répétées et pour lesquelles les neurones se synchronisent à la fréquence de stimulation [16].

De nombreuses mesures de connectivité s'affranchissent de l'aspect temporel, nécessaire à l'estimation de la dynamique attentionnelle, en passant au domaine fréquentiel. Pour pallier cette difficulté, des métriques dépendantes du temps ont été développées : ce sont les métriques dénommées *short-time* [18]. Lorsque plusieurs répétitions d'une même stimulation sont disponibles (un nombre de répétitions $r = 1, 2, \ldots, N_T$), il est possible de définir les mesures de connectivité précédentes en moyennant les matrices de corrélation à travers les différents essais sur de courtes fenêtres (avec N_S points temporels) considérées comme quasi-stationnaires [18]. On obtient alors une matrice de corrélation croisée dépendante du temps ($\widetilde{R}_{ij}(s)$ avec s un délai pré-défini) telle que :

$$\widetilde{R}_{ij}(s) = \frac{1}{N_T} \sum_{r=1}^{N_T} \frac{1}{N_S} \sum_{t=1}^{N_S} X_i^{(r)}(t) X_j^{(r)}(t+s)$$

cette matrice $\widetilde{R}_{ij}(s)$ permet alors d'obtenir une DTF à court terme (Short-time DTF ou SDTF) par exemple.

Dans tous les cas (mesures statiques ou dynamiques), les matrices d'adjacence sont calculées sur les métriques de connectivité les plus appropriées. Ici, chaque graphe défini sur un temps court ou non représente un état défini à l'échelle du participant. Puis, pour une action donnée, les graphes sont comparés à l'échelle du groupe (inter-participants). Les graphes (i.e. les états) similaires au travers des participants pourront alors être agrégés soit par une méthode basée sur une approche de mélange assortatif adaptée des statistiques de graphes pour les données EEG [18], soit par une méthode de clustering [7]. Enfin, les divergences observées à l'échelle du groupe entre les états inter-participants pour chaque condition expérimentale seront capturées par la fonction de transition du modèle MDP.

4.1 Application de l'IRL

La construction des graphes, obtenus sur des courtes fenêtres temporelles, nous permettra de définir les états du MDP sur lequel nous voulons appliquer les algorithmes d'IRL. Les métriques de connectivité de chaque graphe constitueront le vecteur $\boldsymbol{\theta}$ de propriétés (*features*) définissant chaque état, et les réponses des participants aux stimuli constitueront les actions dans le MDP (voir fig. 1). *In fine* nous obtiendrons pour chaque participant l'ensemble de trajectoires $\mathcal{D}$ dont nous avons besoin pour inférer la fonction de récompense $\mathcal{R}$. Le design expérimental présenté ainsi que l'objectif final contraignent le choix des algorithmes d'IRL qu'il est possible d'utiliser. Ni les algorithmes d'*apprenticeship* [1, 32, 29], ni ceux qui envisagent la sous-optimalité du comportement de l'expert [26, 6] ne semblent appropriés puisqu'ici nous souhaitons inférer la fonction de récompense elle-même d'une population, et non purement copier une politique ou encore extrapoler autour d'elle.

Nous supposons que : i) nos espaces d'états et d'actions sont discrets et nos vecteurs $\boldsymbol{\theta}$ de propriétés pour chaque état bien définis par des mesures observées ; ii) il est plus pertinent d'inférer une distribution de probabilité sur l'en-

semble des fonctions de récompense qui peuvent expliquer la dynamique cérébrale et comportementale d'un participant ; et iii) l'expérience permet de fournir en entrée d'un algorithme des trajectoires provenant non pas d'un seul mais d'une trentaine d'agents différents aux performances variées.

