Actes des 20es Rencontres des Jeunes Chercheurs en Intelligence Artificielle

Maxime Guériau

AfIA

Association française
pour l'Intelligence Artificielle

RJCIA

Rencontres des Jeunes Chercheurs
en Intelligence Artificielle

PFIA 2022

Table des matières

Éditorial

Rencontres des Jeunes Chercheurs en Intelligence Artificielle

Les 20èmes Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA'2022) se sont déroulées les deux derniers jours (30 juin et 1er juillet) de l'édition 2022 de la Plate-Forme Intelligence Artificielle (PFIA'2022), qui s'est tenue du 27 juin au 1er juillet 2022 à Saint-Etienne. La conférence RJCIA est soutenue par le Conseil d'Administration de l'AFIA.

Les RJCIA sont destinées aux jeunes chercheurs et chercheuses en IA, doctorant(e)s ou titulaires d'un doctorat depuis moins d'un an. L'objectif de cette manifestation est double :

— permettre aux jeunes chercheurs préparant une thèse en Intelligence Artificielle, ou l'ayant soutenue depuis peu, de se rencontrer et de présenter leurs travaux, et d'ainsi former des contacts avec d'autres jeunes chercheurs et d'élargir leurs perspectives en échangeant avec des spécialistes d'autres domaines de l'intelligence artificielle, et ;

— former les jeunes chercheurs à la préparation d'un article, à sa révision pour tenir compte des observations du comité de programme, et à sa présentation devant un auditoire de spécialistes, leur permettant ainsi d'obtenir des retours de chercheurs de leur domaine ou de domaines connexes.

Pour cette édition 2022 de la conférence, nous avons eu l'honneur de recevoir Pr. Franck Gechter – Université de Technologie de Belfort-Montbéliard, Laboratoire Connaissances et Intelligence Artificielle Distribuée – qui a donné une conférence invitée intitulée « Du contrôle de véhicule autonome à l'optimisation de systèmes de transports : un retour d'expérience sur l'utilisation des systèmes multi-agents réactifs pour le contrôle/management de systèmes Cyber-Physique ».

Concernant les contributions scientifiques, 21 articles ont été soumis. Au total 18 articles ont été acceptés puis présentés et constituent le contenu de ces actes. Ces articles sont de deux types : 14 articles longs qui présentent des contributions originales dans les thèmes de la conférence ; et 4 articles courts qui présentent les sujets de recherche des jeunes chercheurs.

Les RJCIA'22, comme toutes les conférences et événements hébergés de PFIA'22, ont eu la chance de voir le retour de leurs participants en présentiel. Cela favorisera je l'espère les échanges entre les jeunes chercheurs mais aussi avec les membres des autres communautés présents à l'occasion de PFIA'22.

Je profite de ce dernier paragraphe pour remercier toutes les personnes qui ont contribué au succès de ces rencontres : les auteurs, leurs co-auteurs pour leurs contributions ; les participants pour leurs échanges constructifs ; le travail assidu du comité de programme pour proposer des rapports de relecture bienveillants et constructifs ; ainsi que la qualité de l'accueil et de la préparation assurés par le comité d'organisation de PFIA'22.

Maxime Guériau

Comité de programme

Président

— Maxime Guériau (LITIS, INSA Rouen Normandie, France).

Membres

— Arthur Aubret (LITIS, Université Claude Bernard Lyon 1, France) ;
— Zied Bouraoui (CRIL-CNRS, Université d'Artois, France) ;
— Mathieu Chollet (LS2N, IMT Atlantique, France) ;
— Baudouin Dafflon (DISP, Université Claude Bernard Lyon 1, France) ;
— Maxime Devanne (IRIMAS, Université Haute-Alsace, France) ;
— Madeleine El-Zaher (LINEACT, CESI, France) ;
— Maxime Folschette (CRIStAL, Centrale Lille, France) ;
— Abir Beatrice Karami (FGES, Université Catholique de Lille, France) ;
— Pierre-Antoine Laharotte (LICIT, Université Gustave Eiffel, France) ;
— Alexandre Lombard (CIAD, Université de Technologie de Belfort-Montbéliard, France) ;
— Guillaume Lozenguez (Center Digital Systems, IMT Lille Douai, France) ;
— Jean-Guy Mailly (LIPADE, Université de Paris, France) ;
— Mohamed-Lamine Messai (ERIC, Université Lumière Lyon 2, France) ;
— Arianna Novaro (Université Paris 1 Panthéon-Sorbonne, France) ;
— Charlotte Pelletier (IRISA, Université Bretagne Sud, France) ;
— Nicolas Verstaevel (IRIT, Université Toulouse 1 Capitole, France).

Session posters

Gestion Décentralisée de Clefs Cryptographiques dans un Système Multi-Agents Embarqués

Arthur Baudet[1,2], Annabelle Mercier[1], Oum-El-Kheir Aktouf[1], Philippe Elbaz-Vincent[2]

[1] Univ. Grenoble Alpes, Grenoble INP, LCIS, Valence, France
[2] Univ. Grenoble Alpes, CNRS, Institut Fourrier, Grenoble, France

{arthur.baudet, oum-el-kheir.aktouf, annabelle.mercier}@lcis.grenoble-inp.fr
philippe.elbaz-vincent@univ-grenoble-alpes.fr

Résumé

Nous présentons les travaux réalisés et en cours dans le cadre d'une étude menant à la proposition d'une architecture de sécurité pour des systèmes multi-agents embarqués : des systèmes décentralisés et autonomes constitués d'agents coopérant pour réaliser la tâche qui leur est attribuée.

Mots-clés

Système multi-agents, système embarqué, infrastructure à clefs publiques

Abstract

We present an ongoing work on defining a security architecture for multi-agent systems of embedded agents, a kind of decentralized system of autonomous and embedded agents coordinating to fulfill their objectives.

Keywords

Multi-agent system, embedded system, public key infrastructure

1 Contexte et problématique

Nos travaux concernent ce que nous nommons les systèmes multi-agents embarqués (SMAe) ouverts, des systèmes où des agents, des systèmes embarqués autonomes coopérant pour atteindre leurs objectifs, ne venant pas forcément du même constructeur, peuvent se connecter ou se déconnecter du système durant son exécution. On retrouve, par exemple, ce genre de système dans les réseaux de véhicules autonomes connectés et communicants ou les réseaux de capteurs sans fil.

La sécurisation des communications et l'authentification des membres d'un système dépendent fortement de solutions cryptographiques ; notamment afin d'assurer la non-répudiation, la détection d'atteinte à l'intégrité et la confidentialité des échanges ainsi que l'authentification des parties communicantes. Tout cela pouvant être réalisé à l'aide d'une infrastructure à clefs publiques (PKI) telles que la PKIX [**?**]. Néanmoins, dans ce contexte, l'hypothèse de précharger et mettre à jour en temps réel des clefs ou certificats dans chaque agent ne peut pas être satisfaite. De plus, l'absence d'une autorité centrale rend tous les systèmes de gestion de clefs centralisés inapplicables.

Notre problématique est donc la suivante : dans le cas d'un attaquant possédant un contrôle total sur le réseau, pouvant donc intercepter, modifier ou forger des messages sans que nous ayons un a priori sur son comportement, comment élaborer un système de gestion de clefs cryptographiques pour un système multi-agents embarqués ouvert ?

2 Bibliographie

Nous avons réalisé une étude rigoureuse de la littérature récente concernant la problématique de sécurisation des SMAe [**?**]. Elle nous a permis de rendre compte d'une abondance de travaux proposant une protection contre des attaques venant de l'intérieur, notamment par l'usage de système de gestion de confiance ou de détection d'intrusion, mais peu de travaux concernant les attaques venant de l'extérieur, ce qui nécessite généralement l'usage de cryptographie. De plus, nous avons remarqué que ces systèmes de gestion de confiance nécessitent eux-mêmes une base cryptographique sûre pour garantir de l'authenticité et l'intégrité des communications entre agents.

La raison évoquée par les auteurs des différents travaux concernant le peu de travaux sur le sujet de l'usage de cryptographie dans les SMAe est la difficulté de mise en place d'une infrastructure cryptographique dans tels systèmes décentralisés. Nous avançons néanmoins que cela est nécessaire et cherchons à proposer une solution à ce problème.

3 Architecture de sécurité

L'objectif de ces travaux est de fournir la description d'une architecture de sécurité pour SMAe. Comme nous l'avons indiqué précédemment, nous trouvons qu'une solution cryptographique décentralisée est nécessaire à l'utilisation d'outils, tous aussi nécessaires, tels que les systèmes de gestion de confiance.

L'architecture que nous envisageons de proposer doit pouvoir assurer que chaque agent possède une identité et que ses communications soient sécurisées. Plus précisément, dans le cas d'un échange entre deux agents, la confidentialité de l'échange doit être maintenue, les atteintes à l'intégrité

de la communication détectées, l'identité de l'interlocuteur vérifiée et la répudiation des envois impossible.
Dans le cas d'une communication unidirectionnelle entre un agent et un ensemble d'agents, les atteintes à l'intégrité de sa transmission doivent être détectée, son identité liée à cette dernière et la répudiation de l'envoi impossible.

4 Infrastructure

Pour répondre à ces besoins, l'utilisation d'une PKI à certificats semble être le plus approprié.
Chaque agent génèrera une paire de clefs et les utilisera pour signer ses communications ainsi que pour engager des processus d'établissement des clefs de chiffrement éphémères. Son identité sera liée à sa clef publique qu'il devra faire certifier par une autorité afin de pouvoir se connecter au système.
Néanmoins, l'absence de serveurs centraux rend l'implémentation d'une PKI traditionnelle difficile puisqu'il n'y a pas de tierce partie de confiance candidate au rôle d'autorité de certification, d'autorité d'enregistrement et spécialement d'autorité de certification racine.
De plus, n'ayant pas de média de communication structuré, assurer une communication portant sur tout le système est difficile et coûteux.

4.1 Gestion des certificats

Certification et stockage Dans le cas où il n'est pas possible d'avoir une confiance absolue dans les autorités, il semble intéressant pour les agents de posséder plusieurs certificats. Si une autorité n'est plus jugée comme digne de confiance, ses certificats perdront leur valeur et de nouveaux seraient nécessaires pour les remplacer.
Par défaut, chaque autorité aura la responsabilité de stocker et partager (sur demande ou gratuitement) les certificats valides qu'elle aura signés, mais on pourrait imaginer donner cette tâche à certains autres agents du système.

Révocation La révocation d'un certificat, et donc l'exclusion d'un agent sera menée d'une part par l'ajout de l'identité de l'agent à une liste de révocation, et d'autre part, par l'utilisation de certificats à courte date d'expiration qui ne seraient pas renouvelés. La première méthode est rapide et directe mais seule la seconde est définitive.

Usurpation et conservation d'identité Dans un système hétérogène et ouvert, les agents n'ont pas forcément d'à priori sur leurs pairs, il est donc possible de réduire l'identité d'un agent à sa clef publique et ainsi prévenir toute tentative d'usurpation d'identité. Il reste néanmoins une nécessité de coopération entre autorités de certification pour assurer la conservation des identités afin de s'assurer que deux agents ne s'enregistrent pas avec la même clef en deux points différents du système, même si cela est très peu probable.

4.2 Intégration dans un système de gestion de confiance

Afin de proposer une architecture de sécurité plus transversale, le système de gestion de confiance pourrait être en partie intégré dans notre PKI. On peut notamment imaginer les autorités ajouter la valeur de confiance qu'elles ont en les agents certifiés dans leurs certificats. De même, une autorité dont la confiance baisserait pourrait perdre son statut. L'exclusion d'un agent par le système, de par sa réputation trop basse, pourrait aussi se concrétiser par la révocation de son certificat et le refus de lui en attribuer un nouveau.

5 Scénarios d'usage

Scénario A Lors de la mise en place du système, plusieurs agents sont déployés avec l'objectif de servir d'autorité de certification. Ce scénario est le plus simple à mettre en œuvre mais limite l'autonomie du SMAe et peut mener à un goulot d'étranglement ou de point de défaillance unique si le nombre d'autorités de certification n'est pas correctement calibré.
Une partie des agents se voit attribuer le rôle d'autorité de certification. Ces agents auront un rôle important d'un point de vue de sécurité et il serait préférable qu'ils soient les mieux équipés pour réaliser leurs tâches dans le système. Tout agent souhaitant entrer dans le système génèrera une paire de clefs et fera la demande de certificat à, au moins, une autorité. Une fois certifié, l'agent pourra prendre part aux communications du système.

Scénario B Lors de la mise en place du système, aucun agent n'est déployé avec l'objectif de servir d'autorité de certification. Ce scénario, plus « sauvage » ou « libertaire », offre des possibilités de résilience plus grande une fois le challenge du choix et du maintien de la confiance dans les autorités de certification résolu.
Tout comme dans le scénario A, il faut pouvoir s'assurer de la fiabilité de l'autorité délivrant un certificat. Il faut pouvoir motiver les agents à devenir autorité de certification. Cela peut être une obligation dans le comportement des agents dans un premier temps et être lié à un système de réputation dans un second.

6 Conclusion

Ces travaux se poursuivent actuellement et devraient permettre d'obtenir une approche de gestion décentralisée des clefs cryptographiques, adaptée aux SMAe. Une étude de cas sur une application de drones autonomes est prévue comme preuve de concept.

Remerciements

Ce travail a bénéficié d'une aide de l'État gérée par l'Agence Nationale de la Recherche au titre du programme « Investissements d'avenir » portant la référence ANR-15-IDEX-02.

Références

[1] Arthur Baudet, Oum-El-Kheir Aktouf, Annabelle Mercier, and Philippe Elbaz-Vincent. Systematic mapping study of security in multi-embedded-agent systems. *IEEE Access*, 9 :154902–154913, 2021.

[2] Jean-Guillaume Dumas, Pascal Lafourcade, and Patrick Redon. *Architectures de sécurité pour internet-2e éd. : Protocoles, standards et déploiement*. Dunod, 2020.

Panorama de Constraint Answer Set Programming

B. Garreau[1], M. Dieguez Lodeiro[1], E. Monfroy[1], I. Stéphan[1]

[1] Université d'Angers, LERIA

{bryan.garreau, martin.dieguezlodeiro, eric.monfroy, igor.stephan}@univ-angers.fr

Résumé

Constraint Answer Set Programming (CASP) est un paradigme fusionnant Answer Set Programming (ASP) et la programmation par contraintes. CASP a gagné en popularité ces dernières années et beaucoup de techniques de résolutions ont émergé. Dans ce court article nous allons vous présenter quelques une des méthodes de résolution les plus renommées et nous allons voir les avantages et désavantages de celles-ci avant d'aborder les futures pistes de recherches que nous aimerions explorer.

Mots-clés

Constraint Answer Set Programming, Answer Set Programming, Constraint Programming, Knowlege Representation, NonMonotonic Reasoning

1 Introduction

Le paradigme *Constraint Answer Set Programming* (CASP) consiste en un langage déclaratif à la fois riche et simple pour modéliser des problèmes complexes et des mécanismes pour les résoudre. Ce nouveau paradigme est issu de la fusion de deux paradigmes d'intelligence artificielle, *Answer Set Programming* et *Constraint Programming* dont les origines remontent aux travaux de Mellarkod et al. [10]. Cette fusion permet de tirer partie des avantages des deux paradigmes pour obtenir un paradigme plus déclaratif et des solveurs plus performants.

Answer Set Programming (ASP) [5] est un langage de programmation utilisé en représentation des connaissances, en raisonnement non monotone mais aussi pour résoudre des problèmes combinatoires difficiles. Les solveurs ASP acceptent des programmes avec des variables qui doivent être substituées par des symboles propositionnels avant la phase de résolution. Le processus qui consiste à remplacer ces variables s'appelle le *grounding*. Bien qu'il existe des approches qui combinent la phase de *grounding* avec la phase de résolution [9], les solveurs les plus utilisés le font séparément [8].

Tandis que les approches avec *grounding* sont devenues très efficaces ces dernières années, les programmes propositionnels obtenus à l'issue de cette phase peuvent être très volumineux à cause de l'explosion combinatoire des substitutions, particulièrement lorsque les variables s'étendent sur les domaines $\mathbb{N}$, $\mathbb{R}$, etc... Le *grounding* est souvent un frein au processus de résolution. Dans le but d'améliorer et de raffiner le *grounding*, des techniques de programmation par contraintes peuvent être appliquées lors du *grounding* [10] ce qui permet d'éviter la génération de clauses inutiles. Afin d'illustrer le problème du *grounding*, considérons l'exemple suivant :

Exemple 1 *Considérons un circuit électrique composé de deux appareils : un interrupteur et une lampe. L'interrupteur peut être* allumé *(`switchOn`) ou* éteint *(`swicthOff`) de manière non-déterministe. L'état de la lampe (*on*/* off*) est directement déduit de l'état de l'interrupteur. Le prédicat `light` représente la présence de lumière dans l'environnement d'après l'état de l'interrupteur et l'heure de la journée. La proposition `night` représente la partie de la journée ou il n'y a pas de lumière naturelle (entre 22h00 et 7h00) et la proposition `sleep` est une conséquence directe de `switchOff` et `night`.*

Un programme ASP de cet exemple est présenté dans la Liste 1. Le programme *ground* proposé par le grounder `Gringo` contient 40 règles (en extension ou intension).

Liste 1 – Exemple de programme ASP

```
time(0..23).
1 {ctime(X) : time(X)} 1.
switchOn :- not switchOff .
switchOff :- not switchOn .
light :- switchOn.
light :- not night.
night :- X<7, ctime(X).
night :- X>=22, ctime(X).
sleep :- switchOff,night.
```

Constraint Programming (CP) [2] est un autre paradigme de programmation qui permet de modéliser des problèmes sous la forme de triplets (X, D, C) où X est l'ensemble des variables, D représente les domaines associés aux variables et C est l'ensemble des contraintes qui représentent les relations entre les variables. La particularité de CP est l'utilisation d'algorithmes performants de propagation des contraintes permettant de réduire considérablement l'espace de recherche à explorer lors de la résolution d'un problème. CP profite de plus d'un catalogue de contraintes globales qui permettent d'utiliser des algorithmes encore plus efficaces sur des problèmes précis. Les contraintes globales sont un atout majeur de CP puisqu'elles permettent

de rendre un programme plus concis et plus efficace.

Le principal problème des solveurs ASP actuellement est la phase de *grounding* qui peut être lente et produit des modèles qui sont inutiles. Ainsi ASP est souvent peu performant sur les problèmes comprenant un grand nombre de données. L'un des intérêts de CASP est d'apporter des techniques de filtrage et de propagation de contraintes pour optimiser cette phase de *grounding* et rendre possible la résolution de problèmes ASP contraints. C'est notamment le cas de problèmes industriels d'ordonnancement [1][3].

Dans ce papier nous allons vous présenter deux des principales méthodes de résolution de problèmes CASP ainsi que leurs avantages et inconvénients.

2 Approches pour CASP

Parmi les différents solveurs CASP, le plus populaire est Clingcon 3 [4]. Ce solveur n'effectue pas la substitution des variables utilisées dans les contraintes et les traite pendant la phase de résolution. Cette simplification du *grounding* permet de réduire le temps de calcul mais également d'économiser de la mémoire. La phase de résolution bénéficie donc d'algorithmes de CP pour traiter plus efficacement les contraintes. Toutefois, la propagation ne s'applique pas aux contraintes natives à ASP mais seulement aux contraintes sur des variables à domaines finis.

Par exemple, le problème dans la Liste 1 peut être représenté dans le langage de Clingcon 3 par le programme en Liste 2. Dans le premier programme il faut utiliser une règle de choix pour représenter le temps actuel alors que dans le deuxième exemple une variable `x` avec un domaine fini représente le temps actuel. Le *grounding* en est simplifié et ne produit que 8 règles, ce qui est beaucoup moins que les 40 initiales.

Liste 2 – Exemple de programme CASP

```
&dom{0..23} = x.
switchOn :- not switchOff .
switchOff :- not switchOn .
light :- switchOn.
light :- not night.
night :- &sum{x} < 7.
night :- &sum{x} >= 22.
sleep :- switchOff,night.
```

Il est également possible d'écrire des programmes CASP contenant des contraintes globales ce qui les rend plus concis et plus lisibles. De plus, cela permet d'utiliser des algorithmes adaptés pour rendre le traitement de ces contraintes plus efficient. Cependant, cette méthode de résolution, bien qu'efficace, fait une distinction entre les contraintes issue de CP et les contraintes ASP ce qui limite le traitement des variables des contraintes et la coopération entre les méthodes de résolution.

D'autres approches visent à traduire des programmes CASP en problème de *Satisfiability Modulo Theories* (SMT) [12]. Les solveurs SMT traitent des problèmes de satisfiabilité qui contiennent des formules logiques plus complexes comprenant des entiers, des réels ou des structures de données. Ces formules plus complexes appartiennent à des théories et sont traitées par des solveurs de théories pendant la résolution. Les solveurs de SMT, devenus très performants, permettent de résoudre des problèmes CASP de manière efficace. De plus, SMT est flexible et il est possible de rajouter des théories pour traiter des contraintes plus efficacement.

Plusieurs approches devenues récemment populaires ont essayé de donner une sémantique logique aux programmes CASP. Basées sur l'*equilibrium logic* [11], Cabalar et al [6] l'ont étendu pour traiter les contraintes et les agrégats. De plus, Eiter et al [7] combinent l'*equilibrium logic* avec la *weighted logic* pour représenter des contraintes algébriques. Toutes ces approches permettent de traiter les contraintes d'un point de vue logique.

3 Conclusions et ligne de travail

La résolution ASP est une approche de *Generate and Test* alors que la résolution CP utilise une approche *Constraint and Generate*. Cette différence entre les méthodes de résolution rend leur fusion prometteuse mais complexe. CASP est un paradigme assez récent et plusieurs méthodes de résolution ont été proposées. Cependant, la plupart des solveurs sont construits autour de solveurs ASP ou traduisent les programmes CASP dans d'autres langages (en problème SMT par exemple).

L'utilisation de CASP est principalement motivée par les limitations des solveurs ASP actuels tel que le *grounding*. Il est néanmoins possible d'étendre ASP pour traiter par exemples des nombres flottants qui ne sont pas traités nativement par ASP. Cette augmentation du langage permettrait de rendre ASP plus déclaratif qu'auparavant.

Dans le futur il pourrait être utile de s'intéresser à une coopération plus forte entre les solveurs ASP et de CP de façon à réellement les fusionner, et ce de manière plus transparente afin de mieux tirer parti des deux approches.

Références

[1] Dirk Abels, Julian Jordi, Max Ostrowski, Torsten Schaub, Ambra Toletti, and Philipp Wanko. Train scheduling with hybrid ASP. In *International Conference on Logic Programming and Nonmonotonic Reasoning*, pages 3–17. Springer, 2019.

[2] Krzysztof Apt. *Principles of constraint programming*. Cambridge university press, 2003.

[3] Marcello Balduccini. Industrial-size scheduling with ASP + CP. In *International conference on logic programming and nonmonotonic reasoning*, pages 284–296. Springer, 2011.

[4] Mutsunori Banbara, Benjamin Kaufmann, Max Ostrowski, and Torsten Schaub. Clingcon : The next generation. *TPLP*, 17(4) :408–461, 2017.

[5] G. Brewka, T. Eiter, and M. Truszczyński. Answer set programming at a glance. *Communications of the ACM*, 54(12) :92–103, 2011.

[6] Pedro Cabalar, Roland Kaminski, Max Ostrowski, and Torsten Schaub. An ASP semantics for default reasoning with constraints. In Subbarao Kambhampati, editor, *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 1015–1021. IJCAI/AAAI Press, 2016.

[7] Thomas Eiter and Rafael Kiesel. ASP($\mathcal{AC}$) : Answer Set Programming with Algebraic Constraints. *Theory Pract. Log. Program.*, 20(6) :895–910, 2020.

[8] Roland Kaminski, Javier Romero, Torsten Schaub, and Philipp Wanko. How to build your own ASP-based system ? ! *CoRR*, abs/2008.06692, 2020.

[9] Claire Lefèvre, Christopher Béatrix, Igor Stéphan, and Laurent Garcia. Asperix, a first-order forward chaining approach for answer set computing. *Theory and Practice of Logic Programming*, 17(3) :266–310, 2017.

[10] Veena S. Mellarkod, Michael Gelfond, and Yuanlin Zhang. Integrating answer set programming and constraint logic programming. *Ann. Math. Artif. Intell.*, 53(1-4) :251–287, 2008.

[11] D. Pearce. A New Logical Characterisation of Stable Models and Answer Sets. In *Proc. of Non-Monotonic Extensions of Logic Programming (NMELP'96)*, pages 57–70, Bad Honnef, Germany, 1996.

[12] Benjamin Susman and Yuliya Lierler. SMT-based constraint answer set solver EZSMT (system description). In *Technical Communications of the 32nd International Conference on Logic Programming (ICLP 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

Vers une plate-forme de réalité mixte pour les robots mobiles autonomes

I. Argui[1], M. Gueriau[1], S. Ainouz[1]

[1] Normandie Univ, INSA Rouen, LITIS

Résumé

L'entraînement des robots mobiles à la navigation autonome requiert la simulation de multiples scénarios variés auxquels le robot n'est pas habitué. Par conséquent, le transfert d'algorithmes de la simulation vers la réalité peut s'avérer risqué dû à l'écart entre la réalité et la simulation. Dans cet article, nous développons une première version d'une plate-forme de réalité mixte pour les robots mobiles dont la perception est basée sur la vision. Les premiers tests permettent à un robot mobile de visualiser la fusion de deux environnements synchronisés à travers une caméra RGB-D durant la navigation.

Mots-clés

Réalité mixte, robotique mobile, simulation.

Abstract

Training mobile robots to autonomous navigation requires simulation in different scenarios the robot is not familiar with. The transfer of the algorithms from virtual world to reality can be risky due to the reality gap. In this paper, we propose a mixed-reality framework for mobile robots based on RGB-D cameras. The first tests enable a 2-wheeled robot to visualize the fusion of two synchronized environements during navigation.

Keywords

Mixed-reality, mobile robotics, simulation.

1 Introduction

La robotique mobile est un secteur de recherche qui bénéficie des avancées de l'intelligence Artificielle (IA) pour permettre aux robots d'évoluer dans leur environnement avec toujours plus d'autonomie. Les applications de ce domaine sont variées : industrie manufacturière, automobile, exploration planétaire, secteur médical, etc. Les robots mobiles autonomes peuvent s'appuyer sur des algorithmes d'apprentissage automatique pour apprendre et réaliser des tâches spécifiques (navigation autonome, évitement d'obstacles, cartographie, détection d'objets, etc.). Ils suivent alors le processus traditionnel : *(i)* pré-entraînement du modèle d'apprentissage en simulation, *(ii)* portage du modèle pré-entraîné au robot réel, *(iii)* ajustement des paramètres en conditions réelles. L'usage de la simulation est parfois indispensable pour tester certaines situations critiques (dangereuses ou coûteuses), ou difficiles à contrôler. Un problème classique rencontré dans ce type d'approche est que les conditions d'entraînement sont différentes entre l'environnement de la simulation et celui du monde réel. Cet écart est du à la différence des données reçues par les capteurs en simulation et en réalité. L'une des solutions proposées dans la littérature afin de remédier à ce problème est l'entraînement des robots dans des environnements à réalité mixte [2] (RM). La RM consiste à fusionner le monde réel et le monde virtuel pour former un seul environnement dans lequel l'agent pourra agir et percevoir à la fois des objets réels et d'autres virtuels. Ainsi, il devient possible d'ajouter des obstacles dynamiques (piétons, agents) avec lesquels le robot peut interagir sans prendre de risques coûteux. Dans cet article, nous proposons une plate-forme de réalité mixte pour les robots mobiles autonomes dont la perception repose sur de la vision.

2 Travaux passés

La RM est utilisée dans différents champs d'études : éducation, chirurgie, jeux vidéos, architecture [3]. En robotique mobile, il existe plusieurs applications où il est possible de faire appel à la RM (*e.g.* pour le suivi de véhicule virtuel [1]). L'une des premières plate-formes RM réalisée pour les robots mobiles a été présentée par Chen *et al.* [2]. Le principe consiste à fusionner les perceptions des données LIDAR (télémètre laser) de l'environnement réel avec celles de l'environnement virtuel. Des travaux plus récents ont appliqué ce principe sur un véhicule autonome dans le but de traiter des scénarios à risque comme la traversée d'un piéton et d'évaluer le comportement du véhicule. Après la fusion des données reçues par le LIDAR (réel et virtuel), le véhicule est capable de détecter le piéton introduit dans le monde virtuel [4]. Ces travaux, bien qu'ils montrent le potentiel de la RM, se limitent à des données LIDAR, dont la capacité à interpréter les données de scène est limitée, et ne permettent pas d'étudier des plate-formes de robots dont la perception est basée sur la vision. On propose dans cet article un travail préliminaire de réalisation d'une plate-forme RM pour les robots mobiles en utilisant les caméras RGB-D, introduite dans l'outil de simulation Gazebo.

3 Plate-forme de RM pour les robots mobiles autonomes

Habituellement, une plate-forme RM utilise les composants suivants : 1) un environnement réel, 2) un environnement

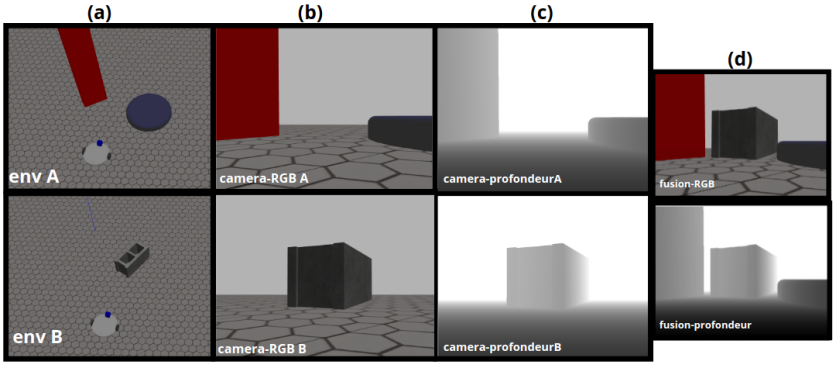


FIGURE 1 – Capture d'écran d'une simulation dans la plateforme de RM. (a) environnement du robot (b) perception du robot en couleur (c) perception du robot en profondeur (d) résultat de la stratégie d'augmentation (environnement en réalité mixte).

virtuel, 3) un robot physique, et 4) un avatar virtuel qui reproduit tous les mouvements du vrai robot. Nous présentons dans cette section une première version de la plate-forme RM développée.

3.1 Outils utilisés

Le simulateur utilisé est Gazebo, choisi pour sa simplicité et sa flexibilité. Deux environnements différents ont été crées, dans lesquels le robot et son avatar peuvent naviguer. La communication entre les différents noeuds s'effectue à travers ROS. Le robot (et son avatar virtuel) sont dotés d'une caméra RGB-D capable de percevoir les images en couleurs ainsi que les informations de profondeur (distance entre chaque élément perçu et la caméra).

3.2 Stratégie d'augmentation

Les images reçues par la caméra sont sous le format de `sensor-msg/Image`, (format propre à ROS). Afin de manipuler ces images, elles ont été d'abord converti en format *OpenCv* grâce à la librairie *CvBridge*. Ensuite, un nouveau noeud ROS permet : a) la souscription aux données de profondeur reçues par les deux caméras, b) la détermination de la différence entre l'image A et l'image B, c) l'ajout de cette différence à l'image A. Cette stratégie d'augmentation permet au robot de percevoir, dans une même image, à la fois les objets présents dans son environnement mais aussi ceux de l'environnement de l'avatar virtuel.

3.3 Résultats obtenus

Pour réaliser une première validation du prototype, nous considérons que les deux environnements sont virtuels : le robot que l'on commande est le robot A, et son jumeau numérique (avatar virtuel) est le robot B. La stratégie d'augmentation est appliquée aux données de la caméra de A. Le résultat final est représenté dans la figure 1 : la partie gauche illustre les deux environnements, où objets variés sont placés dans des positions différentes. On peut voir les images obtenues par chaque caméra séparément dans un premier temps, puis le résultat de la fusion des deux caméras à droite : c'est ce que le robot A perçoit. Un algorithme de contrôle simplifié assurant le couplage du mouvement des deux robots permet de visualiser le résultat de la stratégie d'augmentation en temps-réel.

4 Conclusion et perspectives

Dans cet article, nous avons présenté la première étape vers la réalisation d'une plate-forme basée sur la réalité mixte pour les robots mobiles autonomes. Le prototype réalisé nous permet d'envisager à présent l'utilisation d'un robot réel capable de naviguer dans un environnement physique tout en interagissant avec les obstacles virtuels. Pour cela, nous allons remplacer le robot présent dans l'environnement A par le robot réel, et son avatar suivra ses mouvements au fur et à mesure dans la plate-forme. Cela nécessitera d'avoir un suivi et un couplage des positions/orientation des robots précis (qui pourrait se reposer sur des capteurs débarqués). Par la suite, la plate-forme pourra être utilisée pour entraîner des IA embarquées dans des scénarios critiques qui mixeront environnement (de navigation) réel et obstacles (piétons, véhicules). Nous envisageons d'étudier des approches par apprentissage par transfert, permettant à un véhicule autonome de partager son modèle d'apprentissage avec un ou plusieurs robots virtuels et/ou d'explorer différents environnements en parallèle.

Références

[1] R. Baruffa, J. Pereira, P. Romet, F. Gechter, and T. Weiss. Mixed reality autonomous vehicle simulation : Implementation of a hardware-in-the-loop architecture at a miniature scale. 10 2020.

[2] I. Y.-H. Chen, B. MacDonald, and B. Wunsche. Mixed reality simulation for mobile robots. In *2009 ICRA*, pages 232–237, 2009.

[3] S. Rokhsaritalemi, A. Sadeghi-Niaraki, and S.-M. Choi. A review on mixed reality : Current trends, challenges and prospects. *Applied Sciences*, 10(2), 2020.

[4] M. R. Zofka, M. Essinger, T. Fleck, R. Kohlhaas, and J. M. Zollner. The sleepwalker framework : Verification and validation of autonomous vehicles by mixed reality LiDAR stimulation. In *2018 SIMPAR*, pages 151–157.

Towards Considering Explicit Sensitivity to Augmentation in Visual Instance Discrimination Tasks

A. Devillers[1], M. Lefort[1]

[1] University Lyon 1 Claude Bernard, LIRIS

alexandre.devillers@liris.cnrs.fr
mathieu.lefort@liris.cnrs.fr

Résumé

Les méthodes récentes d'apprentissage autosupervisé de représentations visuelles sont basées sur des tâches de discrimination d'instance visant à apprendre des représentations non triviales insensibles à un ensemble d'augmentations soigneusement choisi. Les performances de ces méthodes se rapprochent rapidement des approches supervisées, et les surpassent même dans certains cas, et ce, sans supervision experte. Cet article donnera un aperçu général de ces méthodes et discutera d'une direction de recherche visant à inclure de la sensibilité à certaines augmentations.

Mots-clés

Apprentissage de représentation visuelle, apprentissage profond autosupervisé, discrimination d'instance.

Abstract

Recent methods of self-supervised visual representation learning are based on instance discrimination tasks aiming at learning non-trivial representations insensitive to a carefully chosen set of augmentations. These methods are closing the gap with the supervised approaches, even outperforming them in some cases, while having the advantage of not requiring expert supervision. This paper will overview some of these methods, and discuss a research direction aiming at including sensitivity to some augmentations.

Keywords

Visual representation learning, deep self-supervised learning, instance discrimination.

1 Introduction

Learning pertinent visual representations is a crucial and challenging problem to achieve good performance on downstream tasks while allowing for better data efficiency. Learning such representations in a self-supervised manner, *i.e.* without human supervision, allows the use of plentiful raw data, opening the application of deep learning to domains suffering from a lack of annotations. Yet, self-supervised learning requires finding a supervisory signal obtainable from the data.

Recent successful approaches in visual representation learning are based on instance discrimination tasks, and more precisely on building augmentations-invariant embeddings [1, 2, 3, 5, 6]. These methods are closing the gap with the supervised approaches, even being competitive in some cases as few-shot learning. Still, learning such an invariant property encourages embeddings to be insensitive to augmentations, *i.e.* to leave the information modifiable by the transformations. This way, the set of possible transformations has to be carefully selected. For instance, if one requires color information to be in the representations, then the set of transformations should avoid color manipulation. Sec. 2 will briefly overview some of these recent methods relying on an invariance task, while in Sec. 3 we will identify why using only insensitivity may be sub-optimal, and show, based on recent work and our preliminary results, how explicit sensitivity can be beneficial to representations. Finally, Sec. 4 will stand for the perspectives and future works we aim for.

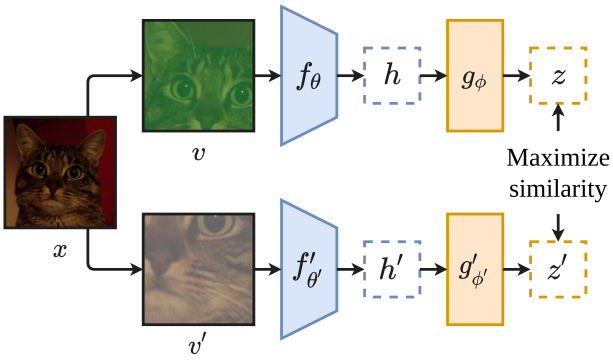


FIGURE 1 – Common base shared by recent methods. The input image x is augmented in two ways to give the views v and v'. These views are then passed through an encoder to give respectively the representations h and h', which are then passed in a projection head that outputs respectively the embeddings z and z' on which the loss is applied.

2 Existing Methods

Recent methods are mostly siamese networks with an instance discrimination task [1, 2, 3, 5, 6]. They aim to build a latent space where two augmentations of the same image

have similar embeddings, see Fig. 1, while using various tricks to avoid simple solution collapse. Thus, the resulting embeddings are insensitive to the possible augmentations, requiring the set of used augmentations to be carefully chosen, as specified in the previous section. Current state-of-the-art methods are almost all using the same set of transformations, which have been constructed experimentally by testing multiple combinations [2]. These transformations are strong enough to make the views very different at the pixel level (i.e. crop, color jitter, etc.), while preserving the original semantic information of the source image, thus forcing the representations to encode this shared semantic to be similar.

On top of this, the embeddings on which the invariance loss is applied are not directly the representations, but a non-linear projection of the representations, as this has shown to improve performance. One hypothesis is that it could allow some augmentation-related information to be present in the representations, as this projection could filter it before the loss. This projection can be seen as an invariance-head, taking the form of a multi-layer perceptron, which is only used by the invariance task during representation learning.

3 Research Hypothesis

The importance of the selected augmentations, and the final projection applied to the representations, shows us that we can separate augmentations into two groups : the ones for which the representations benefit from insensitivity (crop, color jitter, etc.), and the ones for which sensitivity is beneficial (rotation, vertical flip, etc.), more details can be found in [4]. Moreover, this separation experimentally seems the same for all recent methods performing augmentation-invariance tasks. While a large number of recent methods have simply ignored the augmentations requiring sensitivity [1, 2, 3, 5, 6], only one, to the best of our knowledge, has tried to add explicit sensitivity to these augmentations while keeping an invariance task [4].

This last method consists of a framework that proposes to add an extra head with a second task that predicts the transformation, i.e. rotation, that has led to a given view based on its representation. It can be seen as image invariance, as two images augmented in the same way are classified similarly, whereas recent methods perform augmentation invariance. Consequently, it forces the model to encode augmentation-related information into the representations, making them sensitive to these augmentations. Therefore, the transformations used for this task have to be the ones for which sensitivity has shown to be beneficial, such as rotation or vertical flip. This addition has demonstrated to improve existing methods of the state of the art such as [2, 3, 6].

On our side, we have also explored the same idea of using an extra head with a second task to add an explicit sensitivity to some augmentations. We have started experimenting with a task of non-trivial equivariance, aiming at building a latent space in which displacements caused by augmentations in the image space are significant. Therefore, we differ from recent methods that rely on invariance, where such displacements are null. We also differ from [4] as we do not have image invariance and as we structure the latent space so that the displacements are predictable from the augmentations parameters. Altogether, our addition guarantees that some augmentation-related information is present and structured in the representations. Preliminary results using SimCLR [2] as a baseline, on which we have added our extra head and task, have shown to lower the error from around 9% to 7.5% on CIFAR10 using the usual linear classification evaluation.

4 Discussion and Perspectives

We believe explicit augmentation sensitivity has been under-explored in recent visual representation learning methods. We also believe that some discarded augmentations could benefit a non-invariant task. Following recent work and our preliminary results, we suggest that such explicit augmentation sensitivity may be a good research direction and could lead to more state-of-the-art improvements in visual representation learning. Our future work will aim at scaling our experiments to more baselines such as BYOL [5] and more datasets such as ImageNet. We also plan to perform few-shot evaluations while experimenting with the set of augmentations for which explicit sensitivity is significantly beneficial.

Acknowledgements

This work was granted access to the HPC resources of IDRIS under the allocation 2021-AD011013160 made by GENCI.

Références

[1] A. Bardes, J. Ponce, and Y. LeCun. Vicreg : Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv :2105.04906*, 2021.

[2] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[3] X. Chen and K. He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021.

[4] R. Dangovski, L. Jing, C. Loh, S. Han, A. Srivastava, B. Cheung, P. Agrawal, and M. Soljačić. Equivariant Contrastive Learning. *arXiv preprint arXiv :2111.00899*, 2021.

[5] J-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, and M. G. Azar. Bootstrap your own latent : A new approach to self-supervised learning. *arXiv preprint arXiv :2006.07733*, 2020.

[6] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny. Barlow twins : Self-supervised learning via redundancy reduction. *arXiv preprint arXiv :2103.03230*, 2021.

Session "IA & algorithmes"

Building an Operable Graph Representation of a Java Program as a basis for automatic software maintainability analysis

Sébastien Bertrand[1,2], Pierre-Alexandre Favier[1,3], and Jean-Marc André[1,3]

[1]IMS Laboratory, University of Bordeaux, UMR 5218 CNRS, France
[2]onepoint, Sud-Ouest, France
[3]ENSC, Bordeaux INP, France

s.bertrand@groupeonepoint.com, {pierre-alexandre.favier, jean-marc.andre}@ensc.fr

Résumé

Dans le cadre d'un projet de recherche concernant l'évaluation de la maintenabilité logicielle en collaboration avec l'équipe de développement, nous nous sommes intéressés à l'utilisation fréquente de métriques comme prédicteurs. De nombreuses métriques existent, souvent avec des implémentations opaques et discutables. Nous affirmons que les métriques mélangent l'évaluation de la présentation, de la structure et du modèle. Afin de se concentrer sur les vrais défauts de maintenabilité détectables, nous avons calculé des métriques uniquement basées sur la structure du programme. Notre approche a consisté à analyser le code source de programmes Java *comme un graphe, et calculer les métriques dans un langage de requête déclaratif. À cette fin, nous avons développé* Javanalyser *et implémenté 34 métriques en utilisant* Spoon *pour analyser les programmes* Java*, et* Neo4j *comme base de données de graphes. Nous allons montrer que le graphe de programme constitue une base solide pour calculer les métriques et mener de futures études d'apprentissage automatique pour évaluer la maintenabilité.*

Mots-clés

Maintenabilité logicielle, Analyse de programme, Graphe de programme.

Abstract

As a part of a research project concerning software maintainability assessment in collaboration with the development team, we were interested in the frequent use of metrics as predictors. Many metrics exist, often with opaque and arguable implementations. We claim metrics mix the assessment of presentation, structure and model. In order to focus on true detectable maintainability defects, we computed metrics solely based on the structure of the program. Our approach was to parse the source code of Java *programs as a graph, and to compute metrics in a declarative query language. To this end, we developed* Javanalyser *and implemented 34 metrics using* Spoon *to parse* Java *programs and* Neo4j *as graph database. We will show that the program graph constitutes a steady basis to compute metrics and conduct future machine-learning studies to assess maintainability.*

Keywords

Software Maintainability, Program Analysis, Program Graphs.

1 Introduction

Software maintainability is paramount to reducing the cost of systems in time [15, 17]. Our research project concerns software maintainability assessment working in collaboration with the development team [13]. As part of this effort, we began to reproduce a study from Schnappinger *et al.* [33], because their work is based on a recent, high quality software maintainability dataset [32]. This introduction presents what maintainability and metrics are, before continuing on the importance of the program structure and having clear and unambiguous metrics. Finally, we will present *Javanalyser*, the tool we developed to answer our research questions.

Maintainability is defined as the efficiency with which the software can be corrected, improved or adapted to changes in the system or in the specifications, either technical or functional. According to the ISO 25010 [25], maintainability is composed of five subcharacteristics:

- Modularity: degree of decoupling between components;
- Reusability: degree of potential reuse of a component;
- Analysability: degree to which the implementation of a component can be understood and debugged;
- Modifiability: degree to which a component can be modified without introducing defects in other components;
- Testability: degree to which a component can be tested against a set of technical and functional specifications.

Most studies try to predict maintainability by using metrics as predictors [11, 20]. There is a great number of existing

metrics [19], which can be broadly categorized in product metrics relating to the structure of the software, for example the number of lines of code, and process metrics relating to the activity of developing the software, for example the number of hours needed to correct a bug [22]. Product metrics are trying to pinpoint the intrinsic cause of maintainability defects, and process metrics are extrinsic giveaways of these defects. We can metaphorise that product metrics are the disease while process metrics are the symptoms.

The development of a software in general, and maintainability in particular, can also be approached from three angles:

- The presentation of the source code, encompassing all the style rules applied to write the program, for instance how many classes there is per file, where a blank line should be inserted for clarity, or the naming convention of classes and methods;
- The structure of the source code, characterized by the components of the program, their responsibilities, and the relationships that exist between them;
- The model of the problem the program is trying to solve, which can be highlighted or not by the organisation of the code, for instance the Domain Driven Design [21] explicitly puts the model forward.

The presentation of the source code can be a true issue in some context, as shown by obfuscation tools that can modify all named elements to reduce readability. But when developing an application, presentation can be put under control, as there exist many tools that enforce formatting and naming convention, such as Prettier [5] for Javascript or Pylint [6] for Python.
The modelisation of the problem to solve is not only related the implementation of the program, but to the whole development process and the business maturity of the development team. Despite being an important stake in software development, problems related to modelisation go beyond the scope of program analysis. A program that perfectly respects the subcharacteristics of maintainability defined above can implement a model completely out of phase with the business, thus making it very difficult to evolve when new requirements emerge [18]. Typically, in this case, product metrics would remain stable and process metrics would drastically increase. The modelisation angle can be viewed as the goal the team is trying to reach, when the goal is wrong for any reason the risks are high to encounter maintainability problems.
On the other hand, the internal structure of the program can be very different for equivalent modelisation of the business (*i.e.* the same functional scope), thus harbouring maintainability flaws. Analysability is obviously related to the presentation of the code, but bad encapsulation or factorisation can also lead to readability issues, such as methods with too many arguments or very big classes difficult to apprehend. It seems legitimate to relate the modularity, reusability and modifiability to the sole structure of the program, because their main concerns are about the components of the program. Testability is related to both the model and the structure implemented by the source code, because tests are designed to check that the implementation (the program) matches the specifications (the model).
Then, however related to the presentation and the model, it seems that the structural design of the program is the main factor impacting the maintainability.

Moreover, while reproducing the study from Schnappinger *et al.* [33], we encountered many problems collecting metrics. Many tools exist and many metrics have been studied [12, 28]. Some studies are based on unmaintained or deprecated tools, and each tool implements metrics computation differently, often with very little documentation available.

We make the assumption that focusing on the structural analysis of a program will allow us to detect predictable maintainability defects, that depend solely on the program and not an external context such as the chosen modelisation. We developed our tool called *Javanalyser*, that parses the abstract syntax tree of a *Java* program and load its structure as a graph within a graph database (*Neo4j* [4]). We focused on having an operable graph and tested it by implementing a set of 34 product metrics as *Cypher* queries [23], leveraging declarative programming to have concise, mutable and explicit definition of metrics. *Javanalyser* is available under the open-source MIT licence. The goal is to build a steady basis to conduct machine-learning studies to assess maintainability.

2 Method

This section explains in detail how we built *Javanalyser* to process a *Java* program and the design choices we made. Basically, our approach was to parse the source code, compute the corresponding graph, and load it into a graph database. When implementing the metrics, we had to manage external references towards the projects' dependencies and define how to walk along the relationships of the graph. Finally, we were able to compute metrics based solely on the graph.

Parsing Java. We used *Spoon* [31] to parse *Java* programs, it produces an abstract syntax tree designed to be both complete and understandable for *Java* developers. Before, we considered two other parsers. At first, we wanted to use directly the *Eclipse Java Development Tools* [1], which is internally used by *Spoon*, but it was very difficult to make it run outside an Eclipse environment, and its documentation is sparse on this issue. Then, we tried JavaParser [2], which was simple to install within our solution. However, code references to external dependencies (typically specified by the `CLASSPATH`) were not properly parsed by its symbol solver in our tests. On the other hand, *Spoon* is able to parse properly all external references within a *Java* program. This point was paramount for being able to compute a graph from the abstract syntax tree (AST), as we must be able to detect that two leaves from the AST are referencing

the same type (a `Class` or an `Enum` in *Java*). An atomic element of the *Spoon* meta-model is a `CtElement`, which we encapsulated within a `ProgramElement` for easy manipulation. Hereafter, we will call "program element" the nodes from the AST produces by *Spoon*.

The graph. Internally, we implemented a `Scanner` in *Javanalyser* to walk through the AST produced by *Spoon* and compute a graph by collecting additional edges between references and referenced elements. The Scanner from *Spoon* do not automatically walk along these relationships to avoid infinite loop when scanning the AST. In this process, we also implemented some simplifications within the graph, for instance:

- we avoided creating vertices for `TypeReference` by linking the referencing program element to the referenced type directly;
- we chose not to specify obvious relationships such as `ThisAccess` which can be inferred from context;
- we avoided creating extraneous vertices for very simple elements, such as a *VariableAccess* without `Cast`, `Annotation` or `Comment`, that is only a reference to a `Variable`.

We only implemented reversible simplifications, that allow to infer the correct AST from the graph. These simplifications were designed to produce a graph that matches more intuitively the code. However, the trade-off of these simplifications was that the graph produced has a database schema depending on the context. We accepted this matter of fact, as our ultimate goal is to assess maintainability, we wanted to produce a graph that matches more closely the "point of view" of a developer.