Par conséquent, nous portons actuellement notre intérêt sur l'algorithme de Babeş-Vroman [3] qui est l'un des seuls à utiliser à la fois un modèle probabiliste (bayésien) et une méthode de *clustering* (*Expectation-Maximization*) capable de définir plusieurs groupes d'agents selon la distribution de probabilité sur les fonctions de récompense qui leur est associée. En entrée l'algorithme prend l'ensemble des trajectoires $\mathcal{D}$ dont on sait qu'elles proviennent d'intentions différentes (ici de stratégies attentionnelles) et le nombre m de *clusters* maximal supposé (avec $|\mathcal{D}| = n > m$). On initialise le vecteur $\mathbf{\Theta} = (\rho_1, ..., \rho_m, \theta_1, ..., \theta_m)$, où ρ_j sont les priors et θ_j les paramètres de nos fonctions de récompense (précédemment définis) associés à chaque *cluster*. L'idée est ensuite assez intuitive et consiste en trois étapes principales (voir [3] pour plus de détails).

1. On calcule, à chaque itération t :

$$z_{ij}^t = \prod_{(s,a)\in\tau_i} \pi_{\boldsymbol{\theta}_j^t}(s,a)\rho_j^t / Z$$

à savoir la probabilité qu'une trajectoire donnée τ_i ait été générée par une intention j ; Z est la constante de normalisation et $\pi_{\boldsymbol{\theta}_j}(s,a)$ est la politique *Softmax* déterminée par la Q-valeur sous l'hypothèse temporaire $\boldsymbol{\theta}$. Elle est donc définie comme suit :

$$\pi_{\boldsymbol{\theta}_j^t}(s,a) = \frac{e^{\beta Q_{\boldsymbol{\theta}_t}(s,a)}}{\sum\limits_{a'} e^{\beta Q_{\boldsymbol{\theta}_t}(s,a')}}$$

avec $\beta \geq 0$ la constante de Boltzman.

2. On cherche à maximiser :

$$\sum_{l=1}^{m}\sum_{i=1}^{n} \log(\rho_l^t) z_{il}^t + L(\mathcal{D}|\boldsymbol{\theta}_t)$$

à savoir la probabilité que chaque trajectoire appartienne à un cluster donné et qu'à ce cluster soit associée une fonction de récompense hypothétique. La vraisemblance (*log-likelihood*) des trajectoires sachant notre fonction de récompense est calculée comme suit :

$$L(\mathcal{D}|\boldsymbol{\theta}_t) = \sum_{l=1}^{m}\sum_{i=1}^{n} \log(\Pr(\tau_i|\theta_l^t)) z_{il}^t$$

3. On met à jour $\boldsymbol{\theta}$ par la méthode de gradient :

$$\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t + \alpha_t \nabla L(\mathcal{D}|\boldsymbol{\theta}_t)$$

où $\alpha_t > 0$ est, à l'instant t, le pas à appliquer lors de la descente du gradient ; et on réitère (1)-(3) jusqu'à ce que le nombre d'itérations fixé soit atteint.

Nous devrions ainsi obtenir des groupes de participants dont les distributions de probabilité sur les fonctions de récompense, c'est-à-dire les stratégies attentionnelles, se ressemblent. Nous espérons ainsi pouvoir mieux définir des stratégies de l'attention inter-modale plus efficaces que d'autres.

5 Discussion et perspectives

Nous avons présenté un protocole expérimental visant à identifier les marqueurs associés aux changements d'attention focalisée inter-modale à l'aide d'algorithmes d'IRL appliqués sur des données d'activité cérébrale. L'approche novatrice de cette proposition porte non seulement sur les métriques utilisées afin d'extraire les caractéristiques attentionnelles du signal EEG (notamment des mesures court-terme dynamiques) ; mais aussi sur l'utilisation d'algorithmes d'IRL permettant, dans le cadre d'un modèle MDP, d'apprendre la fonction de récompense associée à une politique attentionnelle donnée. Le protocole expérimental que nous avons développé nous permettra par ailleurs d'identifier une population dite "experte" (celle dont les performances sont bonnes) par la fonction de récompense sous-jacente à la politique optimale qu'on y observera.

Pour inférer et classer différentes dynamiques attentionnelles nous pensons qu'un algorithme d'IRL qui combine clustering et modèle probabiliste est le choix le plus cohérent dans le cadre de cette étude. Ce choix doit cependant être nuancé par le fait que nous devrons estimer la fonction de transition $\mathcal{T}$ entre états que nous ne connaissons pas *a priori*. Par ailleurs le modèle $\mathcal{M}$ que nous avons défini n'inclut pas l'existence d'un état final absorbant puisqu'il s'agit avant tout d'une tâche de réaction en continu. Enfin il est possible qu'un agent ne suive pas une mais plusieurs fonctions de récompense variables dans le temps. À cet égard la clusterisation d'une même trajectoire en plusieurs fonctions de récompense subordonnées et l'intégration de cycles dans un modèle bayésien non paramétrique [25] est une piste complémentaire à envisager.

Quant aux perspectives de ce travail il s'agit, dans un premier temps, de tester nos hypothèses en acquérant les données EEG des trente participants prévus. À plus long terme, la définition exacte de l'activité cérébrale associée à un comportement "expert" efficace pourrait permettre : (i) de mettre en place des interfaces cerveau-machine pour détecter des états attentionnels dégradés et lancer des contre-mesures quand, par example, l'activité d'un opérateur dévie d'une activité optimale ; (ii) aux algorithmes d'intelligence artificielle de tirer profit de l'étude des heuristiques mises en place par le cerveau pour optimiser l'allocation de ses ressources ; enfin (iii) d'améliorer la formation des apprentis en comprenant comment maximiser l'utilisation de leurs réseaux attentionnels – des techniques récentes comme celle du *neurofeedback* sont à cet égard prometteuses.

Remerciements

Ce travail est financé par ANITI - *Artificial and Natural Intelligence Toulouse Institute*, Institut 3IA, ANR-19-PI3A-0004.

Références

[1] P. Abbeel and A.Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *ICML*, page 1, 2004.

[2] S. Arora and P. Doshi. A survey of inverse reinforcement learning : Challenges, methods and progress. *arXiv preprint arXiv :1806.06877*, 2018.

[3] M. Babes, V.N. Marivate, K. Subramanian, and M.L. Littman. Apprenticeship learning about multiple intentions. In *ICML*, 2011.

[4] C.L. Baker, R. Saxe, and J.B. Tenenbaum. Action understanding as inverse planning. *Cognition*, 113(3) :329–349, 2009.

[5] A.K.R. Bauer, S. Debener, and A.C. Nobre. Synchronisation of neural oscillations and cross-modal influences. *Trends in cognitive sciences*, 2020.

[6] D. Brown, W. Goo, P. Nagarajan, and S. Niekum. Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations, 2019.

[7] E. Bullmore and O. Sporns. Complex brain networks : graph theoretical analysis of structural and functional systems. *Nature reviews neuroscience*, 10(3) :186–198, 2009.

[8] M.X. Cohen and R. Gulbinaite. Rhythmic entrainment source separation : Optimizing analyses of neural responses to rhythmic sensory stimulation. *Neuroimage*, 147 :43–56, 2017.

[9] M. Corbetta, G. Patel, and G.L. Shulman. The reorienting system of the human brain : from environment to theory of mind. *Neuron*, 58(3) :306–324, 2008.

[10] F. Dehais, H.M. Hodgetts, M. Causse, J. Behrend, G. Durantin, and S. Tremblay. Momentary lapse of control : A cognitive continuum approach to understanding and mitigating perseveration in human error. *NBR*, 100 :252–262, 2019.

[11] F. Dehais, A. Lafont, R. Roy, and S. Fairclough. A neuroergonomics approach to mental workload, engagement and human performance. *Frontiers in Neuroscience*, 14 :268, 2020.

[12] F. Dehais, I. Rida, R.N. Roy, J. Iversen, T. Mullen, and D. Callan. A pbci to predict attentional error before it happens in real flight conditions. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 4155–4160, 2019.