Graph database. We used *Neo4j* [4] as the graph database. *Neo4j* allows to use a flexible property graph schema. A property graph is a labeled directed multigraph, sometimes called labeled multidigraph. A multigraph allows self-loop edges and parallel edges between nodes. A labeled multidigraph has labeled vertices and arcs. In *Neo4j*, vertices can have multiple labels although edges can only have one label called type. That is why *Neo4j* matched perfectly our requirements, each program element of the AST from *Spoon* being precisely typed (for instance `CtClass` or `CtConstructor`), and having a defined role in its parent's program element (for instance `FOR_INIT` or `EXPRESSION`). The trickiest part was to load the data as quickly as possible, because parsing an actual *Java* program from scratch involves a huge graph.

External references. There are two types of references, internal and external to the parsed *Java* program. Internal references are declared in another part of the parsed *Java* program. External references reference program elements declared in dependencies (sometimes called libraries) used by the parsed *Java* program, typically passed along the `CLASSPATH` variable for a *Java* program. We paid particular attention to parse and identify each of these dependencies. If there is a missing Jar within the `CLASSPATH`, the corresponding external references are flagged as broken references. But if these external references are known, we implemented a walk along their parents' nodes within the AST to provide potential useful additional information. Finally, we flagged external references as "shadow", which is the term used by *Spoon*. Identifying external references is useful when implementing metrics, according to their definition.

Walking the graph. There are many types of relationships within the graph. Because we lost the presentation information conveyed by the segregation of code in files, we had to define how to walk along relationships within the graph. In other words, we wanted to be able to query program elements belonging to a class, leaving apart program elements belonging to other classes. Actually, we classified relationships' types in six sets:

- organisational: describing the structure of modules, packages and classes;
- inner: describing how nodes belong to one another, for instance a local variable belongs to a method which belongs to a class;
- type: linking typed elements to their type (`Class`, `Enum`, ...);
- outer: describing references to potential outer elements, which are typically members of other classes;
- comment: linking program elements to their comment;
- flow: describing the control and data flow of the code.

These sets of relationships were instrumental for the design of the metrics queries.

Metrics. We implemented the computation of metrics in *Cypher* [23], the declarative query language for property graphs associated with *Neo4j*. To ensure that the implementation of our metrics was correct, we qualitatively compared our results with *SonarQube* [7] and *SourceMeter* [8] to detect potential defects within our queries. This whole process helped us to iteratively design *Javanalyser*. Moreover, every metric we planned was successfully implemented and tuned to our expectation. This final task showed the flexibility and versatility of the computation of metrics based on the graph.

3 Results

Foremost, the goal is to build a steady basis to conduct machine-learning studies to assess maintainability. As we focus on the structure of the program, we want to have a presentation-independent representation. This is why we build *Javanalyser* to represent code as a graph, depending only on the structure of the code, and allowing to easily extract data such as metrics. *Javanalyser* parses a *Java* program, produces a *Neo4j* [4] graph, and outputs metrics in a CSV file. The source code

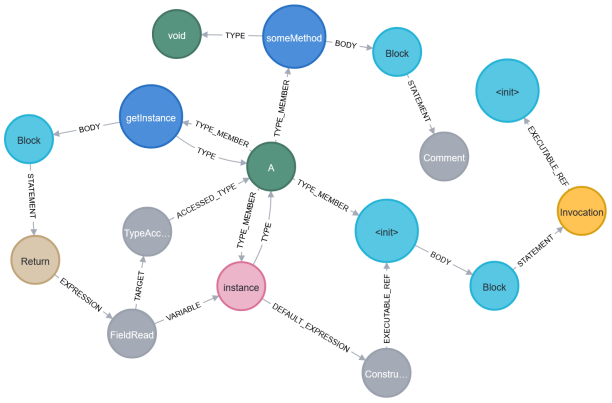


Figure 1: Screenshot of the graph of the singleton pattern

of *Javanalyser* is available under the free (as in freedom) MIT licence on GitLab at `https://gitlab.com/onepoint/research/javanalyser`.

Figure 1 presents an example of the singleton pattern implementation from listing 1. The graph is a labeled directed multigraph. Each node represents an element from the source code. There are 89 existing node labels, for instance there is "`Class`", "`LocalVariable`" or "`While`". Edges represent oriented relationship between nodes, for instance in our example the class "`A`" has four members. There are 93 existing relationship's types, for instance "`TYPE_MEMBER`", "`DEFAULT_EXPRESSION`" or "`EXPRESSION`". All names (node label and relationship type) comes from the *Spoon* [31] meta-model, except "`STATEMENT_ORDER`" which we introduced to obviously keep track of the execution flow within a block.

```
package fr.onepoint.javanalyser.tests.singleton;

public class A {

  private static final A instance = new A();

  private A() {
  }

  public static A getInstance() {
    return instance;
  }

  public void someMethod() {
    // This is a method
  }
}
```

Listing 1: A singleton pattern in *Java*

Thirty-four metrics have been implemented as *Cypher* queries. Listing 2 shows an example with the implementation of the cyclomatic complexity. *Javanalyser* aggregates the results of these metrics for each class of the parsed *Java* program. Then, it produces a CSV file listing classes and associated metrics. Metrics computations are only based on the graph, no pre-computation is done by *Javanalyser*.

```
//Cyclomatic Complexity
CALL {
  MATCH (class:Class)
  OPTIONAL MATCH (class)-[:ANNOTATION|ARGUMENT|ASSIGNED|
    ASSIGNMENT|BODY|CASE|CATCH|CONDITION|
    DEFAULT_EXPRESSION|DIMENSION|ELSE|EXPRESSION|
    FINALIZER|FOREACH_VARIABLE|FOR_INIT|FOR_UPDATE|
    LEFT_OPERAND|NESTED_TYPE|PARAMETER|RIGHT_OPERAND|
    STATEMENT|TARGET|THEN|TYPE_MEMBER|TYPE_PARAMETER|
    VALUE *0..]->(node)
  RETURN class, node
}
WITH class, node
WHERE
  (node:Constructor AND NOT node.implicit)
  OR node:Method
  OR node:AnonymousExecutable
  OR node:If
  OR node:Conditional
  OR node:For
  OR node:ForEach
  OR node:While
  OR node:Do
  OR (node:Case AND EXISTS((node)-[:EXPRESSION]->()))
  OR node:Catch
  OR (node:BinaryOperator AND (node.operator = "AND" OR
    node.operator = "OR"))
RETURN class.id AS id, class.name AS class, count(
    distinct(node)) AS cyc
```

Listing 2: Cyclomatic Complexity Query

Implemented metrics include:

- Number of children [16], which is the number of classes that directly inherit from the class;
- Depth of Inheritance Tree [16], which is the number of parents the class inherits;
- Number of nodes from the class, which is the graph analog of the number of lines of code;
- Maximum of methods' number of nodes;
- Number of nodes within methods;
- Average methods' number of nodes;
- Nesting Level Else-If, which measure complexity as the depth of the maximum embeddedness of its conditional, iteration and exception handling block scopes;
- Cyclomatic complexity [29], which corresponds to the number of linearly independent paths;
- Maximum length of loops;
- Number of loops;
- Number of outgoing invocation, which is actually the number of executable referenced;
- Coupling between objects [16], which counts the number of non-inheritance related classes, *i.e.* the classes that are acted upon by the class or act upon the class (*e.g.* calling methods or holding instance variables);
- Maximum nesting depth;
- Number of methods [27];
- Cognitive complexity [14], which is an evolution of the cyclomatic complexity;
- Number of local attributes and methods declared in the class [27];

- Message-passing coupling [27], which is the number of call statements from the class to other classes;
- Response for a class [16], which is the cardinality of the set of local methods and the methods called by these;
- Lack of cohesion in methods [16, 24], which is the number of sets of methods bound by at least one common instance variable;
- Number of nodes embedded in blocks deeper than 4;
- Data Abstraction Coupling [27], which is the number of variables defined in the class and having an abstract data type.

Moreover, *Javanalyser* outputs two other CSV files. The first lists broken references, which denotes there was some missing dependencies that need to be passed as an argument to the command-line. The second represents the schema of the graph, *i.e.* the effective relationships between nodes' types. These relationships are weighted by counting the number of instances. This weighted schema can be used to compare at a large scale the programming style of different *Java* programs.

Figure 2 presents an extract from the graph produced by parsing *Art of Illusion*[1], which is around 100 kilo lines of code. This extraction presents an overview of the *Java* packages (in brown), and most classes (in green) and interfaces (in pink) from the project. *Javanalyser* parses this project and computes all metrics within approximately 5 minutes on an average computer[2]. A lot of the development effort was put into optimizing the processing speed, ultimately dividing the total execution time by 50. To date, we worked around 130 days and wrote about 1700 lines of code to develop *Javanalyser*, which we shared to the community under the open-source MIT licence. *Javanalyser* is a console application that can be pipelined within a more global process, and more Cypher queries can be easily added. This allows to use this tool for batch processing large datasets such as the *GitHub Java Corpus* [10].

4 Discussion

We designed our work as a unified maintainability-analysis framework, spanning from metrics computation to program representation. This discussion begins with the stakes of the definition of metrics, then we focus on how to count complexity with the case of `Optional` in *Java*. Finally, we will discuss program representation by covering *jQAssistant* [3], ontology-based program analysis, and graph representations.

Definition of Metrics. When we compared our metrics to *SonarQube* [7] and *SourceMeter* [8], it jumped out their implementations of metrics often differ. Even for a metric as plain as the number of lines of code, we counted many

[1] `https://github.com/ArtOfIllusion/ArtOfIllusion`
[2] Intel(R) Core(TM) i7-1185G7 @ 3.00GHz, 16 Go of RAM

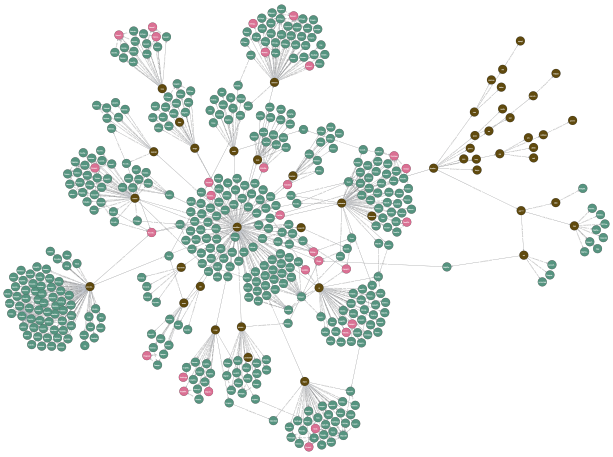


Figure 2: An extract of the graph produced by *Javanalyser* when parsing *Art of Illusion*

variations with none being equal as shown by an example from *Art of Illusion* in table 1. Moreover, reading documentations is nor practical nor always sufficient to grab all the subtleties of metrics computation. For instance the documentation of *SourceMeter* does not state the computation of cyclomatic complexity ignores nested classes, which is most surprising because the declaration on-the-fly of a nested class is clearly a source of complexity, and a backdoor for masking complexity if ignored.
Whether open-source or well documented, imperative implementations of metrics remains opaque, hard to discuss and hard to adapt according to the context. On the other hand, our declarative implementations of metrics within *Cypher* queries [23] embodies formal definitions and are all at once univocal, transparent, easily shareable and mutable.

Tool	Metric name	Value
SonarQube	Lines	480
	Lines of Code	368
	Comment Lines	28
SourceMeter	Lines of Code	314
	Total Lines of Code	453
	Logical Lines of Code[3]	233
	Total Logical Lines of Code[3]	357
	Comment Lines of Code	23
	Documentation Lines of Code	16
	Total Comment Lines of Code	34

Table 1: Example of lines of code metrics on the class `ActorEditorWindow`

How to count complexity. Most product metrics are basically specialized counters of some sort, for instance the cyclomatic complexity [29] counts the number of predicates[4], the depth of inheritance tree [16] counts the number of parent classes, or the message passing coupling [27] counts the number of "foreign" calls. Some metrics use different top-level aggregation operations, like the *average methods' number of nodes* that computes an average of counts. In all cases, these metrics base their computation on simply

[3] A logical line of code correspond to an executable statement.
[4] An operator or function that returns either true or false.

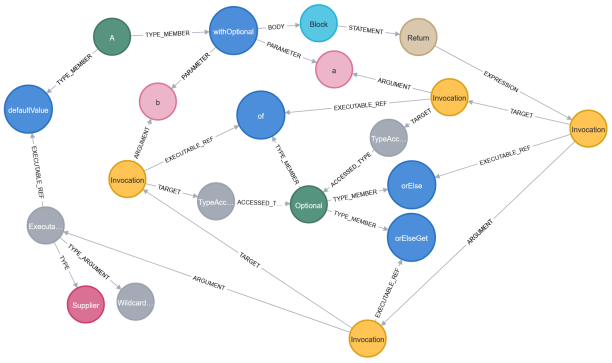


Figure 3: Screenshot of `withOptional` graph

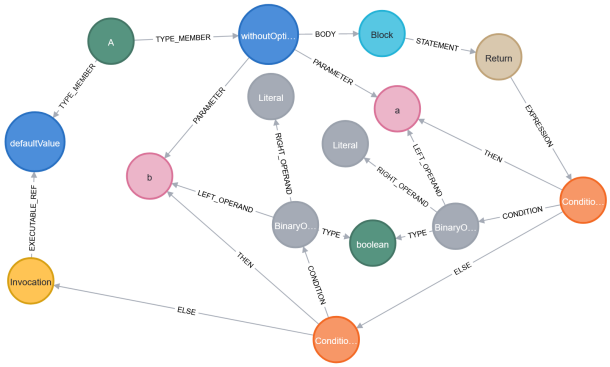


Figure 4: Screenshot of `withoutOptional` graph

counting targeted program elements. Interestingly, the lack of cohesion in methods [24] is more complex, as it is associated with the independent subsets of methods within a class.

However, deciding what to count for a given metrics is not obvious. Do we count indiscriminately each node or a subset of so-called statement nodes? Do we count external references? Do we count nested classes or inner classes? We can go even further by considering the `Optional` in *Java* versus the classic if-then-else structure. Listing 3 presents two methods, `withOptional` and `withoutOptional`. These methods have intrinsically the same behaviour but their computed *cyclomatic complexity* differ. We argue this difference is not legitimate, especially considering that the *cyclomatic complexity* was built to count the number of paths within a piece of code. Figures 3 and 4 show the use of `Optional` does not decrease the number of paths, which is coherent with the fact that these methods implement the same behaviour. The same reasoning can be made with *Java* `Stream` versus classic `for` loop.

Our approach allows to easily take into account new structures contributing to complexity by adding the corresponding selectors to the query (within `WHERE` clauses).

```
package fr.onepoint.javanalyser.tests.optional;

import java.util.Optional;

public class A {

  // Cyclomatic Complexity = 1
  String withOptional(String a, String b) {
    return Optional.of(a).orElse(Optional.of(b).
     orElseGet(this::defaultValue));
  }

  // Cyclomatic Complexity = 3
  String withoutOptional(String a, String b) {
    return a != null ? a : b != null ? b : defaultValue
     ();
  }

  String defaultValue() {
    return "Default Value";
  }
}
```

Listing 3: Code with or without optional

Program Representation. The *cyclomatic complexity* is built by considering the complexity of the associated program graph. Computationally, through graph theory, it can be reduced at counting the number of predicates. But its inception is based on a graph representation of the control flow of the program [29]. Lemberger and Morel use *cyclomatic complexity* to define a scale-invariant measure of complexity [26]. They also emphasize the idea that summation of metrics is not a natural condition to compute higher-level metrics, *i.e.* the complexity of a class is not the summation of the complexity of its methods.

This leads us to consider program representation, which must be first complete and accurate, possibly opinionated, but also freely interpretable to allow discussion and subsequent exploration by machine-learning algorithms. Our approach was to build a complete program graph based on a carefully simplified structure and leaving aside presentation. Müller *et al.* [30] have a similar approach, based on *Neo4j* [4] and *jQAssistant* [3]. However *jQAssistant* does not feed the graph with the content of the methods, making it impossible to compute low-level metrics.

Another approach involves ontology-based program analysis. Graph4Code [9] exploits RDF graphs where each program is represented in a separate graph, which are interconnected by fully-qualified function names (package.classname.function). Documentation resources, as well as StackOverflow and StackExchange posts are also represented in separate graphs and connected to program graphs by fully-qualified function names. Applications such as anti-pattern detection or a debugging tool are implemented via smart SPARQL queries on the RDF graph. Zhao *et al.* [34] propose a solution that uses the SWI-Prolog reasoning engine and custom APIs to exploit and possibly update a knowledge base. The authors also argue that among implementations of program analysis, those based on a declarative approach are easier to implement and maintain than those based on an imperative approach. Actually, our *Cypher* queries were a bit messy, for instance due to the systematic repetition of the set of inner relationships. The use of ontologies could simplify our implementation and be more powerful than mere *Neo4j* queries, as it will permit the introduction of more general concepts.

5 Conclusions

In order to focus on true detectable maintainability defects, we built *Javanalyser* to represent code as a graph, depending only on the structure of the code and allowing to easily extract data such as metrics. Thirty-four metrics were implemented as *Cypher* queries [23], embodying a formal definition of these metrics. Moreover, declarative queries allow taking into account the complexity of up-to-now ignored *Java* structures such as `Optional` or `Stream`. *Javanalyser* foreshadows a unified maintainability-analysis framework to conduct machine-learning studies to assess maintainability.

Future works include testing our graph and metrics against the software maintainability dataset [32]. Leveraging ontology-based program analysis [34] could also simplify our implementation and be more powerful than mere *Cypher* queries, as it will permit the introduction of more general concepts, thus allowing higher level analysis. Finally, studying the graph simplifications we designed would be required to ascertain the graph is faithful to the developer's cognitive representation of the program.

Acknowledgments

We thank our collaborators at *onepoint*[5] for their insightful advices, in particular Damien Bonvillain, Alexandra Delmas, Jérôme Fillioux, and Jérôme Lelong.

References

[1] Eclipse Java Development Tools. `https://www.eclipse.org/jdt/`.

[2] JavaParser. `https://javaparser.org/`.

[3] jQAssistant. `https://jqassistant.org/`.

[4] Neo4j. `https://neo4j.com/`.

[5] Prettier. `https://prettier.io/`.

[6] Pylint. `https://pylint.org/`.

[7] SonarQube. `https://www.sonarqube.org/`.

[8] SourceMeter. `https://www.sourcemeter.com/`.

[9] Ibrahim Abdelaziz, Julian Dolby, James P. McCusker, and Kavitha Srinivas. Graph4Code: A Machine Interpretable Knowledge Graph for Code. *arXiv:2002.09440 [cs]*, May 2020. `https://arxiv.org/abs/2002.09440`.

[10] Miltiadis Allamanis and Charles Sutton. Mining source code repositories at massive scale using language modeling. In *2013 10th Working Conference on Mining Software Repositories (MSR)*, pages 207–216, May 2013.

[5] `https://www.groupeonepoint.com/`

[11] Hadeel Alsolai and Marc Roper. A systematic literature review of machine learning techniques for software maintainability prediction. *Information and Software Technology*, 119:106214, March 2020.

[12] Luca Ardito, Riccardo Coppola, Luca Barbato, and Diego Verga. A Tool-Based Perspective on Software Code Maintainability Metrics: A Systematic Literature Review. *Scientific Programming*, 2020:1–26, August 2020.

[13] Sébastien Bertrand, Pierre-Alexandre Favier, and Jean-Marc André. Pragmatic Software Maintainability Management Using a Multi-agent System Working in Collaboration with the Development Team. In Sara Rodríguez González, Alfonso González-Briones, Arkadiusz Gola, George Katranas, Michela Ricca, Roussanka Loukanova, and Javier Prieto, editors, *Distributed Computing and Artificial Intelligence, Special Sessions, 17th International Conference*, Advances in Intelligent Systems and Computing, pages 201–204, Cham, 2020. Springer International Publishing.

[14] G. Ann Campbell. Cognitive complexity: An overview and evaluation. In *Proceedings of the 2018 International Conference on Technical Debt*, TechDebt '18, pages 57–58, New York, NY, USA, May 2018. Association for Computing Machinery. `https://doi.org/10.1145/3194164.3194186`.

[15] Celia Chen, Reem Alfayez, Kamonphop Srisopha, Barry Boehm, and Lin Shi. Why Is It Important to Measure Maintainability and What Are the Best Ways to Do It? In *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*, pages 377–378, May 2017.

[16] Shyam R. Chidamber and Chris F. Kemerer. Towards a Metrics Suite for Object Oriented Design. In *OOPSLA '91: Conference Proceedings on Object-oriented Programming, Systems, Languages, and Applications*, volume 26, pages 197–211, Phoenix, Arizona, USA, November 1991. Association for Computing Machinery.

[17] Don Coleman, Bruce Lowther, and Paul Oman. The application of software maintainability models in industrial software systems. *Journal of Systems and Software*, 29(1):3–16, April 1995.

[18] Ward Cunningham. The WyCash portfolio management system. *ACM SIGPLAN OOPS Messenger*, 4(2):29–30, December 1992.

[19] Sara Elmidaoui, Laila Cheikhi, and Ali Idri. Towards a Taxonomy of Software Maintainability Predictors. In Álvaro Rocha, Hojjat Adeli, Luís Paulo Reis, and

Sandra Costanzo, editors, *New Knowledge in Information Systems and Technologies*, volume 930 of *Advances in Intelligent Systems and Computing*, pages 823–832. Springer, 2019.

[20] Sara Elmidaoui, Laila Cheikhi, Ali Idri, and Alain Abran. Empirical Studies on Software Product Maintainability Prediction: A Systematic Mapping and Review. *e-Informatica Software Engineering Journal*, 13(1):141–202, 2019.

[21] Eric Evans. *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison-Wesley, Boston, 2004.

[22] Norman E Fenton and Shari Lawrence Pfleeger. *Software Metrics: A Rigorous and Practical Approach*. International Thomson, second edition, 1996.

[23] Nadime Francis, Alastair Green, Paolo Guagliardo, Leonid Libkin, Tobias Lindaaker, Victor Marsault, Stefan Plantikow, Mats Rydberg, Petra Selmer, and Andrés Taylor. Cypher: An Evolving Query Language for Property Graphs. In *Proceedings of the 2018 International Conference on Management of Data*, SIGMOD '18, pages 1433–1445, New York, NY, USA, May 2018. Association for Computing Machinery.

[24] Martin Hitz and Behzad Montazeri. Measuring Coupling and Cohesion In Object-Oriented Systems. In *Proceedings of the International Symposium on Applied Corporate Computing*, page 10, Mexico, Monterrey, 1995.

[25] ISO/IEC. Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models. Standard ISO/IEC 25010:2011, ISO/IEC, 2011. `https://www.iso.org/standard/35733.html`.

[26] Pirmin Lemberger and Médéric Morel. Two Measures of Code Complexity. pages 195–206. January 2013.

[27] Wei Li and Sallie Henry. Object-Oriented Metrics that Predict Maintainability. *Journal of Systems and Software*, 23(2):111–122, November 1993. `http://www.sciencedirect.com/science/article/pii/016412129390077B`.

[28] Rüdiger Lincke, Jonas Lundberg, and Welf Löwe. Comparing software metrics tools. In *Proceedings of the 2008 International Symposium on Software Testing and Analysis - ISSTA '08*, page 131, Seattle, WA, USA, 2008. ACM Press.

[29] Thomas J. McCabe. A Complexity Measure. *IEEE Transactions on Software Engineering*, SE-2(4):308–320, December 1976.

[30] Richard Müller, Dirk Mahler, Michael Hunger, Jens Nerche, and Markus Harrer. Towards an Open Source Stack to Create a Unified Data Source for Software Analysis and Visualization. In *2018 IEEE Working Conference on Software Visualization (VISSOFT)*, pages 107–111, Madrid, September 2018. IEEE.

[31] Renaud Pawlak, Martin Monperrus, Nicolas Petitprez, Carlos Noguera, and Lionel Seinturier. Spoon: A library for implementing analyses and transformations of Java source code. *Software: Practice and Experience*, 46(9):1155–1179, September 2016.

[32] Markus Schnappinger, Arnaud Fietzke, and Alexander Pretschner. Defining a Software Maintainability Dataset: Collecting, Aggregating and Analysing Expert Evaluations of Software Maintainability. In *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 278–289, Adelaide, Australia, September 2020. IEEE.

[33] Markus Schnappinger, Arnaud Fietzke, and Alexander Pretschner. Human-level Ordinal Maintainability Prediction Based on Static Code Metrics. In *EASE 2021: Evaluation and Assessment in Software Engineering*, pages 160–169, Trondheim, Norway, June 2021. ACM.

[34] Yue Zhao, Guoyang Chen, Chunhua Liao, and Xipeng Shen. Towards Ontology-Based Program Analysis. In Shriram Krishnamurthi and Benjamin S. Lerner, editors, *30th European Conference on Object-Oriented Programming*, volume 56 of *Leibniz International Proceedings in Informatics*, pages 26:1–26:25. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016.

Etude des méthodes de détection d'anomalies non supervisées appliquées aux flux de données

K. Ducharlet[1,2], L. Travé-Massuyès[2], M-V. Le Lann[2], Y. Miloudi[1]

[1] Carl Berger-Levrault
[2] LAAS-CNRS, Université de Toulouse, CNRS, INSA

{kevin.ducharlet, youssef.miloudi}@carl.eu
{louise, mvlelann}@laas.fr

Résumé

La détection d'anomalies est un sujet d'intérêt en fouille de données. Ces dernières années, le nombre d'applications reposant sur des flux continus de données se multiplie. Ces flux de données sont accompagnés de spécificités dont les algorithmes de détection d'anomalies doivent tenir compte. Pour cette raison, de nombreuses méthodes adaptées à la détection en ligne ont vu le jour. Cet article présente un tour d'horizon des méthodes de détection non supervisées appliquées aux flux de données et discute de l'insuffisance de métriques permettant d'évaluer et comparer ces méthodes.

Mots-clés

Détection d'anomalies, Flux de données, Etat de l'art, Fouille de données

Abstract

Outlier detection is a subject of interest in data mining. During the last decade, the amount of domains featuring data streams grew a lot. Those data streams come with peculiarities that outlier detection methods have to deal with. That is why various methods adapted to the online outlier detection problem have been developed. This article presents a survey on outlier detection unsupervised methods for data streams and discusses the lack of metrics allowing to evaluate and compare these methods.

Keywords

Outlier detection, Data streams, Survey, Data mining

1 Introduction

La détection d'anomalies est un sujet de recherche d'intérêt dans le cadre de la fouille de données qui touche de nombreux domaines d'applications. Les anomalies peuvent être une source d'information importante ou une nuisance à retirer. Dans tous les cas, les détecter est souvent crucial.

Depuis le début des années 2000, les travaux dans le contexte des flux de données se sont multipliés [1, 30]. En effet, les données sont générées sous forme de flux continus dans de nombreux domaines d'application (réseaux de capteurs, surveillance de l'activité web, étude de données météorologiques, surveillance du trafic réseau, ...). Ces données étant souvent sensibles à l'apparition d'anomalies, il est naturel de vouloir y appliquer des méthodes de détection d'anomalies. Cependant, les flux de données dénotent des spécificités nouvelles auxquelles les méthodes doivent s'adapter [37].

Récemment, un grand nombre de méthodes ont vu le jour pour détecter des anomalies dans les flux de données. L'objectif de cet article est de réaliser un tour d'horizon de l'état de l'art de ce domaine. Dans une première partie, nous décrivons brièvement le problème de la détection d'anomalies. Une seconde partie décrit les spécificités des flux de données. Les sections suivantes présentent les différents types de méthodes pouvant s'appliquer à la détection d'anomalies dans les flux de données. Il est important de noter que ces types ne constituent pas des ensembles disjoints ou incompatibles mais permettent de bien saisir les principes des différentes approches existantes. Enfin, nous discutons du manque d'approches comparatives pour ces méthodes avant de conclure cet article.

Travaux connexes. De nombreuses études ont été réalisées afin de recenser et classifier les méthodes de détection d'anomalies [13, 36]. Certaines d'entre elles mentionnent le problème de la détection en ligne en citant quelques méthodes [47, 41, 45] mais sans en faire un tour d'horizon suffisamment complet.

Il existe tout de même des études spécialisées dans la détection d'anomalies dans les flux de données. Certaines d'entre elles traitent d'une catégorie de méthodes ou un cas d'application spécifique. Tran, Han et Shahabi [44], par exemple, se concentrent sur les méthodes qui utilisent la distance entre les points pour déterminer les anomalies. A notre connaissance, les seules études se concentrant sur la détection d'anomalies dans les flux de données et qui en réalisent un tour d'horizon suffisamment complet sont celle de Thakkar, Vala et Prajapati [43] et celle de Salehi et Rashidi [39].

2 Détection d'anomalies

La détection d'anomalies est un sujet de recherche qui a intéressé différentes communautés depuis la fin du 19ème siècle, à commencer par les statisticiens comme en té-

moignent les travaux de Edgeworth [16]. Aussi, différentes définitions ont été fournies pour désigner le terme "anomalie" selon le domaine d'étude mais aussi le domaine d'application, si bien qu'il est impossible d'en donner une définition unique. Néanmoins, la définition qui revient le plus fréquemment dans la littérature est celle de Hawkins [19] : une anomalie est une observation qui s'écarte tant des autres observations qu'on puisse supposer qu'elle ait été générée par un mécanisme différent. On en distingue en général trois types [13] :

— les anomalies ponctuelles, qui correspondent à des points paraissant anormaux par rapport au reste du jeu de données ;
— les anomalies contextuelles, qui sont des points anormaux dans le contexte, temporel et/ou spatial, dans lequel ils apparaissent ;
— les anomalies collectives, qui sont des points normaux individuellement mais anormaux quand considérés comme un ensemble.

Avec plus d'un siècle de travaux dans le domaine, de nombreuses méthodes ont vu le jour. On les sépare principalement selon les informations qu'elles requièrent lors d'une phase d'apprentissage [13] :

— les méthodes supervisées nécessitent, lors de l'apprentissage, un label étiquetant chaque point comme normal ou anormal ;
— les méthodes semi-supervisées n'apprennent que sur des données labellisées comme normales et peuvent ensuite déterminer si un nouveau point est similaire au jeu d'entraînement (normal) ou s'il est différent (anormal) ;
— les méthodes non supervisées n'ont besoin d'aucun label, leur précision est néanmoins inférieure à celle des autres méthodes et il peut être nécessaire de faire des hypothèses fortes, en fixant par exemple le taux d'anomalies attendu.

Pour chaque observation évaluée, la sortie des méthodes de détection d'anomalies peut être de deux types : 1) dans le cas des méthodes supervisées, on obtient souvent une décision (normal ou anormal), 2) dans les autres cas, le degré d'anomalie d'une observation est évalué à partir d'un score ; on peut cependant se ramener à une décision binaire en appliquant un seuil sur ce score.

Nous traiterons ici du cas non supervisé. En effet, les informations nécessaires aux méthodes supervisées ou semi-supervisées sont rarement disponibles, en particulier dans le cas des flux de données, pour lesquels donner un label au fil de l'acquisition est difficile.

3 Spécificités et difficultés des flux de données

Définition Un flux de données est un jeu de données $\mathcal{D} := \{d_t, t \geq 0\}$ de taille infinie où chaque élément d_t correspond à un couple $d_t := (\tau_t, \mathbf{x}_t)$ d'une valeur p-variée $\mathbf{x}_t$ horodatée par une date unique τ_t. Ce flux est généré par une source avec une périodicité pouvant, selon le cadre d'application, ne pas être fixe ; pour $i \neq j$ et $i, j > 0$, on peut avoir $\tau_i - \tau_{i-1} \neq \tau_j - \tau_{j-1}$. Enfin, à chaque instant t, on ne dispose que d'un flux partiel $\mathcal{D}_t := \{d_i, t - \alpha \leq i \leq t - 1, \alpha \geq 1\}$ de points antérieurs pour évaluer d_t.

Par définition, les flux de données sont proches des séries temporelles. Nous considérons ici la subtilité que l'analyse des *séries temporelles* a pour objectif de prédire les observations à venir à partir d'un unique apprentissage sur les données passées, tandis que les *flux de données* introduisent l'idée d'un flux continu avec la nécessité d'un *apprentissage en ligne ou incrémental*.

Spécificités Sept spécificités des flux de données sont formulés dans l'état de l'art [37] :

— *Etat éphémère* : chaque point d_t a une durée de vie déterminée ; l'intérêt du point n'étant pas durable, il doit être traité par la méthode de détection d'anomalies dès qu'il est généré dans le flux de données.
— *Temporalité* : chaque point d_t étant associé à une date τ_t, la notion d'anomalie définie par le modèle doit tenir compte du contexte temporel des points ; dans les flux de données, on ne cherche pas d'anomalies ponctuelles car l'étude est toujours faite dans un contexte défini.
— *Infinité* : les données sont générées en continu, $\mathcal{D}$ est donc de taille infinie et les approches classiques consistant à stocker tous les points avant de générer le modèle et de rechercher les anomalies ne peuvent pas s'appliquer ; dans le cadre des flux de données, les méthodes doivent travailler sur une représentation sommaire du jeu partiel $\mathcal{D}_t$ et cette représentation sommaire doit pouvoir être incrémentée avec les nouveaux points entrants.
— *Vitesse de génération* : les points arrivant en continu et devant être traités dès qu'ils arrivent, il est nécessaire que le temps d'exécution de la classification d'un nouveau point et de l'incrémentation du modèle soit inférieur à la durée entre l'arrivée de deux points consécutifs. De plus, si la vitesse de génération est variable, alors la vitesse d'exécution doit pouvoir s'y adapter quitte à réduire la précision des résultats en travaillant avec une représentation plus réduite pour accélérer le calcul.
— *Non-stationnarité* : la distribution des données peut évoluer à travers le temps ; les méthodes faisant l'hypothèse d'une distribution fixe ne sont donc pas applicables.
— *Incertitude* : dans certains cas d'application, comme les réseaux de capteurs, les mesures générées ne sont pas fiables car elles peuvent être perturbées par des phénomènes environnementaux ; les mesures de similarité utilisées doivent tenir compte de cette incertitude. Celle-ci touche également la vitesse de génération des points qui peuvent être relevés avec du retard ou ne pas l'être du tout ; dans ce cas, il faut tout de même maintenir l'évaluation des points dans leur contexte temporel.
— *Multi-dimensionnalité* : les flux de données sont sujets aux problèmes usuels en grandes dimensions ;

ces problèmes rendent certaines méthodes d'estimation de densité peu efficaces et forcent l'utilisation de mesures de similarité adaptées.

Il existe également d'autres spécificités dans le cas du traitement de plusieurs flux de données [37], comme par exemple dans l'étude du trafic réseau entre plusieurs appareils. Il faut alors considérer les corrélations entre les flux, leur caractère asynchrone (rendant la contextualisation sur plusieurs flux difficile), l'aspect dynamique des relations (dû au comportement asynchrone et à la non-stationnarité des flux individuels) et enfin l'hétérogénéité des flux (les variables mesurées ne sont pas nécessairement les mêmes). Nous ajoutons également que, afin de respecter les spécificités mentionnées plus tôt, le calcul est désormais souvent embarqué, notamment dans le cadre des réseaux de capteurs. Cette spécificité impose des limites sur l'utilisation CPU et de l'espace mémoire des méthodes embarquées.

4 Adaptation par fenêtrage

Les premières approches pour la détection d'anomalies dans les flux de données ont cherché à adapter les méthodes déjà existantes pour des jeux de données statiques dans un cadre dynamique avec une composante temporelle. Pour ce faire, des fenêtres glissantes ont été utilisées afin de ne prendre en compte qu'une partie restreinte et évolutive du jeu de données, permettant ainsi d'adresser une grande partie des spécificités des flux de données.

Il existe plusieurs approches pour le fenêtrage [39] :

- le fenêtrage par point de repère, entre un point de repère fixé dans le jeu de données et la dernière observation générée ;
- le fenêtrage glissant, pour lequel on fixe la taille de la fenêtre (durée ou nombre d'échantillons) puis on la fait glisser à chaque nouvelle observation ;
- le fenêtrage amorti, où on associe à chaque point un poids selon son ancienneté, ainsi les points les plus récents auront un poids plus élevé que les plus anciens ;
- le fenêtrage adaptatif, similaire à un fenêtrage glissant mais où la taille de la fenêtre dépend de la vitesse à laquelle les données évoluent. La fenêtre sera grande si la distribution est stable et petite si la distribution évolue rapidement.

Néanmoins, à chaque fois que la fenêtre est modifiée, le modèle appris doit être mis à jour en conséquence. Les méthodes les plus adaptées sont donc celles qui ne nécessitent pas de réaliser un nouvel apprentissage complet sur la fenêtre actualisée.

5 Méthodes par erreur de prédiction

Nous avons mentionné en introduction de cette étude le lien entre les flux de données et les séries temporelles. Aussi, les méthodes généralement utilisées pour la détection d'anomalies dans des séries temporelles ont été utilisées pour traiter des flux de données, comme présenté dans la récente étude comparative de Duraj et Szczepaniak [15].

L'analyse des séries temporelles se concentre sur l'identification de tendances (changements de comportements linéaires au cours du temps) et de comportements cycliques ou saisonniers afin de définir une relation entre les observations passées et les observations futures. A partir de ces relations, il est donc possible de prédire les prochaines valeurs. Les méthodes de détection d'anomalies s'appuient sur l'erreur de prédiction ; plus l'erreur de prédiction est grande, plus l'observation est éloignée du modèle et peut être considérée comme anormale.

Exemples de méthodes. Parmi les méthodes reposant sur l'erreur de prédiction, nous citerons :

- les modèles ARIMA (Auto-Regressive Integrated Moving Average), dont la méthodologie est détaillée dans le livre de Asteriou et Hall [7] ;
- les modèles de prédictions utilisant le lissage exponentiel ou EST (Exponential Smoothing State Space Model) [22] ;
- les LSTM (Long Short-Term Memory) [29], réseaux de neurones inspirés des réseaux récursifs prenant en entrée les valeurs passées pour prédire les valeurs futures.

Avantages et inconvénients. La limite des modèles de prédiction vient de la caractéristique non-stationnaire des données. En général, la relation est déterminée à partir de données d'entraînement puis appliquée sans ajustement possible. Mettre à jour le modèle est souvent coûteux et difficile à mettre en place en suivant la contrainte de vitesse de génération des flux de données. Notons aussi que les LSTM, comme la majorité des méthodes utilisant des réseaux de neurones, sont peu adaptés à l'apprentissage en ligne à cause de la complexité même du modèle.

6 Approches de partitionnement dynamique

Les approches de partitionnement, ou clustering, sont des techniques, généralement non supervisées, qui ont pour objectif de regrouper les points dans l'espace selon leur similarité [13] et, à ce titre, elles sont liées au méthodes basées distance décrites en Section 8. Ces méthodes ne sont initialement pas pensées pour la détection d'anomalies mais plusieurs d'entre elles ont historiquement été utilisées à cette fin en faisant l'une des trois hypothèses suivantes :

- les points normaux appartiennent à des groupes, ou clusters, contrairement aux anomalies ; pour pouvoir détecter des anomalies, les méthodes de partitionnement ne doivent donc pas forcer tous les points à appartenir à un groupe (exemple : DBSCAN [17]),
- les points normaux sont proches du plus proche centroïde de cluster tandis que les anomalies en sont éloignées ; il faut calculer l'emplacement des centroïdes, barycentres des points de chaque groupe, et la distance des points aux centroïdes. Néanmoins, si un ensemble d'anomalies forment un groupe isolé, alors ces anomalies seront considérées comme normales (exemple : Smith et al. [40]),
- les points normaux appartiennent à des clusters

denses et de grande taille tandis que les anomalies appartiennent à de petits clusters épars ; contrairement à la seconde hypothèse, les anomalies doivent former des clusters isolés et le nombre de groupes à former doit donc être important (exemple : FindCBLOF [20]).

Dans un contexte statique, les méthodes de partitionnement réalisent un seul apprentissage sur les données puis placent les nouveaux points dans les clusters appris en fonction de leur distance aux centroïdes. Cette approche n'est cependant pas viable dans le cas des flux de données à cause de la non-stationnarité de la distribution. Des approches de partitionnement dynamique ont ainsi été développées pour permettre aux clusters d'évoluer dans le temps.

Exemples de méthodes. Il existe de nombreuses méthodes de partitionnement dynamique, présentées en détails dans différentes études [39, 47]. Nous n'en citerons ici que quelques unes :

- BIRCH [48] et CluStream [2] utilisent des caractéristiques des clusters (CFs) contenant, pour chaque cluster, le nombre de points contenus ainsi que la somme et la somme des carrés des valeurs ;
- DenStream [11], DStream [14] et SDstream [35] sont des améliorations de CluStream modifiant respectivement le paramétrage du nombre de clusters, la séparation des clusters en grille et le regroupement en micro-clusters ;
- DyClee [9] travaille avec des micro-clusters, regroupés en clusters selon leur densité et la distance entre eux, et permet de rejeter des micro-clusters anormaux.

Avantages et inconvénients. Les méthodes de partitionnement dynamique citées résument les caractéristiques des clusters avec un nombre fini de métriques. Cette approche permet de limiter le temps nécessaire pour chercher dans quel groupe se positionnent les nouveaux points et, dans la plupart des cas, facilite l'incrémentation du modèle. Pour les CFs par exemple, ajouter un nouveau point à un cluster nécessite simplement d'incrémenter de un le nombre de points et d'ajouter la valeur du point à la somme des valeurs et la valeur du carré à la somme des carrés. Il n'est donc pas nécessaire de stocker l'entièreté du jeu de données.

Cependant, les méthodes de partitionnement dynamique sont souvent critiquées dans le cas de la détection d'anomalies car leur premier objectif est de regrouper les points et non de détecter des anomalies [43]. Aussi, les méthodes les plus adaptées sont celles qui : 1) ne nécessitent pas de fixer le nombre de clusters comme paramètre et 2) sont capables de créer de nouveaux clusters pouvant être considérés anormaux.

7 Méthodes statistiques

L'approche statistique fait l'hypothèse que les données ont été générées par une distribution statistique. L'objectif de ces méthodes est alors d'estimer empiriquement la distribution statistique en question. Les points normaux apparaissent dans des zones de l'espace où la densité de probabilité est élevée tandis que les anomalies apparaissent dans des zones de faible densité de probabilité.

Les méthodes statistiques sont généralement séparées en deux catégories : les méthodes paramétriques et les méthodes non-paramétriques [47].

7.1 Méthodes paramétriques

Les méthodes paramétriques font l'hypothèse que les données suivent une distribution prédéfinie. Les données à disposition sont ensuite utilisées pour déterminer, de manière empirique, les paramètres de ce modèle en minimisant ou maximisant une métrique choisie.

Un exemple typique est celui des modèles Gaussiens, où l'objectif est de déterminer la moyenne et l'écart-type qui maximisent la vraissemblance. Les modèles à base de mélanges gaussiens (GMM) [8] sont aussi populaires pour la détection d'anomalies et ont notamment été utilisés sur des séries temporelles, couplées à un modèle de régression linéaire [3].

Cependant, puisque les méthodes paramétriques font l'hypothèse que les données suivent une distribution fixée, elles ne sont pas applicables dans le cadre des flux de données [43].

7.2 Méthodes non-paramétriques

Il existe principalement deux catégories de méthodes statistiques non-paramétriques. A l'opposé des méthodes paramétriques, il n'est pas nécessaire de faire d'hypothèses a priori concernant la distribution.

7.2.1 Construction d'histogrammes

En statistiques, les histogrammes sont généralement utilisés pour obtenir une représentation visuelle d'une distribution empirique à une dimension. L'espace est divisé en cellules pour lesquelles des colonnes sont construites. La hauteur des colonnes correspond au nombre d'échantillons dont la valeur tombe dans la cellule. Ainsi, une cellule associée à une forte probabilité aura une colonne plus haute qu'une cellule de faible probabilité. La forme de l'histogramme tend vers celle de la fonction de densité de la distribution quand le nombre d'échantillons grandit. On peut donc naturellement utiliser la hauteur de la cellule pour identifier les anomalies.

Exemples de méthodes. Il existe trois types d'approches pour ces méthodes [47] :

- construction à partir des données normales (semi-supervisée) : avec cette approche, les anomalies sont les points qui tombent dans des cellules vides ;
- construction à partir des anomalies (semi-supervisée) : les anomalies tombent cette fois dans des cellules non-vides, cette approche est principalement utilisée dans des cas d'applications où il n'existe qu'un nombre fini de profils anormaux connus ;
- construction sans labels (non supervisée) : on définit les cellules anormales comme celles dont le nombre d'éléments est inférieur à un seuil, dépendant du nombre d'éléments dans les autres cellules de l'histogramme et de la taille de la cellule ; les

échantillons dans ces cellules anormales sont considérés comme anormaux.

Pour le cas multivarié, il est commun de construire un histogramme par variable puis de calculer un score sous forme d'agrégation de la probabilité estimée sur chaque variable. On peut notamment citer HBOS [18] qui calcule son score comme

$$HBOS(x) = \sum_{i=1}^{p} log(\frac{1}{hist_i(x)})$$

où p est le nombre de variables, x l'échantillon à évaluer et $hist_i(x)$ est la hauteur de la cellule dans laquelle tombe x pour la i-ième variable.

Avantages et inconvénients. Parmi les avantages de ces méthodes, nous pouvons noter qu'elles sont faciles à implémenter mais également faciles à incrémenter en recalculant la hauteur des cellules avec de nouveaux points. Cependant, elles deviennent rapidement limitées quand le nombre de dimensions augmente [47].

7.2.2 Méthodes à noyaux

L'approche non-paramétrique la plus connue est celle de l'estimation de densité par noyau (KDE) ou méthode de Parzen-Rosenblatt [32]. Celle-ci est proche de la construction d'histogrammes mais avec une notion de continuité et elle permet d'obtenir une approximation empirique de la fonction de densité de probabilité associée à la distribution. Formellement, soit $x_1, x_2, ..., x_n$ n échantillons i.i.d. (indépendants et identiquement distribués) d'une variable aléatoire X, l'estimateur de la fonction de densité f est

$$\tilde{f}_h(x) = \frac{1}{Nh} \sum_{i=1}^{n} K(\frac{x - x_i}{h})$$

où K est la fonction noyau (on choisit souvent le noyau Gaussien ou le noyau de Epanechnikov) et h est un paramètre jouant sur la zone d'influence de chaque échantillon, ou autrement dit sur le lissage de la courbe. Choisir un h trop faible génère une courbe où chaque échantillon est représenté par un pic de densité tandis que choisir un h trop grand engendre une courbe trop lisse.

On peut citer [26] parmi les méthodes qui ont cherché à adapter cette approche aux spécificités des flux de données.

Avantages et inconvénients. Les méthodes à noyaux ont l'avantage de donner une meilleure approximation de la densité que les méthodes à base d'histogrammes pour un nombre limité d'observations. La notion de continuité corrige aussi une partie des problèmes liés à l'augmentation du nombre de dimensions. Néanmoins, le paramètre h est connu comme étant sensible à paramétrer pour obtenir de bons résultats, et la complexité de la méthode augmente toujours rapidement avec le nombre de variables.

8 Méthodes s'appuyant sur la distance

Ces méthodes utilisent la distance entre les échantillons dans l'espace pour calculer leur score d'anomalie. Plus un point est isolé et plus il est anormal. La grande majorité des méthodes appartenant à cette catégorie généralisent la notion des plus proches voisins (kNN) aux flux de données en travaillant dans des fenêtres glissantes [44]. Il convient également de citer la méthode des HalfSpaceTrees (HST) [42] qui, d'une certaine manière, adapte la méthode des forêts d'isolation [28] aux flux de données.

8.1 Méthodes basées sur les kNN

Présentation des kNN. Les méthodes des plus proches voisins reposent sur la définition d'anomalie donnée par Knorr et Ng [24]. Selon cette définition, un point est anormal selon un critère de distance (DB-anormal) si la proportion de points du jeu de données se trouvant à une distance supérieure à D est au moins r. On parle alors d'un point $DB(r, D)$-anormal. Cette notion est ensuite simplifiée pour considérer comme anormaux tous les points ayant moins de k voisins à une distance inférieure ou égale à d.

De nombreuses approches découlent de cette définition en étudiant certaines statistiques des k plus proches voisins (kNN) comme la somme des distances [5] ou certaines statistiques du k-ième plus proche voisin (k^{th}NN) comme sa distance seule [34].

Exemples d'adaptations en ligne. Pour adapter l'approche des plus proches voisins aux flux de données, dont la taille n'est pas limitée, les méthodes décrites ici utilisent des fenêtres glissantes. Cette approche facilite la recherche des voisins proches.

L'étude comparative de Tran, Fan et Shahabi [44] décrit et compare cinq de ces méthodes, à savoir : exact-Storm et approx-Storm [6], Abstract-C [46], DUE et MCOD [25], ainsi que Thresh_LEAP [12]. Ces méthodes proposent différentes approches pour indexer les données. Ces stuctures indexées facilitent les trois étapes cruciales des kNN en ligne : retrouver les voisins proches, retirer des points de la structure lorsqu'ils sortent de la fenêtre glissante et en ajouter de nouveaux.

La conclusion de cette étude comparative des méthodes de détection en ligne reposant sur la distance entre les points est que MCOD offre de meilleures performances en général. Il est intéressant de noter que les méthodes ne sont pas comparées selon leur précision mais selon leur temps d'exécution et la mémoire utilisée, des critères importants dans le cas des flux de données.

Avantages et inconvénients. Ces méthodes sont pensées pour répondre aux différentes spécificités des flux de données (état éphémère, infinité, vitesse de génération, non-stationnarité). Cependant, les méthodes se basant sur la distance sont sujettes au fléau de la dimension. De plus, les performances des méthodes reposant sur des fenêtres dépendent grandement du choix de la taille de la fenêtre.

8.2 HST

Méthode des forêts d'isolation. L'algorithme des forêts d'isolation [28] adapte les forêts aléatoires à la détection d'anomalies. On construit un arbre d'isolation en choisissant aléatoirement, à chaque embranchement, une variable et une valeur selon laquelle réaliser une séparation. Chaque noeud contient donc un certain nombre d'obser-

vations. La séparation s'arrête lorsque chaque feuille de l'arbre ne contient qu'un unique point. Intuitivement, plus un point a été rapidement isolé (faible hauteur dans l'arbre), plus il est anormal. On construit ainsi un ensemble d'arbres (forêt) aléatoirement et on calcule pour chaque point la hauteur moyenne à laquelle il est isolé. En pratique, il n'est pas nécessaire de construire l'arbre complet.

Adaptation en ligne Les HST [42] sont une forme d'adaptation des forêts d'isolation au problème de la détection en ligne. La différence dans la construction des arbres vient du fait que seule la dimension à séparer est choisie aléatoirement ; la valeur est quant-à-elle prise au milieu de l'intervalle contenant les points de la dimension retenue. On utilise ensuite des fenêtres consécutives et on évalue les points d'une fenêtre par rapport à l'arbre construit dans la fenêtre précédente.

Avantages et inconvénients. Le modèle de mise à jour des HST est particulièrement rapide et s'adapte donc bien aux spécificités de la détection en ligne. Cependant, les résultats dépendent grandement du choix de la taille des fenêtres consécutives. Des fenêtres trop petites ne permettent pas d'avoir une bonne représentation de la répartition des points dans l'espace tandis que des fenêtres trop grandes induisent un temps de retard dans l'adaptation du modèle en cas de changement de distribution.

9 Méthodes s'appuyant sur la densité

Cette section recense les méthodes généralisant le LocalOutlierFactor (LOF) [10] à l'apprentissage en ligne, avec notamment le LOF incrémental (iLOF) [33]. Elles sont étroitement liées aux méthodes statistiques non-paramétriques en ce sens que le LOF tend vers la densité de probabilité quand le nombre d'échantillons augmente.

Présentation du LOF. Le LOF est une mesure d'anomalie reposant sur la densité locale qui utilise les kNN. Cette mesure est construite à partir de la moyenne du rapport de la concentration de points autour des plus proches voisins d'un point par rapport à la concentration de points autour de ce point. Si les plus proches voisins d'un point x sont dans une zone de l'espace très dense par rapport à la densité de la zone de l'espace dans laquelle se trouve x, alors le rapport moyen de concentration sera élevé ; on obtiendra donc un LOF, mesure d'anomalie, élevé.

iLOF : une adaptation incrémentale. Il est possible de prouver qu'ajouter ou supprimer un point d'un jeu de données n'influence qu'une petite partie de ses plus proches voisins dans le calcul du LOF [33]. iLOF se base sur cette propriété pour rendre le LOF incrémental et adapté à la détection en ligne. La précision de la méthode est similaire à celle obtenue en entraînant un nouveau modèle à chaque fois qu'un point est ajouté, mais en limitant considérablement le temps de calcul.

Variantes. Néanmoins, plusieurs méthodes ont été publiées pour améliorer les performances de iLOF : I-IncLOF [23], MiLOF [38], DILOF [31], TADILOF [21] et GP-LOF [4].

Avantages et inconvénients. A l'image des méthodes s'appuyant sur la distance, ces méthodes respectent une grande partie des spécificités des flux de données. Cependant, comme le prouve le nombre de travaux cherchant à améliorer iLOF, il est difficile de réduire le temps de calcul de ces méthodes.

10 Discussion sur l'évaluation des méthodes en ligne

En parcourant l'état de l'art de la détection d'anomalies en ligne, nous avons noté qu'il n'existait que très peu d'études comparatives de ces méthodes.

L'étude comparative de Duraj et Szczepaniak [15] ne compare qu'un nombre très limité de méthodes tout en mentionnant à quel point la complexité des flux de données et la diversité des cas d'application rendaient toute comparaison difficile. De même, celle de Tran, Fan et Shahabi [44] ne compare que les méthodes de distance et uniquement selon des critères de performance algorithmique.

La première difficulté est en réalité de pouvoir évaluer les méthodes de détection en ligne sur des critères mettant en avant les spécificités des flux de données. A notre connaissance, la contribution la plus complète sur l'évaluation des méthodes en ligne est le Numenta Anomaly Benchmark (NAB) [27]. Le NAB propose une métrique pour évaluer la capacité des méthodes à détecter les anomalies dans une série temporelle en récompensant la détection en amont de l'anomalie labellisée et en pénalisant les faux positifs et faux négatifs. Cette approche nécessite néanmoins des jeux de données labellisés représentatifs du domaine d'application dans lequel les méthodes seront appliquées.

11 Conclusion

Au-delà de la grande diversité de méthodes cherchant à répondre à la problématique de la détection d'anomalies dans les flux de données, cet état de l'art a principalement permis d'identifier deux points :

- proposer une méthode répondant aux spécificités des flux de données n'est pas une tâche aisée ; les contraintes traitées sont l'*état éphémère* des points et la *non-stationnarité*, qui sont les plus cruciales, mais aussi l'*infinité* et la *vitesse de génération* ;
- le domaine manque d'une méthode d'évaluation qui permettrait de vérifier à quel point chaque méthode répond à *chacune des spécificités* et de les comparer entre elles.

Références

[1] D. Abadi, D. Carney, U. Çetintemel, M. Cherniack, C. Convey, C. Erwin, E. Galvez, M. Hatoun, A. Maskey, A. Rasin, A. Singer, M. Stonebraker, N. Tatbul, Y. Xing, R. Yan, and S. Zdonik. Aurora : A data stream management system. In *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, SIGMOD '03, page 666, New

York, NY, USA, 2003. Association for Computing Machinery.

[2] Charu C. Aggarwal, Philip S. Yu, Jiawei Han, and Jianyong Wang. - a framework for clustering evolving data streams. In Johann-Christoph Freytag, Peter Lockemann, Serge Abiteboul, Michael Carey, Patricia Selinger, and Andreas Heuer, editors, *Proceedings 2003 VLDB Conference*, pages 81–92. Morgan Kaufmann, San Francisco, 2003.

[3] Hermine N. Akouemo and Richard J. Povinelli. Probabilistic anomaly detection in natural gas time series data. *International Journal of Forecasting*, 32(3) :948–956, 2016.

[4] Raed Alsini, Omar Alghushairy, Xiaogang Ma, and Terrance Soule. A grid partition-based local outlier factor for data stream processing. In Hamid R. Arabnia, Ken Ferens, David de la Fuente, Elena B. Kozerenko, José Angel Olivas Varela, and Fernando G. Tinetti, editors, *Advances in Artificial Intelligence and Applied Cognitive Computing*, pages 1047–1060, Cham, 2021. Springer International Publishing.

[5] F. Angiulli and C. Pizzuti. Outlier mining in large high-dimensional data sets. *IEEE Transactions on Knowledge and Data Engineering*, 17(2) :203–215, Feb 2005.

[6] Fabrizio Angiulli and Fabio Fassetti. Detecting distance-based outliers in streams of data. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, page 811–820, New York, NY, USA, 2007. Association for Computing Machinery.

[7] Dimitros Asteriou and Stephen G Hall. Arima models and the box–jenkins methodology. *Applied Econometrics*, 2(2) :265–286, 2011.

[8] David Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[9] Nathalie Barbosa Roa, Louise Travé-Massuyès, and Victor Hugo Grisales. DyClee : Dynamic clustering for tracking evolving environments. *Pattern Recognition*, 94 :162–186, October 2019.

[10] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. Lof : Identifying density-based local outliers. *SIGMOD Rec.*, 29(2) :93–104, may 2000.

[11] Feng Cao, Martin Estert, Weining Qian, and Aoying Zhou. Density-based clustering over an evolving data stream with noise. In *Proceedings of the 2006 SIAM international conference on data mining*, pages 328–339. SIAM, 2006.

[12] Lei Cao, Di Yang, Qingyang Wang, Yanwei Yu, Jiayuan Wang, and Elke A. Rundensteiner. Scalable distance-based outlier detection over high-volume data streams. In *2014 IEEE 30th International Conference on Data Engineering*, pages 76–87, March 2014.

[13] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection : A survey. *ACM Comput. Surv.*, 41(3), jul 2009.

[14] Yixin Chen and Li Tu. Density-based clustering for real-time stream data. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, page 133–142, New York, NY, USA, 2007. Association for Computing Machinery.

[15] Agnieszka Duraj and Piotr S. Szczepaniak. Outlier Detection in Data Streams — A Comparative Study of Selected Methods. *Procedia Computer Science*, 192 :2769–2778, 2021.

[16] F.Y. Edgeworth. Xli. on discordant observations. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 23(143) :364–375, 1887.

[17] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press, 1996.

[18] Markus Goldstein and Andreas Dengel. Histogram-based outlier score (hbos) : A fast unsupervised anomaly detection algorithm. *KI-2012 : poster and demo track*, 9, 2012.

[19] Douglas M Hawkins. *Identification of outliers*, volume 11. Springer, 1980.

[20] Zengyou He, Xiaofei Xu, and Shengchun Deng. Discovering cluster-based local outliers. *Pattern Recogn. Lett.*, 24(9–10) :1641–1650, jun 2003.

[21] Jen-Wei Huang, Meng-Xun Zhong, and Bijay Prasad Jaysawal. Tadilof : Time aware density-based incremental local outlier detection in data streams. *Sensors*, 20(20), 2020.

[22] Rob J Hyndman, Anne B Koehler, Ralph D Snyder, and Simone Grose. A state space framework for automatic forecasting using exponential smoothing methods. *International Journal of Forecasting*, 18(3) :439–454, 2002.

[23] Seyed Hesamodin Karimian, Manouchehr Kelarestaghi, and Sattar Hashemi. I-inclof : Improved incremental local outlier detection for data streams. In *The 16th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP 2012)*, pages 023–028, May 2012.

[24] Edwin M. Knorr and Raymond T. Ng. Algorithms for mining distance-based outliers in large datasets. In *Proceedings of the 24rd International Conference on Very Large Data Bases*, VLDB '98, page 392–403, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[25] Maria Kontaki, Anastasios Gounaris, Apostolos N. Papadopoulos, Kostas Tsichlas, and Yannis Manolopoulos. Continuous monitoring of distance-based out-

liers over data streams. In *2011 IEEE 27th International Conference on Data Engineering*, pages 135–146, April 2011.

[26] Matej Kristan, Aleš Leonardis, and Danijel Skočaj. Multivariate online kernel density estimation with gaussian kernels. *Pattern Recognition*, 44(10) :2630–2642, 2011. Semi-Supervised Learning for Visual Content Analysis and Understanding.

[27] Alexander Lavin and Subutai Ahmad. Evaluating real-time anomaly detection algorithms – the numenta anomaly benchmark. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pages 38–44, Dec 2015.

[28] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422, Dec 2008.

[29] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, Puneet Agarwal, et al. Long short term memory networks for anomaly detection in time series. In *Proceedings*, volume 89, pages 89–94, 2015.

[30] Rajeev Motwani, Jennifer Widom, Arvind Arasu, Brian Babcock, Shivnath Babu, Mayur Datar, Gurmeet Singh Manku, Chris Olston, Justin Rosenstein, and Rohit Varma. Query processing, approximation, and resource management in a data stream management system. In *First Biennial Conference on Innovative Data Systems Research, CIDR 2003, Asilomar, CA, USA, January 5-8, 2003, Online Proceedings*. www.cidrdb.org, 2003.

[31] Gyoung S. Na, Donghyun Kim, and Hwanjo Yu. Dilof : Effective and memory efficient local outlier detection in data streams. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '18, page 1993–2002, New York, NY, USA, 2018. Association for Computing Machinery.

[32] Emanuel Parzen. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3) :1065 – 1076, 1962.

[33] Dragoljub Pokrajac, Aleksandar Lazarevic, and Longin Jan Latecki. Incremental local outlier detection for data streams. In *2007 IEEE Symposium on Computational Intelligence and Data Mining*, pages 504–515, March 2007.

[34] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. Efficient algorithms for mining outliers from large data sets. *SIGMOD Rec.*, 29(2) :427–438, may 2000.

[35] Jiadong Ren and Ruiqing Ma. Density-based data streams clustering over sliding windows. In *2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, volume 5, pages 248–252, Aug 2009.

[36] Lukas Ruff, Jacob R. Kauffmann, Robert A. Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G. Dietterich, and Klaus-Robert Müller. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5) :756–795, May 2021.

[37] Shiblee Sadik and Le Gruenwald. Research issues in outlier detection for data streams. *SIGKDD Explor. Newsl.*, 15(1) :33–40, mar 2014.

[38] Mahsa Salehi, Christopher Leckie, James C. Bezdek, Tharshan Vaithianathan, and Xuyun Zhang. Fast memory efficient local outlier detection in data streams. *IEEE Transactions on Knowledge and Data Engineering*, 28(12) :3246–3260, Dec 2016.

[39] Mahsa Salehi and Lida Rashidi. A survey on anomaly detection in evolving data : [with application to forest fire risk prediction]. *SIGKDD Explor. Newsl.*, 20(1) :13–23, may 2018.

[40] R. Smith, A. Bivens, M. Embrechts, C. Palagiri, and Boleslaw Szymanski. Clustering approaches for anomaly based intrusion detection. *Proceedings of Intelligent Engineering Systems Through Artificial Neural Networks*, pages 579–584, 01 2002.

[41] SS Sreevidya et al. A survey on outlier detection methods. *IJCSIT) International Journal of Computer Science and Information Technologies*, 5(6), 2014.

[42] Swee Chuan Tan, Kai Ming Ting, and Tony Fei Liu. Fast anomaly detection for streaming data. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, page 1511–1516. AAAI Press, 2011.

[43] Pooja Thakkar, Jay Vala, and Vishal Prajapati. Survey on outlier detection in data stream. *International Journal of Computer Applications*, 136 :13–16, 02 2016.

[44] Luan Tran, Liyue Fan, and Cyrus Shahabi. Distance-based outlier detection in data streams. *Proc. VLDB Endow.*, 9(12) :1089–1100, aug 2016.

[45] Hongzhi Wang, Mohamed Jaward Bah, and Mohamed Hammad. Progress in outlier detection techniques : A survey. *IEEE Access*, 7 :107964–108000, 2019.

[46] Di Yang, Elke A. Rundensteiner, and Matthew O. Ward. Neighbor-based pattern detection for windows over streaming data. In *Proceedings of the 12th International Conference on Extending Database Technology : Advances in Database Technology*, EDBT '09, page 529–540, New York, NY, USA, 2009. Association for Computing Machinery.

[47] Ji Zhang. Advancements of outlier detection : A survey. *EAI Endorsed Transactions on Scalable Information Systems*, 1(1), 2 2013.

[48] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch : An efficient data clustering method for very large databases. In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, SIGMOD '96, page 103–114, New York, NY, USA, 1996. Association for Computing Machinery.

Session "IA & systèmes complexes"

Système de régulation dynamique de vitesse basé sur un contrôleur PID dans un environnement de trafic connecté

E. Fauchet[1], P.A. Laharotte[1], K. Bhattacharyya[1], Nour-Eddin El Faouzi[1]
[1] Univ Lyon, Univ Gustave Eiffel, ENTPE, LICIT-Eco7, F-69675 Lyon, France

Résumé

L'émergence des Véhicules Connectés (VCs) est associée à de nouvelles stratégies de régulation de trafic. Ce papier se focalise sur l'application d'un système de régulation dynamique de vitesse (VSL) dans un environnement connecté. Il est activé par des antennes de télécommunication en bord de route, et est mis en oeuvre grâce à un bouchon mobile -créé par deux VCs : un contrôleur et un agent asservi par un contrôleur PID- et non plus grâce aux VCs répartis sur le réseau. Cette méthode originale assure un contrôle effectif même à bas taux de pénétration des VCs.

Mots-clés

Systèmes de Transports Intelligents, Vehicules Connectés, Système de Régulation Dynamique de Vitesse, Bouchon mobile, Contrôleur PID

Abstract

The emergence of Connected Vehicles (CVs) is associated with new traffic control strategies. This paper focuses on the application of a Variable Speed Limit system (VSL) in a connected environment. It is activated by roadside telecommunication antennas, and is implemented through a moving bottleneck -created by two CVs : a controller and an agent controlled by a PID controller- rather than through the CVs distributed on the network. This original method ensures effective control even at low CV penetration rates.

Keywords

Intelligent Transport System, Connected Vehicles, Variable Speed Limit, Moving bottleneck, PID controller

1 Introduction

1.1 Contexte

La régulation du trafic vise à optimiser l'écoulement d'un flux de véhicules le long d'un ensemble d'infrastructures routières organisées en réseau. L'objectif consiste généralement à faire coïncider l'offre (la capacité de l'infrastructure) avec la demande locale (la quantité de véhicules en circulation) en limitant les congestions. Diverses stratégies ont été développées pour réguler les flux de trafic depuis le contrôle des flux par des feux de signalisation jusqu'à la régulation dynamique du nombre de voies. Un des objectifs de la régulation du trafic s'intéresse à limiter ou éliminer la propagation d'une onde de congestion, qu'elle soit due à des incidents (accidents, mauvais temps, etc.) ou à des contraintes (goulot d'étranglement, voie d'insertion, etc.) sur le réseau. L'enjeu est de limiter les risques d'accident occasionnés à l'approche de ces mouvements d'accordéons (ondes de congestion) et d'optimiser les temps de parcours tout en réduisant la consommation de carburant. L'objectif suit 3 champs d'action : la sécurité, l'efficacité du trafic et l'environnement. Le principe de fonctionnement usuel consiste à détecter un événement sur le réseau, le caractériser et, si besoin, anticiper sa propagation spatiale et temporelle en vue d'appliquer une régulation adéquate en amont. Pour répondre à ce besoin, la stratégie usuelle et populaire consiste à détecter dynamiquement une onde de choc sur une voie rapide et à appliquer une limitation de vitesse aux véhicules en amont de la zone congestionnée. Dans la littérature, cette approche est appelée Régulation Dynamique des Vitesses, ou Variable Speed Limit (VSL) en anglais [10]. Avec le déploiement des Systèmes de Transports Intelligents et Connectés (C-ITS) et, par conséquent, des Véhicules Connectés (VAC), de nouveaux procédés de mise en œuvre sont disponibles [18], notamment en tirant parti des VACs jouant le rôle de capteur-sonde du trafic et de moyen d'application de la stratégie. De telles approches utilisent les conditions courantes de trafic comme la vitesse ou le flux pour déterminer à quelle vitesse les conducteurs devraient rouler pour éviter la zone de ralentissement. On réduit, ainsi, le flux de véhicules arrivant dans cette zone pour prévenir la propagation de l'embouteillage, voire même le résoudre.

1.2 Etat-de-l'art

D'après Khondaker and Kattan [10], les premiers systèmes VSL sont basés sur une approche réactive, c'est-à-dire que leur déclenchement repose sur des valeurs seuils d'indicateurs de trafic présélectionnées et basées sur des connaissances expertes. Ce type de stratégie a démontré son efficacité pour homogénéiser les vitesses et stabiliser le trafic en réduisant les différences de vitesses entre les conducteurs. Elles permettent ainsi d'améliorer la sécurité sur le réseau. Leur limite principale réside dans leur délai de déclenchement. En effet, le temps que le système soit activé, les conditions de trafic sont souvent déjà trop dégradées pour qu'il puisse agir efficacement. Ainsi, les systèmes VSL basés sur une approche proactive et développés ensuite se fo-

calisent sur l'estimation de la propagation des états de trafic sur le réseau. [7] Ils utilisent des modèles (ex. : Model Predictive Control [9]) pour prédire les conditions de trafic futures et anticiper la propagation des ondes de choc dues à un incident plutôt que de réagir à celui-ci. En complément de ces approches proactives, se sont également développées des approches tirées de l'Intelligence Artificielle, notamment en s'appuyant sur l'Apprentissage par Renforcement (RL) [11]. Leur usage est particulièrement justifié pour identifier et déterminer la politique d'activation de la réduction de vitesse.

Par la suite, nous focalisons notre état de l'art sur la façon de mettre en œuvre, en pratique, la stratégie de régulation des vitesses. Dans un premier temps, les systèmes VSL basés sur une approche proactive ou RL reposaient sur les boucles électromagnétiques pour estimer les conditions de trafic actuelles du réseau, et sur les Panneaux à Messages Variables (PMVs) pour délivrer le contrôle et donner la limitation de vitesse. C'est le cas de l'algorithme du SPECIALIST développé par Hegyi et al. [7], [6]. Cependant, en plus de ne pas garantir des informations continues (du fait de leurs positions discrètes sur le réseau), la combinaison des boucles électromagnétiques et des PMVs n'assure un système de contrôle efficace que s'ils sont en grand nombre, ce qui occasionne un coût d'installation et de maintenance conséquent. Dans ce cadre et avec le développement des C-ITS, certaines études comme Kattan et al. [9] proposent de tirer profit des environnements connectés en utilisant des données collectées par les CVs en plus de celles issues des boucles afin de disposer d'une information continue le long du réseau et d'une réduction des coûts associés aux boucles en réduisant leur nombre. En revanche, le contrôle est toujours délivré par les PMVs. Puis, d'autres études comme Grumert and Tapani [4] ont proposé d'exploiter le potentiel d'un réseau de télécommunication non plus seulement pour collecter des données issues des VACs (en complément des boucles) et estimer les états de trafic, mais aussi pour délivrer le contrôle et imposer la limitation de vitesse grâce aux Unités de Bords de Route (UBR) et à la communication Infrastructure-Véhicule (I2V).

Dans le travail précédent [3], un système VSL reposant entièrement sur les VACs, les UBRs et la communication I2V pour estimer les états de trafic et délivrer le contrôle a été implémenté. Par opposition aux précédentes approches, dites Eulériennes car s'appuyant sur des boucles (ie capteurs localisés en un emplacement précis), cette approche est intégralement basée sur les indicateurs lagrangiens issus des VACs. Les trajectoires individuelles des VACs sont utilisées pour estimer et prédire les conditions de trafic sur le réseau de manière continue pour une meilleure activation du système VSL. Le potentiel de la communication apportée par les VACs et les UBRs est également exploité pour assurer un meilleur déclenchement de la limitation de vitesse tout en réduisant les coûts associés à l'utilisation des boucles et des PMVs. [3] montre qu'un tel système VSL satisfait des performances comparables avec un système VSL basé sur des indicateurs eulériens issus des boucles. Néanmoins, les performances du système tendent à décroître pour de faibles taux de pénétration de CVs. Fort de ces conclusions, nous proposons d'explorer une nouvelle approche s'appuyant sur l'asservissement de deux véhicules connectés et automatisés en vue de créer un bouchon [12; 19] imposant la réduction de vitesse à l'ensemble des véhicules du flux de trafic en amont de la zone congestionnée. Cette stratégie est exposée par la suite et consiste à résoudre l'onde de choc en aval en créant un court intervalle avec une densité de véhicules nulle grâce à un bouchon mobile contrôlé.

Si les bouchons mobiles sont largement étudiés dans la littérature pour caractériser leur impact sur la capacité des routes [19]ou pour modéliser les véhicules lents (comme les transports publics) selon la théorie de l'écoulement du trafic [12], seules quelques études [5; 13; 16] les ont considérés comme une stratégie de régulation du trafic. Pourtant aucune n'a développé ou exploré la séquence complète du processus permettant de contrôler le trafic dans le contexte des Véhicules Connectés. La stratégie VSL basée sur les VCs et les bouchons mobile proposée ci-dessous est une démarche VSL proactive, estimée à partir de considérations tirées de la théorie du trafic : l'approche SPECIALIST [7]. Néanmoins, cette même stratégie pourrait être appliquée en faisant appel à d'autres procédés de détection des ondes de congestion et d'autres politiques d'activation telles que celles générées par les approches d'Apprentissage par Renforcement.

1.3 Positionnement : VACs agissant comme un embouteillage mobile pour réguler le trafic

Si le potentiel de la communication I2V et des VACs pour l'élaboration de nouvelles stratégies de régulation de trafic a été montré, d'après Li et al. [14], le futur déploiement de seulement quelques Véhicules Automatisés et Connectés promet déjà le développement de nouvelles méthodes jusqu'alors irréalisables du fait de la faible compatibilité / réponse / obéissance de la conduite humaine. Ainsi, les VACs automatisés (VACs) agissant comme embouteillage mouvant sur une voie rapide peuvent être considérés comme agents de régulation de trafic. En effet, en contrôlant leur vitesse, on oblige les véhicules en amont à respecter cette limitation en les bloquant derrière un bouchon mobile.

Ce type de stratégie a déjà montré son potentiel pour réguler le trafic d'un réseau à une seule voie dans Han et al. [5], où un VAC (ou plusieurs VACs) qui freine(nt) en amont d'une zone de congestion permettent d'homogénéiser le trafic.

Également, dans Piacentini et al. [16], un seul VAC agissant en bouchon mobile est utilisé pour améliorer les conditions de trafic du réseau, et notamment, de sa voie présentant un goulot d'étranglement fixe. Sa vitesse est donc considérée comme variable de contrôle permettant de réguler les conditions de trafic en amont de la congestion grâce à un Modèle Prédictif de Contrôle (MPC). Cette stratégie permet de réduire les émissions de carburant et les temps de parcours lorsqu'une congestion est observée. Les changements de voie des véhicules en amont potentiellement induits par le ralentissement du (ou des) VAC(s) ne sont, cependant, pas

considérés.
Ainsi, pour appliquer un système VSL sur une autoroute à 3 voies, Li et al. [13] propose d'utiliser une ligne de VACs agissant comme bouchon mobile sur les 3 voies. La formation de ce peloton en ligne est considérée possible grâce à la communication V2V mais son implémentation n'est ni précisée ni étudiée.
La principale contribution de ce papier consiste à développer une méthode, pour un segment d'autoroute à plusieurs voies (appliquée avec 2 voies), permettant d'assurer que 2 VACs se rejoignent et restent côte à côte pour former un bouchon mobile (de 2 VACs sur 2 voies) grâce à l'implémentation d'une boucle de contrôle PID (Proportionnel, Intégral, Dérivé) [8]. Ce procédé a, en effet, déjà montré son potentiel pour contrôler un peloton sur une voie et réguler la distance longitudinale entre deux véhicules -en donnant des consignes de vitesse à adopter au(x) véhicule(s) contrôlé(s) pour qu'il(s) conserve(nt) l'écart souhaité (et fixé) avec le(s) véhicule(s) aval(s) [2]. On l'utilise, donc, ici sur notre peloton latéral de 2 VACs en fixant l'écart (longitudinal) souhaité entre eux à 0m (pour qu'ils soient côte à côte). Tandis que le premier VAC, appelé contrôleur, fixe l'allure en suivant les consignes de vitesse fournies par l'Unité de Bord de Route (UBR) par le biais de la communication I2V ; le second VAC, appelé agent, vise à rejoindre le contrôleur et à suivre son allure grâce aux instructions de vitesse provenant de l'implémentation de la boucle de contrôle PID. Dans ce contexte d'environnement connecté, on suppose que les VACs sont équippés de technologie V2X assurant la communication avec les UBRs et entre les VCs. Le contrôleur et son agent sont sélectionnés et appariés par l'UBR, puis communiquent via la technologie V2V pour déployer la stratégie VSL, alimenter la boucle de contrôle PID et se synchroniser pour se déplacer côte à côte, et pour agir ensemble comme un bouchon mobile. L'objectif sous-jacent serait donc ici de tirer profit de l'équipement et de la coopérativité des VACs pour une application originale du système VSL. La consigne n'est plus délivrée à l'ensemble des VACs disponibles sur le réseau routier, mais seulement à un sous-ensemble (ici une paire) de VACs soigneusement et automatiquement identifiés et sélectionnés par l'UBR. Cette dernière centralise les traces laissés par les Véhicules Connectés (VCs) pour identifier les ondes de choc, puis déclenche le processus VSL composé de quatre étapes principales : (i) calcul d'une solution pour déterminer la distance d'aactivation de la VSL, (ii) recherche d'un couple pertinent de VACs (contrôleur-agent), (iii) appariement et détermination du point de rencontre du contrôleur et de son agent, puis (iv) demande d'application des consignes de vitesse au contrôleur. Il est attendu de cette stratégie qu'elle assure un plus grand respect de la limitation de vitesse, même à faible taux de pénétration, en obligeant tous les véhicules en amont du bouchon mobile à respecter la consigne sans l'avoir directement reçue. On s'attend à observer des performances équivalentes pour tout scénario où le taux de pénétration est supérieur ou égal à 30%. Les performances pour des taux plus faibles sont, en effet, surtout sensibles à la solution de limitation de vitesse qui peut être difficile à générer en raison de la faible quantité de trajectoires de VAC disponibles.
Les principales contributions de ce papier sont les suivantes :

- l'amélioration du processus de détection d'une onde de choc [3] insipré de l'approche du SPECIALIST [7], mais seulement basé sur les traces laissées par les VACs : en particulier, un processus basé sur les écarts est introduit pour appliquer la stratégie VSL.
- l'introduction d'un processus de sélection pour identifier la paire pertinente de VACs utilisés comme régulateurs de trafic.
- l'introduction d'un processus basé sur un contrôleur PID pour faire se rejoindre 2 VACs côte à côte afin de générer un goulot d'étranglement mobile.

Le reste de ce document s'organise de la façon suivante : la section 2 présente la méthodologie employée, la section 3 expose les résultats obtenus, puis la section 4 discute les conclusions et perspectives de ces travaux.

2 Méthodologie

2.1 Implémentation du système VSL : un processus en 3 étapes

Comme illustré dans la Figure 1, le système VSL développé précédemment dans Fauchet et al. [3], appelé VSL-tous VCs est basé sur la théorie des ondes cinématiques selon l'approche traditionnelle de Lighthill-Whitham-Richard (LWR) [17] utilisée pour modéliser les dynamiques macroscopiques du trafic. Elle décrit la loi de conservation du flux ($\frac{\partial \rho(x,t)}{\partial t} + \frac{\partial q(x,t)}{\partial x} = 0$) au temps t et à la position x, et suppose qu'une relation forte existe entre le débit $q(x,t)$ et la densité $\rho(x,t)$. Cette relation, appelée diagramme fondamental, suit une courbe concave, simplifiée en triangle. Le système VSL-tous VCs est composé de 3 étapes principales :

- *L'étape 1* consiste à détecter une onde de choc sur le réseau grâce aux données issues des VACs, c'est-à-dire grâce à la vitesse et à la position de chaque VAC à chaque pas de temps. On considère qu'il y a propagation d'une onde de congestion sur le réseau quand i) au moins 3 VACs ont une vitesse inférieure à la vitesse seuil et que ii) le premier d'entre eux finit par atteindre une phase d'accélération (*a.k.a.* onde de raréfaction), délimitant ainsi la zone de congestion (niveaux 1 à 3 sur la Figure 1.a).
- *L'étape 2* consiste à trouver, si possible, une solution permettant de résoudre l'onde de choc. On cherche donc la distance et le temps d'application de la limitation de vitesse sur le réseau ($[d_{activate}, d_{end}]$, $[t_{deb}, t_{end}]$) (niveaux 4 à 6 sur la Figure 1.a).
- *L'étape 3* consiste à délivrer le contrôle en appliquant cette limitation de vitesse sur la distance et le temps calculés à l'étape 2.

Dans la stratégie VSL-MB, un bouchon mobile (MB) est construit et utilisé comme régulateur de trafic. Il faut donc adapter légèrement le processus, comme l'illustre la figure 1.c&d. Si le processus de détection de l'onde de choc reste

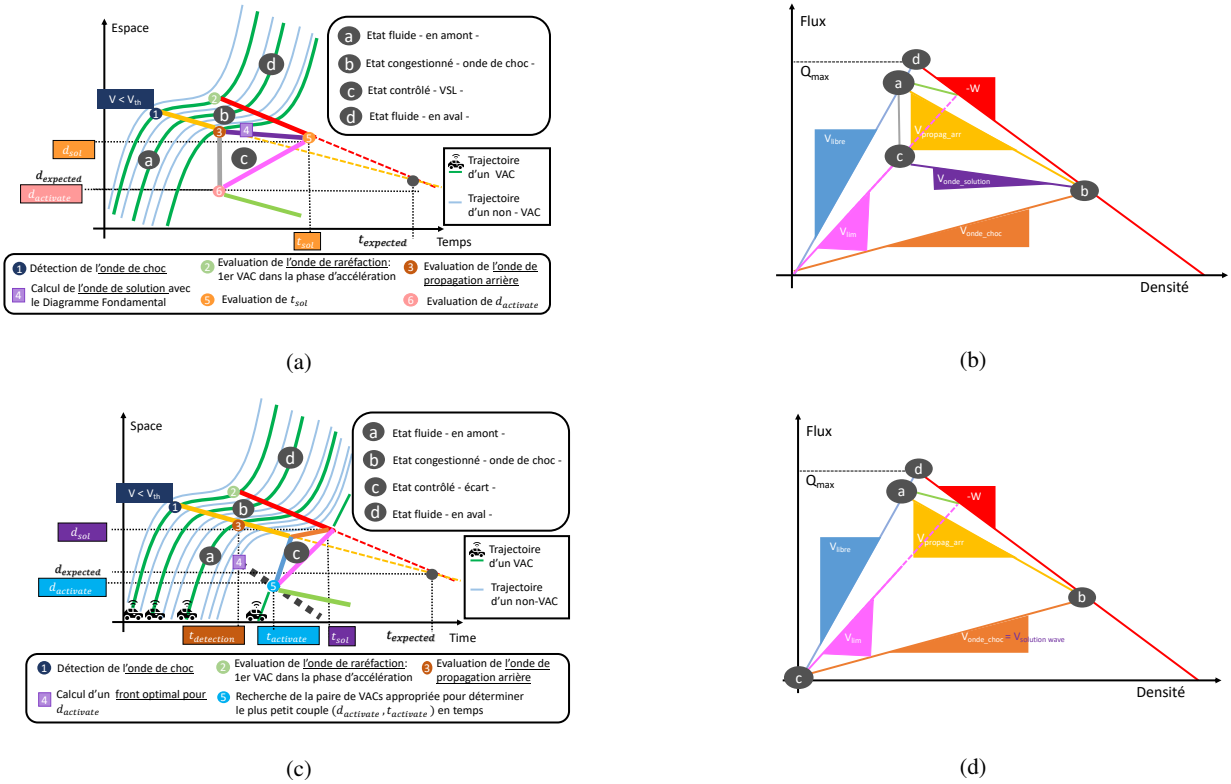


FIGURE 1 – Illustration des systèmes VSL basés sur la théorie des ondes cinématiques et isnpirés par l'algorithme du SPECIALIST [7] : (a) Diagramme Espace-Temps et (b) Diagramme Fondamental de l'approche VSL-tous VCs versus (c) Diagramme Espace-Temps et (d) Diagramme Fondamental de l'approche VSL-MB

inchangé, sa résolution repose sur un écart créé entre les véhicules en aval et les VACs contrôlés. Selon la théorie des ondes cinématiques, les caractéristiques de l'espace à densité nulle, illustré par l'état C sur la Figure 1.c, résultent directement : (i) de la pente de l'onde de propagation arrière (ligne jaune), (ii) de la vitesse moyenne dans l'onde de choc (ligne orange), (iii) de la vitesse d'écoulement libre (ligne bleue) et (iv) de la limite de vitesse appliquée (ligne rose). À partir du temps de détection de l'onde de choc ($t_{detection}$), un front optimal de distances d'activation potentielles ($d_activate$) émerge et évolue avec le temps d'activation attendu. Le défi consiste alors à identifier les paires appropriées de VACs capables de coopérer afin de rouler côte à côte avant d'atteindre le front des distances d'activation. La paire de VACs ayant le temps d'activation le plus court ($t_{activate}$) doit être sélectionnée pour appliquer les instructions de limitation de vitesse à la distance $d_{activate}$.

Par la suite, pour des raisons de simplicité et de comparabilité des résultats entre les systèmes VSL, on suppose que l'étape de résolution de l'onde de choc (étape 2) est connue et partagée entre les systèmes VSL comparés (VSL-tous VCs et VSL-MB). Seule l'étape 3 diffère entre les systèmes VSL. Cela permet de supprimer l'impact de la stochasticité due au calcul des solutions en fonction de la position des véhicules connectés. Dans la section suivante, l'étude se concentre sur la principale contribution du système VSL-MB, à savoir les performances du processus d'appariement des VACs en fonction du taux de pénétration. Au lieu d'imposer la limitation de vitesse à tous les VACs disponibles dans la zone d'activation, le système VSL-MB ne cible que deux VACs, capables de se synchroniser pour former un bouchon mobile et ainsi ralentir les véhicules situés en amont d'une section encombrée.

2.2 Application de la limitation de vitesse à 2 VACs côte à côte

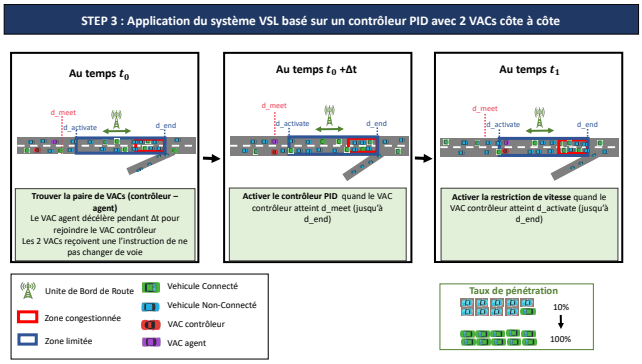


FIGURE 2 – Application de la limitation de vitesse à deux VACs côte à côte (contrôleur PID)

Comme décrit dans la Figure 2, cette troisième étape de notre système VSL se décompose en 3 sous-étapes :

1. Trouver le couple de VACs (contrôleur-agent) apte à se rapprocher côte à côte et à qui on appliquera la limitation de vitesse entre $d_{activate}$ et d_{end}.
2. Appliquer une boucle d'asservissement pour rassembler les deux véhicules de d_{meet} à d_{end}, un contrôle PID est appliqué au VAC agent pour qu'il reste côte à côte du VAC contrôleur
3. Appliquer la restriction de vitesse au VAC contrôleur de $d_{activate}$ à d_{end}.

2.3 Recherche du couple de véhicules (contrôleur-agent)

[h !] Comme illustré dans la Figure 3, le processus de sélection des deux VACs se déroule en 4 étapes :

1. *Construction de la liste de véhicules potentiels :* La liste des véhicules candidats au couple de VACs (contrôleur-agent) est établie. Pour appartenir à la liste, les véhicules doivent être des VACs et être en amont de $d_{activate}$.
2. *Sélection d'un couple de VACs (contrôleur-agent) test :* Le processus de sélection est opéré en déroulant la liste des couples candidats depuis les plus proches de la distance d'application de la réduction de vitesse $d_{activate}$. Pour un couple candidat donné, le VAC le plus en amont des deux est désigné comme étant le contrôleur et l'autre l'agent. C'est donc l'agent qui décélèrera pour rejoindre le contrôleur pendant que ce dernier ne changera pas sa vitesse. Également, le véhicule contrôleur doit être sur la voie de droite et l'agent sur celle de gauche. Si ce n'est pas le cas, la consigne de changer de voie leur est envoyée.
3. *Calcul du temps de décélération (Δt) à appliquer au VAC agent pour qu'il rejoigne le contrôleur.* Au temps t_0, on a :
   — Position et vitesse du véhicule contrôleur : xc_0 et vc_0
   — Position et vitesse du véhicule agent : xa_0 et va_0
   Au temps $t_1 = t_0 + \Delta t$, on a :
   — Position et vitesse du véhicule contrôleur : xc_1 et vc_1
   — Position et vitesse du véhicule agent : xa_1 et va_1
   On cherche la position d_{meet} à laquelle les deux VACs seront côte à côte (si elle existe), soit $d_{meet} = xc_1 = xa_1$. Or on a :

   $$xc_1 = xc_0 + vc_0\Delta t$$

   et

   $$xa_1 = xa_0 + va_0\Delta t - 1/2f\Delta t^2$$

   avec f la valeur de décélération. D'où :

   $$\begin{aligned} & xc_1 = xa_1 \\ \iff & -1/2f\Delta t^2 + (va_0 - vc_0)\Delta t + xa_0 - xc_0 = 0 \end{aligned} \quad (1)$$

   On résout ce trinôme du second degré pour trouver le temps de décélération (s'il existe) Δt pour lequel les deux VACs auront la même position à t_1.
4. *Calcul de la position où les deux VAC (contrôleur-agent) se rejoignent (d_{meet})* Si une solution positive existe pour Δt, alors on calcule d_{meet} avec :

   $$d_{meet} = xc_0 + vc_0\Delta t \quad (2)$$

   Si d_{meet} est positive et inférieure ou égale à $d_{activate}$, alors notre couple candidat de VACs (contrôleur-agent) devient notre couple (contrôleur-agent). Sinon, on teste un autre couple test (contrôleur-agent) de la liste. S'il n'y a plus de véhicules à tester dans la liste de véhicules potentiels, on attend le pas de temps suivant et on reconstruit une nouvelle liste.

2.4 Application de la limitation de vitesse au VAC contrôleur et d'un contrôle PID au VAC agent

Une fois le couple contrôleur - agent trouvé, on applique :

1. *Un contrôle PID* au véhicule agent, entre d_{meet} et d_{end}, avec pour objectif de minimiser la distance entre les 2 VACs. le rôle du PID est double : (i) compenser l'écart entre les 2 VACs, et (ii) compenser les différences de vitesse entre les 2 VACs lorsqu'ils se rencontrent à d_{meet} et jusqu'à ce qu'ils atteignent d_{end} ;
2. *Une consigne de ne pas changer de voie* aux 2 VACs entre d_{meet} et d_{end}
3. *La limitation de vitesse* au véhicule contrôleur entre $d_{activate}$ et d_{end}

Un contrôleur Proportionnel Intégral Dérivé (PID) [8] est une boucle de contrôle fermée permettant d'appliquer une correction à une fonction de contrôle. On l'utilise ici pour contrôler l'espace entre le VAC contrôleur et le VAC agent. À chaque pas de temps, on mesure un terme d'erreur ($e(t)$) égal à la différence entre la distance souhaitée entre les deux VACs (valeur consigne SP) et la distance réelle (valeur mesurée $PV(t)$), on a donc

$$e(t) = SP - PV(t)$$

. Trois composantes interagissent les unes avec les autres :
— La composante Dérivée (D) basée sur la vitesse (dérivée) à laquelle l'action est prise pour atteindre l'instruction cible
— La composante intégrale (I) basée sur l'impact cumulé (intégrale) des actions qui ont déjà été prises pour atteindre l'instruction cible
— La composante Proportionnelle (P) proportionnelle au terme d'erreur

Pour calculer la valeur d'accélération/décélération à appliquer au véhicule agent, on applique le PID au terme d'erreur. On a donc :

$$u(t) = k_p * e(t) + k_i * \int_0^t e(t')\,\mathrm{d}t' + k_d * \frac{\partial e(t)}{\partial t} \quad (3)$$

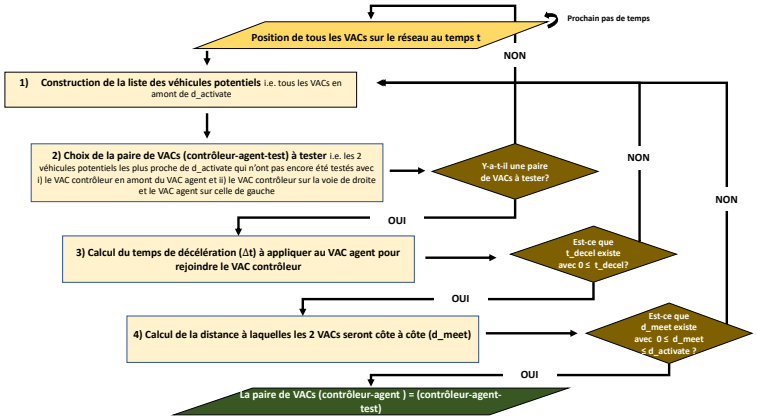


FIGURE 3 – Proccesus pour trouver le couple de véhicules (contrôleur-agent) qui seront côte à côte à qui on appliquera la limitation de vitesse

la fonction de contrôle, au pas de temps t, avec k_p, k_i et k_d les coefficients des termes proportionnel, intégral et dérivé. On utilise la méthode de Ziegler Nichols [1; 20] pour calibrer ces constantes. On ne considère que l'action proportionnelle qu'on augmente jusqu'à ce que le signal en sortie de la boucle fermée $u(t)$ oscille de manière stable et régulière. Le but est de trouver K_u le gain maximum et T_u la période d'oscillation de la boucle fermée. On applique notre PID à 2 VACs sur un réseau pour qu'ils restent côte à côte en ne considérant que la composante proportionnelle que l'on augmente jusqu'à ce que la valeur d'accélération à appliquer au VAC agent oscille de manière régulière et stable. On part de $K_u = 0.005$ et on obtient $K_u = 0.020$ et $T_u = 73.8s$. Puis, on calcule les constantes du PID avec :

$$k_p = 0.3 * K_u$$

$$k_i = 1.2 * K_u/T_u$$

$$k_d = 3 * K_u * T_u/40$$

3 Résultats

Une fois le contrôleur PID calibré, notre approche (VSL-MB) est prête à être évaluée et comparée à une stratégie VSL alternative (VSL-tous VCs). L'analyse est réalisée grâce à simulateur de trafic microscopique, appelé SUMO [15], et capable de reproduire le comportements de conduite (latéral et longitudinal) d'un usager de la route et ses intéractions avec l'environnement. Ce simulateur est composé de plusieurs couches de composants qui interagissent les uns avec les autres : (i) la couche environnement décrivant la configuration de la route et la demande de trafic, (ii) la couche agent comprenant des caractéristiques liées à tout agent actif, des conducteurs et voitures jusqu'au gestionnaire de la route, (iii) la couche capteur, et (iv) la couche communication. La couche de communication (V2X, I2V) est développée en Python et fonctionne via une architecture client/serveur basée sur TCP, appelée Traffic Control Interface (TraCI). À chaque étape, les données sur l'état du véhicule sont collectées par SUMO à l'aide de TraCI. Ensuite, de nouvelles instructions peuvent être données, basées sur différentes logiques VSL, pour modifier l'état actuel du véhicule au cours du même pas de temps.

3.1 Scénario considéré

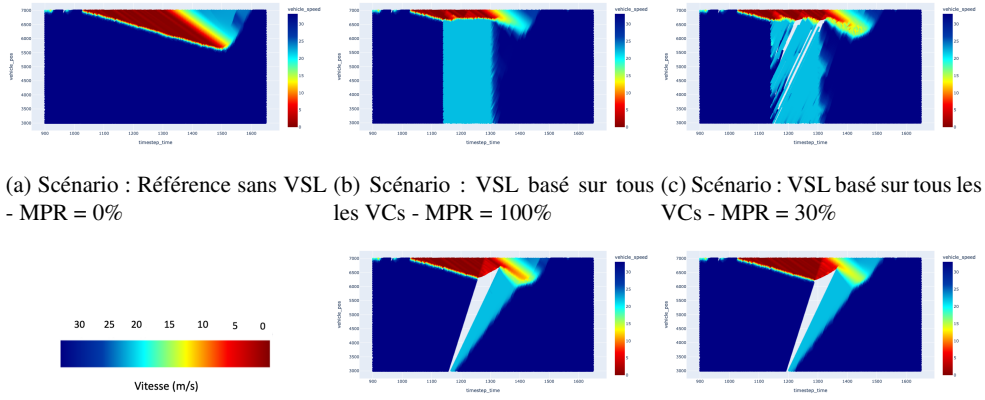
Le réseau considéré est décrit dans la Figure 2, il s'agit d'une portion de 11km d'une autoroute à deux voies comprenant une voie d'insertion au 9e km. Ici, l'analyse porte uniquement sur la manière de délivrer le contrôle VSL, la solution de limitation de vitesse est donc fixée, pour tous les scénarios, à la solution trouvée lorsque l'environnement est complètement connecté (taux de pénétration de 100%). On fixe donc $[d_{activate}, d_{end}]$, $[t_{deb}, t_{end}]$).Pour la solution de limitation de vitesse fixée, on considère différents scénarios avec un taux de pénétration de VACs (Market Penetration Rate en anglais ou MPR) variant de 0% (scénario de référence avec un environnement non connecté et donc sans aucun contrôle) à 100% (environnement entièrement connecté). Pour couvrir les cas de trafic mixte avec un environnement plus ou moins connecté, le MPR prend les valeurs discrètes suivantes (20%, 30%, 50%, 75%, 100%).

3.2 Performances de ce système

Pour les différents taux de pénétration considérés et pour les deux méthodes d'application du système VSL, on obtient les résultats suivants :

- Les temps de parcours moyens des véhicules observés sur le réseau en secondes (Table 1)
- Les diagrammes espace-temps pour MPR = 0%, MPR = 30% et MPR = 100% (Figure 4)
- Les indicateurs de déclenchement de la limitation de vitesse dans le cas de l'approche basée sur les 2 VACs formant un bouchon mobile avec $d_{triggering}$ la distance entre $d_{activate}$ et d_{meet} et Δt le temps pour que les deux véhicules se rejoignent côte à côte (Tableau 2).

Les deux applications du système VSL montrent, pour tous les scénarios considérés, une réduction des temps de parcours moyens des véhicules par rapport au scénario de référence sans contrôle (MPR=0%) (Table 1). Dans le cas de



(a) Scénario : Référence sans VSL - MPR = 0%

(b) Scénario : VSL basé sur tous les VCs - MPR = 100%

(c) Scénario : VSL basé sur tous les VCs - MPR = 30%

(d) Scénario : VSL basé sur les 2 VACs côte à côte avec le PID - MPR = 100%

(e) Scénario : VSL basé sur les 2 VACs côte à côte avec le PID - MPR = 30%

FIGURE 4 – Diagrammes espaces-temps en fonction de l'application de la restriction de vitesse

TABLE 1 – Temps de parcours moyens (ATT) des véhicules en fonction de l'application du système VSL et du taux de pénétration (MPR)

ATT(s)	**0%**	**100%**	**75%**	**50%**	**30%**	**20%**
VSL-tous VCs	112	95	96	100	101	101
VSL-MB	112	103	104	104	105	105

TABLE 2 – Indicateurs pour le déclenchement de la limitation de vitesse dans le cas de l'approche basée sur les 2 VACs formant un bouchon mobile avec $d_{triggering}$ la distance entre $d_{activate}$ et d_{meet} et Δt le temps pour que les deux véhicules se rejoignent côte à côte

MPR	**100%**	**75%**	**50%**	**30%**	**20%**
$d_{triggering}$ (m)	39	77	77	499	376
Δt (s)	0	15	15	21	26

l'application de la limitation de vitesse à tous les VACs du réseau (VSL-tous VCs), le gain de temps de parcours moyen (ATT) par véhicule par rapport à la situation de référence diminue de 6s entre MPR=100% (17s) et MPR 30% (11s), et dans le cas de la VSL basée sur les 2 VACs avec le contôleur PID, le gain de temps moyen ne diminue que de 2s entre MPR=100% (9s) et MPR 30% (7s). Il est intéressant de noter que, comme prévu, le VSL basé sur le CV MB avec PID fournit des performances plus stables (écart-type - StD = 0,84) avec les variations du taux de pénétration du marché que l'approche originale (StD = 2,88).
En outre, la figure 4 illustre les performances des approches de limitation de vitesse variable basées sur tous les VCs du réseau et sur les 2 VACs pour résoudre l'onde de choc. Elle met particulièrement en évidence :

— la stabilité des performances de l'approche introduite en ce qui concerne le taux de pénétration des VACs ;
— les différences de mise en œuvre entre les deux approches VSL (VSL-tous VCs et VSL-MB) : alors que l'approche basée sur tous les VCs aura un impact immédiat sur l'onde de choc, certains retards sont observés dans l'approche VSL-MB en raison du temps écoulé pour créer un écart comme le montre 2 avec Δt le délai nécessaire pour que les 2 VACs soient côte à côte et $d_{triggering}$ la distance à parcourir avant le déclenchement de la restriction de vitesse. Cela pourrait expliquer les avantages limités de l'approche basée sur les 2 VACs en ce qui concerne les gains en temps de parcours, puisque nous avons calculé dans cette étude la distance d'activation pour résoudre l'onde de choc selon les résultats de l'approche basée sur tous les VCs.