[13] S.M. Doesburg, A.B. Roggeveen, K. Kitajo, and L.M. Ward. Large-scale gamma band phase synchronization and selective attention. *Cerebral cortex*, 18(2) :386–396, 2007.

[14] G. Durantin, F. Dehais, N. Gonthier, C. Terzibas, and D.E. Callan. Neural signature of inattentional deafness. *Human brain mapping*, 38(11) :5440–5455, 2017.

[15] D. Fougnie, J. Cockhren, and R. Marois. A common source of attention for auditory and visual tracking. *Attention, Perception, & Psychophysics*, 80(6) :1571–1583, 2018.

[16] C.S. Herrmann. Human eeg responses to 1–100 hz flicker : resonance phenomena in visual cortex and their potential correlation to cognitive phenomena. *Experimental brain research*, 137(3) :346–353, 2001.

[17] M.J. Kaminski and K.J. Blinowska. A new method of the description of the information flow in the brain structures. *Biological cybernetics*, 65(3) :203–210, 1991.

[18] M.J. Kaminski, A. Brzezicka, J. Kaminski, and K.J. Blinowska. Coupling between brain structures during visual and auditory working memory tasks. *International journal of neural systems*, 29(3), 2019.

[19] A. Khosla, P. Khandnor, and T. Chand. A comparative analysis of signal processing and classification methods for different applications based on eeg signals. *Biocybernetics and Biomedical Engineering*, 40(2) :649–690, 2020.

[20] E. Klein, B. Piot, M. Geist, and O. Pietquin. Structured classification for inverse reinforcement learning. *JMLR*, 2012 :1–14, 2013.

[21] A. Kolobov. Planning with markov decision processes : An AI perspective. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1) :1–210, 2012.

[22] J.Z. Kolter, P. Abbeel, and A.Y. Ng. Hierarchical apprenticeship learning with application to quadruped locomotion. In *Advances in Neural Information Processing Systems*, pages 769–776, 2008.

[23] D. Lakens. Calculating and reporting effect sizes to facilitate cumulative science : a practical primer for t-tests and anovas. *Frontiers in psychology*, 4 :863, 2013.

[24] G.W. Lindsay. Attention in psychology, neuroscience, and machine learning. *Frontiers in computational neuroscience*, 14 :29, 2020.

[25] B. Michini and J.P. How. Bayesian nonparametric inverse reinforcement learning. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 148–163, 2012.

[26] G. Neu and C. Szepesvári. Apprenticeship learning using inverse reinforcement learning and gradient methods, 2012.

[27] A.Y. Ng and S.J. Russell. Algorithms for inverse reinforcement learning. In *IMSL*, volume 1, page 2, 2000.

[28] D. Ramachandran and E. Amir. Bayesian inverse reinforcement learning. In *IJCAI*, volume 7, pages 2586–2591, 2007.

[29] N.D. Ratliff, J.A. Bagnell, and M.A. Zinkevich. Maximum margin planning. In *ICML*, pages 729–736, 2006.

[30] E.E. Smith and J. Jonides. Working memory : A view from neuroimaging. *Cognitive psychology*, 33(1) :5–42, 1997.

[31] R.S. Sutton and A.G. Barto. *Reinforcement learning : An introduction*. MIT press, 2018.

[32] U. Syed and R.E. Schapire. A game-theoretic approach to apprenticeship learning. In *Advances in neural information processing systems*, pages 1449–1456, 2008.

[33] J.J. Todd, D. Fougnie, and R. Marois. Visual short-term memory load suppresses temporo-parietal junction activity and induces inattentional blindness. *Psychological science*, 16(12) :965–972, 2005.

[34] N.J. Williams, I. Daly, and S.J. Nasuto. Markov model-based method to analyse time-varying networks in eeg task-related data. *Frontiers in computational neuroscience*, 12 :76, 2018.

[35] B.D. Ziebart, A.L. Maas, J.A. Bagnell, and A.K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pages 1433–1438, 2008.