4 Conclusions et Perspectives

Nous avons introduit dans cet article une nouvelle approche, appelée VSL-MB, et dédiée à l'application du VSL pour amortir les ondes de choc. Cette approche consiste à former un embouteillage mobile en maintenant côte à côte deux Véhicules Automatisés et Connectés (VACs) en utilisant un contrôleur PID. L'objectif est de créer un espace entre les VACs contrôlés et les véhicules en aval afin de briser la propagation d'une onde de choc de congestion en aval. La principale contribution réside dans la mise en œuvre de la stratégie s'appuyant sur la communication V2I pour diffuser l'information, tandis que le PID est construit pour assurer la décélération d'un VAC afin de construire le front mobile et assurer sa stabilité jusqu'à ce qu'il atteigne la zone congestionnée. Un tel processus d'implémentation permet d'assurer un contrôle stable par rapport au taux de pénétration des CAVs.
Les performances de la méthode introduite sont comparées à celles d'une approche alternative [3]. Comme prévu, une forte amélioration en termes d'homogénéité des temps de parcours, lorsque le taux de pénétration des VACs varie, est mise en évidence avec la nouvelle approche. Cependant, les

avantages absolus en termes de temps de parcours sont réduits dans cette étude, ce qui pourrait s'expliquer par le fait que la distance d'activation du VSL a été optimisée pour la référence [3] afin d'éviter les stochasticités dues au calcul de la solution. En outre, le processus de sélection de la paire de CAV est construit sur l'hypothèse qu'un seul véhicule est contrôlé pour décélérer et atteindre son partenaire. Une telle hypothèse restreint l'espace des solutions disponibles et peut affecter négativement les performances.
Les travaux futurs pourraient se concentrer sur :

- améliorer et évaluer la méthodologie basée sur les 2 VACs formant un bouchon mobile pour définir correctement la distance et le temps d'activation ;
- explorer des algorithmes alternatifs pour rechercher une paire appropriée de VACs en relâchant les hypothèses sur les actions disponibles prises par les VACs contrôlés (pour que les 2 VACs se rejoignent plus rapidement, le contrôleur peut accélérer en même temps que l'agent décélère) ;
- introduire la consommation de carburant dans le critère lié au processus de fusion des deux VACs afin de limiter la surconsommation pour l'agent suivant le contrôleur.

Références

[1] H. O. Bansal, R. Sharma, and P.R. Shreeraman. PID controller tuning techniques : A review. *Journal of Control Engineering and Technology*, 2 :10, 2012.

[2] S. Dasgupta, V. Raghuraman, A. Choudhury, T.N. Teja, and J. Dauwels. Merging and splitting maneuver of platoons by means of a novel pid controller. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8, 2017.

[3] E. Fauchet, K. Bhattacharyya, P.A. Laharotte, and N.E. El Faouzi. A variable speed limit strategy based on lagrangian traffic indicators resulting from connected vehicles : Application to the SPECIALIST algorithm. *Transportation Research Board Conference Proceedings*, 2022.

[4] E.F. Grumert and A. Tapani. Using connected vehicles in a variable speed limit system. *Transportation Research Procedia*, 27 :85–92, 2017.

[5] Y. Han, D. Chen, and S. Ahn. Variable speed limit control at fixed freeway bottlenecks using connected vehicles. *Transportation Research Part B : Methodological*, 98 :113–134, 2017.

[6] A. Hegyi and S.P. Hoogendoorn. Dynamic speed limit control to resolve shock waves on freeways - field test results of the SPECIALIST algorithm. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 519–524. IEEE, 2010.

[7] A. Hegyi, S.P. Hoogendoorn, M. Schreuder, H. Stoelhorst, and F. Viti. SPECIALIST : A dynamic speed limit control algorithm based on shock wave theory. In *2008 11th International IEEE Conference on Intelligent Transportation Systems*, pages 827–832. IEEE, 2008.

[8] M.A. Johnson and M.H. Moradi. *PID control*. Springer, 2005.

[9] L. Kattan, B. Khondaker, O. Derushkina, and E. Poosarla. A probe-based variable speed limit system. *Journal of Intelligent Transportation Systems*, 19(4) : 339–354, 2015.

[10] B. Khondaker and L. Kattan. Variable speed limit : an overview. *Transportation Letters*, 7(5) :264–278, 2015.

[11] K. Kušić, E. Ivanjko, M. Gregurić, and M. Miletić. An overview of reinforcement learning methods for variable speed limit control. *Applied Sciences*, 10(14), 2020.

[12] L. Leclercq, S. Chanut, and J.B. Lesort. Moving bottlenecks in lighthill-whitham-richards model : A unified theory. *Transportation Research Record*, 1883 (1) :3–13, 2004.

[13] Z. Li, X. Zhu, X. Liu, and X. Qu. Model-based predictive variable speed limit control on multi-lane freeways with a line of connected automated vehicles. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 1989–1994. IEEE, 2019.

[14] Z. Li, M.W. Levin, R. Stem, and X. Qu. A network traffic model with controlled autonomous vehicles acting as moving bottlenecks. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE, 2020.

[15] P.A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner. Microscopic traffic simulation using sumo. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2575–2582. IEEE, 2018.

[16] G. Piacentini, P. Goatin, and A. Ferrara. Traffic control via moving bottleneck of coordinated vehicles. *IFAC-PapersOnLine*, 51(9) :13–18, 2018.

[17] P.I. Richards. Shock waves on the highway. *Operations research*, 4(1) :42–51, 1956.

[18] F. Vrbanić, E. Ivanjko, K. Kušić, and D. Čakija. Variable speed limit and ramp metering for mixed traffic flows : A review and open questions. *Applied Sciences*, 11(6), 2021.

[19] X. Wei, C. Xu, W. Wang, M. Yang, and X. Ren. Evaluation of average travel delay caused by moving bottlenecks on highways. *Plos one*, 12(8), 2017.

[20] J. G. Ziegler and N. B. Nichols. Optimum settings for automatic controllers. *Journal of Dynamic Systems, Measurement, and Control*, 115(2) :220–222, 1993.

Éléments d'état de l'art sur la cartographie sémantique et son applicabilité en environnement industriel

A. Achour[1,3], H. Al-Assaad[1], M. El Zaher[1],Y. Dupuis[2]

[1] CESI campus de Toulouse, LINEACT CESI
[2] CESI, LINEACT CESI
[3] ENSAM, 75013 Paris, France

Résumé

La sécurité dans la collaboration homme-robot est l'un des principaux défis de l'industrie 5.0. Les robots doivent à présent s'adapter à l'humain. Ils évaluent et améliorent le système de navigation sur la base de technologies de cartographie sémantique qui permet de conduire un raisonnement clair. Ces cartes permettent aux robots de comprendre l'environnement qui les entoure pour assurer la sécurité des humains. L'objectif de cet article de synthèse est de présenter les différentes méthodes de cartographie sémantique mono-robot et l'évolution de ces systèmes, existant dans la littérature, et basés sur des techniques d'IA. Dans cette perspective, nous mettrons en avant les méthodes récentes et applicables dans le domaine industriel.

Mots-clés

Cartographie sémantique mono-robot, Fusion multi-sources, Collaboration homme-robot, Environnement industriel.

Abstract

Safety in human-robot collaboration is one of the main challenges of Industry 5.0. Now, robots must adapt to humans. They evaluate and improve the navigation system based on semantic mapping technologies that enable clear reasoning. These maps allow robots to understand the environment around them to ensure human safety. The objective of this review article is to present the different single-robot semantic mapping methods and the evolution of these systems, existing in the literature, and based on AI techniques. In this perspective, we will highlight the recent methods applicable in the industrial domain.

Keywords

Single-robot semantic mapping, Multi-source fusion, Human-Robot collaboration, Industrial environment.

1 Introduction

Dans la littérature, il existe de nombreux travaux qui s'intéressent au problème de la cartographie d'intérieur afin de proposer de nouvelles approches capables de générer une meilleure représentation de l'environnement. La majorité des approches proposées sont basées sur des données de perceptions des robots mobiles. Elles permettent de générer deux grandes catégories de cartes : les cartes métriques et topologiques. Ces cartes sont principalement utilisées pour permettre au robot de se localiser dans son environnement et de planifier sa trajectoire. Les cartes métriques fournissent une représentation géométrique des objets de l'environnement dans un cadre de référence global. Un problème majeur de ces cartes est l'incertitude des mouvements des robots due à l'accumulation des erreurs de perception et de localisation. Les cartes topologiques représentent l'environnement sous la forme d'un graphe, où les nœuds représentent des lieux et les arêtes représentent les relations entre ces lieux. D'après [1], l'avantage principal des cartes topologiques est qu'elles font abstraction des problèmes d'incertitude dans le mouvement du robot : comme il navigue localement entre les nœuds, il n'y a pas une accumulation globale d'erreurs. Par ailleurs, ce manque des informations métriques peut être considéré comme l'inconvénient principal de ces cartes. Afin de surmonter les limites de ces deux cartes, des cartes hybrides combinant les deux ont été proposées. Ces cartes permettent d'utiliser la carte métrique pour une localisation précise et un graphe topologique global pour se déplacer d'un endroit à un autre [2]. Mais, malgré les avantages de ces différentes cartes, elles sont toutes considérées comme des représentations de bas niveau de l'environnement.

Depuis quelques années, afin de faire face aux nouveaux besoins dans le domaine de la robotique mobile, un nouveau concept, appelé cartographie sémantique, est proposé. Ce dernier permet d'ajouter des informations sémantiques à la carte afin d'avoir des robots plus efficaces, avec une diversité d'actions planifiées et capables de réagir à des événements inattendus [3]. La carte sémantique représente l'environnement avec des concepts de haut niveau et ajoute une signification sémantique aux éléments cartographiés. *Nüchter et al.* [4] la définissent comme une carte qui contient, en plus des informations spatiales, une association des éléments physiques cartographiés avec des entités de classes connues. Une base d'informations supplémentaires sur ces entités, structurée et indépendante de la carte, est également nécessaire pour permettre au robot de raisonner. Plusieurs

travaux sur la cartographie sémantique ont été réalisés, mais la plupart d'entre eux se concentrent sur la représentation d'environnements statiques. En effet, l'intérêt de la majorité de ces travaux est de déployer une carte sémantique dans le système de navigation des robots de service afin que les utilisateurs puissent les faire fonctionner à l'aide de commandes de haut niveau, par exemple en utilisant des commandes vocales ou gestuelles [5].
Récemment, un nouveau contexte industriel, l'industrie 5.0, où les humains et les robots partagent le même espace de travail [6], se met en place. En effet, les barrières physiques entre les humains et les robots sont totalement ou partiellement supprimées. Ainsi, en plus de la collaboration directe, où l'humain et le robot travaillent simultanément sur une tâche commune, un nouveau mode de collaboration apparaît, où l'humain et le robot travaillent à proximité l'un de l'autre pour effectuer des tâches différentes. Il est donc question d'améliorer les capacités des robots pour les rendre plus adaptés à cette nouvelle façon de coopération homme-robot. En effet, il est nécessaire de disposer d'une carte sémantique à jour des éléments statiques et dynamiques de l'environnement [3]. Par exemple, cette carte peut être utilisée par les robots mobiles pour éviter les collisions homme-robot / robot-robot ou pour effectuer des tâches plus complexes, pour analyser et optimiser le flux des déplacements des robots et des opérateurs, et pour fournir un inventaire en cas d'accident ou d'incendie. Alors, notre objectif est de proposer un état de l'art préliminaire des méthodes de cartographie sémantique pour étudier leur applicabilité dans un environnement industriel.
Dans ce travail, nous présentons, dans une première section, le processus de cartographie sémantique. Nous étudions les différentes méthodes d'acquisition de données géométriques et de données sémantiques proposées, ainsi que les méthodes de représentation de ces données. Ensuite, dans une deuxième section, nous étudions les caractéristiques et les besoins de l'environnement industriel. Cette étude vise à identifier les méthodes de cartographie utilisées dans le domaine domestique, qui seront applicables dans le domaine industriel.

2 Cartographie sémantique

La cartographie sémantique est le processus qui consiste à générer une carte sémantique à partir de données provenant de sources différentes, potentiellement hétérogènes. Ces données sont traitées pour générer une représentation géométrique et obtenir des informations sémantiques de l'environnement (Fig.1). Ensuite, cette représentation et les informations sémantiques obtenues sont organisées dans une structure pour obtenir une carte sémantique. Il existe des approches de cartographie sémantique mono-robot et des approches multi-robots. Les approches étudiées dans cette section sont uniquement celles basées sur un seul robot qui peut être assisté par un humain.

2.1 Acquisition et traitement de données

Dans cette section, les trois types d'acquisition de données utilisés dans le processus de cartographie sémantique sont

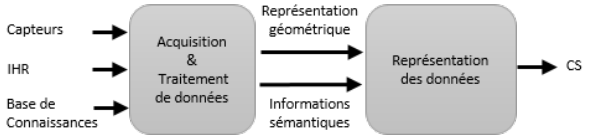


FIGURE 1 – Le processus de cartographie sémantique (CS = Carte Sémantique)

présentés : par les capteurs, par l'interaction homme-robot (IHR) et par le raisonnement.

2.1.1 Acquisition des données par les capteurs

Les robots mobiles sont dotés de différents capteurs pour réaliser la cartographie. Les données acquises par ces capteurs sont utilisées pour la représentation géométrique de l'environnement et l'extraction d'informations sémantiques, permettant de relier les entités physiques, tels que les lieux et les objets, à une base de données contenant des informations supplémentaires sur l'environnement. En général, la cartographie géométrique est basée sur les méthodes classiques SLAM (Simultaneous Localization And Mapping), qui consistent à estimer simultanément l'état d'un robot et à construire un modèle de l'environnement perçu par ses capteurs [7]. Il convient de noter que ce travail ne couvre pas les systèmes qui utilisent des informations sémantiques, telles que des informations sur les objets dans l'environnement, au profit du SLAM. Ils s'intéressent toutefois aux approches qui utilisent les systèmes SLAM classiques avec d'autres techniques telles que la segmentation et la reconnaissance d'objets pour créer une carte sémantique de l'environnement.
Qi et al. [8] proposent une approche pour créer une carte d'occupation sémantique d'un environnement domestique qui permet aux utilisateurs finaux de robots de spécifier les tâches de navigation d'une manière plus pratique en utilisant le langage naturel. Ils utilisent la méthode SLAM hors ligne basée sur des capteurs sonars, proposée dans [9], pour la création d'une grille d'occupation. Ensuite, grâce à l'odométrie et à une caméra stéréo, ils utilisent une méthode basée sur la détection d'objets et la triangulation pour y ajouter les espaces topologiques des objets détectés avec leurs étiquettes. Dans [8], les capteurs sonars ont été préférés aux capteurs laser, plus précis, car ces derniers sont peu coûteux, légers et nécessitent peu de ressources de calcul. Ce travail compare la carte obtenue à une autre créée en utilisant des capteurs laser. Le résultat final montre que la carte obtenue par le laser est plus fine, mais elle manque certains obstacles puisque le capteur scanne un plan 2D. Par contre, la carte obtenue par les capteurs sonars représente parfaitement ces obstacles car ils scannent en 3D.
Li et al. proposent dans [10] une méthode de reconstruction 3D et de segmentation de l'environnement à partir d'un flux vidéo destinée aux robots opérant dans des environnements intérieurs et extérieurs. En effet, une méthode SLAM monoculaire est utilisée car elle permet de représenter un environnement semi-dense sans avoir besoin d'informations de profondeur. De plus, elle permet de surmonter les limites

des méthodes basées sur des capteurs à petit échelle, tels que les caméras stéréo et les caméras RGB-D, qui ne fournissent une représentation fiable de l'environnement que dans leur plage de fonctionnement limitée et perdent en efficacité lors du passage d'un environnement à un autre. Dans ce travail, un modèle CNN (Convolutional Neural Network) pré-entraîné est utilisé pour l'extraction d'informations sémantiques en parallèle à la reconstruction 3D. *Niko et al.* [11] ont également proposé une approche qui utilise une caméra RGB-D pour cartographier un environnement 3D et reconstruire à la volée des modèles d'objets détectés. En effet, ORB-SLAM2 est utilisé pour la cartographie et la localisation de la caméra sur chaque image RGB-D, en parallèle les objets sont détectés sur les images RGB et le nuage de points associé est segmenté en 3D. Les travaux [10, 11] utilisent des techniques d'apprentissage profond basées sur l'image pour l'acquisition d'informations sémantiques grâce aux grands progrès réalisés par ces derniers dans la détection d'objets et la segmentation basée sur les images 2D. Dans les travaux de [12], les auteurs proposent également une méthode de détection d'objets basée sur une caméra RGB-D de type Kinect pour effectuer une navigation sémantique. L'approche proposée fusionne deux techniques de détection d'objets : la détection des contours à partir de l'image de profondeur et la détermination de la similarité basée sur les descripteurs à partir des images RGB. Par ailleurs, *Zender et al.* [13] proposent un système qui crée une représentation multi-couches d'un environnement d'intérieur. Ils utilisent un algorithme EKF-SLAM basé sur le laser pour créer les couches de bas niveau de la représentation et l'algorithme de reconnaissance d'objets RFCH (Receptive Field Cooccurrence Histograms) basé sur la caméra RGB pour établir un lien avec la couche de haut niveau.

En conclusion, le choix des capteurs ne dépend pas seulement du type de carte que l'on souhaite obtenir, notamment une carte 2D/3D, un nuage de points ou une carte en couleurs. Mais, il dépend également de niveau de la description de l'environnement que l'on souhaite obtenir. De plus, les capteurs les plus populaires pour la cartographie géométrique sont les lasers 2D/3D, et récemment les caméras RGB-D sont de plus en plus utilisées [14]. Ces caméras sont relativement peu coûteuses et peuvent fournir simultanément une image couleur et une carte de profondeur précise et de haute résolution sur une portée de 5 à 7 mètres, caractérisant la distance des objets perçus dans l'image. D'autre part, les capteurs visuels sont fréquemment choisis pour l'acquisition d'informations sémantiques, car ils permettent une représentation de l'environnement proche de celle perçue par l'œil humain. Ils permettent également de collecter des données de bas niveau, telles que des points, et des données de haut niveau, comme les catégories d'objets [2].

2.1.2 Acquisition des données sémantiques par l'interaction homme-robot

Selon [5], l'extraction automatique d'informations sémantiques à partir de capteurs est limitée et n'est pas assez robuste. En effet, la reconnaissance d'objets ou de lieux est une tâche complexe pour les robots. Pour ces raisons, certaines techniques intègrent l'humain dans le processus d'identification d'objets ou de lieux pour la création d'une carte augmentée [3, 14].

Dans le travail de [5], un utilisateur guide le robot dans une visite de l'environnement pour sa cartographie. L'algorithme RBPF-SLAM basé sur des données laser et des données d'odométrie est utilisé pour générer en temps réel une carte d'occupation. En même temps, l'utilisateur dispose d'une application vocale basée sur IFLYTEK sur son téléphone portable pour labelliser les lieux sur cette carte. Afin d'éviter les erreurs dans la reconnaissance des commandes vocales données par l'opérateur, une étape de confirmation des informations est appliquée. Dans la même optique [13], le robot possède des connaissances préalables sur les concepts spatiaux, et le rôle de l'utilisateur est de l'assister dans le processus de labellisation des lieux. En effet, tout en marchant avec le robot, l'utilisateur exprime ce qu'il considère comme pertinent, par exemple : « C'est le couloir » ou « C'est la station de recharge ».

Par ailleurs, les auteurs de l'article [15] proposent une approche où le robot est guidé par l'utilisateur dans une visite de l'environnement opérationnel. Il perçoit son environnement et détecte l'objet pointé par un laser. Ensuite, il le segmente dans l'image et estime sa position et son orientation. Enfin, il reçoit sa description de l'utilisateur via un module de reconnaissance vocale et l'associe à sa position pour construire la représentation de l'objet à ajouter à la carte. Dans [16], *Bastianelli et al.* ont amélioré cette approche en implémentant un système qui permet d'acquérir de nouveaux objets dans la représentation grâce à une interaction continue et en ligne avec l'utilisateur après la création initiale de la carte sémantique. Les connaissances sont alors acquises selon les besoins et ajoutées progressivement à la représentation de l'environnement par le robot. Par exemple, si l'utilisateur donne une commande vocale indiquant un emplacement inconnu pour le robot, un processus d'acquisition intégré par un réseau de Petri est lancé. Pendant ce processus, l'utilisateur peut guider le système avec des commandes vocales telles que « Tournez à droite », « Suivez-moi » ou « Allez à la cuisine ». Lorsque le robot se trouve devant un nouvel emplacement ou objet à mémoriser, l'utilisateur peut pointer l'objet et indiquer au robot sa référence, par exemple « C'est la porte de secours ».

Pronobis et Jensfelt [17] proposent un algorithme qui combine des informations sur la présence d'objets avec des propriétés sémantiques des lieux telles que la taille et l'apparence pour classer les pièces. Pendant le processus de cartographie, l'utilisateur peut introduire des informations supplémentaires sur les objets présents dans la pièce à l'aide d'une interface sur le PC utilisé par le robot. Si l'utilisateur fournit une information sur l'existence d'un objet, le robot la traite comme une autre source d'information. Les travaux de *Crespo et al.* [18] ont mis en œuvre des dialogues en langage naturel entre les utilisateurs et le robot par une interface vocale et le clavier pour ajouter les catégories d'objets et leurs relations sémantiques à la carte. Par exemple, le robot peut interroger l'utilisateur sur les utilisations possibles

d'un objet ou sur les possibilités d'interaction avec les objets de l'environnement.

En conclusion, les méthodes d'acquisition d'informations par l'interaction homme-robot ajoutent une variété d'informations sémantiques, notamment des données sur les objets de la scène (la catégorie de l'objet, l'espace qu'il occupe et son occurrence dans une pièce), des données sur les lieux (la catégorie de la pièce et ces caractéristiques), et des données sur les relations sémantiques des objets (par exemple l'utilité d'un objet). De plus, il existe de nombreux travaux pour l'acquisition d'informations sémantiques dans des scènes statiques, mais seuls quelques travaux abordent le problème de l'acquisition de données dans une scène dynamique [16].

2.1.3 Acquisition des données sémantiques par le raisonnement

La troisième méthode d'acquisition de données est celle par raisonnement. Elle consiste à raisonner, en utilisant les données déjà obtenues par les autres méthodes et les connaissances fournies par une base de connaissances de sens commun, afin d'obtenir de nouvelles informations sur l'environnement. Les connaissances de sens commun sont celles que tous les humains possèdent et qu'ils ont acquises depuis leur naissance sans même en avoir conscience [19]. Cette base est généralement représentée comme une structure composée de concepts décrivant l'environnement liés par des lois.

Par exemple, *Galindo et al.* [20] proposent une structure hiérarchique d'informations conceptuelles modélisée par le système NeoClassic. Le niveau le plus bas de cette structure est composé de symboles liés à des éléments physiques détectés dans l'environnement. La structure conceptuelle est composée de lois qui relient ces symboles à des concepts et relient les concepts entre eux pour permettre un raisonnement sur les symboles de catégorie connue. Par exemple, si un élément physique détecté est lié au symbole *« four-1 »*, que ce dernier est lié au concept *« four »*, et que le concept *« four »* est lié au concept *« cuisine »* par un lien *« contient »*, alors le lieu actuel du robot peut être déduit comme étant la cuisine. D'autre part, dans [13], les connaissances de sens commun sur un environnement intérieur d'un bureau sont modélisées dans une ontologie. L'ontologie constitue un modèle de données représentant un ensemble de concepts dans un domaine, ainsi que les relations entre ces concepts. Sur la base de cette description, le système utilise un logiciel de raisonnement pour déduire des catégories plus spécifiques pour les zones topologiques connues. En effet, si une zone est classée comme une pièce et qu'elle contient un canapé selon les informations acquises, donc grâce aux connaissances conceptuelles données dans l'ontologie, cette zone peut être classée comme une instance de salon. De même, une ontologie est utilisée dans la carte sémantique de [21] pour créer une stratégie de planification des tâches pour les robots de service. Dans cette ontologie, le concept *« objets »* est séparé en *« objets statiques »* et *« objets dynamiques »*. Ensuite, les objets statiques sont liés aux objets dynamiques par des relations probabilistes qui permettent d'obtenir par raisonnement les positions approximatives des objets dynamiques. En outre, les auteurs de [17] proposent une structure conceptuelle probabiliste où les lois reliant les concepts peuvent être prédéfinies, acquises ou déduites et peuvent être probabilistes ou pas. Les catégories et les propriétés des pièces sont estimées ensemble à l'aide d'un modèle de graphe de chaîne probabiliste. Les propriétés de la pièce telles que les objets, l'apparence, la surface et la forme sont extraites des capteurs, puis utilisées avec les relations pièces-objets fournies par la base de connaissances « Open Mind Indoor Common Sense » pour déduire la catégorie de la pièce. Les auteurs de [18] proposent un modèle relationnel qui permet de stocker facilement des informations dans des tables gérées par un moteur de raisonnement. La conception de ce modèle définit implicitement les relations entre les entités. Il n'est donc pas nécessaire de définir des lois entre elles comme pour d'autres structures. En utilisant ce modèle, si le robot identifie des objets dans la même pièce qui ont une utilité commune, il est possible de déduire que ces objets sont situés dans une pièce où le robot pourrait trouver d'autres objets ayant des utilités connexes. Par conséquent, de nouvelles catégories de pièces sont créées de manière autonome. L'identification des objets et des pièces permet également de détecter d'éventuelles incohérences.

Alors, dans le processus de cartographie sémantique, un système de raisonnement peut être mis en œuvre pour intégrer des informations supplémentaires dans la représentation de l'environnement. Ce système comprend une base de données contenant des connaissances de sens commun sur l'environnement d'application et un moteur de raisonnement pour exploiter ces données. La base de données est composée d'informations prédéfinies avant le processus de cartographie et d'informations acquises par les capteurs du robot. En utilisant ces informations structurées et le moteur de raisonnement, de nouvelles informations sont déduites pour enrichir la représentation.

2.2 Représentation de la carte sémantique

Afin d'exploiter les données sémantiques extraites par les méthodes présentées précédemment, il est nécessaire de les organiser dans une structure adaptée à l'application.

La méthode la plus simple est d'attacher directement les informations sémantiques aux éléments physiques associés dans la carte géométrique. Par exemple, *Sunderhauf et al.* [11] utilisent ORB-SLAM2 pour créer un nuage de points de l'environnement, et les points associés à chaque objet détecté sont modélisés par une couleur différente. Dans la même optique, une carte d'occupation sémantique est créée en ajoutant des MBRs (Minimum Bounding Rectangles), qui représentent les espaces occupés par les objets, à leurs zones topologiques dans la carte [8]. Chaque nouvelle catégorie d'objet est représentée par une couleur différente. *Zhao et al.* [5] utilisent également le module Gmapping de ROS pour créer une carte d'occupation de l'environnement. Au cours de ce processus, lorsque le robot reçoit une information sémantique, il la combine avec sa position et ajoute un nouveau nœud sémantique à la carte pour la représenter.

Les cartes obtenues par ces méthodes permettent une interprétation directe des informations sémantiques en position. Par conséquent, elles peuvent être facilement déployées dans les systèmes de navigation des robots mobiles pour effectuer des tâches de navigation sémantique. Cependant, ces représentations simples ne permettent pas au robot d'effectuer des tâches complexes telles que le raisonnement de mission. Par conséquent, d'autres représentations plus intéressantes ont été proposées, à savoir des représentations hiérarchiques, où les informations spécifiques sont placées dans les niveaux inférieurs et les informations abstraites dans les niveaux supérieurs.

Pronobis et al. [17] proposent une représentation qui donne au robot la capacité de déduire les catégories sémantiques des pièces en utilisant les propriétés de l'espace, notamment l'apparence, la surface, la forme et les objets. Ils représentent la connaissance spatiale par une structure hiérarchique à 4 couches : la couche sensorielle, la couche topologique, la couche des catégories et la couche conceptuelle. Le niveau le plus bas de la structure contient les données de perception des capteurs et le 4^{eme} niveau contient les connaissances conceptuelles abstraites. La couche sensorielle contient une carte métrique de l'environnement. La couche topologique contient un graphe topologique avec des nœuds représentant des lieux et des arêtes codant le chemin vers un autre nœud. La couche des catégories contient les objets reconnus et les propriétés de l'espace obtenues par des modèles de classification géométriques et visuels. Enfin, la couche conceptuelle contient une ontologie statique de connaissances de sens commun et des relations reliant les concepts de cette ontologie aux connaissances de bas niveau perçues dans les trois autres couches. Cette structure permet non seulement au robot de classer les pièces, mais aussi de prédire l'existence d'objets, les propriétés de l'espace et les espaces inexplorés. Par ailleurs, il existe d'autres travaux qui séparent les connaissances spatiales des connaissance conceptuelles pour obtenir une représentation plus organisés de l'environnement. Par exemple, dans [20], un modèle de représentation avec deux structures hiérarchiques est proposé pour permettre au robot d'exploiter les informations sémantiques recueillies par ces capteurs dans les tâches de navigation. La première structure est une hiérarchie qui représente les connaissances spatiales par trois niveaux d'abstraction. Le premier niveau, le plus bas, contient des cartes d'occupation locale, des lieux et des images, stockés par le robot. Le deuxième niveau contient un graphe topologique de l'environnement. Et le troisième niveau contient un nœud abstrait qui représente l'ensemble de l'environnement. La deuxième structure est une hiérarchie qui représente les connaissances conceptuelles par quatre niveaux. Elle est codée manuellement en amont avec le langage NeoClassic. Le niveau le plus haut contient un nœud appelé *« Thing »* à partir duquel sortent deux branches *« Pièces »* et *« Objets »*. Le troisième niveau contient les catégories d'objets et de pièces, et le niveau le plus bas contient les instances d'objets et de lieux reconnus par le robot. Les deux hiérarchies sont liées par le processus d'ancrage qui consiste à lier les entités reconnues à différents niveaux de la hiérarchie spatiale aux symboles qui les désignent dans la hiérarchie conceptuelle. Ces liens permettent au robot d'exploiter les informations sémantiques pour déterminer les erreurs de localisation en raisonnant sur les emplacements attendus des objets et pour effectuer des tâches de navigation sémantique. Également, dans [16], la connaissance du robot est divisée en deux parties : la connaissance du monde, qui représente la connaissance spécifique d'un certain environnement, et la connaissance du domaine, qui est la connaissance générale du domaine d'application. La représentation de la connaissance du monde contient les éléments suivants : une grille d'occupation, des cartes cellulaires pour représenter les emplacements locaux, un graphe topologique de l'environnement global et des instances d'éléments physiques reconnus avec leurs catégories et propriétés. D'autre part, la connaissance du domaine est composée comme dans [20] par les concepts impliqués dans l'environnement ainsi que leurs propriétés et relations. Cependant, pour ce travail, contrairement à [20, 17], l'objectif de l'information sémantique ajoutée par la connaissance du domaine n'est pas de classer automatiquement les pièces ou d'effectuer une navigation sémantique, mais plutôt d'obtenir et d'associer à chaque objet perçu par le robot ces propriétés spatiales et fonctionnelles afin de pouvoir raisonner sur les tâches liées à ces objets.

En conclusion [14], les représentations qui consistent à visualiser les informations sémantiques par différentes couleurs sur la carte permettent de montrer les résultats de l'acquisition des connaissances sémantiques, mais elles ne sont pas faciles à implémenter dans les tâches des robots. Par contre, les représentations par structure hiérarchique ne permettent pas de visualiser ces connaissances sur une carte, mais elles permettent de structurer et d'organiser les informations par niveaux d'abstraction pour rendre le robot capable de raisonner.

3 Synthèse

Cette section présente une synthèse des méthodes de cartographie sémantique proposées dans ce travail et étudie leur applicabilité dans un environnement industriel d'intérieur. En effet, ces environnements présentent des caractéristiques et des contraintes spécifiques à prendre en compte pour leur cartographie sémantique. Ils sont généralement de grande taille, dynamiques et à structure changeante [22]. Ils possèdent leurs propres concepts et informations sémantiques. De plus, dans le contexte de l'industrie 5.0, où l'homme et le robot partagent le même espace de travail, le renforcement de la sécurité de la collaboration homme-robot est un besoin majeur de cet environnement.

Comme le montre le tableau 1, la plupart des approches dans la littérature [13, 16, 5, 10, 18, 8, 21], sont proposées pour cartographier des environnements domestiques. Ceci est évident puisque ces dernières années, plusieurs travaux ont porté sur la facilitation de l'intégration des robots de service dans les environnements domestiques et sur l'amélioration de leur fonctionnement en intégrant la navigation

Réf	Environnements	Approche	Sources d'informations sémantiques	La sémantique ajouté	
				Sur les objets	Sur les lieux
[16]	- Env. domestique - Env. de bureaux - Dynamique - Petite échelle	Approche incrémentale par interaction homme-robot	IHR	Catégories, positions, tailles et caractéristiques	Catégories, positions du robot
[18]	- Env. domestique	_	Raisonnement	Catégories, instances utilités, caractéristiques, relations avec les lieux	Catégories et instances
[10]	- Env. d'intérieur et d'extérieur - Moyennement dense	LSD-SLAM monoculaire, Segmentation 2D avec CNN	Capteurs	Catégories	_
[17]	- Env. de bureaux - Large et non structuré	EKF-SLAM , Reconnaissance et Raisonnement d'instances, Classification des propriétés	Capteurs - IHR - Raisonnement	Catégories, relations avec d'autres concepts de l'environnement	Apparence, surface et forme
[8]	- Env. domestique - Petite échelle	SLAM basé sur des capteurs sonars [9], Détection et triangulation d'objets	Capteurs	Catégories	Catégories
[11]	- Env. de bureaux - Espacé	RGBD-SLAM (ORB-SLAM2), Détection d'objets, Segmentation 3D avec CNN	Capteurs	Catégories et instances	_
[21]	- Env. domestique - Dynamique - Non structuré	Monocular SLAM, Détection d'objets	Capteurs - Raisonnement	Catégories, relations avec d'autres concepts de l'environnement	Catégories
[13]	- Env. domestique - Petite échelle	SLAM à base de vision, Reconnaissance d'objets	Capteurs - IHR - Raisonnement	Catégories et instances	Catégories et instances
[5]	- Env. domestique - Petite échelle	RBPF-SLAM basé sur les données laser et odométrie, Reconnaissance vocale	IHR	_	Catégories

TABLE 1 – Résumé des différentes approches de cartographie sémantique citées dans cet article (Env. = Environnement, IHR = Interaction Homme-Robot).

sémantique [3]. D'autres travaux [17, 11], se sont intéressés aux environnements de bureaux ayant les mêmes caractéristiques qu'un environnement domestique. Contrairement aux environnements industriels, ces derniers sont généralement de petite taille et leur structure ne varie pas aussi rapidement que celle d'un environnement industriel.

Les méthodes SLAM sont choisies en fonction du type de l'environnement à cartographier, par exemple, SLAM semi-dense est utilisé dans [10] pour cartographier un environnement moyennement dense. D'autre part, EKF-SLAM est utilisée dans [17] pour cartographier un grand environnement de bureaux non structuré, et RBPF-SLAM est utilisée dans [5] pour cartographier un environnement domestique de petite taille. Les méthodes SLAM basées sur des capteurs laser sont considérées comme les plus robustes dans l'environnement industriel [22], car elles peuvent fournir un positionnement précis dans un environnement changeant et en présence d'obstacles dynamiques. Alors, les approches [13, 17, 16, 5] utilisant ces capteurs conviennent le mieux à cet environnement. En particulier, l'approche de *Pronobis et al.* [17] qui traite un environnement large et non structuré.

En ce qui concerne les informations sémantiques collectées, elles dépendent principalement de l'application à réaliser. Si l'objectif est de permettre la navigation sémantique comme dans [8, 5], une carte d'occupation sémantique, où les catégories d'objets et les lieux topologiques sont associés à leurs positions dans la grille d'occupation, est suffisante. Dans les travaux récents, ces informations sont collectées par des techniques de détection/reconnaissance d'objets ou des techniques de segmentation basées sur l'apprentissage profond à partir d'images RGB ou RGB-D grâce à leurs avancées. Par contre, pour permettre au robot de raisonner, par exemple de déterminer des lieux à partir d'objets [17] ou d'analyser une situation pour prendre une décision [18], la représentation de la carte doit être plus riche en informations sémantiques, par exemple la description des objets et leur utilité. Ces données sont généralement fournies par une base de connaissances commune ou bien introduites par l'intéraction homme robot.

Toutes les méthodes d'acquisition de données sémantiques mentionnées dans cet article peuvent être adaptées à l'environnement industriel. Les méthodes d'acquisition basées sur l'interaction homme-robot [13, 17, 5, 16] peuvent être transposées telles quelles, puisque les informations sont fournies par des humains. Pour les méthodes basées sur des capteurs [17, 10, 11, 8, 21], il est nécessaire de collecter des données de l'environnement industriel pour entraîner les modèles de détection/reconnaissance d'objets ou les modèles de segmentation 2D/3D. Par ailleurs, pour l'acquisition basée sur le raisonnement, il est nécessaire de créer de nouvelles structures de connaissances adaptées à l'environnement industriel. En effet, les ontologies de connaissances conceptuelles proposées [13, 17, 21] sont composées de concepts et de liaisons liées aux environnements domestiques, par exemples les catégories de pièces (cuisine, salon, etc..), les catégories d'objets (fourniture) et les liaisons (le four se trouve dans la cuisine). Ces ontologies ne permettent pas le raisonnement dans des environnements industriels. Donc, il faut créer d'autres ontologies avec de nouveaux concepts et relations. Cependant, le modèle relationnel sémantique proposé dans [18] peut être adapté directement à l'environnement industriel. Ce dernier comprend la représentation physique et conceptuelle des objets et des lieux, les utilités des objets et les relations sémantiques liant les objets et les lieux. Son avantage est que les relations entre les concepts sont incluses implicitement par la conception du modèle. Il suffit donc de modifier les informations sur l'environnement domestique par des informations sur l'environnement industriel.

Concernant la représentation sémantique des objets dynamiques, il existe peu de travaux qui s'intéressent à ce problème. Les approches [13, 17, 5, 10, 11, 18, 8] s'intéressent

uniquement à la cartographie sémantique initiale de l'environnement, et par conséquent ne permettent pas la représentation d'objets dynamiques. L'approche [16], prévoit après cette cartographie initiale, un système de mise à jour en ligne de la carte sémantique avec les nouveaux objets introduits dans l'environnement. Le processus d'acquisition de nouvelles données sémantiques est déclenché par une commande vocale pendant le fonctionnement du robot, mais l'opérateur doit l'assister tout au long de ce processus. Cette approche permet de mettre à jour la carte, mais pas en temps réel et nécessite la présence permanente d'un opérateur pour y procéder. Elle n'est pas adaptée à l'environnement industriel car l'espace de travail dans ce dernier est très dynamique et changeant. Il est partagé entre les robots et les humains, et des objets sont introduits et d'autres sont retirés à chaque instant. Donc, la représentation des objets dynamiques doit être mise à jour en temps réel pour assurer la sécurité dans les collaborations homme-robot et robot-robot. Par ailleurs, dans [21], le système de planification de tâches basé sur une carte sémantique, traite également les objets dynamiques. Ce dernier permet de déterminer les positions approximatives de ces objets à travers des liens sémantiques probabilistes qui les relient aux objets statiques, mais sans pour autant les représenter ou les mettre à jour sur la carte.

En conclusion, certaines des approches de cartographie sémantique mono-robot peuvent être adaptées pour représenter de grands environnements industriels. En effet, afin de cartographier les éléments statiques de cet environnement, il est nécessaire d'opter pour des approches basées sur des algorithmes SLAM adaptés aux environnements denses et de grande taille. En outre, les méthodes basées sur le laser sont considérées comme les plus robustes pour l'environnement industriel car elles permettent un positionnement précis dans un environnement changeant et en présence d'obstacles. Pour l'acquisition de données sémantiques, des techniques de segmentation, de reconnaissance d'objets et d'IHR peuvent être utilisées. De plus, il est nécessaire de disposer d'une base de connaissances conceptuelles adaptée à cet environnement et à la mission du robot afin de lui fournir d'autres informations sémantiques. En ce qui concerne la représentation des éléments dynamiques, il existe seulement quelques méthodes qui considèrent l'environnement dynamique et la mise à jour de la carte. Les systèmes proposés sont basés sur l'IHR et ne sont pas capables de mettre à jour les positions de ces éléments en temps réel. Cependant, cette condition est nécessaire dans les environnements industriels, où la structure change rapidement, les objets sont introduits ou retirés à tout moment, et les opérateurs sont à proximité des robots. Par conséquent, les approches présentées sont capables de créer une carte sémantique initiale de l'environnement avec une représentation des objets statiques et une représentation initiale des objets dynamiques. Mais, elles ne sont pas en mesure de mettre à jour la carte avec leurs nouvelles positions et les nouveaux éléments introduits en temps réel.

Pour répondre à ce besoin, les extensions possibles des différentes méthodes citées dans cet article sont les méthodes

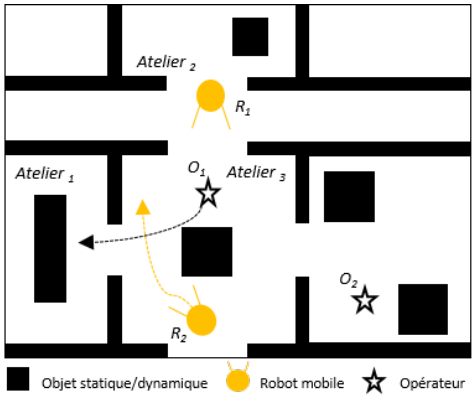


FIGURE 2 – Cas d'utilisation de la cartographie sémantique multi-robots dans un environnement industriel

de cartographie sémantique multi-robots. La Figure 2 présente un exemple de situation rencontrée dans un environnement industriel. Les trajectoires à suivre par le robot R_2 et l'opérateur O_1 illustrées par des flèches, sont concordantes. Dans ce cas, R_2 ne peut pas voir O_1 à cause de l'obstacle au milieu de l'$Atelier_3$. Il est donc nécessaire de partager des informations entre R_1 et R_2 pour rendre O_1 visible à R_2. Ainsi, R_2 peut anticiper un accident avec O_1 par un raisonnement et une perception augmentée et partagée avec R_1. Il y a peu de travaux qui se concentrent sur les problématiques de la cartographie multi-robots [23, 24, 25]. En effet, selon [24], la majorité des travaux existants s'intéressent soit à la cartographie sémantique mono-robot, soit à la cartographie géométrique collaborative [26].

4 Conclusion

Cet article propose un état de l'art des méthodes de cartographie sémantique mono-robot. Dans la littérature, les méthodes SLAM classiques sont généralement utilisées pour générer la représentation géométrique de l'environnement. D'autre part, plusieurs méthodes sont utilisées pour l'acquisition d'informations sémantiques, notamment des méthodes basées sur les capteurs, sur les IHR et sur le raisonnement. Cette étude montre que la plupart des travaux de cartographie sémantique se concentrent sur les environnements domestiques afin d'y faciliter l'intégration des robots de service. Le présent travail étudie donc la capacité de transposition de ces méthodes dans des environnements industriels de grande taille, dynamiques et structurellement changeants, où les robots et les humains partagent le même espace de travail.

En conclusion, il se révèle que les méthodes mono-robot sont efficaces pour cartographier un environnement statique, et peuvent donc être utilisées pour la cartographie sémantique initiale d'un environnement industriel. Cependant, elles ne seront pas en mesure d'établir une cartographie en temps réel de cet environnement dynamique. Une piste possible pour adresser cette problématique est la fusion de cartes sémantiques individuelles de différents robots pour créer une carte sémantique partagée, représentant

en temps réel les objets statiques et dynamiques de l'environnement.

Références

[1] F. Wang, C. Zhang, F. Tang, H. Jiang, Y. Wu, and Y. Liu, "Lightweight object-level topological semantic mapping and long-term global localization based on graph matching," *ArXiv*, vol. abs/2201.05977, p. 9, 2022.

[2] Q. Liu, R. Li, H. Hu, and D. Gu, "Extracting semantic information from visual data : A survey," *Robotics*, vol. 5, no. 1, p. 8, 2016.

[3] J. Crespo Herrero, J. C. Castillo Montoya, Ó. Martínez Mozos, and R. I. Barber Castaño, "Semantic information for robot navigation : a survey," *Applied Sciences*, vol. 10, no. 2, p. 497, 2020.

[4] A. Nüchter and J. Hertzberg, "Towards semantic maps for mobile robots," *Robotics and Autonomous Systems*, vol. 56, no. 11, pp. 915–926, 2008.

[5] C. Zhao, W. Mei, and W. Pan, "Building a grid-semantic map for the navigation of service robots through human–robot interaction," *Digital Communications and Networks*, vol. 1, no. 4, pp. 253–266, 2015.

[6] D. TIHAY and N. Perrin, "Robotique collaborative : perception et attentes des industriels," *Hygiéne et Sécurité du Travail, INRS*, no. 250, pp. 50–57, 2018.

[7] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping : Toward the robust-perception age," *IEEE Transactions on robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[8] X. Qi, W. Wang, M. Yuan, Y. Wang, M. Li, L. Xue, and Y. Sun, "Building semantic grid maps for domestic robot navigation," *International Journal of Advanced Robotic Systems*, vol. 17, no. 1, p. 172988141990006, 2020.

[9] K. Lee, S.-J. Lee, M. Kölsch, and W. K. Chung, "Enhanced maximum likelihood grid map with reprocessing incorrect sonar measurements," *Autonomous Robots*, vol. 35, no. 2, pp. 123–141, 2013.

[10] X. Li and R. Belaroussi, "Semi-dense 3d semantic mapping from monocular slam," *arXiv preprint arXiv :1611.04144*, 2016.

[11] N. Sünderhauf, T. T. Pham, Y. Latif, M. Milford, and I. Reid, "Meaningful maps with object-oriented semantic mapping," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5079–5085, IEEE, 2017.

[12] C. Astua, R. Barber, J. Crespo, and A. Jardon, "Object detection techniques applied on mobile robot semantic navigation," *Sensors*, vol. 14, no. 4, pp. 6734–6757, 2014.

[13] H. Zender, O. M. Mozos, P. Jensfelt, G.-J. Kruijff, and W. Burgard, "Conceptual spatial representations for indoor mobile robots," *Robotics and Autonomous Systems*, vol. 56, no. 6, pp. 493–502, 2008.

[14] X. Han, S. Li, X. Wang, and W. Zhou, "Semantic mapping for mobile robots in indoor scenes : A survey," *Information*, vol. 12, no. 2, p. 92, 2021.

[15] G. Randelli, T. M. Bonanni, L. Iocchi, and D. Nardi, "Knowledge acquisition through human–robot multimodal interaction," *Intelligent Service Robotics*, vol. 6, no. 1, pp. 19–31, 2013.

[16] E. Bastianelli, D. D. Bloisi, R. Capobianco, F. Cossu, G. Gemignani, L. Iocchi, and D. Nardi, "On-line semantic mapping," in *2013 16th International Conference on Advanced Robotics (ICAR)*, pp. 1–6, IEEE, 2013.

[17] A. Pronobis and P. Jensfelt, "Large-scale semantic mapping and reasoning with heterogeneous modalities," in *2012 IEEE international conference on robotics and automation*, pp. 3515–3522, IEEE, 2012.

[18] J. Crespo, R. Barber, and O. Mozos, "Relational model for robotic semantic navigation in indoor environments," *Journal of Intelligent & Robotic Systems*, vol. 86, no. 3, pp. 617–639, 2017.

[19] K. Darlington, "Common sense knowledge, crucial for the success of ai systems," *OpenMind BBVA*, 2020.

[20] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J.-A. Fernandez-Madrigal, and J. González, "Multi-hierarchical semantic maps for mobile robotics," in *2005 IEEE/RSJ international conference on intelligent robots and systems*, pp. 2278–2283, IEEE, 2005.

[21] Z. Wang and G. Tian, "Hybrid offline and online task planning for service robot using object-level semantic map and probabilistic inference," *Information Sciences*, vol. 593, pp. 78–98, 2022.

[22] S. Macenski and I. Jambrecic, "Slam toolbox : Slam for the dynamic world," *Journal of Open Source Software*, vol. 6, no. 61, p. 2783, 2021.

[23] G. S. Martins, J. F. Ferreira, D. Portugal, and M. S. Couceiro, "Modsem : modular framework for distributed semantic mapping," *Poster Papers*, p. 12, 2019.

[24] Y. Yue, C. Zhao, R. Li, C. Yang, J. Zhang, M. Wen, Y. Wang, and D. Wang, "A hierarchical framework for collaborative probabilistic semantic mapping," in *2020 IEEE international conference on robotics and automation (ICRA)*, pp. 9659–9665, IEEE, 2020.

[25] V. Tchuiev and V. Indelman, "Distributed consistent multi-robot semantic localization and mapping," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4649–4656, 2020.

[26] R. Dubois, *Méthodes de partage d'informations visuelles et inertielles pour la localisation et la cartographie simultanées décentralisées multi-robots*. PhD thesis, École centrale de Nantes, 2021.

Evaluation des cas d'usages des véhicules automatisés et connectés : Vers une approche basée sur les scénarios visant à réduire la quantité de tests en simulation ou environnement réel

Hugues Blache[1], Pierre-Antoine Laharotte[1], Nour-Eddin El Faouzi[1]

[1]Univ Lyon, Univ Gustave Eiffel, ENTPE, LICIT-Eco7, F-69675 Lyon, France

Résumé

Avec le développement des Véhicules Automatisés et Connectés (VAC), se pose la question de la validation et de l'homologation de ces nouveaux outils. La quantité de tests à réaliser pour évaluer les cas d'usages relatifs aux VACs est conséquente. Dans cet article, nous esquissons une démarche de réduction du nombre de cas d'usage visant à rassembler ceux faisant appel à des scénarios similaires. Pour ce faire, en s'appuyant sur les définitions issues de la littérature, nous identifions les cas d'usages existants, les caractérisons par des variables, puis exploitons ces variables pour rattacher les cas d'usage à un ensemble de scénarios.

Mots-clés

Cas d'Usage, Scénario, Véhicule Automatisé et Connecté, Réduction, Classification.

Abstract

With the development of Automated and Connected Vehicles (CAVs), the question of validation and approval of these new tools arises. The quantity of tests to be carried out to evaluate the use cases relating to CAVs is considerable. In this article, we outline an approach to reduce the number of use cases by gathering those involving similar scenarios. To do this, based on definitions from the literature, we identify existing use cases, characterise them by variables, and then use these variables to link the use cases to a set of scenarios.

Keywords

Use case, Scenario, Automated and Connected Vehicles, Clustering.

1 Introduction

Afin de couvrir un ensemble de conditions environnantes suffisamment vastes pour inclure les principales situations à risque, il est estimé qu'il faudrait environ 275 millions de miles [19] pour valider la fiabilité d'une voiture autonome (ou Véhicule Automatisé et Connecté - VAC). Concrètement, ceci nécessiterait la circulation en continu pendant 12,5 ans de 100 véhicules autonomes à une vitesse de 25mph (soit environ 40 km/h). **Une telle durée d'évaluation, excédant la décennie, serait-elle acceptable pour les pouvoirs publics ou les constructeurs automobiles ?** Cette latence aurait en plus des répercussions sur l'intégration de toute nouvelle fonctionnalité aux VAC. En effet, chacune des nouvelles fonctionnalités devrait justifier du même niveau d'évaluation. Ces évaluations, qui ont pour but de déterminer le nombre de kilomètres à parcourir jusqu'à la validation ou l'émergence du prochain accident, sont communément appelées "approches basées sur la distance" (*Distance-based Approach* en anglais) [27]. Néanmoins, une autre approche, plus prometteuse et parcimonieuse, émerge dans la littérature relative à l'évaluation des VACs. Cette approche est "basée sur les scénarios" (*Scenario-based Approach* en anglais). [27]. Le but de cette approche est de drastiquement réduire la quantité de tests à réaliser sur le terrain ou en simulation . Elle se base, par exemple, sur le regroupement des scénarios s'appuyant sur des paramètres similaires, puis elle cible les scénarios nécessaires à la validation des systèmes de VAC [7]. C'est cette approche que nous avons décidé de déployer pour nos recherches.

2 Positionnement

La réduction des scénarios et des tests n'est pas propre aux transports, mais est un sujet d'étude multidisciplinaire [5, 8, 22]. La réduction est notamment pleinement ancrée dans ces disciplines dans le but d'économiser le temps et le coût tout en s'appuyant sur des analyses fines d'un système. Plusieurs approches existent dans la littérature relative aux VACs :

- **Les plans d'expériences** : Cette approche a pour but de prédire les réponses d'un système depuis un minimum d'expériences réalisées. Pour cela, des modèles mathématiques sont mis en place selon le nombre de facteurs (variables) pris en compte et selon leurs niveaux (valeurs) [16]. Dans le domaine du transport, cette approche est employée par [4].
- **Les plans d'échantillonnages** : Cette approche permet de tirer, à partir d'une population, un sous-ensemble représentatif de cette population. Ces plans d'échantillonnage parcimonieux visent à définir combien d'observations sont nécessaires pour décrire un phénomène [2]. On retrouve cette pro-

blématique dans le domaine du transport [3].

— **La recherche de cas nominaux** : Cette approche vise à cibler les cas nominaux et permet, dans un premier temps, de visualiser les performances du système dans des situations "usuelles". D'après le domaine de l'ingénierie du logiciel [5], il existe plusieurs techniques pour éliminer les tests. L'une d'elles, la classification non-supervisée [18, 20] est intéressante à prendre en compte pour l'évaluation des VACs.

— **La recherche des scénarios critiques** : Cette approche vise à cibler les scénarios considérés comme critiques, *i.e.* les scénarios ayant une probabilité élevée d'aboutir à un accident [21, 27]. Il existe différentes manières et indicateurs pour identifier les cas critiques. Au sein du projet Pegasus [1], les indicateurs s'appuient sur les valeurs du time-to-collision (TTC) ou du time-to-collision-speed (TTC_VCOL).

— **La décomposition fonctionnelle du système** : Cette démarche a pour but de décomposer des fonctions ou des systèmes complexes en "sous-sytèmes" moins complexes. Pour les VAC, d'après le Graads [6], ces systèmes peuvent être décomposés en 6 couches. Cependant, distinguer problématique des transports et problématiques des télécommunications/conception du véhicule est difficile dans ce cas.

À ce jour, il est à noter l'existence de quelques travaux exploratoires visant à l'identification et/ou la réduction des scénarios [28, 34]. Cependant, d'après nos connaissances, il n'existe pas, dans la littérature, de procédé d'analyse et de réduction drastique d'une multitude de cas d'usage des VACs en vue de leur évaluation. Le but est donc d'identifier et regrouper les cas d'usages sollicitant les mêmes scénarios. La démarche innovante de notre approche propose, à la fois, de réduire les scénarios à tester, mais également de réduire en amont le cas d'usage à analyser pour proposer des plans d'expériences exhaustifs. Ces plans d'expériences tenteront de prendre un maximum de cas d'usage existant.
Avant d'aborder la démarche adoptée dans la section 4, il est important de poser des définitions pour la compréhension du document. Ces définitions sont expliquées dans la section 3.

3 Définitions élementaires

La figure 1 illustre les définitions élémentaires de nos recherches. On remarque alors que le cas d'usage englobe le tout, dont le système à évaluer (*i.e.* le véhicule automatisé, ou véhicule *ego*). Le cas d'usage constitue la donnée d'entrée alimentant la démarche de réduction. Ainsi, la notion de **cas d'usage** englobe :

1. Le **système d'étude** de par la définition de la **plage fonctionnelle**.
2. Un **comportement désiré** du cas d'usage.
3. Les **limites de fonctionnalités**.
4. Un ensemble de **scénarios**, eux-mêmes, composés de **scènes**.

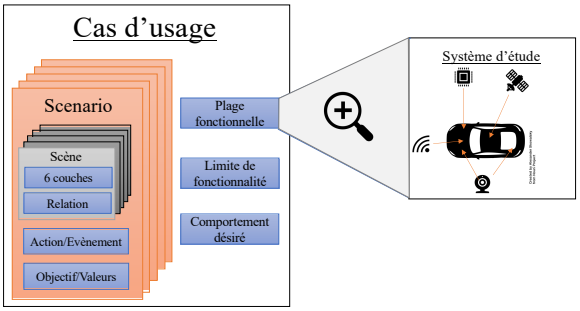


FIGURE 1 – Definition générale, figure inspirée de [13, 32]

3.1 Scène

Une **scène** décrit un *"instantané de l'environnement, y compris le paysage et les éléments dynamiques, ainsi que les représentations de soi de tous les acteurs et observateurs (compétences et aptitudes (c.-à-d. champs de vision ou occlusion)), et les relations entre ces entités. Seule, une représentation scénique dans un monde simulé peut être globale (scène objective, vérité, terrain). Dans le monde réel, la scène est incomplète, incorrecte, incertaine, et d'un ou plusieurs points de vue des observateurs (scène subjective)"* [32].

Une scène est composée de "scenary" et d'éléments dynamiques. Pour les représenter, nous adoptons la décomposition en 6 couches, sans priorisation, explicitée par [10, 26, 29] et détaillée sur la figure 2. Pour illustrer la décomposition en 6 couches des scènes, on peut faire une allégorie à l'hamburger. Schématiquement, un hamburger est composé de différentes couches : du pain, de la sauce, une salade, un steak, du fromage, de la tomate puis du pain. Il est tout à fait possible de remplacer le steak de viande par une galette végétale ou de remplacer la tomate Roma par de la cœur de bœuf, cela ne changera pas la définition même de l'hamburger. Cette notion de changement est ainsi applicable dans la notion de scène, on peut fixer les valeurs des couches et changer une valeur de chaque couche pour faire varier les scènes.

3.2 Scénario

Un **scénario** est une *"séquence de scènes ordonnées temporellement. Chaque scénario commence par une scène initiale et finit par une scène finale. Des actions et des événements, ainsi que des objectifs et des valeurs peuvent être spécifiés pour caractériser le déroulement temporel d'un scénario. Contrairement à une scène, un scénario s'étend sur un certain laps de temps."* [32]. Les éléments du scénario sont décrits sur la Tab 2.
sp Pour se raccrocher au concept de validation d'un système de VAC et être en adéquation avec le modèle en V [2],

1. https ://www.pegasusprojekt.de/en/pegasus-method

2. https ://fr.wikipedia.org/wiki/Cycle_en_V

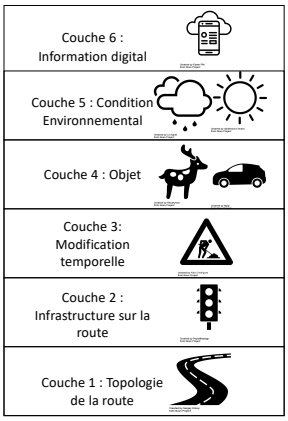


FIGURE 2 – Répartition en couches dans une scène, inspirée de [10, 29]

TABLE 1 – Définition des éléments des scènes

Nom	Définition
Véhicule *ego*	le Véhicule, lui-même, le conducteur et les options d'automatisation ou connectivité.
Objet dynamique et acteur	Objet qui est en mouvement dans le scenario. Un acteur est un objet dynamique de la scène.
Objet statique et paysage	Élément stationnaire et environnement statique

il possible de distinguer trois types de scénarios [25].

- **Scénarios fonctionels** : *Les scénarios fonctionnels incluent des scénarios d'exploitation au niveau sémantique. Les entités du domaine et les relations de ces entités sont décrites via une notation de scénario linguistique. Les scénarios sont cohérents et un vocabulaire est utilisé pour la description.*
- **Scénarios logiques** : *Les scénarios logiques incluent des scénarios de fonctionnement au niveau de l'espace d'état. Les scénarios logiques représentent les entités et les relations de ces entités à l'aide de plages de paramètres dans l'espace d'état. Les plages de paramètres peuvent éventuellement être spécifiées :*
  - *par des distributions de probabilité ;*
  - *à l'aide de corrélations, ou ;*
  - *de conditions numériques.*

  . Un scénario logique comprend une notation formelle du scénario.
- **Scénarios concrets** : *Des scénarios concrets décrivent distinctement des scénarios de fonctionnement au niveau de l'espace d'état. Les scénarios concrets représentent des entités et les relations de ces entités à l'aide de valeurs concrètes pour chaque paramètre dans l'espace d'état*

TABLE 2 – Définition des éléments des scènarios

Nom	Définition
Action & Evènement	Actions effectuées par le véhicules *ego* Evenement : Action faites par l'environnement de la voiture [12, 13]
Objectif et valeurs	Tâche données aux "conducteur ou à l'automatisme". Cette objectif est présent le long du scénario pour le conducteur ou l'automate est influe sur "les actions de navigation, guidage et de stabilisation" [15]

3.3 Cas d'usage

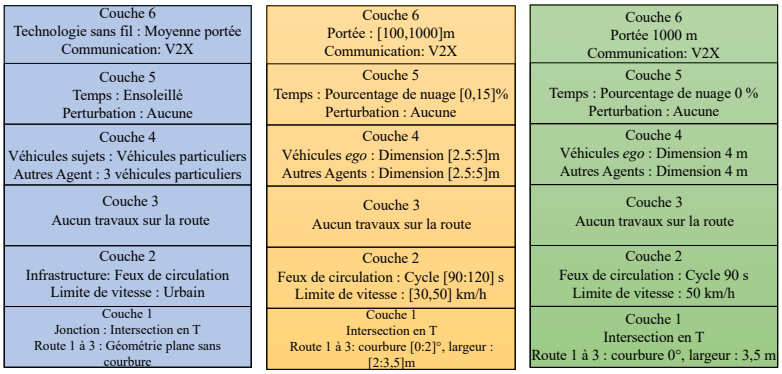
Le **cas d'usage** est une "*combinaison/description fonctionnelle du système, comprenant la limite du système, le comportement désiré et la plage fonctionnelle du système, ainsi que des scénarios satisfaits par ce système. Ainsi, un cas d'usage est une manière d'utiliser le système pour un usage de l'utilisateur."* [32].éléments des cas d'usage sont décrits sur la Tab 3

4 Démarche globale adoptée

La démarche globale vise à proposer un procédé de réduction de la quantité de cas d'usage et de scénarios à explorer en vue de valider les nouvelles fonctionnalités embarquées dans les Véhicules Automatisés et Connectés (VACs). Dans un premier temps, nous nous focalisons sur la compréhension du lien entre cas d'usage et la réalisation d'une scène de trafic, que ce soit sur le terrain ou en environnement simulé. L'objectif est de comprendre l'enchaînement des liens logiques conduisant du cas d'usage à une instance de l'expérimentation. Pour ce faire, nous proposons de décrire ce cheminement sous la forme d'un graphe (ou arbre) dépeint en Figure 4. La lecture descendante peut se faire de la manière suivante :

Pour un Cas d'usage (U_c) donné, la validation peut-être opérée sur un set de j scénarios fonctionnels (S_f) avec $j \leq i$ (i étant le nombre de U_c totaux), communément appelé relation 1 à n. Chaque scénario fonctionnel, c'est-à-dire adoptant une forme verbale, peut-être traduite par un set de paramètres se présentant sous la forme d'un scénario logique (S_l) (relation 1 à 1). Outre ces scénarios fonctionnels, on peut identifier d scénarios concrets (S_c) avec $k \leq d$ (relation 1 à n), k étant le nombre total de scénarios logiques. Un scénario concret correspond dès lors à la réalisation d'un

3. Le Green Light Optimal Speed Advice (GLOSA) est un service basé sur un système de communication Infrastructure-to-Vehicles (I2V). Il permet d'envoyer des informations relatives à l'état du feu de circulation (appelées SPATEM en Europe) vers les véhicules connectés en approche. L'objectif de cette communication est de permettre au véhicule connecté de tirer parti des informations relatives au statut des feux de circulation pour l'aider à optimiser sa vitesse d'approche de l'intersection [30].



(a) Scénarios fonctionnels (b) Scénarios logiques (c) Scénarios concrets

FIGURE 3 – Description des 3 types de scénarios

scénario logique. Cette distinction entre scénario concret et scénario logique est proche à celle adoptée en statistiques pour distinguer une variable aléatoire et sa réalisation. Finalement, l'ensemble des scénarios concrets sont constitués de p différentes scènes (dernière couche) avec $m \leq p$ (relation 1 à n), m étant le nombre total de scénarios concrets. La démarche inverse, ascendante, est possible en adoptant le même cheminement de pensées, mais en remplaçant les relations 1 à n en n à 1.
Par la suite, les deux cheminements, ascendants et descendants, seront adoptés en vue de s'assurer de l'exhaustivité des scénarios considérés.

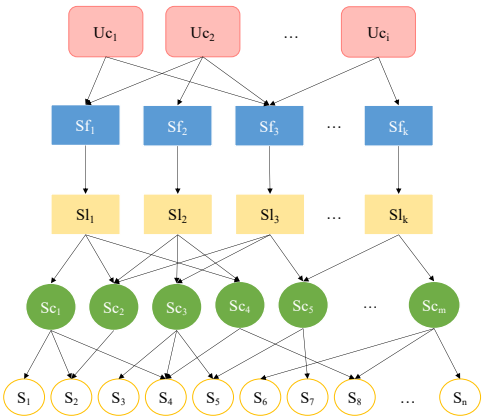


FIGURE 4 – Cheminement hiérarchique de la démarche d'identification des scénarios et des cas d'usage.

La démarche adoptée se décline alors en 4 étapes visant à peupler/renseigner cet arbre :

1. l'identification et la caractérisation des nœuds de l'arbre ;
2. la détermination des connexions entre les nœuds, c'est-à-dire l'établissement des arcs et de leur pondération ;
3. la réduction du graphe de l'arbre en vue de regrouper les cas d'usage similaires ;
4. le développement de plans d'expériences parcimonieux en scénarios à tester (*i.e.* faisant appel à un nombre réduit de scénarios à tester).

4.1 Caractérisation des nœuds du modèle - Identification des cas d'usages, des scénarios fonctionnels, logiques et concrets (étape 1)

Dans cette première étape du procédé, l'objectif est de déterminer et caractériser les nœuds du graphe illustré en Figure 4.

4.1.1 Identification des cas d'usage (U_c)

Cette étape se basera principalement sur la littérature et sur les rapports déjà existants, que ce soit du côté étatique (Scoop [1] , C-Roads [17] , Pegasus ...) que dans les recherches effectuées par des institutions (ETSI, etc.). Il est également envisagé de prendre en considération la génération de cas d'usage à l'aide de mots-clefs, en s'appuyant sur les principes des ontologies [13]. Cette identification se doit d'être la plus exhaustive possible, d'abord en rassemblant un maximum de cas d'usage (même similaires), puis, en identifiant les plages fonctionnelles des systèmes étudiés, les limites des systèmes et les comportements désirés pour lequel le système a été conçu. Une attention particulière sera portée sur les lieux géographiques pris en compte. Nous considérerons une architecture permettant d'intégrer l'ensemble des études, mais les travaux (ex : études de réduction et sélection des cas d'usage) se focaliseront sur les situations européennes, voire françaises. En complément, la couverture adéquate de l'ensemble des scénarios existants par l'approche adoptée reste un point à étudier. De même, il est pertinent de savoir si tous les U_c existants que l'on identifiera recouvrent bien les aspects de la mobilité intelligente. Finalement, au vu du nombre conséquent de U_cs existant, l'étude de la validation lors de la réduction des scénarios ne se focalisera que sur un petit nombre d'U_c.

4.1.2 Identification des scénarios fonctionnels (S_f)

Cette partie dépendra des cas d'usages identifiés et des limites de fonctionnalité prises en compte. Des vocabulaires spécifiques peuvent et doivent être utilisés pour décrire

TABLE 3 – Definition des éléments des cas d'usages

Nom	Définition
Plage de fonctionnalité du système	Correspond à la plage/gamme de fonctionnalités du système. Par exemple, le système GLOSA [3] évalué ne peut pas communiquer à plus de 1000m.
Limite d'un système	Correspond à la difficulté de réalisation d'une tâche par un système pour lequel il est conçu. Cette notion peut être ralliée à celle de Operational Design Domain (ODD). Par exemple, le système de GLOSA pris en compte ne s' applique que sur les intersections munies de feux de circulation.
Comportement désiré	Correspond à l'attente en termes de comportement que l'on a fixé pour le système. Par exemple, pour le système GLOSA, il s'agit de l'adaptation de la vitesse du conducteur selon les phases.

les différentes situations des VACs, plusieurs techniques sont présentées dans la littérature, comme l'utilisation des Langages de Descrition des Scénarios (SDL) [37], de langage d'ontologie web [4](OWL) [23] ou via des bases comme OpenScenario. Nous adopterons le OWL à l'aide du logiciel *protégé* et en prenons des termes émis par des normes ODD, comme le PAS 1883 :2020.

4.1.3 Identification des scénarios logiques (S_l)

Il existe plusieurs façons de générer les scénarios logiques :

- **Traduction des scénarios fonctionnels (sémantiques) en scénarios logiques (plages de valeurs de plusieurs indicateurs)**. Cette approche est étudiée dans [24]. En pratique, les mots clefs utilisés dans l'ontologie sont traduits par des plages de paramètres.
- **Rassemblement des scénarios concrets en cluster pour la création de scénarios logiques.** Plusieurs pistes sont à envisager pour le regroupement des scénarios concrets (S_c), mais nous tenterons au maximum de limiter les biais de perception mentale explicités dans la littérature [18, 20]. Une approche de clustering, qui a attiré notre attention, est celle du fuzzy-c-means (ou, de façon générale, les approches de clustering "flou"). Ce type de clustering est en mesure d'affecter à un objet (individu) plusieurs clusters, contrairement au k-means, qui affecte une unique classe à un objet (individu). Comme illustré par la figure 4, les scénarios concrets sont susceptibles d'appartenir à plusieurs scénarios logiques. En effet, plusieurs scénarios logiques sont susceptibles de couvrir une même plage de paramètres, c'est-à-dire que les scénarios logiques ne sont pas disjoints. Dès lors, les approches basées sur les regroupements "flou" semblent une piste pertinente à explorer.

De plus en s'inspirant de l'article [9], un des points clef dans le choix des données et de leur sélection (détaillée par la suite) est le lieu géographique. En effet, que nous soyons en Amérique du Nord, en Europe (selon les pays) ou même en Chine les lois et les règles routières seront totalement différentes. Pour peu que le sens de circulation change, un système doit être appliqué selon un lieu géographique donné pour éviter tout problème de sécurité et de validation. Un autre élément qui nécessitera une grande attention est le milieu dans lequel les données sont prélevées et observées. Par exemple, le cas des milieux autoroutiers ou urbains répond à des codes et des comportements du trafic différents. Une grande attention doit être portée à ce type de cas par la suite afin d'identifier les limites des cas d'usage et de sélectionner les scénarios qui y répondent. Un autre paramètre à intégrer consiste en l'état du trafic : par exemple, pour un trafic dense, il faut voir le nombre de scénarios qui y répondent pour au final identifier si un pour un cas d'usage donné, la réalisation est possible. Finalement, il est possible que certaines informations ne soient pas disponibles selon les couches des scènes (figure 2). Par exemple, lors des données, les informations relatives aux données météo peuvent venir à manquer. Cependant, quelques articles s'intéressent à la prise en considération des aspects relatifs aux conditions environnementales [14].

4.1.4 Identification / qualification des scénarios concrets (S_c)

Les scénarios concrets dépendront de plusieurs facteurs :

- **Données récoltées** : Comme nous avons vu dans le document précédent, il existe différents types de données disponibles. Nous en avons ciblé quelques-unes :
  - Données Geostationnaire : Ces données récoltées par drone permettraient de définir sur des lieux fixes les occurrences de plusieurs scénarios.
  - Nuscene [11] : Ces données, libres d'accès et recueillies à partir de CAVs, permettraient d'identifier des scénarios inhabituels et non rencontrés dans les bases de données précédemment citées.
- **Début et fin d'un scénario** : Une attention doit-être portée sur ce point pour définir quand commence un scénario et quand il s'achève. Nous supposons que l'attention peut se porter sur les manœuvres des véhicules pour délimiter les scénarios. Cependant, une manœuvre ne se suffit pas à elle-même, dans le sens où la dangerosité de la manœuvre dépend

4. https ://fr.wikipedia.org/wiki/Web_Ontology_Language

des conditions initiales : on peut faire la même manœuvre du point de vue du véhicule *ego* dans l'espace et le temps, pour autant, les conséquences ne seront pas les mêmes suivant la position et la densité de véhicules obstacles dans son environnement. D'autre part, il y a également la notion d'enchaînement de plusieurs manœuvres, qui peut conduire à des situations plus ou moins complexes. Dans l'absolu, le partage en manœuvres est une option envisageable car simplificatrice du problème. Il est toujours nécessaire de simplifier le problème, c'est ce qui fait les différences entre les modèles." Ainsi on peut définir le début et la fin d'un scénario comme l'accomplissement de l'objectif du véhicule *ego* Par exemple, sur la figure 5, le début du scénario serait marquée par le véhicule *ego* sur la route 1 et la fin par le véhicule sur la route 2.

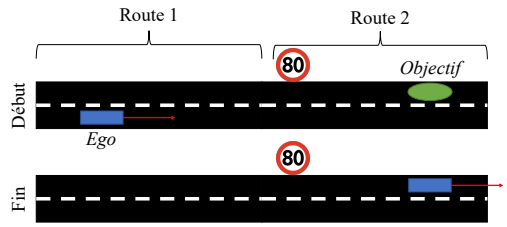


FIGURE 5 – Représentation du début et de la fin d'un scénario

4.2 Connexions entre les nœuds (étapes 2)

Les arcs du graphe (Figure 4) et leur pondération seront déterminés de deux manières distinctes :

- **Pour les arcs localisés entre les nœuds "Cas d'usage" et les nœuds "Scénarios fonctionnels"**, on s'appuiera sur les approches par le haut (Top-Down), développées dans la littérature. Il s'agit pour l'essentiel de se baser sur l'ontologie construite, puis d'établir des correspondances entre le vocabulaire employé pour caractériser le cas d'usage et le vocabulaire à l'œuvre dans le scénario fonctionnel. Pour ce faire, l'établissement d'un arc (lien/connexion) entre un cas d'usage (U_c) et un scénario fonctionnel (S_f) est opéré entre la limite des fonctionnalités du U_c et les couches du scénario fonctionnel.
- **Pour les arcs localisés entre les nœuds "Scénarios concrets" et "Scènes"**, on s'appuiera sur les approches par le bas (Bottom-Up) consistant à regrouper des scènes à partir de données observées ou simulées. Les connexions entre les scènes et les scénarios concrets (Sc) seront établies par des probabilités d'occurrence résultant des observations.

4.3 Réduction des cas d'usage (étape 3)

La démarche envisagée part du constat suivant : Si deux cas d'usages font ressortir un même set de scénarios ou des scénarios similaires, alors la démarche de réduction des scénarios à étudier devrait être similaire. Il est donc inutile de construire deux plans d'expérience pour chacun. L'effort serait à la fois double et « inutile » car le plan d'expérience développé pour le premier cas d'usage est réutilisable pour le second cas d'usage.

Un procédé de regroupement des cas d'usage s'appuyant sur les approches issues de l'apprentissage automatique et nécessitant le développement d'une métrique particulière devra être mis en oeuvre pour répondre à cette problématique. Cette étape suivra la sélection des scénarios fonctionnels des différents cas d'usage. On identifiera les similarités des différents cas d'usages selon s'ils regroupent un pourcentage de scénarios fonctionnels similaires avec des techniques d'apprentissage non-supervisé. Une piste additionnelle sera d'introduire des métriques entre les mots ou les branches des S_f pour déterminer la similarité entre chaques S_f. Pour ce faire, on déterminera des taux d'importance entre chaque critère moins discriminant que d'autres. Dans le domaine de la sémantique, le regroupement de textes [36] existe. On se penchera sur les notions que l'on peut nommer *clustering-semantic* [35] ou bien sur le *Natural Language Processing Clustering*.

En guise d'illustration, si nous reprenons les cas d'usage : GLOSA et Red Light Violation Warning [5]. Il est fort à parier que les scénarios fonctionnels seront par nature similaires, car les deux cas d'usage font référence à des situations contenant des feux de circulation en milieu urbain. Identifier les scénarios pertinents pour évaluer le cas d'usage GLOSA, permettra d'identifier les scénarios nécessaires d'explorer pour la validation du cas d'usage Red Light Violation Warning.

4.4 Réduction des scénarios : vers des plans d'expérience parcimonieux en scénarios à tester (étape 4)

Au vu d'un nombre élevé de scénario générer par les étapes précédentes, tester tous les scénarios reviendrai à de l'approche par distance, ainsi la nécessite de réduire les scénarios est une voie indispensable. Nous pourrons adopter la technique proposée par Sumaili et al [31], dans le domaine de l'éolien :

- Ils déterminent une distance entre chaque scénario. Cette distance peut être par exemple euclidienne.
- Ils déterminent une valeur seuil (valeurs de tolérance) pour déterminer si deux scénarios sont voisins
- Ils déterminent dans chaque groupe de scénarios, le scénario ayant le plus de voisins comme le scénario représentatif (scénario attractif) de ces voisins
- Ils réitèrent la démarche en ne gardant que le scénario représentatif et ils le comparent avec les autres scénarios représentatifs.

L'algorithme s'achève quand l'ensemble des distances interclasses sont inférieures au seuil de tolérance. Cette démarche nécessite une fois encore de développer une mé-

5. Cas d'usage "qui diffuse la phase et la synchronisation du signal (SPat) et d'autres données sur l'appareil embarqué, permettant des avertissements pour les violations imminentes des feux rouges" [33]

trique qualifiant la distance entre deux scénarios, puis de faire appel à des méthodes tirées de l'Intelligence Artificielle pour réduire la quantité de scénarios à explorer.

5 Conclusion

Dans cet article, nous avons défini les notions et développé les jalons d'une démarche visant à aboutir à la production de plans d'expériences parcimonieux en scénarios à explorer, mais couvrant un large spectre de cas d'usages en vue de l'évaluation de nouveaux services VACs. En particulier, le procédé développe une architecture basée sur un graphe logique assurant la génération et qualification de l'ensemble des scénarios à explorer, avant de proposer des procédés de réduction et d'élagage du graphe. La prochaine étape de nos recherches est double.

— Générer les scénarios fonctionnels à l'aide d'une ontologie.
— Identifier les cas d'usage et leurs scénarios fonctionnels.

Une limite à notre démarche est la possibilité d'engendrer de nombreux biais mentaux lors de l'approche descendante s'appuyant sur des ontologies. Pour réduire ce risque, nous tenterons au maximum d'interagir avec différents chercheurs et experts du domaine. Nous nous appuierons également sur les ressources de données disponibles pour compléter le graphe et assurer l'exploration la plus exhaustive possible des citations.

Références

[1] C-ITS French Use Cases Catalogue – Functional Description. Technical report, Ministry for an Ecological and Solidary Transition – Directorate General for Infrastructure, Transport and the Sea (DGITM), 2020.

[2] Guide d'élaboration de plans d'échantillonnage temporel et de reconstitution de données. Technical report, ADEME, 2021.

[3] Yasuhiro Akagi, Ryosuke Kato, Sou Kitajima, Jacobo Antona-Makoshi, and Nobuyuki Uchida. A risk-index based sampling method to generate scenarios for the evaluation of automated driving vehicle safety. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 667–672. IEEE, 2019.

[4] Ali P Akgüngör and Osman Yıldız. Sensitivity analysis of an accident prediction model by the fractional factorial method. *Accident Analysis & Prevention*, 39(1) :63–68, 2007.

[5] Marwah Alian, Dima Suleiman, and Adnan Shaout. Test Case Reduction Techniques - Survey. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 7(5), June 2016. Number : 5 Publisher : The Science and Information (SAI) Organization Limited.

[6] Christian Amersbach and Hermann Winner. Functional decomposition : An approach to reduce the approval effort for highly automated driving. In *8. Tagung Fahrerassistenz*, 2017.

[7] Christian Amersbach and Hermann Winner. Defining Required and Feasible Test Coverage for Scenario-Based Validation of Highly Automated Vehicles*. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 425–430, October 2019.

[8] Margaret Armstrong, Aziz Ndiaye, Rija Razanatsimba, and Alain Galli. Scenario reduction applied to geostatistical simulations. *Mathematical Geosciences*, 45(2) :165–182, 2013.

[9] Johannes Bach, Jacob Langner, Stefan Otten, Eric Sax, and Marc Holzäpfel. Test scenario selection for system-level verification and validation of geolocation-dependent automotive control systems. In *2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, pages 203–210, 2017.

[10] Gerrit Bagschik, Till Menzel, and Markus Maurer. Ontology based Scene Creation for the Development of Automated Vehicles. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1813–1820, June 2018. ISSN : 1931-0587.

[11] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes : A multimodal dataset for autonomous driving. In *CVPR*, 2020.

[12] Wei Chen. *Formal Modeling and Automatic Generation of Test Cases for the Autonomous Vehicle*. phdthesis, Université Paris-Saclay, September 2020.

[13] Wei Chen and Leïla Kloul. An Ontology-based Approach to Generate the Advanced Driver Assistance Use Cases of Highway Traffic :. In *Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, pages 75–83, Seville, Spain, 2018. SCITEPRESS - Science and Technology Publications.

[14] Ying Chen, Jiwon Kim, and Hani S. Mahmassani. Pattern recognition using clustering algorithm for scenario definition in traffic simulation-based decision support systems. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 798–803, October 2014. ISSN : 2153-0017.

[15] Sebastian Geyer, Marcel Baltzer, Benjamin Franz, Stephan Hakuli, Michaela Kauer, Martin Kienle, Sonja Meier, Thomas Weißgerber, Klaus Bengler, Ralph Bruder, Frank Flemisch, and Hermann Winner. Concept and development of a unified ontology for generating test and use-case catalogues for assisted and automated vehicle guidance. *IET Intelligent Transport Systems*, 8(3) :183–189, 2014. _eprint : https ://onlinelibrary.wiley.com/doi/pdf/10.1049/iet-its.2012.0188.

[16] Jacques Goupy. *Introduction aux plans d'expériences : avec applications*. Dunod, 2013.

[17] Damaris Anna Gruber and Wolfgang Kernstock. Milestone 30 - Definition use cases and location including a deployment plan for C-ITS elements for all pilot sites. Technical report, C-ROADS, 2020.

[18] Florian Hauer, Ilias Gerostathopoulos, Tabea Schmidt, and Alexander Pretschner. Clustering Traffic Scenarios Using Mental Models as Little as Possible. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1007–1012, October 2020. ISSN : 2642-7214.

[19] Nidhi Kalra and Susan M. Paddock. Driving to safety : How many miles of driving would it take to demonstrate autonomous vehicle reliability ? *Transportation Research Part A : Policy and Practice*, 94 :182–193, December 2016.

[20] Jonas Kerber, Sebastian Wagner, Korbinian Groh, Dominik Notz, Thomas Kühbeck, Daniel Watzenig, and Alois Knoll. Clustering of the Scenario Space for the Assessment of Automated Driving. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 578–583, October 2020. ISSN : 2642-7214.

[21] Arne Lamm and Axel Hahn. Towards Critical-Scenario Based Testing With Maritime Observation Data. In *2018 OCEANS - MTS/IEEE Kobe TechnoOceans (OTO)*, pages 1–10, May 2018.

[22] Zukui Li and Christodoulos A Floudas. Optimal scenario reduction framework based on distance of uncertainty distribution and output performance : I. single reduction via mixed integer linear optimization. *Computers & Chemical Engineering*, 70 :50–66, 2014.

[23] Till Menzel, Gerrit Bagschik, Leon Isensee, Andre Schomburg, and Markus Maurer. From Functional to Logical Scenarios : Detailing a Keyword-Based Scenario Description for Execution in a Simulation Environment. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 2383–2390, June 2019. ISSN : 2642-7214.

[24] Till Menzel, Gerrit Bagschik, Leon Isensee, Andre Schomburg, and Markus Maurer. From functional to logical scenarios : Detailing a keyword-based scenario description for execution in a simulation environment. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 2383–2390. IEEE, 2019.

[25] Till Menzel, Gerrit Bagschik, and Markus Maurer. Scenarios for development, test and validation of automated vehicles. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1821–1827. IEEE, 2018.

[26] Demin Nalic, Tomislav Mihalj, Maximilian Bäumler, Matthias Lehmann, and Stefan Bernsteiner. scenario-based testing of automated driving systems : a literature survey. page 11, 2020.

[27] Stefan Riedmaier, Thomas Ponn, Dieter Ludwig, Bernhard Schick, and Frank Diermeyer. Survey on Scenario-Based Safety Assessment of Automated Vehicles. *IEEE Access*, 8 :87456–87477, 2020. Conference Name : IEEE Access.

[28] Stefan Riedmaier, Daniel Schneider, Daniel Watzenig, Frank Diermeyer, and Bernhard Schick. Model Validation and Scenario Selection for Virtual-Based Homologation of Automated Vehicles. *Applied Sciences*, 11(1) :35, January 2021. Number : 1 Publisher : Multidisciplinary Digital Publishing Institute.

[29] Maike Scholtes, Lukas Westhofen, Lara Ruth Turner, Katrin Lotto, Michael Schuldes, Hendrik Weber, Nicolas Wagener, Christian Neurohr, Martin Herbert Bollmann, Franziska Körtke, Johannes Hiller, Michael Hoss, Julian Bock, and Lutz Eckstein. 6-Layer Model for a Structured Description and Categorization of Urban Traffic and Environment. *IEEE Access*, 9 :59131–59147, 2021. Conference Name : IEEE Access.

[30] Marcin Seredynski, Bernabe Dorronsoro, and Djamel Khadraoui. Comparison of green light optimal speed advisory approaches. In *16th international IEEE conference on intelligent transportation systems (ITSC 2013)*, pages 2187–2192. IEEE, 2013.

[31] Jean Sumaili, Hrvoje Keko, Vladimiro Miranda, Zhi Zhou, Audun Botterud, and Jianhui Wang. Finding representative wind power scenarios and their probabilities for stochastic models. In *2011 16th International Conference on Intelligent System Applications to Power Systems*, pages 1–6, September 2011.

[32] Simon Ulbrich, Till Menzel, Andreas Reschka, Fabian Schuldt, and Markus Maurer. Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 982–988, September 2015. ISSN : 2153-0017.

[33] USDT. Intelligent Transportation Systems - CV Pilot Deployment Program, 2022.

[34] Lennart Vater, Andreas Pütz, Levasseur Tellis, and Lutz Eckstein. Test Case Selection Method for the Verification of Automated Driving Systems. *ATZelectronics worldwide*, 16(11) :40–45, November 2021.

[35] Wei Wang, Romaric Besançon, Olivier Ferret, and Brigitte Grau. Regroupement sémantique de relations pour l'extraction d'information non supervisée. In *20eme Conférence sur le Traitement Automatique des Langues Naturelles*, 2013.

[36] Tingting Wei, Yonghe Lu, Huiyou Chang, Qiang Zhou, and Xianyu Bao. A semantic approach for text clustering using wordnet and lexical chains. *Expert Systems with Applications*, 42(4) :2264–2275, 2015.

[37] Xizhe Zhang, Siddartha Khastgir, and Paul Jennings. Scenario Description Language for Automated Driving Systems : A Two Level Abstraction Approach. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 973–980, October 2020. ISSN : 2577-1655.

Session "IA & apprentissage"

Safety through Intrinsically Motivated Imitation Learning

Henrique Donâncio[1], Laurent Vercouter[1]

[1] Normandie Université, INSA Rouen
LITIS

Abstract

Deep Reinforcement Learning methods require a large amount of data to achieve good performance. This scenario can be more complex, handling real-world domains with high-dimensional state space. However, historical interactions with the environment can boost the learning process. Considering this, we propose in this work an imitation learning strategy that uses previously collected data as a baseline for density-based action selection. Then, we augment the reward according to the state likelihood under some distribution of states given by the demonstrations. The idea is to avoid exhaustive exploration by restricting state-action pairs and encourage policy convergence for states that lie in regions with high density. The adopted scenario is the pump scheduling for a water distribution system where real-world data and a simulator are available. The empirical results show that our strategy can produce policies that outperform the behavioral policy and offline methods, and the proposed reward functions lead to competitive performance compared to the real-world operation.

Keywords

Learning from Demonstrations, Deep Reinforcement Learning, Pump Scheduling Optimization

1 Introduction

Over the past years, Reinforcement Learning (RL) approaches combined with function approximators have been applied in different tasks such as games [1, 2] to control [3, 4]. The appeal of this approach is the ability to support decision-making and leverage scalability for complex domains. However, exploration in large state spaces as found in the real world can be costly, inefficient, and even infeasible. For example, in scenarios such as healthcare systems and autonomous driving, trial and error methods are not an option due to safety constraints. A way to mitigate these problems is using historical interactions with the environment, an approach called Offline Reinforcement Learning[1].

In the Offline RL [5] settings, the experiences of the agent are limited to collected data, without the possibility of further exploration. The reasons for this limitation include the complexity of building accurate simulators and safety constraints for exploring the environment. Even when online data collection is reasonable, the use of prior datasets capable of generalizing to efficient policies can be attractive due to the costs to interact with the environment. Offline RL methods such as [6, 7] rely upon the idea of constraining the policy to the dataset to mitigate overestimation caused when facing out-of-distribution state-action pairs.

On the other hand, in Online RL, agents interact with the environment in an exploration-exploitation trade-off fashion. A widely applied exploration strategy is the epsilon-greedy [8], in which a given probability trade-off between exploring the environment or exploiting the policy learned. The exploration can also be encouraged by intrinsic motivation as curiosity or disagreement in the state's estimation by augmenting the reward function. For instance, some works [9, 10, 11] propose strategies based on counting occurrences of states/actions to encourage exploration of unfamiliar areas of the state-action space. In [12], the reward is augmented based on the disagreement of an ensemble of parametric Q-functions. Finally, methods to perform safe exploration are discussed in [13]. In particular, incorporating demonstrations and restricting the exploration to meaningful states can produce safe policies.

This work proposes the *Safety through Intrinsically Motivated Imitation Learning (SIMIL)* strategy that uses the distribution of historical interactions (*demonstrations*) as a guideline for action selection. The approach works as follows: given a current state, action selection depends on choosing the one that occurs most frequently in the most similar states found in the demonstrations. Later, we augment the immediate reward with an intrinsic motivation [14] according to the state likelihood under some distribution of states. The underlying idea is to constrain the policy to state-action pairs found in expert demonstrations using k-Nearest Neighbors (k-NN) to avoid exhaustive exploration. Also, we encourage states that lie in high-density regions under the demonstrations distribution using Kernel Density Estimation (KDE) [15].

We apply this imitation learning strategy in a scenario of pumping scheduling for water distribution systems (WDS). For that, it is available a dataset of three years of data collected in timesteps of one minute from a real-world operation. The pump scheduling is the process to decide when, and in some cases at which speed, the pump(s) should operate regarding the forecasting of the water demand. Yet,

[1] Some works uses the term *Batch* instead of *Offline*

some requirements must be satisfied, including safety constraints of water level in the tanks and pressure in the network's nodes. Some works have addressed these questions through several methods, including linear optimization, evolutionary and branch-and-bound algorithms, and recently Deep RL [16, 17, 18, 19]. This work uses a Deep Q-Networks (DQN) [20, 1]-based approach to handle the pump scheduling problem. The contributions presented in this work are the following:

- A formulation of the pumping scheduling problem using Partially Observable Markov Decision Process (POMDP) is presented, with definitions of system states/observations, actions, and reward function. These definitions allow the system to operate by achieving the constraints and minimizing the associated costs;
- An imitation learning strategy using real-world/offline data. The empirical results demonstrated that the obtained policies achieved competitive average cumulative rewards compared with fully-offline training.
- To evaluate the proposed scheduling, we compare the results with the real-world water distribution system regarding the electricity consumed, pumps use distribution and the tank level profile. The results showed that our approach achieved competitive performance with real-world operation.

The organization of the paper is as follows. Section 2 describes some related works. Section 3 introduces the pump scheduling problem and its formalization as a POMDP; Section 4 presents technical aspects. Section 5 describes the imitation learning strategy proposed. Section 6 describes the conducted experiments and shows the obtained results. Finally, are presented conclusions in Section 7.

2 Related Works

Offline RL methods rely on the capacity to exploit and generalize from static datasets to efficient policies. Although, leveraging the learning process using prior experiences can be challenging due to the distributional shift issue and overly optimism in the face of uncertainty [6, 5]. The Random Ensemble Mixture (REM) [21] used in this paper addresses this problem by using a convex combination of Q-values to mitigate overestimation under the assumption of a diverse and large dataset. Also, it can be adopted orthogonally to other sampling methods allowing further online data collection. Similarly, Jaques and colleagues [7] handle the overestimation issue by applying a dropout-inspired Q-learning and penalizing divergence from prior data distribution through KL-control. Batch Constrained Deep Q-Learning (BCQ) [6, 22] also constrains the policy regarding actions found in the dataset using as a baseline a generative model.

Some other works have used demonstrations as a pre-training step. In [23] is presented Deep Q-learning from Demonstrations (DQfD) that uses demonstrations as a pre-training and later improves the learned policy with self-generated experiences. The pre-training phase applies a supervised loss to ground values from unseen actions regarding the demonstrations. After the pre-training, DQfD interacts with the environment through the learned policy. In [24] the method incorporates the actor-critic algorithm DDPG, and a loss is applied to tie the policy to the offline data.

As an imitation learning strategy, this work interacts with the environment by performing a density-based action selection using the demonstrations as a distribution. Some approaches instead use this distribution to create models to perform exploration. For instance, Model-based Offline Policy Optimization (MOPO) [25] builds a model using supervised learning and then penalizes the uncertainty in further interactions based on the model's error estimation. Therefore, MOPO balances the return and risk in collecting experiences out-of-distribution of the support data. Similarly, the Model-Based Offline Reinforcement Learning (MOReL) [26] proposes to learn a policy for a pessimist MDP (P-MDP) using offline data. This P-MDP partitions the state space according to the uncertainty, applying a reward penalty to unknown areas. In [27] the authors propose a model-based approach that can mix online data collection with prior offline data. For that, a model is built and incrementally improved through Monte Carlo Tree Search rollouts.

Imitation Learning approaches [28] aims to mimic the behavior observed in demonstrations. In [29] is proposed a hierarchical method for action selection with self-improvement over time. The first step is to select a primitive that corresponds to some behavior. The second step is selecting a sub-goal achieved by performing the chosen primitive. Finally, an action generator picks a policy to execute the primitive. The underlying idea is to improve the action generator by practicing. Similar to our work and [30], this approach uses k-NN queries to retrieve a primitive from demonstrations given a current state. The difference lies in the fact that we propose a passive sampling approach, letting the evaluation of the state-action pairs to the learning method.

Our work has a basis on the wealth of literature on imitation learning and offline RL. However, the intersection between these branches remains underexplored. While offline methods rely upon methods to improve the exploitation of static datasets, the exploration often seeks to uncover areas in the state space. Yet, static datasets may not be large and diverse, and exploring unknown states can produce undesirable behaviors. Thus, the premise of our contribution is to increase the sample efficiency. We consider expert demonstrations as an underlying model for action selection and encourage policy convergence to high-density regions under the demonstration distribution through intrinsic motivation.

3 Modeling the pump scheduling problem

In water distribution systems, pump scheduling is a decision process about when operating pumps to supply water while limiting electricity consumption. Therefore some constraints must be respected, including a minimum pressure within the network, safety water level in the tanks, and avoiding frequent switches in pump operation to protect the assets. To that end, distinct strategies can be used according to the particularities of the system. For instance, pumping water in off-peak hours when the price of electricity has different tariffs throughout the day or reducing the tank level in periods of low consumption to preserve the water quality, and so on.

The water distribution system used here is located in Worms, Germany, and supplies water for about 120000 citizens[2]. The composition of this system is one station with four pumps (NP1, NP2, NP3, NP4), with distinct settings and fixed speed (ON/OFF). The flow Q through those pumps is proportional to the electricity consumption kW, being NP1 $>$ NP2 $>$ NP3 $>$ NP4. In other words, using pump NP1 supplies more water in the network than pump NP2 but also corresponds to higher electricity consumption. Also, two storage tanks with different capacities are placed and provide water for the end consumers. Among the constraints and requirements established in the operation settings for this system are the following:

- It is desirable to avoid frequent switches and distribute pump operations to protect the assets;
- It is imposed a boundary condition of the tank level and, once achieved, the minimum pressure is guaranteed;
- It is desirable to provide water exchange in the tank during one day of operation to keep the water quality.

The tank is located 47m above the pumps and has a 10m length. Thus, the tank levels considered are in the range of [47, 57]m. We consider only one tank once that the second has the level *stable* along with the operation. The specialists assume a safety operation guarantee with at least 3m filled with water. Besides this, the system does not have sensors measuring the water's quality. Thus, to *ensure* the exchange and preserve the water's quality, we assume that in one operation day, the level must decrease below half of the total capacity. Finally, the upper boundary constraint overlaps the physical limit.

As with many real-world tasks, the scenario of pumping scheduling is partially observable. In other words, the agent has a noisy or incomplete observation of the environment. For instance, most of the state's features are noisy once it has been gathering by sensors. Also, the water demand has variance along the hours, days, and seasons, even following a pattern. A POMDP is an extension of MDP that considers uncertainty regarding the current state of the environment. Formally, the POMDP can be defined as [8]:

$$POMDP = < S, A, P, R, \Omega, O, \gamma > \quad (1)$$

where the set S correspond to the **States** of the environment; A is defined as the set of **Actions** available; P is the **Transition Probability** which defines the probability being in some state $s_t \in S$, taking an action $a_t \in A$, resulting a next state $s'_{t+1} \in S$; the **Reward** $r_t \in R$ is the return to be in some state s_t and perform an action a_t; O is the set of conditional probabilities of take an action a_t in some state s_t and receive an **Observation** $o_t \in \Omega$ about the next state s'_{t+1}; Finally, γ is the **Discount Factor** $\in [0, 1]$ which determines the relevance of immediate rewards over rewards in the future.

The **States** S and the **Observations** Ω are interchangeable in the context of this work as adopted in [31] and represented by:

- The water level in the tank and water consumption;
- The previous action performed (currently being applied);
- The cumulative time that the pumps operated in a horizon length of 24 hours, the month and time t;
- A binary value called water quality indicating whether on the current day of operation the system has reached a certain minimum in the tank level.

Actions A are defined by the set of binary values that represent if some pump is operating (value 1) or not (value 0) once the pumps have fixed speed. At each timestep, only one pump is running or none of them.

Finally, two **Reward** functions are designed to choose the most *efficient* pump at a given time t, as well as respect the boundary conditions of the tank level, preserve the water quality, and make use of different pumps. The immediate rewards are defined by the Equations 2 and 3:

$$r_t = e^{1/(-Q_t/kW_t)} - B * \psi + \log(1/(P + \omega)) \quad (2)$$

$$r_t = -e^{(-1/kW_t)} - B * \psi + \log(1/(P + \omega)) \quad (3)$$

where at the time t, Q_t is the flow rate through the active pump, and kW_t is the respective electricity consumption; B is the achievement of lower/upper restrictions of the water level in the tank. These lower/upper values are defined by specialists in the system and in case of not achievement, $B = 1$ in case of overflow and $B = abs(level\_of\_the\_tank_t - boundary\_condition) \in (0, 1]$ in case of (near) shortage, being $\psi = 10$, otherwise $B = 0$. Also, B has an exception, being -1 strictly for the timestep when the tank level reaches the water quality condition. P is a penalty that increases with accumulated pump run time. The penalty P increases +1 at each timestep of cumulative operating time, and for the Equation 3 it also hold for the

[2]The dataset has been provided by the IoT.H2O project (IC4WATER JPI funding)

action *(NOP)*. In the case of switching to a pump that has already been running throughout the day, ω equals 30 for the respective timestep of the switch, otherwise 1. If no pumps are running, neither $-e^{(-1/kW_t)}$ nor $e^{1/(-Q_t/kW_t)}$ are considered.
Which differentiates the Equation 2 is efficiency regarding the pumps through Q_t/kW_t, when the Equation 3 directly penalize the electricity consumption through the term e^{-1/kW_t}. As a consequence, this leads to a different perception regarding the actions. As the agent tries to maximize these rewards along its trajectory, the result is the emergence of some behavior applying the policy learned through these distinct returns. Thus, by designing two reward functions, we aim to analyze the adequacy of those behaviors regarding the goals established.

4 Deep Reinforcement Learning

The DQN [20, 1] combines Q-Learning [32] with Deep Neural Networks. The state-input can be, for instance, a set of images or continuous values, and the output is an estimation of how good is be in that state s and perform an action a, called Q-value. During the learning process, DQN tries to approximate the optimal Q* for each state-action pair performing updates through the Bellman equation. This approach achieves higher scalability compared to other methods once that is not necessary to keep a vast search space. Later, Hausknecht and Stone [31] introduced long short-term memory (LSTM) in this structure to handle partially observable environments, and van Hasselt and colleagues adopt a Double DQN [33] to tackle the optimistic nature of the original Q-Learning.

4.1 Learning Process

Using a simulator of the environment and real-world data of the water consumption at determined time t, the simulator can calculate at timestep t the values of flow Q, pressure H, and electricity consumed kW, as well the tank level at $t+1$. The dynamic of this simulator is first to define the state s and then use some strategy to choose an action to be performed. Once this action is applied, a reward is given, and the next state is perceived, constituting a transition $T = < state, action, reward, next\ state >$.
During this process, new transitions feed the *Experience Replay*. The *Experience Replay* [34] and the *Target Network* are two techniques applied in [1], to improve the performance of DQN. The former break the correlation of data, and the latter makes the learning process more stable. Transitions stored in the *replay memory* consist of a batch. Then, this batch is split into mini-batches and shuffled to break the correlation between the data. Finally, the states of these mini-batches are inputs in the neural network, which aims to approximate Q*(s, a) through the Bellman Equation 4 [8].

$$Q^*(s,a) = E[R(s,a) + \gamma max_{a'} Q'(s',a')], \qquad (4)$$

where an expectation is defined regarding the future returns, discounting it through the factor $\gamma \in [0,1]$. The Q-Learning approach establishes a convergence for the optimality, updating the Bellman equation through Equation 5.

$$Q(s,a) = Q(s,a) + \alpha[R(s,a) + \gamma max_{a'} Q'(s',a') - Q(s,a)]. \qquad (5)$$

In order to update the Q(s, a), every state-action pair is recorded and updated iteratively in the tabular form of Q-learning [32]. This approach suffers from a problem called *curse of dimensionality* [8] as the number of possible states and actions grows. This can be even more complicated when considered continuous values, that must be discretized in some way. For that, DQN combines Q-learning with neural networks as a function approximator with weights θ to estimate the Q-values. This is accomplished by minimizing the loss δ at each time step i, as shown in Equation 6.

$$\delta_i(\theta_i) = E[R(s,a) + \gamma max_{a'} Q'(s',a',\theta_{i-1}) - Q(s,a,\theta_i)]^2, \qquad (6)$$

where the weights θ_{i-1} are those fixed in the target network that in turn, are periodically updated copying weights θ. The frequency that the target network updates can be seen as a hyperparameter, being with the replay memory properties an object of study in the performance of DQN and variants [35].
The problem of traditional Q-learning is that it tends to overestimate state-action pairs out of the distribution when exploiting a fixed dataset [6]. REM mitigates this using an ensemble of models to improve the generalization through the Equation 7.

$$\delta_i(\theta_i) = E[R(s,a) + \gamma max_{a'} \left\{ \sum_k \alpha_k Q'_k(s',a',\theta^k_{i-1}) - \left\{ \sum_k \alpha_k Q_k(s,a,\theta^k_i) \right]^2, \qquad (7)$$

where for each mini-batch, α is a set of weights randomly generated such that $\left\{ \sum_k \alpha_k = 1$. Thus, REM is a convex combination of Q-values, converging for itself [21].

4.2 Sample Efficiency

The performance of the family of DQN-based approaches is strongly correlated with sample efficiency. This section describes the strategies adopted to provide richer observation of the current state, improve training performance, and make better use of samples.

4.2.1 State stacking

In the original approach of DQN, n last previous states (frames) are concatenated [1]. Thus, the input provides a richer observation of the current state, such as the system's dynamic.

4.2.2 Training data scale

The state-input values have different ranges that differ substantially. It is applied normalization in both states and re-

ward values for the range [0, 1] using Equation 8. The feature is the value x, and max/min was defined considering historical observations.

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{8}$$

4.2.3 Prioritized Experience Replay (PER)

Schaul and colleagues present in [36] an improvement regarding the Experience Replay, prioritizing samples more "unexpected". In other words, samples that provide the highest values $|\delta_i|$ through the Equations 6 are those much to learn from [37]. Then, every transition in the mini-batch is associated with the correspondent magnitude of the loss, such as $T = < state, action, reward, next\ state, |\delta| >$. Finally, to balance the bias introduced by the prioritization of samples, PER applies Importance Sampling (IS) weights.

5 Imitation Learning

The imitation learning strategy *Safety through Intrinsically Motivated Imitation Learning (SIMIL)* present in this work assumes that offline data is available and online data collection is feasible. The underlying idea is to use the offline dataset distribution as a model to constrain the action selection and enhance the sample efficiency while encouraging the policy's convergence to states that lie in high-density regions under the same prior distribution.

The imitation learning strategy works as a follows: given a current state s_t and demonstrations $\mathcal{D}$, select the action a mostly applied in the *k-most* similar states to s_t in $\mathcal{D}$. For that, we make use of k-Nearest Neighbors (k-NN), where the parameter *k* can be chosen such that minimizes the distance $min \int_{\mathcal{D}} d(\tau, \tau^{\mathcal{D}})$, regarding trajectories $\tau^{\mathcal{D}} \in \mathcal{D}$. The objective is to keep new transitions tied to the previously collected data, mitigating overestimation facing unseen state-action pairs. Finally, a reward bonus $\rho\eta(s_t)$ is added to the immediate reward according to the *Kernel Density Estimation* (KDE) for s_t through Equation 9, being ρ the importance factor for the bonus. Thus, we encourage policy convergence to states with high density under prior dataset distribution.

$$\eta(s_t) = \frac{1}{N} \int_{i=1}^{N} K\left(\frac{s_t - s_i^{\mathcal{D}}}{h}\right). \tag{9}$$

In Equation 9, $K(s_t) \geq 0$ is the kernel that estimates the density for the current state s_t over the states $s^{\mathcal{D}}$ found in the demonstrations. The parameter h is the bandwidth that trade-off the results between balance and variance. In this work, we adopt the *k-NN* based on *Manhattan distance* once it can provide suitable metric for real-values without parameter tuning and KDE with a *gaussian kernel* from Scikit-learn [15]. The Algorithm 1 summarizes the strategy proposed.

In particular, we reduce the dimensionality of the state's representation for the meaningful features regarding the current status of the WDS and skip some of them for the *k-NN* queries. That is because the timesteps are strongly correlated, and skipping some of them reduces the computational overhead due to *k-NN* query. Thus, the state representation used to calculate the reward bonus and perform k-NN queries has the reduced form of $\phi(s_t) = < tank\ level, water\ consumption, current\ time, month >$.

Algorithme 1 : Safety through Intrinsically Motivated Imitation Learning (SIMIL)

Input : set of Q-Networks with weights θ^Q, set of Target Q'-Networks with weights $\theta^{Q'} \leftarrow \theta^Q$, replay memory $\mathcal{D}'$, demonstrations $\mathcal{D}$, frequency which update target net λ, importance factor ρ;
Output : Policy π

for $t \in \{1, 2, ...\}$ **do**
  Sample state s_t
  Select action a_t using *k-NN*(s_t) in $\mathcal{D}$
  Play (s_t, a_t), observe the reward r_t and the next state s'_t
  Calculate $\eta(s_t)$, sum it to a final reward $r'_t = r_t + \rho\eta(s_t)$
  Store transition (s_t, a_t, r'_t, s'_t) into $\mathcal{D}'$
  $s_t \leftarrow s'_t$
end
for $t \in \{1, 2, ...\}$ **do**
  Sample a mini-batch of *n* transitions from $\mathcal{D}'$
  Calculate loss $\delta(\theta^Q)$
  Perform a gradient descent step to update θ^Q
  if *t* ***mod*** $\lambda = 0$ **then**
    Update the set of weights $\theta^{Q'} \leftarrow \theta^Q$
  end
end

6 Policy Evaluation

6.1 Experimental Setup

In this work, we aim to evaluate if (1) the proposed imitation learning strategy can generate policies that outperform offline methods baselines; (2) the proposed POMDP can obtain policies that offer a competitive performance relative to that observed in the real world. To this end, we conducted the experiments using the real-world dataset divided into one year for the learning process and one year for the evaluation. Both Offline RL methods and SIMIL use the same amount of data for learning. For accurate comparisons, all samples interact with the simulator for both training and evaluation. This means that the evaluation of the offline dataset is done through interactions with the simulator. We compare the policies BCQ, REM, and SIMIL + REM using 5 models for each reward function due to the stochasticity in the learning process [38].

6.2 Results

To analyze the performance, we call the set of policies obtained using the Equations 2 and 3 by Π_1 and Π_2 respec-

Policy	*Electricity Consumption (kW)*
REM Π_1	-1.11 $\pm$ 9.78
SIMIL + REM Π_1	-4.05 $\pm$ 1.97
BCQ Π_1	-3.54 $\pm$ 2.71
REM Π_2	4.08 $\pm$ 7.93
SIMIL + REM Π_2	-3.33 $\pm$ 5.77
BCQ Π_2	-1.40 $\pm$ 3.33

Table 1: Average electricity consumption (%) $\pm$ standard deviation compared to real-world operation.

Policy	*NOP*	*NP1*	*NP2*	*NP3*	*NP4*
Real-world	30.47	8.30	43.42	8.31	9.50
REM π_1^*	11.38	4.93	0.87	82.82	0.0
SIMIL + REM π_1^*	17.05	0.17	28.54	5.29	48.95
BCQ π_1^*	22.87	17.79	8.13	51.09	0.12
REM π_2^*	32.64	25.85	0.04	41.47	0.0
SIMIL + REM π_2^*	28.08	3.12	36.04	4.89	27.87
BCQ π_2^*	37.11	37.48	0.06	25.35	0.0

Table 2: Action distribution (%)

tively. We show in Figure 1 the min, max, and average cumulative reward along with the episodes using the 5 policies obtained. The results show that SIMIL has lower variance and competitive performance relative to cumulative rewards compared to fully-offline policies. The lower peaks in performance are mainly due to not meeting the tank level safety constraints.

The three sub-goals: electricity consumption, distribution of pump usage, and tank level are the counterparts of the policy. Thus, a suitable policy performs with lower electricity consumption/higher efficiency, reduces switches, and distributes the pump operation while respecting the tank level constraints. We show in Figure 2 the performance of policies π^* with a better average cumulative reward for Offline RL and SIMIL. Tables 1 and 2 present a comparison between the policies using as baseline the real-world statistics for the evaluation data. Table 1 compares the electricity consumption for Π regarding real-world operation while Table 2 shows the action distribution for π^*. The results show that SIMIL policies achieve competitive results with real-world operations considering electricity consumption. Finally, generally, the policies presented an operation in the safety range of tank levels.

7 Conclusions

This work presents *Safety through Intrinsically Motivated Imitation Learning (SIMIL)*, an imitation learning strategy using density-based action selection and intrinsic motivation to constrain policies to expert demonstrations. Our contribution lies in the idea that SIMIL, while retrieving expert demonstrations behavior, also allows the possibility of extrapolating it in favor of states that lies in high-density regions. That could represent a means to deploy safe deep RL approaches in real-world applications. Finally, the results show that SIMIL can lead to policies that could even outperform fully-offline methods.

We present a real-world problem called pumping scheduling for water distribution utilities as an evaluation scenario. The contributions of this work extend to this domain. The proposed reward functions lead to policies that satisfy the safety constraints, protect the assets and lead to electricity savings. The authors hope that this representation of the pumping scheduling problem can help other researchers in different WDS settings.

Acknowledgments

We would like to thank Harald Roclawski (TU Kaiserslautern), Aloysio P. M. Saliba (UFMG), Benjamin Dewals (ULiège), Anika Theis (TU Kaiserslautern), Thomas Pirard (ULiège), and Thomas Krätzig (Dr. Kraetzig) for their comments and suggestions regarding this work. The authors acknowledge FAPEMIG, Federal Ministry of Education and Research of Germany, Agence Nationale de la Recherce de France and Fonds de la Recherce Scientifique Belge, for funding this research by the project IoT.H2O (ANR-18-IC4W-0003) on the IC4Water JPI call.

References

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[2] G. Lample and D. S. Chaplot, "Playing fps games with deep reinforcement learning," 2017.

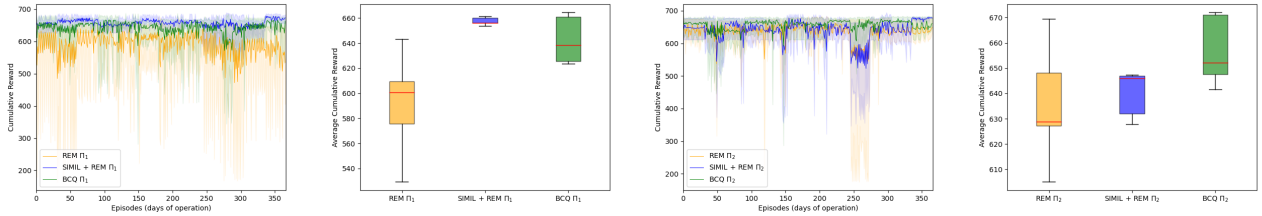
[3] Y. Yang, J. Hao, M. Sun, Z. Wang, C. Fan, and G. Strbac, "Recurrent deep multiagent q-learning for autonomous brokers in smart grid," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pp. 569–575, 7 2018.

[4] T. Wei, Y. Wang, and Q. Zhu, "Deep reinforcement learning for building hvac control," in *Proceedings of the 54th Annual Design Automation Conference 2017*, DAC '17, (New York, NY, USA), 2017.

[5] S. Levine, A. Kumar, G. Tucker, and J. Fu, "Offline reinforcement learning: Tutorial, review, and perspectives on open problems," *CoRR*, vol. abs/2005.01643, 2020.

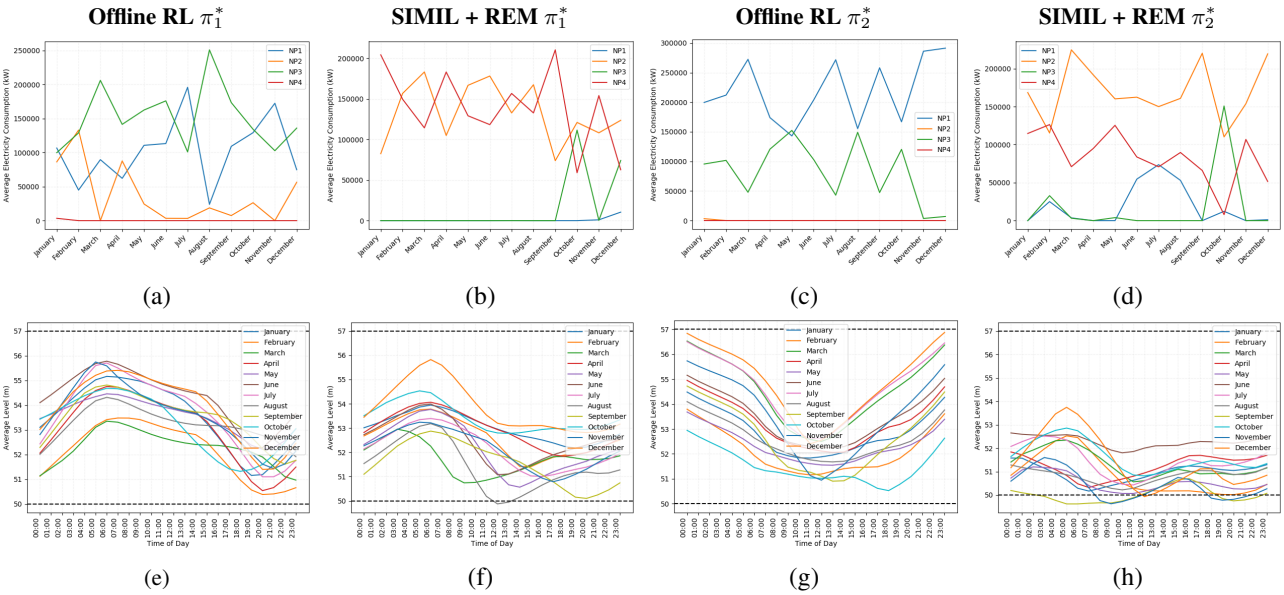
[6] S. Fujimoto, D. Meger, and D. Precup, "Off-policy deep reinforcement learning without exploration," in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97 of *Proceedings of Machine Learning Research*, pp. 2052–2062, PMLR, 09–15 Jun 2019.

[7] N. Jaques, A. Ghandeharioun, J. H. Shen, C. Ferguson, À. Lapedriza, N. Jones, S. Gu, and R. W. Picard, "Way off-policy batch deep reinforcement learning of implicit human preferences in dialog," *CoRR*, vol. abs/1907.00456, 2019.



(a) (b) (c) (d)

Figure 1: (a) and (c) are respectively the min, max, and average cumulative reward for the set of policies Π_1 and Π_2. (b) and (d) shows the average of the cumulative rewards along with the episodes for Π_1 and Π_2.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 2: Shows the performance of the policies that achieve the highest average cumulative reward for the sets Π_1 and Π_2 for Offline RL and SIMIL + REM. (a), (b), (c), and (d) present the average electricity consumption per day. (e), (f), (g), and (h) show the average tank level per time of day.

[8] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. 2018.

[9] H. Tang, R. Houthooft, D. Foote, A. Stooke, O. Xi Chen, Y. Duan, J. Schulman, F. DeTurck, and P. Abbeel, "#exploration: A study of count-based exploration for deep reinforcement learning," in *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017.

[10] D. Abel, A. Agarwal, F. Diaz, A. Krishnamurthy, and R. E. Schapire, "Exploratory gradient boosting for reinforcement learning in complex domains," *CoRR*, vol. abs/1603.04119, 2016.

[11] M. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, and R. Munos, "Unifying count-based exploration and intrinsic motivation," in *Advances in Neural Information Processing Systems*, vol. 29, Curran Associates, Inc., 2016.

[12] D. Pathak, D. Gandhi, and A. Gupta, "Self-supervised exploration via disagreement," in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97 of *Proceedings of Machine Learning Research*, pp. 5062–5071, PMLR, 09–15 Jun 2019.

[13] J. García and F. Fernández, "A comprehensive survey on safe reinforcement learning," *Journal of Machine Learning Research*, vol. 16, no. 42, pp. 1437–1480, 2015.

[14] S. Singh, A. Barto, and N. Chentanez, "Intrinsically motivated reinforcement learning," in *Advances in Neural Information Processing Systems*, vol. 17, MIT Press, 2005.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python,"

Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[16] L. H. M. Costa, B. de Athayde Prata, H. M. Ramos, and M. A. H. de Castro, "A branch-and-bound algorithm for optimal pump scheduling in water distribution networks," *Water resources management*, vol. 30, no. 3, pp. 1037–1052, 2016.

[17] S.-C. Georgescu and A.-M. Georgescu, "Pumping station scheduling for water distribution networks in epanet," *UPB Sci. Bull, Series D*, vol. 77, no. 2, pp. 235–246, 2015.

[18] F. T. Abiodun and F. S. Ismail, "Pump scheduling optimization model for water supply system using awga," in *2013 IEEE Symposium on Computers Informatics (ISCI)*, pp. 12–17, 2013.

[19] G. Hajgató, G. Paál, and B. Gyires-Tóth, "Deep reinforcement learning for real-time optimization of pumps in water distribution systems," *Journal of Water Resources Planning and Management*, vol. 146, p. 04020079, nov 2020.

[20] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller, "Playing atari with deep reinforcement learning," *CoRR*, vol. abs/1312.5602, 2013.

[21] R. Agarwal, D. Schuurmans, and M. Norouzi, "An optimistic perspective on offline reinforcement learning," in *Proceedings of the 37th International Conference on Machine Learning*, vol. 119 of *Proceedings of Machine Learning Research*, pp. 104–114, PMLR, 13–18 Jul 2020.

[22] S. Fujimoto, E. Conti, M. Ghavamzadeh, and J. Pineau, "Benchmarking batch deep reinforcement learning algorithms," *arXiv preprint arXiv:1910.01708*, 2019.

[23] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband, G. Dulac-Arnold, J. Agapiou, J. Leibo, and A. Gruslys, "Deep q-learning from demonstrations," 2018.

[24] A. Nair, B. McGrew, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Overcoming exploration in reinforcement learning with demonstrations," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6292–6299, 2018.

[25] T. Yu, G. Thomas, L. Yu, S. Ermon, J. Y. Zou, S. Levine, C. Finn, and T. Ma, "Mopo: Model-based offline policy optimization," in *Advances in Neural Information Processing Systems*, vol. 33, pp. 14129–14142, 2020.

[26] R. Kidambi, A. Rajeswaran, P. Netrapalli, and T. Joachims, "Morel: Model-based offline reinforcement learning," vol. 33, pp. 21810–21823, 2020.

[27] J. Schrittwieser, T. Hubert, A. Mandhane, M. Barekatain, I. Antonoglou, and D. Silver, "Online and offline reinforcement learning by planning with a learned model," 2021.

[28] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, "Imitation learning: A survey of learning methods," *ACM Comput. Surv.*, vol. 50, apr 2017.

[29] D. C. Bentivegna, C. G. Atkeson, and G. Cheng, "Learning tasks from observation and practice," *Robotics and Autonomous Systems*, vol. 47, no. 2, pp. 163–169, 2004. Robot Learning from Demonstration.

[30] L. Cardamone, D. Loiacono, and P. L. Lanzi, "Learning drivers for torcs through imitation using supervised methods," in *Proceedings of the 5th International Conference on Computational Intelligence and Games*, CIG'09, p. 148–155, 2009.

[31] M. Hausknecht and P. Stone, "Deep recurrent q-learning for partially observable mdps," in *AAAI Fall Symposium on Sequential Decision Making for Intelligent Agents (AAAI-SDMIA15)*, November 2015.

[32] C. J. C. H. Watkins and P. Dayan, "Q-learning," in *Machine Learning*, pp. 279–292, 1992.

[33] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," 2016.

[34] L. J. Lin, "Self-improving reactive agents based on reinforcement learning, planning and teaching," *Mach. Learn.*, vol. 8, pp. 293–321, 1992.

[35] W. Fedus, P. Ramachandran, R. Agarwal, Y. Bengio, H. Larochelle, M. Rowland, and W. Dabney, "Revisiting fundamentals of experience replay," in *Proceedings of the 37th International Conference on Machine Learning*, vol. 119 of *Proceedings of Machine Learning Research*, pp. 3061–3071, PMLR, 13–18 Jul 2020.

[36] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," in *4th International Conference on Learning Representations, ICLR 2016*, 2016.

[37] M. Hessel, J. Modayil, H. van Hasselt, T. Schaul, G. Ostrovski, W. Dabney, D. Horgan, B. Piot, M. Azar, and D. Silver, "Rainbow: Combining improvements in deep reinforcement learning," 2018.

[38] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, "Deep reinforcement learning that matters," 2018.

Apprentissage continu et étiquetage automatique de données pour améliorer un réseau de neurones incertain

Q. Christoffel[1], A. Ayadi[1], A. Deruyver[1], A. Jeannin-Girardon[1],

[1] Université de Strasbourg, Laboratoire ICube UMR 7357

{q.christoffel, ali.ayadi, aline.deruyver, anne.jeannin}@unistra.fr

Résumé

Les réseaux de neurones artificiels s'inspirent du fonctionnement du cerveau humain, mais sont encore très loin d'imiter le comportement humain. Cet article propose nos premières réflexions pour développer un modèle capable d'agir en toute autonomie dans le cadre de la classification d'images. L'approche proposée permet au réseau d'exprimer son incertitude, et de l'utiliser pour détecter les nouveautés qui lui sont présentées. En utilisant des graphes de connaissances, l'approche permettrait d'étiqueter automatiquement les nouveautés. Ces données étiquetées seront utilisées dans le cadre d'un apprentissage continu du modèle, afin de l'améliorer et de réduire son incertitude.

Mots-clés

Apprentissage continu, Incertitude, Détection de nouveautés, Vectorisation de connaissances, Graphes de Connaissance

Abstract

Artificial neural networks are inspired by the functionning of the human brain, but they are still far from imitating human behaviour. This paper proposes our first thoughts to develop a model capable of acting autonomously in the context of image classification. The proposed approach allows the network to express its uncertainty, and to use it to detect the novelties presented to it. By using Knowledge Graphs, the approach would allow novelties to be labelled automatically. This labelled data will be used in a continual learning setting of the model to improve it and reduce its uncertainty.

Keywords

Continual Learning, Uncertainty, Novelty Detection, Knowledge Embedding, Knowledge Graph

1 Introduction

L'être humain est capable d'apprendre pendant toute la durée de son existence. Dès le plus jeune âge, le cerveau humain fait appel à une grande partie de ses fonctions (vision, attention, mémoire, *etc.*) pour que nous puissions acquérir de nouveaux savoirs et savoir-faire. De manière simplifiée, l'apprentissage humain peut être vu comme un processus cognitif dynamique composé de deux étapes : l'acquisition de nouvelles connaissances à partir d'une succession d'expériences et leur stockage en mémoire sans les oublier [14]. Dans l'optique de simuler le comportement de l'apprentissage humain, des chercheurs ont présenté les réseaux de neurones artificiels [35], inspirés du fonctionnement des neurones humains. Les réseaux de neurones sont très efficaces pour apprendre une représentation des données et faire une prédiction à partir des données [2], mais ils n'ont pas l'autonomie d'un humain [20]. Dans ce domaine, différentes approches ont été proposées pour rendre l'apprentissage des réseaux de neurones plus proche de l'apprentissage humain :

1. Contrairement aux humains, un réseau de neurones est généralement entraîné à faire une classification parmi un nombre fixé de classes. Pour permettre à un modèle d'être capable d'apprendre à prédire de nouvelles classes, l'*apprentissage continu* [30] a été proposé. Cela permet à un modèle d'apprendre à résoudre de nouvelles tâches de manière séquentielle.
2. De plus, une personne est capable de se rendre compte qu'elle se trouve face à une nouveauté, qu'il faut donc analyser, apprendre et mémoriser une chose en plus. Pour qu'un modèle soit capable de détecter qu'il se trouve dans une situation inhabituelle, il est possible d'associer une mesure d'*incertitude* à ses prédictions [15, 29]. Pour détecter spécifiquement qu'une donnée est nouvelle, il est possible d'utiliser des méthodes de détection de nouveautés [4].
3. Dans le cas où une personne ignore quelque chose, elle peut demander de l'aide à quelqu'un de plus expérimenté pour apprendre. L'*apprentissage actif* [37] permet à un modèle de « demander » des informations concernant l'étiquette des données à un utilisateur, ou en utilisant d'autres sources d'information.

Dans ce contexte de grand manque d'autonomie de la part des réseaux de neurones, nous soulevons deux problématiques. La première vise à détecter lorsqu'un modèle « ne sait pas résoudre » une tâche afin d'améliorer ce modèle en lui apprenant davantage, tout en réduisant son incertitude. La deuxième problématique, porte sur la capacité à *étiqueter* automatiquement des données grâce aux connaissances structurées contenues dans des graphes de connaissances, en passant par une représentation commune entre les don-

nées et les connaissances.
L'approche que nous proposons dans cet article de positionnement consiste à, dans un premier temps, exploiter l'incertitude exprimée par un modèle pour détecter des données sur lesquelles le modèle « ne sait pas » faire une prédiction. Notre approche se base sur l'hypothèse que, si le modèle n'est pas totalement certain ni incertain face à des données, c'est qu'il a réussi à reconnaître des attributs dans ces données, mais qu'il n'a pas été capable de les exploiter pour faire une prédiction, donc nous considérons que le modèle est face à des nouveautés. Dans un second temps, l'incorporation de connaissances externes, provenant de graphes de connaissances, permettra d'étiqueter ces nouveautés. L'idée est de trouver une représentation commune entre les données et les connaissances afin de les comparer et trouver l'étiquette correcte. Ces nouvelles données annotées seraient ensuite utilisées dans le cadre d'un apprentissage continu pour permettre au modèle à la fois de réduire son incertitude, et aussi d'apprendre de nouvelles tâches.

2 État de l'art

2.1 Incertitude et apprentissage automatique

Pour donner à un modèle d'apprentissage automatique plus d'autonomie, en permettant au modèle de dire qu'il « ne sait pas résoudre » une tâche, une approche consiste à introduire la notion d'incertitude. Il faut d'abord distinguer deux concepts : l'*exactitude* et la *certitude* d'une prédiction. Si la prédiction faite par le modèle est celle qui est attendue, elle est exacte. Cependant, cette prédiction n'est généralement pas accompagnée d'une mesure de confiance : il se peut que le modèle ait fait une prédiction exacte, mais sans en être certain, d'où la nécessité de pouvoir mesurer l'incertitude d'une prédiction. L'incertitude peut ensuite être utilisée dans le cas où elle est trop élevée, pour donner au modèle la possibilité de dire qu'il « ne sait pas résoudre » le problème qui lui a été donné. Cette mesure d'incertitude peut donc être très utile dans des domaines où les prédictions peuvent avoir des conséquences sur la vie des personnes, comme en médecine ou dans le cas de la conduite autonome.
Il existe deux types d'incertitude [15]. L'incertitude *aléatoire* est inhérente aux données, elle peut être due par exemple à un capteur qui introduit une erreur de mesure. L'incertitude *épistémique* est causée par un manque d'informations ou de connaissances d'un domaine. Contrairement à l'incertitude aléatoire, l'incertitude épistémique peut généralement être réduite avec plus de données, si celles-ci couvrent mieux le domaine étudié.
Au vu des définitions ci-dessus, un objectif à atteindre est d'avoir une incertitude élevée lorsque des données hors distribution ou bruitées sont présentées au modèle, et une incertitude plus faible si le modèle a bien appris à traiter des données d'un domaine similaire.
Généralement, les modèles font une prédiction en un point, ce qui permet de prédire une valeur proche de la moyenne des vraies valeurs possibles, mais cela ne permet pas d'avoir une information sur l'incertitude du modèle. Au lieu de prédire uniquement un point, une solution consiste alors à obtenir une distribution de prédictions, qui exprime directement l'incertitude d'un modèle. On peut ensuite extraire de cette distribution différentes mesures statistiques (moyenne, médiane, *etc*). En particulier, la variance permet de quantifier l'incertitude du modèle, car elle mesure la dispersion des prédictions : plus la variance est faible, plus le modèle est certain, et plus elle est élevée, plus le modèle est incertain.
Il y a plusieurs méthodes pour obtenir une distribution de prédiction. Certaines se basent sur les statistiques Bayésiennes qui permettent, grâce au théorème de Bayes, de mettre à jour des probabilités à partir de connaissances antérieures et de données. Blundell *et al*. [7] ont présenté un modèle où chaque poids et chaque biais est représenté par une distribution, ce qui autorise une incertitude au niveau des paramètres. En théorie, l'apprentissage de ces distributions est possible grâce à l'inférence Bayésienne, mais en pratique, l'inférence Bayésienne exacte est insoluble à cause du nombre élevé de paramètres présent dans un réseau de neurones [7]. Les auteurs présentent leur algorithme *Bayes by Backprop* qui permet d'approximer l'inférence Bayésienne. Un avantage de leur approche est que, bien que le modèle requiert deux fois plus de paramètres, le résultat permet d'avoir un ensemble infini de modèles à travers l'échantillonnage de chaque distribution. Puisque les poids ne sont pas fixes, le modèle ne fait pas toujours la même prédiction sur une même entrée et ceci permet d'estimer à quel point il est certain ou non de sa prédiction. Il faut pour cela répéter plusieurs fois la prédiction sur les mêmes données et analyser la distribution des résultats prédits. Ce type de réseau avec des poids représentés par des distributions est appelé un *réseau de neurones Bayésien*.
Pour éviter d'entraîner un réseau de neurones Bayésien, qui a plus de paramètres, et s'entraîne plus lentement, et converge moins vite [10], Gal *et al*. [10] ont présenté une méthode permettant d'approximer un comportement Bayésien à partir d'un réseau de neurones. Leur approche consiste à ajouter du *dropout* [40] entre chaque couche du modèle. Le *dropout* est une méthode de régularisation utilisée pendant l'entraînement d'un modèle qui permet de désactiver des neurones dans les couches du réseau. La méthode proposée dans [10] consiste à continuer à utiliser le dropout après l'entraînement. Par conséquent, à chaque prédiction, des neurones différents sont utilisés et la sortie du modèle est non-déterministe comme dans le cas du réseau de neurones Bayésien.
Une autre méthode consiste à entraîner un ensemble de modèles profonds, où chaque modèle est entraîné sur les mêmes données [18]. Comme les modèles peuvent être entraînés en parallèle, cela ne demande pas plus de temps d'entraînement. L'utilisation de l'ensemble des modèles pour faire une prédiction permet d'avoir une distribution de résultats.
L'incertitude exprimée par un modèle à propos d'une prédiction nous permet de savoir s'il était certain de la classe prédite, ou s'il a prédit la classe par hasard. Dans ce dernier cas, nous voulons réduire l'incertitude du modèle en lui ap-

prenant à mieux connaître cette classe, voire à en apprendre une nouvelle. Nous cherchons donc à améliorer notre modèle, comme l'évoque notre première problématique. Pour améliorer le modèle, nous envisageons d'utiliser l'apprentissage continu, dont une définition et un état de l'art sont présentés dans la section suivante.

2.2 Apprentissage continu

L'apprentissage continu a pour but d'entraîner un modèle de manière à ce qu'il puisse résoudre des tâches qui lui seront présentées de manière séquentielle. Pour citer un exemple basique d'un tel problème, le jeu de données *MNIST* [19] peut être découpé en différentes tâches à apprendre : d'abord les classes $\{0, 1\}$ puis $\{2, 3\}$, $\{4, 5\}$, *etc*. Le modèle se voit donc présenté uniquement les données de la première tâche à apprendre, puis les données de la deuxième tâche et ainsi de suite pour toutes les tâches. L'objectif est d'obtenir un modèle qui arrive à apprendre les nouvelles tâches et qui parvient toujours à résoudre les tâches précédentes. La difficulté de cette approche se situe dans l'ajout de connaissances au modèle pour qu'il résolve les nouvelles tâches sans oublier ce qui a déjà été appris précédemment. Le problème étant que si on ne veille pas à conserver ce que le modèle a appris, il peut se produire de l'*oubli catastrophique* [16], qui est une perte soudaine de performance sur les tâches déjà apprises lors de l'apprentissage de la tâche courante. On parle ici de compromis *plasticité/stabilité* [22], définissant la capacité de s'adapter à de nouvelles tâches (plasticité) tout en conservant les performances atteintes sur d'autres tâches (stabilité).

Plusieurs objectifs peuvent être atteints avec l'apprentissage continu [31] : le principal objectif est qu'on ne veut pas que le modèle oublie ce qu'il a appris. Il est, de plus, souhaitable que l'utilisation de la mémoire et des ressources de calcul par le modèle soit fixe ou augmente légèrement lorsque de nouvelles tâches sont apprises. Il faut aussi faire un choix concernant deux points importants : l'autorisation (ou non) d'avoir du *Forward Transfer* ou du *Backward Transfer* dans le modèle. Imaginons que nous avons un modèle déjà entraîné sur les tâches 0 à $t-1$ et qu'on veut apprendre la tâche t. Le Forward Transfer est le fait que l'apprentissage des tâches précédentes ait une influence sur l'apprentissage de la tâche t. Cela peut avoir un effet positif dans le cas où le modèle présente de meilleures performances sur la tâche t, et un effet négatif si le modèle n'arrive pas à bien apprendre la tâche t. Le Backward Transfer représente la situation inverse : on considère ici l'influence de l'apprentissage de la tâche t sur les performances du modèle sur les tâches précédentes. L'effet est positif si les performances sont améliorées. Un dernier objectif consiste à ne pas conserver les données des tâches précédentes, ou du moins aussi peu que possible. Dans le cas d'une application réelle de l'apprentissage continu qui traiterait un flux de données contenant de nouvelles tâches à apprendre, il pourrait être impossible de conserver toutes les données pour des raisons d'espace de stockage ou des raisons légales [1].

Pour atteindre certains de ces objectifs, plusieurs méthodes ont été proposées. Par exemple, la méthode de *Model Growing* proposée par [36] consiste dans un premier temps à définir une structure de base pour le réseau qui sera capable de traiter une tâche. Puis, à chaque nouvelle tâche, la capacité du modèle est augmentée en ajoutant cette structure de base au réseau déjà existant. Cette méthode permet d'utiliser du Forward Transfer en liant les neurones de l'ancienne structure à la nouvelle, mais pas de Backward Transfer, car les neurones de la nouvelle structure ne sont pas connectés à l'ancienne. D'autres méthodes existent comme l'*isolation de paramètres* [25], la *régularisation* [16], et la *distillation de connaissances* [21]. Il existe aussi des méthodes basées sur la *répétition* des données, qui consistent à stocker ou générer un certain nombre de données pour chaque tâche [23, 38]. Ces méthodes permettent de limiter l'oubli des tâches précédentes en les rappelant au modèle.

L'apprentissage continu est la méthode qui nous permettra d'améliorer notre modèle. Il faut pour cela avoir de nouvelles données avec leur étiquette pour les fournir à notre modèle. Notre deuxième problématique porte donc sur le fait d'étiqueter automatiquement des données grâce à des connaissances contenues dans des graphes de connaissances, en utilisant une représentation commune entre les données et les connaissances. La section suivante présente les graphes de connaissances et différentes méthodes pour changer leur représentation.

2.3 Vectorisation des connaissances

Bien que les graphes de connaissances ne soient pas récents, ils ont gagné en popularité et sont désormais un élément clé dans de nombreuses applications d'intelligence artificielle liées à la recherche rapide et contextuelle d'information ainsi qu'à la prise de décision, citons à titre d'exemples DBpedia [33], Google Knowledge Graph [27], Wikidata [41], etc. Comme les ontologies, ces graphes fournissent une représentation de la connaissance relative à un domaine particulier sous une forme facilement exploitable par la machine. Ils représentent un ensemble d'entités reliées entre elles, composés de nœuds qui décrivent les entités (objets et concepts), et d'arcs modélisant les relations entre ces entités. Les relations peuvent être enrichies par des attributs ou des valeurs quantitatives représentant le poids de la relation. Les graphes de connaissances sont relativement faciles à développer et à interpréter, ce qui en fait un important outil pour décrire la sémantique de grands volumes de données issus de multiples sources, hétérogènes ou incomplètes [9].

Une représentation commune permettrait d'établir un lien entre des données et les connaissances stockées dans les graphes de connaissances. On peut par exemple utiliser une représentation sous forme vectorielle. Pour cela, il existe différentes méthodes qui se basent sur un même principe : il faut dans un premier temps transformer le graphe de connaissance en une séquence de mots qui pourra ensuite être vectorisée avec des méthodes comme Word2Vec [12]. Pour générer ces séquences, la technique *graph walks* consiste à parcourir un graphe en partant de chaque sommet et en parcourant tous les chemins qui y sont liés avec une profondeur définie. Chaque sommet et chaque relation

parcourus permettent de former une phrase. Par exemple, avec les sommets « chat » et « félin » reliés par la propriété « est un type de » allant de « chat » vers « félin » la phrase générée serait « un chat est un type de félin ». D'autres algorithmes de parcours de graphe comme la méthode de *Weisfeler Lehman Subtree RDF Graph Kernel* [8] permettent de se focaliser sur des sous-graphes particuliers. Ces deux méthodes de transformation de graphe en séquences de mots ont été utilisées par Ristoki et Paulheim pour créer l'outil RDF2Vec [34] permettant de transformer des entités et des relations en une représentation vectorielle. Un autre moyen d'obtenir des séquences de mots à partir d'un graphe est d'utiliser un raisonneur pour inférer de nouveaux axiomes logiques. Cette méthode a été utilisée par Smaili *et al*. dans leur approche Onto2Vec [39].

Après avoir transformé ces structures de connaissances en représentations vectorielles, se pose la question de l'évaluation de leur qualité. Initialement, l'évaluation était extrinsèque, utilisant la vectorisation générée comme attributs d'entrée d'un modèle. Ainsi, la qualité des représentations vectorielles est estimée en fonction de la qualité des résultats du modèle. Si les résultats ne sont pas satisfaisants, la représentation ne l'est pas non plus. Alshargi *et al*. [5] ont présenté des métriques pour évaluer la qualité de ces vectorisations intrinsèquement, sans devoir les utiliser en entrée d'un modèle. Une première métrique consiste à mesurer la *catégorisation* des entités par rapport à leur concept, en calculant la moyenne des représentations de toutes les entités typées par un concept, puis en mesurant la distance entre cette moyenne et la représentation du concept. Une distance faible reflète une représentation de bonne qualité. Une seconde métrique, qui a cette fois pour but d'évaluer la conservation du comportement hiérarchique, est l'*erreur sémantique absolue*. Le calcul de cette métrique nécessite d'avoir une mesure de similarité entre *deux concepts*. On calcule ensuite la similarité entre *les représentations de ces concepts*. Avec une représentation correcte, on peut s'attendre à ce qu'il y ait une corrélation entre ces deux mesures. Alshargi *et al*. [5] ont notamment évalué l'approche RDF2Vec [34] et conclu que les méthodes de vectorisation actuelles des graphes de connaissances ne permettent pas d'obtenir une vectorisation d'aussi bonne qualité qu'une vectorisation effectuée sur un corpus de texte, permettant, elle, d'obtenir un contexte plus riche et donc une meilleure vectorisation. De plus, les auteurs montrent qu'il n'y a pas de méthode de vectorisation meilleure que toutes les autres : chaque méthode capture des éléments spécifiques des concepts et il faut de ce fait choisir la méthode appropriée en fonction des besoins de la tâche extrinsèque qui utilisera les représentations.

Ce tour d'horizon de différents domaines et méthodes donne un aperçu des notions sur lesquelles notre approche s'appuie.

3 Approche proposée

Dans cette section, nous présentons nos idées et réflexions concernant une approche répondant aux deux problématiques présentées en introduction. La première portant sur la détection des cas où un modèle ne sait pas résoudre une tâche et la deuxième sur l'étiquetage de données en utilisant des connaissances issues de graphes de connaissances.

Lors de la conférence *Computer Vision and Pattern Recognition* (CVPR) 2021, Aljundi [4] a présenté une vision d'ensemble de la mise en place d'un agent autonome. Cela consiste dans un premier temps à entraîner un modèle, puis le déployer dans un environnement où il sera face à des situations changeantes qui nécessitent une adaptation du modèle. Pour que le modèle puisse s'adapter, il faut détecter quand il est confronté à des nouveautés. Il y a ensuite une sélection des nouveautés, pour garder et annoter celles qui permettront au modèle d'être amélioré grâce à l'apprentissage continu.

Nous nous positionnons dans un cadre similaire, mais alors qu'Aljundi présente une approche générique, nous discutons d'une approche plus bas niveau, en faisant des choix d'implémentation guidés par nos problématiques et une application dans un domaine précis : la classification d'images. Notre approche s'organise en 4 modules interconnectés comme l'illustre la figure 1. Ces modules sont détaillés dans les sections suivantes, mais n'ont pas encore été implémentés.

3.1 Modèle incertain

Notre module de base est un modèle qui est capable d'exprimer son incertitude. Cette particularité nous permet d'accorder plus de confiance aux prédictions de notre modèle et de voir ses faiblesses. Nous prévoyons aussi d'utiliser l'incertitude pour détecter les nouveautés (voir Section 3.2).

Ovadia *et al*. [29] ont évalué le comportement de différentes méthodes de prédiction d'incertitude pour vérifier si on pouvait effectivement *faire confiance à l'incertitude du modèle*. La méthode ayant obtenu les meilleurs résultats sur la plupart des métriques est celle utilisant les ensembles de modèles profonds [18]. De plus, ils ont montré que de bons résultats pouvaient être obtenus avec un petit ensemble de cinq réseaux. Leurs expériences ont aussi montré que les réseaux de neurones Bayésiens (SVI dans l'article, pour *Stochastic Variational Inference*) sont prometteurs sur des petits jeux de données comme MNIST, mais plus difficiles à utiliser avec des jeux de données comme ImageNet et des architectures complexes comme les LSTM. Dans notre cas, nous choisissons d'utiliser un réseau de neurones Bayésien, pour développer leur utilisation, car ils permettent en théorie d'entraîner un ensemble infini de modèles.

Dans les réseaux de neurones Bayésiens, les poids et les biais du modèle sont représentés par des distributions. Cette particularité permet de faire des prédictions non déterministes. Ainsi, en faisant plusieurs prédictions sur une même donnée, le résultat correspond à une distribution des valeurs en sortie pour chaque classe. Si le modèle est certain de sa prédiction, la distribution de la classe prédite a une faible variance, car le modèle prédit toujours qu'il s'agit de cette classe. Dans le cas où le modèle est incertain, toutes les distributions des classes ont une variance élevée. Pour déterminer quelle classe a été prédite par le réseau, la médiane

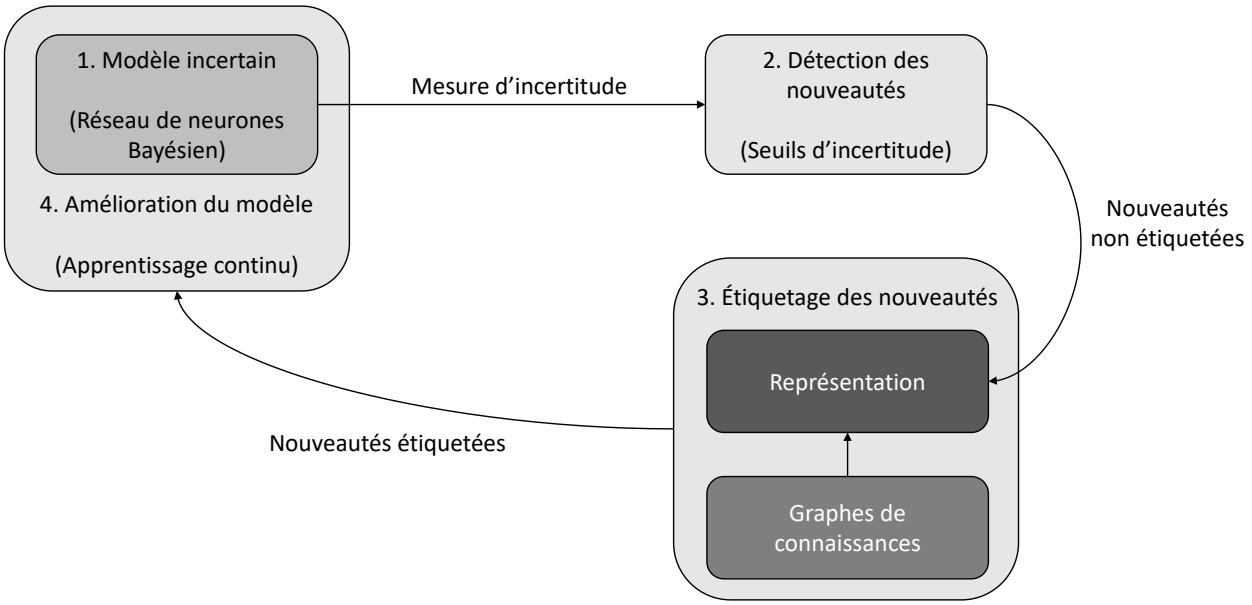


FIGURE 1 – Aperçu de l'approche proposée découpée en différents modules interconnectés.

de la distribution de chaque classe est calculée. Nous privilégions l'utilisation de la médiane puisqu'elle considère les observations et est robuste face à des valeurs extrêmes. La classe avec la médiane la plus élevée représente la prédiction finale du modèle. La médiane nous donne aussi une information sur l'incertitude, car si le modèle ne fait pas la même prédiction assez souvent, cela se reflétera sur la médiane qui sera plus basse.

3.2 Détection de nouveautés

Après l'obtention d'un modèle capable d'exprimer son incertitude, nous souhaitons détecter si des données d'une classe inconnue du modèle sont passées en entrée de celui-ci. Pour cela, nous prévoyons initialement d'utiliser l'incertitude pour détecter les nouveautés, ce qui a déjà été expérimenté dans de nombreux articles dont [32]. Nous utilisons l'incertitude exprimée par le modèle pour permettre au modèle de dire qu'il « ne sait pas » à quelle classe correspond l'image qu'il a eu en entrée si l'incertitude est trop élevée.
Nous proposons de fixer deux seuils auxquels comparer la médiane de la classe prédite par le modèle. Un premier seuil élevé, qui permet de déterminer les cas où le modèle reconnaît effectivement les données. Ainsi, si la médiane est supérieure à ce seuil, le modèle a su classifier les données. Dans le cas contraire, on considère que le modèle répond qu'il « ne sait pas » de quelle classe il s'agit. Un deuxième seuil, plus faible, permet de déterminer si le modèle est complètement incertain face aux données. Si la médiane est inférieure à ce seuil, c'est que le modèle n'a pas su faire de prédiction sur les données et on les considère comme étant hors distribution. Nous rappelons que notre hypothèse principale est que si le modèle n'est pas totalement incertain et n'est pas certain non plus, il a pu reconnaître des attributs dans les données, mais n'a pas réussi à les exploiter pour prédire une classe. Nous portons par conséquent un intérêt particulier aux données qui ont été prédites avec une médiane entre ces deux seuils et nous considérons que le modèle devrait apprendre à faire des prédictions sur ces données par l'apprentissage d'une nouvelle classe. Les données détectées comme étant hors distribution ne seront pas utilisées, bien qu'elles pourraient contenir de nouvelles classes qui n'avaient tout simplement pas de rapport avec les classes que le modèle connaissait déjà. Ces données rejetées pourraient être classées de manière non-supervisée avant d'être présentées à un expert chargé de déterminer leur importance. Ces seuils seront dans un premier temps recherchés de manière empirique avec des jeux de données appropriés aux différentes catégories que l'on veut séparer : des données connues du modèle, des données qu'il devra apprendre et des données hors distribution. Il est possible que ces seuils soient dépendants du nombre de classes, par exemple, lors d'une classification entre 1000 classes, une médiane autour de 0.5 aura plus d'importance que lors d'une classification avec 10 classes. Par la suite, nous réfléchirons à une alternative utilisant la variance de la distribution de la classe prédite, en plus de la médiane.
Dans le cas où l'utilisation de l'incertitude ne serait pas suffisante pour détecter les nouveautés, il serait possible d'utiliser d'autres méthodes. Dans un contexte similaire à ce que l'on propose, Aljundi *et al.* [3] ont étudié différentes méthodes de détection de nouveautés appliquées à un modèle entraîné grâce à l'apprentissage continu.

3.3 Étiquetage des nouveautés

Une fois que le modèle a détecté des données inconnues (mais pas complètement hors distribution) grâce à l'incertitude, l'objectif est de réussir à identifier ces données grâce à des connaissances contenues dans des graphes de connais-

sances. Par souci de clarté, dans la suite le mot « données » est utilisé pour parler des données inconnues du modèle et le mot « connaissances » est utilisé pour parler des graphes de connaissances. C'est ici que se trouve notre principale problématique : pour identifier ces données, nous devons d'abord déterminer une représentation commune entre les données et les connaissances. Plusieurs possibilités existent : dans un premier temps, il est possible de transformer à la fois les données et les connaissances sous une représentation vectorielle afin de les comparer. Pour cela, nous proposons de générer des représentations vectorielles des connaissances à l'aide des approches de transformation de graphes comme RDF2Vec [34] ou Onto2Vec [39]. Ces représentations seront évaluées grâce aux métriques intrinsèques présentées par [5]. Étant donné que nous nous intéressons à des images, il peut être envisageable de générer une description textuelle de l'image [13], au coût d'un modèle supplémentaire. Cette description pourra ensuite être vectorisée pour être comparée aux connaissances vectorisées. Le modèle de description doit être générique au sens où il doit être capable de décrire les éléments de base présents sur l'image et leur organisation. Si ce modèle pouvait décrire des éléments de haut niveau, cela voudrait dire qu'il a déjà connaissance de la nouvelle classe qu'on recherche, ce qui rendrait caduque la démarche de découverte de nouvelles connaissances. L'espace de représentation latent de notre modèle pourrait aussi être comparé aux représentations vectorielles des connaissances, mais comme notre modèle est non déterministe, il faudra tenir compte du fait que la représentation latente des données ne sera pas toujours la même.

La deuxième méthode envisagée pour arriver à une représentation commune consiste à transformer les images sous forme d'un graphe [6], les connaissances étant déjà stockées sous forme de graphe. Une piste à explorer consiste à voir s'il est possible d'appliquer cette approche de transformation en graphe pour d'autres types de données.

Initialement, nous supposerons que les nouvelles classes à ajouter au modèle sont présentes et décrites dans les graphes de connaissances utilisés. Mais dans une application réelle, il est possible que les graphes de connaissances ne contiennent pas d'informations concernant les nouvelles données. Dans ce cas, une piste envisagée est un apprentissage conjoint entre le modèle et les structures de connaissances, ce qui peut également nécessiter de transformer les données en un graphe. Dans un dernier recours, il reste toujours la possibilité de faire intervenir un expert pour identifier les données ou enrichir les graphes de connaissances.

Après avoir transformé les données et les connaissances dans une représentation commune, il faudra trouver une manière de faire un lien entre les représentations dans le but d'attribuer une étiquette aux données. On peut par exemple penser à une mesure de similarité entre deux représentations vectorielles, en utilisant la similarité cosinus ou en passant par un modèle supplémentaire chargé d'apprendre la similarité [26]. Il existe aussi des méthodes de mesure de similarité entre graphes [24].

3.4 Amélioration du modèle

Une fois que les données ont été annotées, nous voulons ajouter cette connaissance acquise dans le modèle : si les données correspondent à une nouvelle classe, il faudra que le modèle l'apprenne, si la classe était déjà connue, cela permettra au modèle de renforcer sa connaissance. L'objectif de cet ajout de connaissances est de réduire l'incertitude du modèle quand il sera confronté à des données de cette classe. Dans cette optique d'ajout de connaissance, il est possible de se limiter à une modification du modèle seulement si un nombre suffisant de données ont préalablement été annotées. Le cas opposé, où seul un petit ensemble de données annotées permet la modification du modèle revient à faire de l'apprentissage en *few-shot* [42]. Pour améliorer le modèle, une approche d'apprentissage continu peut être mise en place. En particulier, utiliser une méthode qui permet le Forward Transfer est important, puisque notre approche se base sur l'hypothèse que le modèle connaît déjà des attributs présents dans les données, il faut que ce que le modèle a déjà appris puisse avoir une influence positive sur la nouvelle tâche qu'il va apprendre. Il faudra étudier dans quelle mesure le Backward Transfer pourra être utilisé, car les données annotées grâce aux connaissances peuvent introduire des erreurs dans le modèle et on ne veut pas que la nouvelle tâche réduise les performances du modèle sur les tâches précédemment apprises. Les approches de type *model growing* seront potentiellement à éviter, car dans un scénario réel le nombre de tâches n'est pas censé être connu à l'avance, donc la taille du modèle pourrait fortement augmenter. Dans le contexte que nous présentons, il est possible qu'il y ait naturellement de la *répétition* des données des tâches précédentes : si le modèle est devenu trop incertain dans ses prédictions d'une classe (s'il y a eu de l'oubli), alors une donnée de cette classe avec son étiquette pourra tout à fait être présentée au modèle. Nguyen *et al*. [28] présentent une méthode d'apprentissage continu appliquée au cas particulier des réseaux de neurones bayésiens (dont les poids sont représentés par des distributions), en utilisant les poids appris après chaque tâche comme distribution antérieure des poids de la tâche suivante.

D'autres approches [17], travaillent plutôt sur l'injection de connaissances d'un domaine à partir de graphes de connaissances directement dans les couches du modèle. Ces approches, connues sous le nom de « *Knowledge Infused Learning* » [11], permettraient de relier des caractéristiques couvrant différentes dimensions contextuelles d'un problème, en relevant les défis lexicaux et sémantiques spécifiques au domaine, tels que la rareté, l'ambiguïté et le bruit pour la classification selon une échelle d'évaluation informée par des experts du domaine. Ces travaux s'intéressent à deux questions. La première, comment décide-t-on d'injecter ou non des connaissances à un stade particulier de l'apprentissage, et comment mesurer cette injection de connaissances. La deuxième porte sur la manière dont on peut combiner les représentations de données situées entre les couches du modèle avec des représentations de connaissances externes issues de graphe de connaissances.

D'après les auteurs, cette approche d'injection de connaissances aborde des défis fondamentaux de l'IA, à savoir la réduction des données volumineuses, renforcer l'explicabilité des décisions du modèle, et améliorer la couverture des données et connaissances spécifiques à un domaine qui ne seraient pas considérées autrement.

4 Conclusion

Dans cet article, nous nous positionnons sur une approche divisée en plusieurs modules, permettant d'obtenir un modèle capable d'agir en toute autonomie dans le cadre de la classification d'images. Nous proposons d'utiliser un réseau de neurones Bayésien afin d'avoir un modèle capable d'exprimer son incertitude lors d'une prédiction. Cette incertitude est ensuite utilisée pour faire de la détection de nouveautés. Nous considérons que si le modèle n'est pas totalement certain, ni totalement incertain lors d'une prédiction, c'est qu'il a reconnu des attributs dans les données et qu'il s'agit donc de nouveautés qu'il peut être utile d'apprendre. Ces nouveautés doivent d'abord être étiquetées automatiquement en passant par une représentation commune aux données et aux connaissances stockées dans des graphes de connaissances. L'étape finale consiste à améliorer le modèle par apprentissage continu, grâce à l'utilisation des données étiquetées, ce qui permettra au modèle de réduire son incertitude.

Nos objectifs futurs consistent à mettre en place les différentes étapes de cette approche, en commençant par évaluer l'influence de l'apprentissage continu sur la mesure d'incertitude. La problématique de l'annotation automatique des données est le point le plus important, qui nécessitera des travaux approfondis.

Dans ce processus autonome qui se rapproche du raisonnement humain, il pourrait être intéressant d'intégrer un module supplémentaire, qui ajouterait de l'explicabilité au niveau des différents modules. Cela permettrait d'améliorer l'approche grâce à une meilleure compréhension, et donnerait encore plus de transparence et de confiance au processus.

Références

[1] General Data Protection Regulation (GDPR) – Official Legal Text. https ://gdpr-info.eu/.

[2] Oludare Isaac Abiodun, Aman Jantan, Abiodun Esther Omolara, Kemi Victoria Dada, Nachaat AbdElatif Mohamed, and Humaira Arshad. State-of-the-art in artificial neural network applications : A survey. *Heliyon*, 4(11) :e00938, November 2018.

[3] Rahaf Aljundi. Continual learning : A story line and wider view. Conference on Computer Vision and Pattern Recognition, 2021. page 18.

[4] Rahaf Aljundi, Daniel Olmeda Reino, Nikolay Chumerin, and Richard E. Turner. Continual Novelty Detection. *arXiv :2106.12964 [cs]*, June 2021.

[5] Faisal Alshargi, Saeedeh Shekarpour, Tommaso Soru, Amit P. Sheth, and Uwe Quasthoff. Concept2vec : Metrics for evaluating quality of embeddings for ontological concepts. *CoRR*, abs/1803.04488, 2018.

[6] Pedro H. C. Avelar, Anderson R. Tavares, Thiago L. T. da Silveira, Cláudio R. Jung, and Luís C. Lamb. Superpixel Image Classification with Graph Attention Networks. *arXiv :2002.05544 [cs, stat]*, November 2020.

[7] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning*, pages 1613–1622. PMLR, 2015.

[8] Gerben Klaas Dirk De Vries and Steven De Rooij. Substructure counting graph kernels for machine learning from rdf data. *Journal of Web Semantics*, 35 :71–84, 2015.

[9] Lisa Ehrlinger and Wolfram Wöß. Towards a definition of knowledge graphs. *SEMANTiCS (Posters, Demos, SuCCESS)*, 48(1-4) :2, 2016.

[10] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation : Representing Model Uncertainty in Deep Learning. *arXiv :1506.02142 [cs, stat]*, October 2016.

[11] Manas Gaur, Ugur Kursuncu, Amit Sheth, Ruwan Wickramarachchi, and Shweta Yadav. Knowledge-infused deep learning. In *Proceedings of the 31st ACM Conference on Hypertext and Social Media*, pages 309–310, 2020.

[12] Martin Grohe. Word2vec, node2vec, graph2vec, x2vec : Towards a theory of vector embeddings of structured data. In *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS'20, pages 1–16, New York, NY, USA, 2020. Association for Computing Machinery.

[13] MD Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CsUR)*, 51(6) :1–36, 2019.

[14] Knud Illeris. *Contemporary Theories of Learning : Learning Theorists... in Their Own Words*. Routledge, 2009.

[15] H. M. Dipu Kabir, Abbas Khosravi, Mohammad Anwar Hosen, and Saeid Nahavandi. Neural Network-Based Uncertainty Quantification : A Survey of Methodologies and Applications. *IEEE Access*, 6 :36218–36234, 2018.

[16] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13) :3521–3526, March 2017.

[17] Ugur Kursuncu, Manas Gaur, and Amit Sheth. Knowledge infused learning (k-il) : Towards deep incorporation of knowledge in deep learning, 2020.

[18] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[19] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324, 1998.

[20] Timothée Lesort, Vincenzo Lomonaco, Andrei Stoian, Davide Maltoni, David Filliat, and Natalia Díaz-Rodríguez. Continual learning for robotics : Definition, framework, learning strategies, opportunities and challenges. *Information Fusion*, 58 :52–68, 2020.

[21] Zhizhong Li and Derek Hoiem. Learning without Forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12) :2935–2947, December 2018.

[22] Guoliang Lin, Hanglu Chu, and Hanjiang Lai. Towards better plasticity-stability trade-off in incremental learning : A simple linear connector. *CoRR*, abs/2110.07905, 2021.

[23] David Lopez-Paz and Marc' Aurelio Ranzato. Gradient Episodic Memory for Continual Learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[24] Guixiang Ma, Nesreen K Ahmed, Theodore L Willke, and Philip S Yu. Deep graph similarity learning : A survey. *Data Mining and Knowledge Discovery*, 35(3) :688–725, 2021.

[25] Arun Mallya and Svetlana Lazebnik. PackNet : Adding Multiple Tasks to a Single Network by Iterative Pruning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7765–7773, Salt Lake City, UT, June 2018. IEEE.

[26] Stefano Melacci, Lorenzo Sarti, Marco Maggini, and Monica Bianchini. A neural network approach to similarity learning. In *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*, pages 133–136. Springer, 2008.

[27] Casey Newton. Google's knowledge graph tripled in size in seven months. *CNET. CBS Interactive*, 2012.

[28] Cuong V Nguyen, Yingzhen Li, Thang D Bui, and Richard E Turner. Variational continual learning. *arXiv preprint arXiv :1710.10628*, 2017.

[29] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua V. Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can You Trust Your Model's Uncertainty ? Evaluating Predictive Uncertainty Under Dataset Shift. *arXiv :1906.02530 [cs, stat]*, December 2019. Comment : Advances in Neural Information Processing Systems, 2019.

[30] German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks : A review. *Neural Networks*, 113 :54–71, May 2019.

[31] Razvan Pascanu. Continual learning the challenge. CVPR, 2021.

[32] Xuming Ran, Mingkun Xu, Lingrui Mei, Qi Xu, and Quanying Liu. Detecting out-of-distribution samples via variational auto-encoder with reliable uncertainty estimation. *Neural Networks*, 145 :199–208, January 2022.

[33] Daniel Ringler and Heiko Paulheim. One knowledge graph to rule them all ? analyzing the differences between dbpedia, yago, wikidata & co. In *Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz)*, pages 366–372. Springer, 2017.

[34] Petar Ristoski and Heiko Paulheim. Rdf2vec : Rdf graph embeddings for data mining. In *International Semantic Web Conference*, pages 498–514. Springer, 2016.

[35] F. Rosenblatt. The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6) :386–408, 1958.

[36] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv :1606.04671*, 2016.

[37] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[38] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual Learning with Deep Generative Replay. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[39] Fatima Zohra Smaili, Xin Gao, and Robert Hoehndorf. Onto2vec : Joint vector-based representation of biological entities and their ontology-based annotations. *Bioinformatics (Oxford, England)*, 34(13) :i52–i60, 2018.

[40] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout : A simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1) :1929–1958, 2014.

[41] Andra Waagmeester, Gregory Stupp, Sebastian Burgstaller-Muehlbacher, Benjamin M Good, Malachi Griffith, Obi L Griffith, Kristina Hanspers, Henning Hermjakob, Toby S Hudson, Kevin Hybiske, et al. Science forum : Wikidata as a knowledge graph for the life sciences. *Elife*, 9 :e52614, 2020.

[42] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples : A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3) :1–34, 2020.

Session "IA distribuée"

Une Simulation Multi-Agent Basée sur l'Affordance pour Contraindre l'Emergence

B. Doussin[1], N. Verstaevel[2], B. Gaudou[2], E. Kaddoum[3], F. Amblard[2]

[1] IRIT, Université de Toulouse, CNRS, Toulouse INP, UT3, Toulouse, France
[2] IRIT, Université de Toulouse, CNRS, Toulouse INP, UT3, UT1C, Toulouse, France
[3] IRIT, Université de Toulouse, CNRS, Toulouse INP, UT3, UT2J, Toulouse, France

benoit.doussin@irit.fr

Résumé

Dans ce papier nous décrivons un modèle dans lequel certaines contraintes ou interactions désirées sont directement décrites dans l'environnement. L'idée est de permettre à l'utilisateur de spécifier directement dans le modèle les interactions qu'il veut pouvoir observer à un niveau macroscopique en définissant deux rôles d'agents, les Enabler qui offrent certaines potentialités d'actions, et les Consumer qui peuvent les réaliser. Nous basons notre modèle sur le concept d'affordance et proposons une première implémentation sur le Campus de l'Université Paul Sabatier à Toulouse.

Mots-clés

Affordance, Simulation Multi-Agents, Emergence

Abstract

In this paper we describe a model in which some desired constraints or interactions are directly described in the environment. The idea is to allow the user to specify directly in the model the interactions that he wants to be able to observe at a macroscopic level by defining two agent roles, the Enabler, which offer certain action potentialities, and the Consumer who can achieve them. We base our model on the concept of Affordance and propose a first implementation on the Paul Sabatier Campus.

Keywords

Affordance, Multi-Agent Simulation, Emergence

1 Introduction

Les métropoles sont aujourd'hui marquées par un phénomène de croissance urbaine et de densification, conduisant à d'importants problèmes de congestion et de mobilité. L'aménagement urbain de nouveaux quartiers dans la ville nécessite donc une planification en amont des différents moyens d'accès, en particulier en terme de mode de transports publics ou partagés. Les outils d'analyse de la structure du tissu urbain et des différents réseaux de transport mais aussi de leur évolution [2] sont maintenant des outils indispensables aux urbanistes pour appréhender ces problèmes. Pour aller plus loin et être capable de tester l'évolution des comportements des habitants à différents choix d'aménagements, la modélisation et simulation à base d'agents [3] semble maintenant l'outil approprié [5].

L'approche de modélisation à base d'agents permet de représenter au niveau individuel un phénomène et de générer des phénomènes émergents au niveau macroscopique ou à des niveaux mésoscopiques. A titre d'exemple, la simulation des mobilités individuelles des utilisateurs d'un campus universitaires (étudiants, enseignants...) va faire émerger des patterns d'affluence dans les bâtiments d'enseignement ou de restauration, réguliers à l'échelle de la semaine. Ces patterns peuvent être considérés comme nécessaires à conserver, malgré l'évaluation de différents scénarios en termes de transports individuels, publics et partagés. Le problème principal de ces phénomènes émergents est qu'ils sont difficiles à anticiper, comprendre et contrôler [13]. C'est encore plus le cas au niveau mésoscopique : comment assurer que certaines contraintes fixées par le modélisateur sur des zones de l'environnement soient satisfaites, tout en conservant l'autonomie des agents individuels (niveau microscopique) dans leurs choix de comportement ? D'autre part, comment permettre au modélisateur d'exprimer de manière souple l'ensemble de ces contraintes à l'échelle mésoscopique ?

Nous proposons dans cet article un méta-modèle générique à base d'agents basé sur le concept d'affordance de Gibson [8] afin de permettre au modélisateur de contrôler l'apparition ou la conservation de certains patterns à l'échelle mésoscopique. En effet, pour Gibson, les affordances sont les actions possibles offertes par les objets perçus par un acteur à un moment et à un emplacement donné. Nous proposons donc de modéliser nos agents comme des entités proposant ou ayant besoin de services et cherchant à coopérer ensemble pour que chaque besoin d'interaction (chaque contrainte exprimée par le modélisateur sur une zone de l'environnement) soit satisfait [4].

L'article est organisé comme suit. Après une présentation des travaux existants sur l'utilisation de l'affordance dans les modèles à base d'agents (Section 2), nous introduisons en Section 3 le méta-modèle conceptuel, basé sur l'affordance et une approche écologique de la perception visuelle. Nous proposons ensuite en Section 4 une première implé-

mentation du modèle sur un cas d'application concernant l'étude de la mobilité sur un campus universitaire ainsi que les résultats obtenus lors de son exploration (Section 5). Enfin nous discuterons les perspectives de ce travail en Section 6.

2 Etat de l'art

2.1 Modélisation et simulation à base d'agents

Il existe dans la littérature plusieurs définitions d'un Système Multi-Agents (SMA). Selon Ferber [7], un SMA est constitué d'un ensemble d'objets passifs, d'un environnement, dans lequel sont situés les objets, ainsi que d'un ensemble d'agents autonomes. Ferber précise également que chaque agent peut disposer d'objectifs et de compétences qui lui sont propres.

L'approche multi-agent a permis de voir naître plusieurs axes de recherches, dont notamment la modélisation et simulation à base d'agents [16]. Là où dans les approches de modélisation à base d'équations, la dynamique du système est décrite par des équations décrivant l'évolution d'entités agrégées (souvent des stocks de population ayant une caractéristique commune), l'approche de modélisation à base d'agents représente explicitement le comportement de chaque entité individuelle. La dynamique globale du système, observée à un niveau macroscopique est alors le résultat des interactions de l'ensemble de ces agents dont les comportements sont décrits à un niveau microscopique[6].

Les simulations multi-agents permettent de conserver l'hétérogénéité du système, évite d'avoir à passer par une vue agrégée et sont particulièrement bien adaptés pour représenter des phénomènes localisés et distribués. Cette approche est maintenant appliquée dans la plupart des domaines de recherche : de la planification urbaine [5] ou la représentation du trafic [12]. Un des problèmes récurrents de ce type de systèmes concerne la maitrise par le concepteur du système des changements d'échelles entre des comportements spécifiés à l'échelle individuelle, des contraintes mésoscopiques et des phénomènes émergents observés à l'échelle macroscopique. Nous proposons dans ce travail sur la notion d'affordance afin de représenter ces contraintes.

2.2 Affordance et approche écologique de la perception visuelle

Le concept d'Affordance à été initialement introduit par James Jerome Gibson [9] et vient s'inscrire dans ses travaux sur la théorie de la perception écologique. Les affordances représentent ce qu'un élément de l'environnement peut offrir en terme de potentialité d'action à un animal (incluant l'être humain).

> "The affordances of the environment are what it offers the animal, what it provides or furnishes, whether for good or ill" [8]

Gibson décrit ces affordances comme des relations innées et complémentaires. La notion de complémentarité est importante car un même objet ne permettra pas nécessairement les mêmes actions à deux agents n'ayant pas les mêmes caractéristiques. Par exemple, là où une chaise affordе un humain de s'asseoir, elle affordе une fourmi de grimper. Gibson suggère alors de considérer la niche écologique d'un animal comme un ensemble d'affordances. De plus, il avance que ces affordances sont des données invariantes de l'environnement et qu'elles peuvent être perçues directement par l'animal et ne sont donc pas le résultat d'une inférence. En d'autres termes, nous ne percevons pas simplement une pomme ou une pêche, nous percevons un objet que l'on pourrait manger. Et dans la mesure où les affordances sont directement perçues par l'animal, elles le guident et le contraignent dans ses décisions. C'est cette notion de contrainte qu'il est pertinent d'utiliser dans notre modèle, car elle nous permettrait de guider les agents vers les phénomènes souhaités.

2.3 L'affordance dans les simulations

Le concept d'affordance a été largement repris au cours des dernières années dans le but d'imiter les comportements humains, notamment dans le domaine de la robotique [14]. On le retrouve également appliqué aux modèles à base d'agents avec pour objectif de reproduire les processus cognitifs humains.

Kapadia et al. [10] appliquent ce concept dans un modèle de déplacements de piétons et definissent la notion d'"*Affordance fields*", qui représentent un choix de chemin potentiel pour les agents. Les affordance fields sont détectés après une phase de perception et l'agent sélectionnera le chemin le plus intéressant pour lui.

Les travaux de thèse d'Afoutni [1] considèrent l'affordance comme un tuple à 3 éléments <*actuator, passiveObject, act*>. L'actuator peut être un agriculteur, l'objet passif un tracteur et l'action labourer. L'action labourer est le résultat du système $\{actuator, passivObject\}$ et ne peut être réaliser seul. Dans ce travail, elle considère les actuators et les objets passifs comme des agents non-autonomes appelés $environnemental\_entities$. Les affordances sont détectées par des agents abstraits appelés $place - agent$, qui demandent ensuite aux actuators d'exécuter les actions.

Enfin Klügl et Timpf [11] utilisent la notion d'affordance pour capturer de manière plus explicite le choix des partenaires d'interaction des agents. Ils prennent le parti de formaliser l'affordance comme un 4-tuples : $< a, e, act, p >$, où a est un agent souhaitant réaliser une action act, e un object de l'environnement et p une préférence, permettant de determiner quel est le partenaire le plus intéressant du point de vue de l'agent. Ils proposent aussi ce qu'ils appellent un "Affordance Schemata" permettant de faire émerger ces affordances. Un agent possède, pour chaque action qu'il peut être amené à vouloir réaliser, un schéma d'affordance de la forme : $< EType, condition, fpriority >$, où $EType$ est l'ensemble des types d'objets permettant l'action, e.g. les bancs ou les murets pour l'action s'asseoir, $condition$ exprime les contraintes selon lesquelles une affordance entre l'agent a et l'objet e peut exister et $fpriority$ détermine l'intérêt de l'agent à choisir cet objet.

3 Un Modèle conceptuel basé sur l'Affordance

Dans le but de réaliser un modèle permettant à l'utilisateur de spécifier les contraintes qu'il souhaite voir émerger, plusieurs notions doivent être introduites. Nous considérons que nos agents peuvent avoir deux rôles différents : le rôle d'*Enabler* et celui de *Consumer*. Dans le modèle, les *Enabler* sont des agents qui fournissent des potentialités d'actions aux *Consumer*. Ce sont sur ces *Enabler* que les utilisateurs vont être capables de projeter des contraintes. Les agents dotés du rôle de *Consumer* sont des agents capables de réaliser une action, mais sous l'influence des *Enabler* et een interaction avec eux. Par exemple, dans ce modèle, la chaise pourra être un *Enabler*, qui permettrait alors à la fourmi, le *Consumer*, de grimper. L'objectif est alors d'avoir un modèle permettant à l'utilisateur de spécifier, par exemple le nombre de fourmis qu'il voudrait voir monter sur cette chaise.

3.1 Description du Rôle *Enabler*

Les *Enabler* sont donc des agents qui $affordent$ aux agents *Consumer* un certain nombre d'actions mais les contraignent aussi dans leur choix d'action, car ils ne peuvent pas être amenés à réaliser une action que ne leur permettrait pas leur environnement. Pour ce faire, les *Consumer* doivent être capables de réfléchir directement sur les actions que leur permet leur environnement. Nous proposons donc dans ce modèle que les *Enabler* aient, pour chaque type de *Consumer*, un ensemble de potentialités d'action à lui offrir. Plus formellement, tout agent i ayant le rôle d'*Enabler*, dispose d'un ensemble de potentialités d'actions $Afford_i$ tel que :

$$Afford_i = \{Type : Act, \forall Type \in TYPE \text{ and } Act \subset ACTION\}$$

avec $TYPE$ est l'ensemble des types de *Consumer* et $ACTION$ l'ensemble des actions possibles On aurait alors par exemple : $Afford_{Chaise} = \{Humain : \{S'assoir\}, Fourmis : \{Grimper\}\}$.

C'est sur ces *Enabler* que l'utilisateur va aussi pouvoir préciser des contraintes qu'il voudra voir émerger à un niveau mésoscopique en spécifiant comment sont consommés les services proposés. Pour ce faire, nous introduisons une notion de besoin, qui permet de quantifier l'interêt d'un *Enabler*, i, à ce qu'une de ses potentialités d'action soit exécutée par un agent *Consumer* :

$$fBesoin : ENA \times ACTION \times TYPE \rightarrow \mathbb{R}$$

avec ENA l'ensemble des agents doté du rôle *Enabler*.

On pourrait imaginer que deux *Enabler*, fournissant un même service à un type de *Consumer*, aient chacun une préférence sur la manière dont sera exécuté une action. Ceci représente également le fait qu'un *Enabler* est doté d'une préférence pour interagir avec un type d'agents plus qu'un autre.

Cette fonction sera instanciée pour chaque type d'agents (cf. Section 4).

3.2 Description du Rôle *Consumer*

Les agents pourvus du rôle de *Consumer* sont les agents qui réalisent les actions affordées par les *Enabler*. Ils sont définis par un type (parmi tous les types disponibles définis par $TYPE$).

Nos agents seront dotés d'un cycle de perception-décision-action [17]. Dans notre modèle, la phase de perception sera focalisée sur la perception des potentialités d'actions provenant des *Enabler*. De plus, la phase de décision contiendra deux étapes principales : l'évaluation des potentialité d'actions disponibles selon leur désirabilité pour l'agent et la selection du couple action,*Enabler* le plus utile (cf. Figure 1.

- *Perception.* Nous considérons qu'un agent est influencé et contraint par son environnement dans les actions qu'il va effectuer : les agents dotés du rôle de *Consumer* ne peuvent effectuer que des actions perçues (donc affordées) de l'environnement. Dans cette première étape, nous considérons que les *Consumer*, quand ils perçoivent leur environnement, perçoivent en réalité un ensemble d'Affordances, i.e. un ensemble de potentialités d'action offertes par les *Enabler*.

$$Percues : CON \times ENA^n \rightarrow ACTIONS^m$$

où CON est l'ensemble des agents dotés du rôle *Consumer*.

Cette phase est très similaire à la perception directe initialement suggérée par Gibson [9]. C'est sur cet ensemble d'actions que va être amené le *Consumer* à réfléchir

- *Filtre par désirabilité.* Parmi ces actions possibles, toutes ne sont pas nécessairement intéressantes à réaliser pour l'agent à l'instant t. Il doit donc pour cela être capable de déterminer lesquelles lui sont $Desirables$. L'agent doit alors filter les affordances perçues pour n'en conserver qu'un sous-ensemble d'actions lui sont désirables. Il ne s'agit pas ici d'en choisir une seule, mais plutôt de faire abstraction des affordances n'ayant pas d'intéret pour lui.

$$Desirable : CON \times ACTION^i \rightarrow ACTION^j \text{ où } ACTION^j \subseteq ACTION^i$$

Pour être capable de déterminer quelles actions lui sont plus désirables que d'autres, un *Consumer* doit donc pouvoir exprimer un ordre de priorité entre les actions pour déterminer parmi celles qu'il perçoit lesquelles sont intéressantes ou non. Dans ce modèle nous proposons donc de quantifier le besoin qu'a un *Consumer* à réaliser une action donnée : nous considérons que chaque action possible est associée à une valeur de désirabilité.

$$fDesirabilite : CON \times ACTION \rightarrow \mathbb{R}$$

- *Sélection d'une action par utilité.* Parmi les actions que l'agent a pu déterminer comme désirables, il doit encore décider laquelle il souhaite réaliser et quel partenaire d'interaction choisir. Dans cette phase nous considérons donc que sa décision n'est pas nécessairement égoïste, c.-à-d. qu'elle ne dépend pas seulement de ses besoins mais aussi des besoins des *Enabler* qui lui permet de réaliser les actions qu'il a jugé désirables.

$$Utile : CON \times ACTION^j \times ENA^k \rightarrow ACTION \times ENA$$

Nous faisons l'hypothèse que l'on peut orienter les résultats de la simulation vers les phénomènes émergents souhaités en incitant les *Consumer* à choisir les partenaires d'interaction en fonction de leur besoin de réaliser une action mais en respectant aussi les $fBesoin$ des *Enabler*, résultant des contraintes imposées par l'utilisateur en entrée.

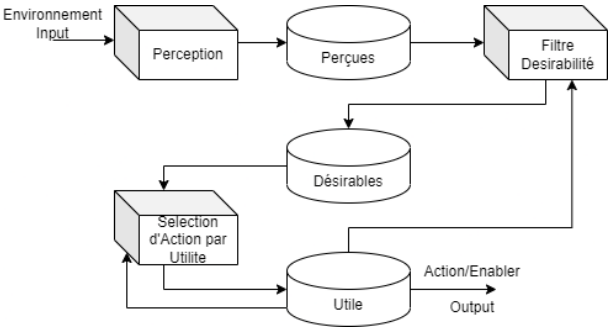


FIGURE 1 – Architecture PDU

Les trois étapes du cycle de perception et décision de l'agent sont résumés dans le schéma présenté sur la Figure 1.

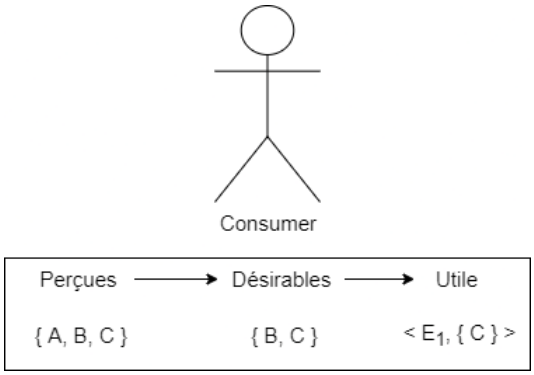


FIGURE 2 – Phases PDU

La Figure 2 représente une description d'un cas d'application simple. Un *Consumer* perçoit la possibilité de réaliser 3 actions différentes ($\{A, B, C\}$). Après les avoir filtré par leur désirabilité, l'agent détermine que seulement deux $\{B, C\}$ lui sont désirables. Finalement, même si l'action B lui serait plus désirable, il choisira finalement l'action C car la $fBesoin$ de l'*Enabler* est élevé.
Il est intéressant de noter qu'en ne considérant plus la fonction d'utilité, et en permettant à un *Consumer* de satisfaire son besoin le plus critique, on aurait alors des agents égoïstes. A contrario, si les *Consumer* ne choisissent plus en fonction de leur désirabilité, mais uniquement selon les besoins des *Enabler*, on aurait alors des agents altruistes.

4 Expérimentation

Nous proposons une implémentation de notre modèle sur un cas pratique : nous cherchons à étudier la mobilité sur une partie du campus de l'Université Paul Sabatier à Toulouse. Pour se faire les différents usagers du campus vont se déplacer de bâtiment en bâtiment en fonction de l'heure de la journée. Nous souhaitons reproduire l'occupation des différents bâtiments, contraintes que nous imposons à notre modèle.
Nous choisissons d'utiliser la plateforme GAMA [15] pour implémenter ce modèle, car elle permet de représenter de manière très riche l'environnement, et en particulier au moyen d'une bonne gestion des données géographiques.

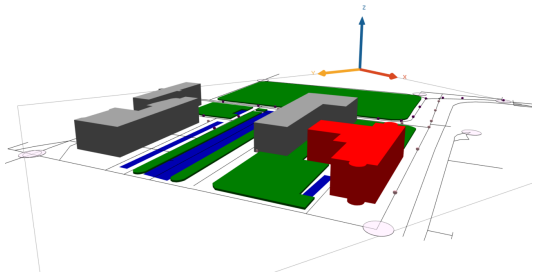


FIGURE 3 – Capture d'écran de la simulation faisant apparaitre l'environnement de la simulation avec les salles de cours (en gris), le restaurant en rouge et les générateurs de flux (cercles roses clairs).

4.1 Instanciation des agents *Enabler*

Dans notre implémentation du modèle, nous considérons plusieurs types d'agent ayant le rôle *Enabler*, chacun ayant différentes listes d'Affordances :

1. $Les\ salles\ de\ cours$: Ces objets, permettent aux Etudiants de venir $Etudier$, mais permettent aux Professeurs de venir $Travailler$. Nous en avons 3 dans la zone, représentés en gris dans la Figure 3. De plus, elles ont comme attributs une fréquentation Espérée et une fréquentation Maximale.
2. $Les\ restaurants$: Ils permettent à tous les *Consumer* de $Manger$. Il y a un restaurant, représenté en rouge (Figure 3). Il a lui aussi une fréquentation Espérée et Maximale.
3. $Les\ generateurs\ de\ Flux$: Ils représentent les entrées et sorties de la zone d'étude. Ils permettent de générer les agents arrivant dans la zone et permettent à ceux voulant la quitter de sortir. On considère donc qu'ils $Afford$ de $Partir$ à tous les agents *Consumer*. Il y a 6 générateurs de flux, tous situés sur les contours de la zone, placés sur les grands axes d'entrée et de sortie. Ils possèdent comme attribut le nombre d'agents qu'il doivent créer dans la zone à chaque pas de simulation (exprimé comme un nombre d'agents créés par heure).

La fréquentation escomptée, précisé en paramètre nous permet de définir la fonction $fBesoin$ des agents *Enabler*. Elle est définie comme la distance entre la fréquentation observée (F_{Obs}, calculée comme le nombre d'agents *Consu-*

mer dans le bâtiment considéré) et la fréquentation espérée F_{Esp} (si elle est supérieure à la fréquentation observée) et la fréquentation maximale (F_{Max}, que l'on normalisera. Ainsi on peut définir $fBesoin$ comme :

$$fBesoin = \begin{array}{l} \bigwedge \frac{Freq_{Esp}-Freq_{Obs}}{Freq_{Esp}} \text{ si } Freq_{Obs} < Freq_{Esp} \\ 1 - \frac{Freq_{Max}-Freq_{Obs}}{Freq_{Max}-Freq_{Esp}} \text{ sinon.} \end{array}$$

4.2 Instanciation des agents *Consumer*

Dans l'application de notre modèle au problème de la mobilité sur le campus, nous proposons d'implémenter deux types d'agents ayant le rôle *Consumer* :

1. $Les\ Etudiants$: Présents sur la zone dans le but d'étudier
2. $Les\ Professeurs$: Présents pour travailler / enseigner

Distinguer les deux est intéressant ici car le besoin d'un *Enabler* en $Etudiant$ n'est pas nécessairement le même que celui en $Professeur$. Un bâtiment peut avoir besoin de 100 étudiants dans l'heure là ou il n'aura besoin que de 5 étudiants.

Initialement vide de tout agent *Consumer*, la zone simulée se remplit pas de simulation après pas de simulation par des agents *Consumer* arrivant par les générateurs de flux. Au cours de la simulation, les agents vont se déplacer de bâtiments en bâtiments en fonction des affordances perçues dans l'environnement.

5 Résultats

Dans ce premier cas d'application, nous cherchons à étudier l'impact du mode de calcul de l'utilité chez les agents *Consumer* sur la satisfaction des besoins des agents *Enabler*.

Nous avons donc lancé un plan d'expérience dans lequel nous faisons varier 2 paramètres : le taux d'agents égoïstes et la fréquentation de la zone. Le taux d'agents égoïstes va déterminer la probabilité d'un qu'agent *Consumer* nouvellement créé soit égoïste (il cherche alors seulement à maximiser ses gains personnels) ou altruiste (il prend alors ses décisions en fonction de son utilité personnelle mais également des besoins des *Enabler*). Le paramètre de fréquentation de la zone détermine le nombre d'agents *Consumer* créé au cours de la simulation. Par la suite, nous explorons les valeurs suivantes :

1. le taux d'agents égoïstes prend une valeur dans $\{0\%, 1\%, 10\%, 25\%, 50\%, 75\%, 100\%\}$. L'échantillonnage n'a pas été fait régulièrement entre 0% et 100%, car il est apparu que la zone la plus intéressante se situait pour des faibles pourcentages.
2. la fréquentation de la zone prend une valeur dans $\{0, 100, ..., 3800, 3900\}$ (toutes les valeurs entre 0 et 3900 avec un pas de 100). La fréquentation de la zone représente le nombre de *Consumer* passant dans cette zone au cours de la simulation.

En sortie des simulations, nous allons observer le taux de satisfaction des agents *Enabler*. Ce taux est calculé en sommant les $fBesoin$ des *Enabler* au cours de la simulation de manière cumulative. A la fin de chaque heure, on observe la fréquentation de l'*Enabler* au cours de l'heure afin de déterminer si la fréquentation observée a été respectée.

La simulation est lancé à 7 heures du matin. Elle se termine à 20 heures le même jour. Le pas de temps de 1 min.

La figure 4 illustre les résultats obtenus par une exploration exhaustive de l'espace des paramètres.

On constate tout d'abord que dans le cas où on a 0% d'agents *Consumer* égoïstes, plus l'affluence est forte, plus on réussit à se rapprocher d'une criticité nulle (c'est-à-dire que les contraintes des bâtiments sont toutes remplies), là où dans un modèle avec 100% d'agents égoïstes n'arrive pas à satisfaire les besoins des agents *Enabler*, quelque soit la valeur de la fréquentation de la zone. On peut pas ailleurs noter que plus la fréquentation de la zone augmente, plus les résultats tendent à se stabiliser, montrant l'impact limité de la dimension stochastique de la simualtion pour de fortes affluences, alors qu'il semble fort à de faibles influences.

En ce qui concerne les valeurs intermédiaires du taux d'agents égoïstes, on observe que pour des petites fréquentation, moins il y a d'agents égoïstes et plus les besoins des bâtiments sont satisfaits. Néanmoins pour des grandes valeurs de fréquentation, les besoins des agents *Enabler* tendent à ne pas être satisfaits. Les agents égoïstes auront tendances à augmenter la criticité du système car ils ne respecteront pas la $fBesoin$ des *Enabler*. En faisant augmenter la fréquentation, on augmente aussi la quantité d'agents non-utiles au système (et même contre-productifs), ce qui explique que la criticité augmente, malgré différentes proportions d'agents cherchant à être utiles au système.

Nous constatons que sur une zone particulièrement fréquentée, même si la majorité des *Consumer* cherchent à se rendre utiles, il y a une forte augmentation de la criticité. Cette augmentation est dûe au fait qu'une fois passée la fréquentation espérée, les agents cherchant aussi à être utile ne peuvent plus compenser les décisions prises par des agents égoïstes, ce qui mène à une augmentation de la criticité globale.

6 Conclusion et Perspectives

Dans cet article, nous proposons un modèle conceptuel dans lequel l'utilisateur est capable de spécifier les contraintes qu'il souhaite voir émerger à un niveau mésoscopique ou macroscopique. En définissant des rôles d'agents nous pouvons créer un modèle dans lequel le choix de partenaire d'interaction suit le besoin global du système pour mener la simulation vers les phénomènes souhaités. Ce papier est une première étape avec pour objectif de pouvoir ensuite rendre explicable les phénomènes émergeants, souhaités ou non. Etre capable d'ainsi directement décrire dans l'environnement les règles d'interactions semblent être plus naturel dans le cadre de l'emménagement urbain et doit permettre d'expliquer les dynamiques globales du système.

Un prochain travail sera d'appliquer un tel modèle sur une

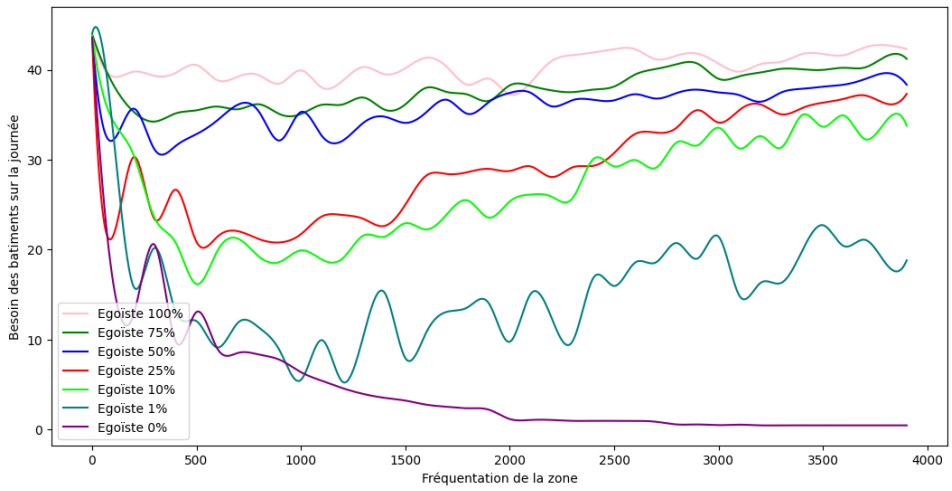


FIGURE 4 – Impact du taux d'agents égoïstes et de la fréquentation dans la zone sur la satisfaction des besoins des bâtiments

zone plus grande en regardant la cohérence des actions réalisées par nos agents. Il pourrait être aussi intéressant de regarder la pertinence d'avoir des *Consumer* possédant deux types différents.

Remerciements

Ces travaux ont été menés avec le soutien du Gouvernement Français dans le cadre du programme Territoire d'Innovation, une action du Grand Plan d'Investissement adossé à la 3eme vague du Programme d'investissement d'Avenir (PIA 3), de Toulouse Métropole et du GIS neOCampus de l'Université Toulouse III Paul Sabatier.

Références

[1] Z. Afoutni, R. Courdier, and F. Guerrin. Représentation de l'action humaine basée sur l'affordance vue comme une propriété émergente du couple acteur/environnement. In R. Courdier and J. Jamont, editors, *Principe de Parcimonie - JFSMA 14 - Vingt-deuxièmes Journées Francophones sur les Systèmes Multi-Agents, Loriol-sur-Drôme, France, Octobre 8-10, 2014*, pages 129–138. Cepadues Editions, 2014.

[2] M. Barthelemy. *The structure and dynamics of cities*. Cambridge University Press, 2016.

[3] E. Bonabeau. Agent-based modeling : Methods and techniques for simulating human systems. *Proceedings of the national academy of sciences*, 99(suppl 3) :7280–7287, 2002.

[4] D. Capera, J.-P. Georgé, M.-P. Gleizes, and P. Glize. The amas theory for complex problem solving based on self-organizing cooperative agents. In *WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on Enabling Technologies : Infrastructure for Collaborative Enterprises, 2003.*, pages 383–388. IEEE, 2003.

[5] A. T. Crooks, A. Patel, and S. Wise. Multi-agent systems for urban planning. In *Technologies for urban and spatial planning : virtual cities and territories*, pages 29–56. IGI Global, 2014.

[6] H. V. Dyke Parunak, R. Savit, and R. L. Riolo. Agent-based modeling vs. equation-based modeling : A case study and users' guide. In *International workshop on multi-agent systems and agent-based simulation*, pages 10–25. Springer, 1998.

[7] J. Ferber. *Les systèmes multi-agents : vers une intelligence collective*. InterEditions, 1997.

[8] J. J. Gibson. The theory of affordances. *Hilldale, USA*, 1(2) :67–82, 1977.

[9] J. J. Gibson. *The ecological approach to visual perception : classic edition*. Psychology Press, 2014.

[10] M. Kapadia, S. Singh, W. Hewlett, and P. Faloutsos. Egocentric affordance fields in pedestrian steering. In *Proceedings of the 2009 symposium on Interactive 3D graphics and games - I3D '09*, page 215, Boston, Massachusetts, 2009. ACM Press.

[11] F. Klügl and S. Timpf. Towards More Explicit Interaction Modelling in Agent-Based Simulation Using Affordance Schemata. In S. Edelkamp, R. Möller, and E. Rueckert, editors, *KI 2021 : Advances in Artificial Intelligence*, volume 12873, pages 324–337. Springer International Publishing, Cham, 2021. Series Title : Lecture Notes in Computer Science.

[12] F. Ksontini. *Modèle d'agent fondé sur les affordances : application à la simulation de trafic routier*. PhD thesis, Université de Valenciennes et du Hainaut-Cambresis, 2013.

[13] R. Lamarche-Perrin, Y. Demazeau, and J.-M. Vincent. Observation macroscopique et émergence dans les sma de très grande taille. In *SMA SYSTÈMES MULTI-AGENTS JFSMA 11*, pages 53–62, 01 2011.

[14] E. Şahin, M. Cakmak, M. R. Doğar, E. Uğur, and G. Üçoluk. To afford or not to afford : A new formalization of affordances toward affordance-based robot control. *Adaptive Behavior*, 15(4) :447–472, 2007.

[15] P. Taillandier, B. Gaudou, A. Grignard, Q.-N. Huynh, N. Marilleau, P. Caillou, D. Philippon, and A. Drogoul. Building, composing and experimenting complex spatial models with the GAMA platform. *GeoInformatica*, 23(2) :299–322, 2019.

[16] J.-P. Treuil, A. Drogoul, and J.-D. Zucker. *Modélisation et simulation à base d'agents : exemples commentés, outils informatiques et questions théoriques*. Dunod, 2008.

[17] M. Wooldridge. *An introduction to multiagent systems*. John wiley & sons, 2009.

Négociation de contenu sémantique pour l'échange de connaissances entre systèmes hétérogènes

Y. Taghzouti[1], A. Zimmermann[1], M. Lefrançois[1]

[1] Mines Saint-Étienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, F - 42023 Saint-Étienne France

yousouf.taghzouti@emse.fr, antoine.zimmermann@emse.fr, maxime.lefrancois@emse.fr

Résumé

Les ressources sur le Web sont identifiées par des identificateurs uniformes de ressources (URI). Chaque ressource peut avoir plusieurs représentations la décrivant, que nous appelons variantes. Un utilisateur (client) souhaitant une représentation particulière de cette ressource fait une requête à son URI avec un ensemble de contraintes. La négociation de contenu (NC) est le processus par lequel un serveur fait correspondre les préférences du client à l'ensemble des variantes de la ressource. Pendant longtemps, le type de média et la langue étaient les principales contraintes qui différenciaient les variantes. Plus tard, des variantes ont été créées pour représenter différentes versions évolutives dans le temps de la ressource avec le même type de média et la même langue. De plus, aujourd'hui, avec l'utilisation des technologies du web sémantique, notamment RDF, une ressource peut être décrite à l'aide de plusieurs vocabulaires et ontologies et en adhérant à différents profils de langage d'ontologie web. Une négociation plus fine de contenu est donc nécessaire, et la réponse du serveur doit être flexible, notamment pour répondre à une requête si aucune représentation ne valide parfaitement toutes les contraintes. Dans ce travail, nous présentons les défis associés à la résolution du problème de la négociation fine de contenu dans un environnement web hétérogène. Nous présentons nos questions de recherche pour résoudre ce problème et les hypothèses proposées. Enfin, nous décrivons notre méthodologie, y compris un plan d'évaluation, et fournissons les résultats préliminaires obtenus et la direction que nous envisageons de prendre.

Mots-clés

Négociation de contenu, web sémantique, contrainte.

Abstract

Resources on the Web are identified by uniform resource identifiers (URIs). Each resource can have several representations describing it, which we call variants. A user (client) wanting a particular representation of that resource makes a request to its URI along with a set of constraints. Content negotiation is the process by which a server matches the client's preferences to the set of variants of the resource. For a long time, media type and language were the primary constraints that differentiated the variants. Later, variants were created to represent different evolving versions over time of the resource with the same media type and language. Furthermore, today, with the use of semantic web technologies, especially RDF, a resource can be described using multiple vocabularies and ontologies and adhering to different web ontology language profiles. A finer negotiation of the content is therefore necessary, and the server response must be flexible, especially to answer a request if no representation perfectly validates all the constraints. In this work, we present the challenges associated with solving the fine-grained content negotiation problem in a heterogeneous web environment. We present our research questions to solve this problem and the proposed hypotheses. Finally, we describe our methodology, including an evaluation plan, and provide preliminary results obtained and the direction we plan to take.

Keywords

Content negotiation, semantic web, constraint.

1 Introduction

Ouvert, distribué, accessible et hétérogène sont quelques-unes des caractéristiques fondamentales du Web [4]. Bien que le fait que n'importe qui puisse accéder au Web de n'importe où dans le monde ait grandement contribué à son développement et à son enrichissement grâce à son ouverture, cela a eu l'effet indésirable d'avoir une abondance de ressources Web et des difficultés à fournir le meilleur contenu pour chaque client ; un exemple simple est celui de deux personnes parlant des langues différentes accédant à la même ressource. Dans ce cas, le serveur avec la ressource devrait être capable de fournir à chaque client une version compréhensible. Pour remédier à cela, une solution a été imaginée dès le départ, avec une couche de négociation entre le client et le serveur [3]. Elle est décrite dans le document Architecture of the World Wide Web comme l'un des composants essentiels de la conception du Web [14, section 3.2].

La négociation, en tant que concept, est une communication aller-retour destinée à atteindre un accord lorsque deux ou plusieurs parties ont des intérêts communs et d'autres opposés [9, p. 1]. Appliqué au Web, elle devient alors le

mécanisme permettant de sélectionner la représentation appropriée lors du traitement d'une requête. Dans HTTP, on peut exprimer et transmettre des contraintes appelées préférences dans [8, section 5.3]. Et avec cela, en plus de trouver et de transmettre des informations, il est possible de sélectionner des formats et des langages plus spécifiques.
Avec l'ère du mobile, un nouveau défi est apparu, le contenu déjà disponible était conçu pour s'adapter aux écrans d'ordinateurs et non aux téléphones. Une fois encore, il a fallu négocier le contenu pour savoir ce qui convenait à ces appareils en fonction de leurs caractéristiques [15].
Les ressources sur le Web étaient principalement destinées aux humains, pas aux machines. L'absence d'un contenu sémantiquement compréhensible a empêché leur exploitation complète, mais cela a changé avec le développement de langages web sémantiques pour décrire le contenu du Web et fournir un moyen pour les machines de le comprendre. Pour ce faire, on a d'abord utilisé le Resource Description Framework (RDF) [12]. Puis, en utilisant le Web Ontology Language (OWL) et une variété de vocabulaires [11]. Néanmoins, cette diversité a révélé le besoin d'une négociation fine qui va au-delà du simple format ou langage tel qu'il était.
Dans ce travail de thèse, nous nous intéressons à la manière de rendre la négociation de contenu (NC) plus fine dans/avec le web sémantique. Nous voulons utiliser le web sémantique pour faire de la négociation (1) et utiliser la négociation quand il s'agit du web sémantique (2).

(1) Dans la négociation, nous utilisons les technologies du web sémantique, par exemple les métadonnées décrivant une ressource dans différents vocabulaires.

(2) Lorsque nous voulons disposer de ressources web sémantiques spécifiques, par exemple en demandant une ressource à l'aide du vocabulaire ou un profil.

Le reste de cet article est structuré comme suit : La section 2 présente le problème, des cas d'utilisation le motivant, les questions de recherche et les hypothèses, le cadre de notre recherche et les contributions attendues, suivies de l'état de l'art et des travaux connexes dans la section 3. La section 4 décrit brièvement l'approche proposée, tandis que la section 5 décrit la méthodologie de recherche prévue. Les résultats préliminaires sont décrits dans la section 6, suivis de la conclusion et des travaux futurs dans la section 7.

2 Problème

2.1 Problématique

Une ressource disponible sous un identifiant de ressource uniforme (URI) spécifique peut avoir différentes représentations que nous appelons variantes ou alternatives comme dans [7, section 1.3]. La NC est le mécanisme permettant de choisir la meilleure représentation parmi les variantes disponibles. Le client inclut un ensemble de contraintes dans sa requête tandis que le serveur délivre une représentation sélectionnée lorsqu'elle correspond à ses propres contraintes [8].
Les questions qui se posent immédiatement sont : quelles sont ces contraintes ? comment les exprimer et comment les faire correspondre ? Au fil des années, les contraintes ont pris de nombreuses formes, en commençant par le type de média, la langue et l'encodage [8], en passant par des contraintes liées à la capacité du dispositif à traiter une certaine représentation [15] jusqu'à être plus complexes pour indiquer une interprétation sémantique [2].
Pour un serveur, outre l'interprétation et la correspondance des contraintes, un autre problème est l'explication du choix, soit de la représentation sélectionnée, soit des alternatives fournies qui ont été jugées suffisamment proches de celle demandée.

2.2 Cas d'utilisation

2.2.1 La négociation du vocabulaire

Chloé gère un portail d'information sur des artefacts anciens (indiquant p.ex. le créateur, la date de création, les matériaux utilisés) dont les données sont récoltés de différentes sources, telles que des musées ou Wikidata. Elle a remarqué que généralement chaque source utilise des vocabulaires différents, parfois personnalisés.
Dernièrement, elle a reçu de nombreuses demandes pour trouver un moyen de rechercher des données dans un vocabulaire spécifique, ou de spécifier les vocabulaires souhaités de manière ordonnée. Par exemple, exposer les données des créateurs en utilisant le vocabulaire FOAF (Friend Of A Friend), Schema.org ou DCMI (Dublin Core Metadata Initiative).
Actuellement, les graphes de données disponibles sur les API utilisent le même type de média : *text/turtle*. L'utilisateur doit interroger manuellement tous les graphes de données pour sélectionner ceux qui utilisent le vocabulaire souhaité.

2.2.2 Négociation des formes RDF

Alexandre est un chercheur qui s'intéresse à l'évolution du chômage des jeunes dans différentes sociétés ; il a besoin de données sous forme de graphes de données RDF. Pour ce faire, il interroge les graphes de données disponibles dans diverses API Web interrogées par le portail de l'université. Ces scénarios sont plausibles :
Scénario 1 - La forme est également importante : Alexandre a besoin d'une représentation qui se conforme à une forme spécifique. Par conséquent, la négociation du vocabulaire n'est pas suffisante car il devrait valider manuellement tous les graphes de données retournés avec les vocabulaires souhaités.
Scénario 2 - Flexibilité vs rigidité : Dans le scénario 1, la négociation peut être rigide dans le cas où Alexandre veut que *toutes* les contraintes soient valides, et préfère ne pas avoir de réponse autrement. Sinon, la négociation peut être flexible dans le cas où il accepte de recevoir une représentation même si elle ne satisfait pas toutes les contraintes.
Scénario 3 - Les contraintes n'ont pas la même importance : Pour Alexandre, toutes les contraintes de forme n'ont pas le même degré d'importance. Il veut donc un

moyen d'exprimer cette importance pour chaque contrainte et d'obtenir la représentation qui minimise le taux de violation en la prenant en compte.

2.3 Questions de recherche et hypothèses

Dans cette recherche, nous cherchons à répondre aux questions suivantes :

RQ1 Quelles sont les caractéristiques d'un cas d'utilisation de la NC, et comment peut-on les comparer et les classer ?

RQ2 Comment formaliser de manière uniforme les différents styles et dimensions de la NC ?

RQ3 Comment utiliser la validation sémantique pour demander la meilleure représentation qui valide partiellement la requête parmi un ensemble de variantes ?

RQ4 Comment évaluer la faisabilité et la qualité des algorithmes et méthodes proposés ?

Nos hypothèses sont directement dérivées des questions de recherche :

H1 La création d'une ressource ciblant la documentation sur la NC, en plus de fournir des cas d'utilisation de la NC avec leurs solutions existantes ou potentielles, encouragerait l'utilisation de la NC.

H2 Les technologies du web sémantique peuvent contribuer au mécanisme de la NC.

H3 Shapes Constraint Language (SHACL) n'est pas seulement utile pour valider les graphes RDF mais peut être utilisé pour introduire une certaine flexibilité dans le processus de choix de la meilleure représentation.

2.4 Cadre de la recherche

À ce stade de la thèse, nous avons fixé le cadre de la recherche pour inclure la NC basée sur les préférences de l'utilisateur et l'interprétation sémantique, ce qui signifie que, par exemple, les conditions du réseau, les caractéristiques du dispositif, les capacités du dispositif, l'état de la batterie, le coût monétaire sont exclus pour le moment. Cependant, bien que nous nous intéressions au mécanisme de la NC en général, une fois que nous aurons instancié notre modèle général, nous le ferons principalement à l'aide des sources RDF [5].

Ce cadre se manifeste dans la formalisation, nous avons commencé par la classe de document, puis ses sous-classes, cependant nous ne sommes pas intéressés à ce stade par la vitesse des flux de données par exemple. Il convient de mentionner que ce cadre ne nous empêchera pas de spécifier des cas d'utilisation employant le mécanisme de négociation dans la ressource Content Negotiation Theoretical Framework (CNTF) que nous développons dans le contexte de ce travail, par exemple, le cas d'utilisation de la négociation de la fréquence de mise à jour des objets (fraîcheur des données) dans un environnement Web des objets (WoT). Cependant, nous ne les prendrons pas en compte dans la partie formalisation du problème.

2.5 Contributions attendues

Les contributions attendues de cette recherche sont les suivantes :

C1 Un site web qui répertorie, catégorise et relie les techniques de la NC, les cas d'utilisation et les travaux connexes. En plus d'avoir le rôle de plateforme de diffusion pour nos futures contributions.

C2 Une formalisation du problème de la NC du général (négociation basée sur les documents) au spécifique (graphe RDF).

C3 Une solution pour la NC fine en utilisant des langages de validation sémantique, tels que SHACL.

C4 l'implémentation et la validation de la solution proposée.

3 État de l'art

La NC a été proposée comme une couche essentielle de l'architecture Web depuis le début [3], et a été implémentée dans le protocole HTTP en fournissant les moyens de négocier des variantes par le biais d'en-têtes entre autres : *accept* pour exprimer une contrainte sur le type de média de la représentation, et *accept-language* pour sélectionner la langue préférée [8]. Ainsi qu'un ensemble de codes d'état de réponse à utiliser pour indiquer si une demande HTTP spécifique a été satisfaite. Mais dans la plupart des cas, ces codes sont génériques et ne fournissent aucune explication. Avec HTTP, différents styles de la NC ont émergé, notamment la NC proactive, qui rend le serveur responsable du choix de la meilleure alternative, et la NC réactive, dans laquelle le serveur fournit une liste d'alternatives et c'est au client de choisir la meilleure. Il convient également de mentionner la NC transparente [13], qui permet aux mandataires de choisir au nom du serveur en tirant parti de l'entête HTTP *vary* [8]. Notre travail récent détaille davantage cette partie de l'état de l'art [29].

Malheureusement, les en-têtes HTTP de base ne sont pas suffisants, mais le protocole peut être étendu avec des nouveaux en-têtes. Des exemples d'en-têtes personnalisés sont *prefer* pour demander que certains comportements soient utilisés par un serveur [24], *accept-presentation* pour négocier la présentation RDF [17], *accept-schema* pour demander comment la ressource doit être structurée [26]. Cette approche résout le problème mais n'est pas évolutive compte tenu du fait qu'un nouvel en-tête est créé pour chaque nouvelle exigence. De plus, l'interopérabilité n'est pas atteinte puisque le nouvel en-tête doit être connu à l'avance.

Les capacités sont une autre dimension qui doit être négociée. Pour l'agent utilisateur, l'en-tête *user-agent* était utilisé, mais lorsque les appareils mobiles sont introduits, une nouvelle approche a été nécessaire, CC/PP (Composite Capabilities/Preference Profiles) et UAProfile ont été proposées pour résoudre ce problème [15, 21]. Mais aujourd'hui encore, alors que l'utilisation du Web des objets se développe, la négociation doit tenir compte de leurs limites, comme la faible puissance du processeur et de la batterie.

RDF est destiné à décrire les ressources sur le web en utilisant des vocabulaires et des ontologies. Et pour négocier ces

représentations, le groupe de travail sur l'échange de données a proposé le vocabulaire de profil [2], et des moyens de négocier les profils qui pourraient prendre la forme de ressources dans le langage de contrainte, par exemple SHACL ou ShEx [28, 27, 16, 23]. Cependant, cela manque de flexibilité, par exemple dans le cas où un serveur doit choisir entre deux représentations qui ne valident que partiellement un profil.

Pour répondre aux impacts sur les performances et la confidentialité de l'envoi d'en-têtes qui ne sont pas utilisés de manière fiable dans le traitement d'une requête client, une proposition récente est celle des Client Hints [10]. Un en-tête de réponse *Accept-CH* est introduit que les serveurs peuvent utiliser pour annoncer leur utilisation des en-têtes de requête pour la négociation proactive de contenu. Cependant, à notre avis, l'adoption d'en-têtes contribuant à la NC est insuffisante en raison de l'absence d'une ressource d'orientation. Cette ressource devrait documenter les approches et les caractéristiques de la NC, et les présenter de manière digeste dans le contexte de cas d'utilisation. Ces derniers comprendraient leurs solutions respectives ou potentielles et des pointeurs vers d'autres liens pour une investigation plus approfondie.

4 Approche proposée

Dans cette section, nous discutons de l'approche adoptée pour répondre aux différentes questions de recherche, qui comprend un cadre théorique prenant la forme d'un site web, une formalisation et un algorithme pour augmenter la flexibilité de la NC.

4.1 Cadre théorique de la négociation de contenu

Nous pensons que la NC est l'un des piliers du Web et mérite d'être étudiée plus en profondeur. Nous prévoyons de commencer par une documentation et une présentation bien conçues pour les nouveaux venus qui veulent apprendre la NC, ses caractéristiques et ses cas d'utilisation, ou pour les experts qui veulent se tenir au courant des dernières techniques et technologies utilisées [**RQ1**]. A cette fin, nous proposons de créer un cadre théorique pour catégoriser et évaluer les différents cas d'utilisation de la NC, et nous voulons matérialiser cela dans une ressource prenant la forme d'un site web que nous présentons dans la section suivante.

CNTF (Content Negotiation Theoretical Framework) est le site web que nous développons dans le cadre de ce travail pour permettre la catégorisation des caractéristiques de la NC en différents groupes : style, dimension, etc. CNTF vise à collecter des cas d'utilisation de la NC, à mettre en évidence les solutions existantes si elles sont disponibles, ou suggérer des moyens plausibles de les faire progresser. Ce sont les principaux objectifs de CNTF. Plus tard, elle sera utilisée pour la diffusion de nos nouvelles propositions pour faire avancer la NC. CNTF tente de répondre à ces exigences :

Conception navigable : CNTF doit avoir un design navigable qui inclut la possibilité de passer d'un concept à un autre, par exemple d'un cas d'utilisation à la dimension ou au style de la NC utilisé via des liens.

Extensible : L'un des principaux contrastes entre un document sur l'état de l'art traditionnel et la ressource CNTF est qu'elle doit être extensible en permettant l'ajout de nouveaux concepts de la NC, par exemple une nouvelle dimension, et être à jour avec les terminologie et les définitions différentes.

Catégorisable : CNTF doit fournir les moyens de catégoriser les différents cas d'utilisation et techniques de la NC pour permettre une évaluation comparative. Elle doit également prévoir un regroupement et une modélisation bien pensés des différents concepts de la NC pour en faciliter la compréhension.

Maintenable : CNTF devrait favoriser la maintenabilité en ajustant le modèle utilisé, par exemple les vocabulaires recommandés par la communauté, et en tenant compte des réactions et des commentaires fournis par les utilisateurs de CNTF pour clarifier et rectifier le contenu.

4.2 Une formalisation ascendante de la NC

Une autre façon d'étudier la NC sémantique est d'avoir plus de précision et de rigueur dans sa formalisation, pour cela nous suggérons une formalisation ascendante de la NC [**RQ2**]. Bien qu'il y ait eu des tentatives de formalisation de la NC, cela a été fait principalement pour s'adapter à un certain contexte, comme l'adaptation de contenu [19]. Dans notre démarche de formalisation de la NC dans des contextes sémantiques, nous prévoyons d'adopter une approche progressive, depuis la négociation de documents Web de base jusqu'à la négociation de documents de graphes spécialisés en utilisant des langages de validation sémantique, par exemple SHACL.

Le but est de définir de manière formelle ce qu'est une requête client avec ses contraintes, ce qu'un serveur doit fournir comme réponse, quelles sont certaines stratégies pour choisir la représentation à retourner. Ensuite, nous ajoutons progressivement de la complexité, par exemple comment exprimer une redirection lorsque le serveur ne peut pas satisfaire la demande mais peut fournir une URI qui aiderait le client. Et plus tard, nous pouvons définir les protocoles de manière formelle.

4.3 Un pas vers la NC sémantique

Pour résoudre les cas d'utilisation déjà présentés et parvenir à une NC flexible de manière pratique [**RQ3, RQ4**], nous utilisons SHACL. Plus précisément, nous utilisons l'en-tête *accept-profile* récemment introduit pour demander une variante qui valide un ensemble de contraintes sous la forme de documents SHACL [27]. Un client fait une demande avec un document SHACL. Le serveur, quant à lui, dans la procédure traditionnelle, dispose de l'ensemble des profils correspondant aux variantes. Si une variante est conforme au profil demandé, elle est servie, sinon un code d'état 406 (Not Acceptable), 404 (Not found) ou 300 (Multiple Choices) est renvoyé en fonction de la configuration

du serveur [1]. Dans notre approche, nous proposons de valider à la volée le profil demandé avec la liste des variantes disponibles, et de fournir la variante la plus proche si plusieurs valident partiellement les contraintes. Nous avons développé un algorithme simple pour montrer comment cela peut être réalisé : il prend en entrée une liste d'URI de documents SHACL S qui représentent les contraintes du client, chacune avec une valeur numérique optionnelle q_s indiquant la préférence de ce profil. Le serveur dispose d'une liste de graphes de données (variantes) G parmi lesquels il peut choisir. Le résultat de cet algorithme est un graphe de données dont le nombre de violations est minimum. Pour chacun des documents SHACL $s \in S$, nous validons les graphes de données disponibles et enregistrons le nombre de contraintes testées n_c ainsi que les contraintes valides v_c. Ensuite, nous calculons la mesure de validation avec la formule :

$$v_m = \frac{v_c}{Max(1, n_c)} \times q_s$$

Une fois que chaque paire (document de contraintes, graphe de données) a une mesure de validation, nous livrons celle qui a le meilleur score.
Une autre approche pour augmenter la flexibilité de la NC est l'endiguement des contraintes (constraint containment) [22, 18, 25, 1]. En prenant le document SHACL comme contraintes, nous pouvons dire qu'une forme est contenue dans une autre forme si chaque nœud d'un graphe satisfaisant les contraintes de la première forme satisfait également les contraintes de la seconde.
Considérons trois graphes de formes (SG) :

$$SG_1 = \{S_1, S_2, S_3\}$$

$$SG_2 = \{S_1, S_2, S_3, S_4, S_5\}$$

$$SG_3 = \{S_1, S_3, S_6\}$$

Nous pouvons imaginer trois cas de négociation :

- **Case 1 :** le client demande SG_1 qui est contenu dans SG_2 car tous les S_n de SG_1 sont contenus dans SG_2.
- **Case 2 :** le client demande SG_2 qui est partiellement contenu dans SG_1 car certains S_n de SG_2 sont contenus dans SG_2 mais SG_1 n'a pas de S_n supplémentaires.
- **Case 3 :** le client demande SG_1 ou SG_3. Dans les deux cas, nous avons des S_n supplémentaires, SG_1 est partiellement contenu dans SG_3 et SG_3 est partiellement contenu dans SG_1.

Les comportements suivants du serveur peuvent répondre à ces cas :

- **Case 1 :** Comme la SG demandée est entièrement contenue dans la SG disponible, nous validons les données avec celles demandées et les retournons.
- **Case 2 :** Puisque la SG demandée est partiellement contenue dans la SG disponible, nous pouvons soit valider les formes supplémentaires au graphe de données, soit retourner uniquement le graphe de données avec la SG disponible appliquée avec précision et rappel.
- **Case 3 :** Nous calculons la précision et le rappel entre le graphe de formes demandé et le graphe disponible. Ensuite, nous construisons un nouveau SG incluant uniquement les formes dont le test d'endiguement est positif. Enfin, nous le renvoyons à l'utilisateur avec les graphes de forme originaux et les mesures de précision et de rappel.

1. De multiples discussions ont eu lieu au sein du groupe de travail sur l'échange de données pour traiter ces questions, par exemple `https://github.com/w3c/dx-connegp/issues/5`

5 Méthodologie

La méthodologie adoptée dans l'élaboration de ce travail de doctorat respecte les tâches suivantes :

1. Investigation de l'état de l'art de la recherche relative au problème identifié. Cela inclut l'étude de la littérature sur les techniques de la NC, la formulation et la validation des contraintes dans le domaine du Web sémantique.
2. Formalisation du problème de la NC.
3. Création de la structure et des catégories de CNTF.
4. Collecte des techniques de la NC motivées par des cas d'utilisation et les ajouter à CNTF.
5. Fourniture des algorithmes pour la NC flexible et injection des implémentations dans un espace de test dans CNTF.

Après avoir réalisé un état de l'art approfondi, nous avons choisi de créer un site web qui supporte la catégorisation et l'évaluation comparative des caractéristiques de la NC. Nous avons collecté des cas d'utilisation, et identifié les styles, les dimensions et les exigences de ces cas d'utilisation.
Les techniques de la NC flexible seront d'abord évaluées à l'aide d'un ensemble de données et de profils synthétiques, puis de données réelles pour les travaux futurs. Les performances et l'évolutivité seront mesurées à l'aide du temps de réponse pour un ensemble de graphes et de profils de données prédéfinis. Le temps de réponse sera également mesuré en remplissant les graphes de données avec de plus en plus de triplets, afin de comparer les performances avec un nombre croissant de triplets (par exemple 500, 1000, 1500 triplets). La même technique de mesure sera appliquée pour un nombre croissant de contraintes dans les profils en ajoutant des formes dans le cas de SHACL.

6 Résultats préliminaires

6.1 Le site web CNTF

un site web a été développé [2], qui, au moment de la rédaction, classe la NC en catégories, un ensemble de cas d'utilisation de la NC, de styles, de dimensions ont été collectés

2. `https://ci.mines-stetienne.fr/cntf`

et ajoutés à la ressource et liés les uns aux autres, la section *updates* dans CNTF fournit la liste des fonctionnalités déjà ajoutées et celles à venir.

6.2 Formalisation de la NC

Une première version de la formalisation de la NC des documents a été réalisée et sera ajoutée à CNTF par la suite. Du côté du serveur, un URI est associé à des documents ayant chacun une valeur de qualité. Pour illustrer, considérons le scénario dans lequel un serveur dispose de deux représentations d'une ressource : d_1 en Turtle avec une préférence de 0,7 et d_2 en texte brut avec une préférence de 0,9. Un client souhaite demander une représentation Turtle. Dans l'ensemble de tous les documents $\mathcal{D}$, nous avons d_1 et d_2. La préférence du client pour un document est indiquée par une valeur de qualité q dans l'intervalle $[0, 1]$ [3]. Nous définissons l'ensemble $\mathcal{C}$ des préférences comme :

$$c \in \mathcal{C} \mid c : \mathcal{D} \rightarrow [0, 1] \tag{1}$$

En pratique, un client attribue généralement une valeur q à un *type* de documents. Ceci peut être modélisé comme une préférence c telle que $c(d) = q$ pour tous les documents d du type demandé, et $c(d) = 0$ pour tous les autres. Par exemple, si une requête possède l'en-tête `accept: text/turtle;q=0.9`, cela peut être traduit par la préférence $c(d) = 0.9$ pour tous les documents Turtle, et $c(d) = 0$ pour tous les autres.
Nous modélisons le Web comme une fonction de l'ensemble des IRIs $\mathcal{U}$ à l'ensemble des contraintes $\mathcal{C}$, formellement :

$$\mathcal{W} : \mathcal{U} \rightarrow \mathcal{C}$$

Le serveur de notre scénario peut servir deux documents :

$$\mathcal{W}(u) : \begin{cases} d_1 \mapsto 0.7 \\ d_2 \mapsto 0.9 \\ \mathcal{D} \setminus \{d_1, d_2\} \rightarrow 0 \end{cases} \tag{2}$$

Nous modélisons la réponse *RESponse* à une requête *REQuest* comme un document, en réalité la réponse contient des informations supplémentaires telles que des en-têtes. Dans notre modèle, RES est une fonction définie comme :

$$RES : \mathcal{RESQ} \rightarrow \mathcal{D} \tag{3}$$

La réponse dépend de la stratégie du serveur. Considérons une requête d'un client vers un IRI u et avec une contrainte c_1 définie comme $\langle u, c_1 \rangle$ dans $\mathcal{RESQ}$. Et un serveur qui peut servir une réponse à partir de l'IRI demandé u mais qui a aussi une contrainte c_2, on peut donc écrire $c_2 = \mathcal{W}(u)$. Nous définissons trois réponses possibles :

$$RES_s : \langle u, c_1 \rangle \rightarrow argmax(\{\mathcal{W}(u)(d) \mid d \in \mathcal{D} \wedge \mathcal{W}(u)(d) \neq 0\})$$

En utilisant la fonction RES_s, le serveur sert la réponse qui maximise ses contraintes, sans tenir compte des contraintes du client.

$$RES_c : \langle u, c_1 \rangle \rightarrow argmax(\{c_1(d) \mid d \in \mathcal{D} \wedge c_1(d) \neq 0 \wedge \mathcal{W}(u)(d) \neq 0\})$$

En utilisant la fonction RES_c, le serveur sert la réponse qui maximise les contraintes du client, c'est-à-dire, $RES_{noAnswer} = \emptyset$.
Le serveur n'envoie pas de réponse s'il existe un conflit entre ses contraintes et celles du client.

6.3 Un pas vers la négociation sémantique de contenu

Un autre résultat qui mérite d'être mentionné est une démonstration fonctionnelle de la NC à l'aide de profils qui prennent la forme de documents SHACL. L'implémentation a été faite en utilisant Java, et Spring Framework pour gérer les requêtes et intercepter les en-têtes de requête. Jena Framework a été utilisé pour gérer les graphes RDF, les documents SHACL et la validation. Une fois que nous aurons validé l'évolutivité des résultats, nous ajouterons l'implémentation à CNTF.

7 Conclusions et travaux futurs

La NC est très importante et constitue un mécanisme fondamental du Web. Les avantages et la nécessité d'exploiter la NC sont soulignés par le groupe de travail Spatial Data on the Web [30], ainsi que par le groupe de travail Web Data Best Practices [6] dans leurs documents sur les meilleures pratiques.
Dans cet article, nous analysons l'état des travaux existants en matière de la NC et indiquons une nouvelle direction : l'utilisation de langages de validation pour exprimer des contraintes plus fines. Nous pensons que dans les applications du monde réel, ces contraintes plus fines, lorsqu'elles sont explorées de manière appropriée, rendraient le processus de négociation beaucoup plus flexible. Nous avons proposé la ressource CNTF, la formalisation générale du processus de négociation, et un algorithme pour la négociation de contraintes utilisant SHACL. Notre travail futur se concentrera sur l'enrichissement de CNTF avec plus de cas d'utilisation et la finalisation des caractéristiques pour satisfaire les exigences mentionnées ci-dessus, ainsi que l'extension de la formalisation pour inclure la négociation de documents spécifiques tels que les graphes RDF en utilisant la négociation de profil. Les résultats de cette étude peuvent être utilisés pour valider et affirmer que SHACL peut être utilisé pour introduire de la flexibilité dans le processus de choix de la meilleure représentation.
Nous prévoyons de suivre ces étapes pour nos travaux futurs :

1. Continuer à évaluer et à étendre la négociation de profils.

3. Dans la RFC7231 [8, Section 5.3.1]. 0 signifie non acceptable, 0,001 est le moins préféré et 1 est le plus préféré.

2. Enrichir la formalisation pour inclure les solutions proposées.
3. Étudier plus en profondeur l'endiguement des contraintes [22, 18, 25, 1], puis proposer un algorithme qui en tire parti.
4. Prendre un cas d'utilisation réel d'un portail comme Europeana [4] qui sert de contenu provenant de différentes sources.
5. Explorer le cas d'utilisation de la NC du web des objets/hypermédia dans un environnement de bâtiment intelligent.
6. Examiner l'adaptation de contenu [20].

Références

[1] A. Abbas, P. Genevès, C. Roisin, and N. Layaïda. SPARQL Query Containment with ShEx Constraints. In *Proc. 21st European Conference on Advances in Databases and Information Systems*, volume 10509 of *LNCS*, pages 343–356. Springer, 2017.

[2] R. Atkinson and N. Car. The Profiles Vocabulary, W3C Working Group Note 18 December 2019. W3c working group note, W3C, December 18 2019.

[3] T. Berners-Lee, R. Cailliau, J. Groff, and B. Pollermann. World-Wide Web : The Information Universe. *Electronic Networking : Research, Applications and Policy*, 2(1) :74–82, 1992.

[4] N. Choudhury. World Wide Web and Its Journey from Web 1.0 to Web 4.0. *Int. Journal of Comp. Sci. and Information Tech.*, 5(6) :8096–8100, 2014.

[5] R. Cyganiak, D. Wood, and M. Lanthaler. RDF 1.1 Concepts and Abstract Syntax, W3C Recommendation 25 February 2014. W3c recommendation, W3C, February 25 2014.

[6] B. Farias Lóscio, C. Burle, and N. Calegari. Data on the Web Best Practices, W3C Recommendation 31 January 2017. W3c recommendation, W3C, January 31 2017.

[7] R. Fielding, J. Gettys, J. Mogul, H. Nielsen, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext Transfer Protocol - HTTP/1.1. RFC 2616, IETF, 1999.

[8] R. Fielding and J. Reschke. Hypertext Transfer Protocol (HTTP/1.1) : Semantics and Content. RFC 7231, IETF, June 2014.

[9] R. Fisher, W. Ury, and B. Patton. *Getting to yes : Negotiating agreement without giving in*. Penguin, 2011.

[10] I. Grigorik and Y. Weiss. HTTP Client Hints. RFC 8942, IETF, February 2021.

[11] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview (Second Edition), W3C Recommendation 11 December 2012. W3c recommendation, W3C, December 2012.

[12] P. Hayes and P. Patel-Schneider. RDF 1.1 Semantics, W3C Recommendation 25 February 2014. W3c recommendation, W3C, February 25 2014.

[13] K. Holtman and A. Mutz. Transparent Content Negotiation in HTTP. Technical report, IETF, March 1998.

[14] I. Jacobs and N. Walsh. Architecture of the World Wide Web, Volume One, W3C Recommendation 15 December 2004. W3c recommendation, W3C, December 15 2004.

[15] G. Klyne, F. Reynolds, C. Woodrow, H. Ohto, J. Hjelm, M. Butler, and L. Tran. Composite Capability/Preference Profiles (CC/PP) : Structure and Vocabularies 1.0, W3C Recommendation 15 February 2004. W3c recommendation, W3C, January 15 2004.

[16] H. Knublauch and D. Kontokostas. Shapes Constraint Language (SHACL), W3C Recommendation 20 July 2017. W3c recommendation, W3C, July 20 2017.

[17] M. Lefrançois. RDF presentation and correct content conveyance for legacy services and the web of things. In *Proceedings of the 8th International Conference on the Internet of Things, IOT 2018, Santa Barbara, CA, USA, October 15-18, 2018*, pages 43 :1–43 :8. ACM Press, October 2018.

[18] M. Leinberger, P. Seifer, T. Rienstra, R. Lämmel, and S. Staab. Deciding SHACL Shape Containment Through Description Logics Reasoning. In *The Semantic Web - ISWC 2020 - 19th ISWC, Athens, Greece, November 2-6, 2020, Proceedings, Part I*, volume 12506 of *LNCS*, pages 366–383. Springer, 2020.

[19] S. Lerouge. *Personalizing quality aspects for video communication in constrained heterogeneous environments*. PhD thesis, Ghent University, 2006.

[20] W. Lum and F. Lau. User-Centric Content Negotiation for Effective Adaptation Service in Mobile Computing. *IEEE Trans. on Soft. Eng.*, 29(12) :1100–1111, 2003.

[21] Open Mobile Alliance. User Agent Profile. Technical report, Open Mobile Alliance, May 2003.

[22] P. Pareti, G. Konstantinidis, F. Mogavero, and T. Norman. SHACL Satisfiability and Containment. In *The Semantic Web - ISWC 2020 - 19th ISWC, Athens, Greece, November 2-6, 2020, Proceedings, Part I*, volume 12506 of *LNCS*, pages 474–493. Springer, 2020.

[23] E. Prud'hommeaux, I. Boneva, J. Labra Gayo, and G. Kellogg. Shape Expressions Language 2.1, Final Community Group Report 8 October 2019. W3c community group report, W3C, 2019.

[24] J. Snell. Prefer Header for HTTP. RFC 7240, IETF, 2014.

[25] S. Staworko and P. Wieczorek. Containment of Shape Expression Schemas for RDF. In *Proc. 38th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 303–319. ACM Press, 2019.

4. https://www.europeana.eu/

[26] L. Svensson. An http Header for Metadata Schema Negotiation. In *W3C Workshop on Smart Descriptions & Smarter Vocabularies (SDSVoc)*. W3C, November 2016.

[27] L. Svensson, R. Atkinson, and N. Car. Content Negotiation by Profile, W3C Working Draft 26 November 2019. W3C Working Draft, W3C, November 26 2019.

[28] L. Svensson, R. Verborgh, and H. Van de Sompel. Indicating, Discovering, Negotiating, and Writing Profiled Representations. Internet draft, IETF, March 2021.

[29] Y. Taghzouti, A. Zimmermann, and M. Lefrançois. Négociation de contenu sur le web : un état de l'art. In *Journées Francophones d'Ingénierie des Connaissances (IC 2022) @ Plate-Forme Intelligence Artificielle (PFIA 2022)*, 2022. to appear.

[30] L. van den Brink, P. Barnaghi, J. Tandy, G. Atemezing, R. Atkinson, B. Cochrane, Y. Fathy, R. García-Castro, A. Haller, A. Harth, K. Janowicz, S. Kolozali, B. van Leeuwen, M. Lefrançois, J. Lieberman, A. Perego, D. Le Phuoc, B. Roberts, K. Taylor, and R. Troncy. Best practices for publishing, retrieving, and using spatial data on the web. *Semantic Web Journal*, 10(1) :95–114, 2019.

Recherche coopérative d'optimum global

D. Vergnet[1], E. Kaddoum[1], N. Verstaevel[1], F. Amblard[1]

[1] IRIT, Université de Toulouse, CNRS, Toulouse INP, UT3, **UT1**, **UT2**, Toulouse, France

damien.vergnet@irit.fr

Résumé

Dans cet article nous proposons une nouvelle métaheuristique basée sur la coopération pour chercher l'optimum global de fonctions à optimiser. Elle repose sur deux processus de recherche locale et semi-locale coopérative. Ses performances sont comparées à quatre autres métaheuristiques sur des problèmes d'optimisation mono-objectif sans contraintes. Les résultats montrent que l'approche proposée est capable de trouver le minimum global des fonctions testées plus rapidement que les méthodes comparées, tout en nécessitant un nombre d'itérations et d'appels à la fonction objectif plus faibles.

Mots-clés

coopération locale, décision collective, optimisation par métaheuristique, recherche locale

Abstract

This paper proposes a new cooperation-based metaheuristic for searching global optima of optimization functions. It relies on a local search process coupled with a cooperative semi-local search process. Its performances are compared against four other metaheuristics on unconstrained mono-objective optimization problems. Results show that the proposed metaheuristic is able to find the global minimum of the tested functions faster than the compared methods while reducing the number of iterations and the number of calls of the objective function.

Keywords

local cooperation, collective decision, metaheuristic optimization, local search

1 Introduction

La simulation est un outil précieux pour la compréhension du comportement et des limites de systèmes. Plusieurs études ont pour objectif de reconstruire des systèmes virtuels, appelés jumeaux numériques, pour simuler et vérifier le comportement de systèmes spécifiques [**?**]. De tels systèmes peuvent être mis en œuvre dans le domaine de la mobilité ou les études de catastrophes naturelles dans le but de reproduire des conditions de simulation spécifiques et comprendre les raisons de tels phénomènes [4]. La conception d'un jumeau numérique qui reproduirait le comportement exact d'un système réel n'est pas une tâche facile [**?**]. Étant donné que les système réels sont généralement des systèmes complexes qui présentent des interdépendances non-linéaires entre leurs paramètres, il est difficile de trouver la meilleure fonction à utiliser dans le modèle et d'adapter en temps réel ses paramètres pour garder le comportement de la simulation le plus proche possible de celui du système réel. De nombreuses études ont formalisé le problème de la calibration en un problème d'optimisation dans lequel les paramètres de la fonction modélisée sont ajustés par l'optimisation d'une fonction objectif : les paramètres de simulation deviennent des variables de décision et les sorties pertinentes du modèle sont intégrés dans des fonctions objectifs [2, 8]. Tout ceci implique la nécessité d'avoir un système d'optimisation capable de s'adapter rapidement aux changements qui pourraient survenir dans le système réel.

Plusieurs méthodes d'optimisation existent et pourraient être utilisées pour résoudre ce problème mais elles présentent d'important inconvénients tels qu'une tendance à converger vers des optimums locaux ou un manque de rapidité [6, 11, 14].

Dans ce papier, nous proposons une nouvelle métaheuristique d'optimisation locale appelée **CoBOpti**, pour ***Co****operation-****B****ased* ***Opti****mization* (Optimisation basée sur la coopération). Elle est basée sur une hypothèse de continuité locale de la dynamique de la fonction objectif, c'est-à-dire que la valeur de la fonction objectif ne varie pas significativement lorsque la valeur des variables de décision varie peu. Par rapport aux méthodes standards de l'état de l'art, CoBOpti atteint les solutions optimales en un nombre réduit d'itérations et d'évaluations de la fonction objectif.

Les principales contributions sont les suivantes :

— Nous introduisont une **nouvelle métaheuristique d'optimisation locale basée sur une hypothèse de continuité et de coopération**. Cette hypothèse permet de modéliser le problème de la recherche d'optimum global comme un **problème de coopération** où un point détermine le prochain point à explorer en exploitant l'information accumulée dans son voisinage.

— Nous menons des expériences visant à comparer notre approche sur des problèmes d'optimisation mono-objectif sans contraintes avec une seule variable de décision dans le but de démontrer que **l'approche proposée permet d'atteindre un optimum global tout**

en minimisant le nombre d'évaluations de la fonction objectif.

Ce papier est organisé de la manière suivante : la section 2 discute des limites de certaines métaheuristiques. La section 3 présente notre approche et comment elle répond à ces limites. Dans la section 4, nous introduisons les résultats de nos expériences qui sont ensuite discutées dans la section 5, avant de conclure avec ses limites actuelles et quelques suggestions de recherches futures.

2 État de l'art

[3] définit les problèmes d'optimisation comme la recherche d'un vecteur $\bar{x}_n^* = (x_1^*, ..., x_n^*)$ qui optimise une fonction objectif

$$\bar{f}_k(\bar{x}_n) = (o_1(\bar{x}_n), ..., o_k(\bar{x}_n)) \quad (1)$$

où $\bar{x}_n = (x_1, ..., x_n)$ est un vecteur de n variables de décision.

De nombreuses méthodes de résolution de problèmes d'optimisation existent, chacune émettant des hypothèses sur la nature du problème. Ces méthodes sont généralement classées par catégorie, nous nous intéressons à celle appelée métaheuristique. [6] définit les métaheuristiques comme des méthodes qui présentent deux niveaux de recherche, local et de plus haut niveau, et qui sont capables de sortir d'optimums locaux. Cette définition inclut entre autres les méthodes qui mettent en œuvre le concept de voisinage. Le voisinage d'une solution s est l'ensemble des solutions atteignables depuis s.

Les métaheuristiques sont intéressantes pour résoudre des problèmes d'optimisation car elles sont conçues pour explorer efficacement les espaces de recherche complexes [6]. Sörensen *et al.* [12] ajoutent que la grande majorité des problèmes d'optimisation réels sont plus facilement résolus par des métaheuristiques, d'où notre focalisation sur ces méthodes dans ce papier.

Les métaheuristiques reposent sur deux notions importantes : l'**intensification** et la **diversification**. L'intensification est un processus qui focalise la recherche sur les régions de l'espace de recherche qui semblent prometteuses, c'est-à-dire celles dans le voisinage de la meilleure solution actuelle. La diversification est un processus dont l'objectif est l'exploration de régions encore inconnues de l'espace de recherche dans l'espoir de trouver de meilleurs solutions. Ces notions reposent généralement sur la mémorisation des solutions visitées [5].

Il existe de nombreuses métaheuristiques, chacune avec leurs propres hypothèses. Étant donné que notre approche est destinée à être mise en œuvre dans la calibration en temps réel, elle doit s'appuyer sur des algorithmes rapides et capables de gérer l'ensemble des solutions visitées. Les méthodes présentées ci-dessous sont donc celles reposant sur un processus de **recherche locale** et/ou basées sur la **notion de population**.

Les algorithmes de **recherche locale** explorent l'espace de recherche en visitant le voisinage immédiat de la solution courante s et en sélectionnant la solution voisine qui a une valeur objectif plus petite que s. Afin de sortir des optimums locaux, ils présentent une phase de *hill-climbing* qui autorise une dégradation temporaire de la fonction objectif. Ces méthodes incluent le recuit simulé (RS), le *Generalized Simulated Annealing* (GSA), la recherche locale itérée (*Iterated Local Search*), la recherche locale guidée (*Guided Local Search*), etc. [6]. Le principale avantage de ces méthodes est leur rapidité. Elles présentent cependant une limite importante : elles ont tendance à se coincer dans des optimums locaux [11]. Certaines métaheuristiques locales, telles que la recherche tabou, utilisent une mémoire des solutions visitées pour contourner cette limite [6].

Une autre catégorie de métaheuristiques regroupe les **algorithmes à base de population**. Ces méthodes s'appuient sur un ensemble de solutions, appelé population. L'espace de recherche est exploré en évaluant chaque solution et en les modifiant avec une ensemble de règles simples. Il existe deux sous-groupes : les méthodes évolutionnaires et les méthodes s'inspirant de la nature [6].

Les **algorithmes évolutionnaires** (AE) sont des méthodes itératives basées sur la notion de *fitness*. La *fitness* d'une solution représente sa qualité d'après la fonction objectif. Au cours de chaque itération, appelée un *génération*, la fitness de chaque solution est évaluée. Les solutions possédant une fitness suffisament élevée sont conservées pour la génération suivante, toutes les autres sont mises de côté. De nouvelles solutions sont générées par croisement et mutation stochastique des meilleurs solutions issues de la phase de sélection. Cette catégorie inclut des méthodes telles que les algorithmes génétiques, l'évolution différentielle (ED) et la programmation génétique [6, 9]. Contrairement aux méthodes de recherche locale, les AE sont moins susceptibles de se coincer dans des optimums locaux grâce à des tailles de population importantes. Cet avantage induit néanmoins un autre inconvénient : plus la taille de la population augmente, plus l'algorithme requière une puissance de calcul élevée et donc des temps de résolution plus longs.

Il existe d'autres méthodes basées sur des populations qui diffèrent des AE. Elles s'inspirent de systèmes biologiques complexes tels que les nuées d'oiseaux ou les colonies de fourmis. Elles présentent le même avantage que les AE, à savoir une plus faible tendance à rester coincé dans des optimums locaux que les méthodes locales, mais souffrent aussi de temps de calcul plus longs et couteux [6]. D'autres méthodes, telles que PSO (*Particle Swarm Optimization*), présentent cependant une pauvre distribution de l'information au sein de la population de solutions, ce qui implique une tendance à converger trop rapidement vers des optimums locaux [14].

Dans notre méthode, nous proposons de combiner la vitesse de la recherche locale et la distribution de l'information des méthodes à base de population. Afin d'atteindre ce but, nous empruntons les notions de voisinage et de raisonnement collectif de ces méthodes. En se basant sur l'hypothèse que la **dynamique de la fonction objectif ne varie pas significativement entre deux points très proches**, nous proposons un système qui **recherche un optimum global en s'appuyant sur le raisonnement collectif des**

solutions déjà visitées.
Des métaheuristiques de recherche locale et basées sur des populations ont été présentées ainsi que leurs limites dans le contexte de la calibration en temps réel. La prochaine section décrit notre méthode, CoBOpti, que nous évaluons dans la section 4.

3 CoBOpti : optimisation par coopération

Dans cette section nous introduisons CoBOpti, une métaheuristique d'optimisation basée sur la coopération. La méthode que nous proposons combine la vitesse de la recherche locale et la distribution de l'information des algorithmes à base de population.
La section 3.1 décrit le principe général de l'approche en présentant un aperçu des différentes phases de recherche. La section 3.2 détaille le processus de recherche locale. La section 3.3 détaille le processus de recherche semi-locale et comment il permet de sortir des optimums locaux. Enfin, la section 3.4 décrit comment les points coopèrent pour résoudre quelques situations spécifiques.

3.1 Principe général

L'objectif de CoBOpti est d'explorer itérativement la surface d'une fonction objectif dans le but de trouver un optimum global. Au cours de chaque itération, le système doit déterminer le prochain point à explorer. Un **point** p_i est défini par une paire $p_i = (x_i, o_i)$ où x_i est la valeur de la variable de décision et o_i la valeur de la fonction objectif pour x_i. La succession des points visités lors d'une recherche locale est appelée une **chaine**. L'algorithme est constitué de quatre phases (figure 1).
L'algorithme combine deux étapes de recherche différentes : une **recherche locale** (**phases 1, 2 et 3**) dont l'objectif est de découvrir un minimum local, et une **recherche semi-locale** (**phase 4**) qui exploite l'ensemble des minimums locaux déjà découverts pour chercher un minimum global.
Le but de la **recherche locale** (**phase 1**) est de trouver un minimum local. Chaque itération t débute avec une chaine contenant des points déjà visités $p(t), p(t-1)$, etc. Parmi tous les points de la chaine, le système choisit deux points pour déterminer dans quelle direction il doit explorer (**phases 2 et 3**). Ce processus continue jusqu'à ce qu'un minimum local soit trouvé, c'est-à-dire jusqu'à ce que la distance selon l'axe x entre les deux points avec la valeur objectif la plus basse de la chaine courante soit inférieure à ε_{dist}.
L'objectif de la **recherche semi-locale** est d'explorer la fonction en direction d'un minimum global. Ce processus doit décider quel point $p(t+1)$ il faut explorer en se basant sur les minimums locaux déjà visités (**phase 4**). Une fois que le processus a décidé quel point explorer en suivant, une nouvelle chaine est créée et la recherche locale reprend à partir de ce nouveau point.
La recherche s'arrête lorsque qu'un minimum local a une valeur objectif inférieure à un seuil prédéfini ε_{obj}.

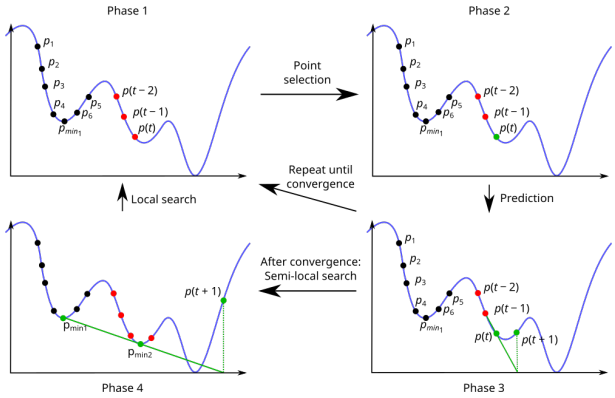


FIGURE 1 – Les phases de recherche de CoBOpti : sélection d'un point, recherche locale, recherche semi-locale

La notion de chaines est importante car elle permet l'isolation de groupes de points (points noirs et rouges dans la figure 1). Il n'est en effet pas désirable que des points distants interagissent lors du processus de recherche locale à cause de potentielles erreurs importantes entre la valeur réelle de la fonction et son estimation. L'utilisation de chaines implique que des points distants ne peuvent pas être utilisés pour calculer les approximations linéaires lors de la recherche locale (cf. section 3.2) et réduit donc le risque d'erreurs. Plusieurs chaines sont créées tout au long du processus d'optimisation.
Les sections suivantes détaillent la manière dont les points sont sélectionnés et comment $p(t+1)$ est calculé. La section 3.2 décrit comment le processus de recherche sélectionne les points successifs pour atteindre un minimum local. La section 3.3 décrit comment le système sort de minimums locaux et chercher un optimum global. Enfin, la section 3.4 décrit comment les points coopèrent pour résoudre quelques situations difficiles.

3.2 Recherche locale

L'objectif de la recherche locale est de suivre la courbe de la fonction objectif pour trouver un minimum local. Durant chaque itération t, le prochain point $p(t+1)$ à explorer est déterminé en calculant une approximation linéaire de la fonction objectif à partir de deux points de la chaine courante.
Étant donné que ces deux points sont proches relativement au domaine de définition de la variable de décision, nous considérons que les fonctions linéaires sont une approximation acceptable de la fonction objectif dans ce contexte.
Le premier point sélectionné est celui qui possède la plus faible valeur objectif au temps t, noté p_{min}. Le second point sélectionné est un des voisins de p_{min}. Deux points p_1 et p_2 d'une chaine sont considérés **voisins** s'ils sont à côté l'un de l'autre, c'est-à-dire s'il n'existe pas de point p_3 entre eux selon l'axe des x. Un point peut avoir un maximum de deux voisins. Par exemple, sur la figure 1, les points p_1 et p_2 sont voisins alors que p_2 et p_4 ne le sont pas.
Puisque p_{min} est le point avec la valeur objectif la plus

basse, il possède à tout moment un ou deux voisins.
Les phases 2 et 3 de la figure 1 illustrent la première situation où $p(t) = p_{min}$ (point vert) a un seul voisin $p(t-1)$. La composante x du prochain point $p(t+1)$ est calculbe par approximation linéaire de la fonction objectif entre $p_{min} = (x_{min}, o_{min})$ et son seul voisin $p(t-1) = p_n = (x_n, o_n)$:

$$x(t+1) = x_n + \frac{-o_n(x_{min} - x_n)}{o_{min} - o_n} \quad (2)$$

Cette équation calcule la composante x du point qui aurait une valeur objectif nulle d'après l'approximation linéaire de la fonction objectif.
Afin de s'assurer que l'hypothèse initiale sur la dynamique de la fonction reste vraie, le prochain point ne doit pas être distant de plus de k_{dist} fois la distance entre p_{min} et p_n. Si c'est le cas, $x(t+1)$ est fixé à $x_{min} + k_{dist}(x_{min} - x_n)$. Dans nos expérimentations, $k_{dist} = 5$ a été utilisée.
Dans la seconde situation, p_{min} a deux voisins p_l et p_h, tout deux ayant une valeur objectif plus élevée que p_{min}. Ceci implique qu'un minimum local se situe quelque part entre p_l et p_h. $x(t+1)$ est donc déterminé par :

$$x(t+1) = \frac{x_{min} + x_n}{2} \quad (3)$$

où x_n est la composante x de p_l ou p_h alternativement. La figure 1 montre un exemple de cette situation (points noirs). Le point p_6 a été calculé en utilisant cette équation avec p_4 en tant que p_{min} et p_5 en tant qu'un de ses voisins.
Nous tenons à faire remarquer que la fonction objectif n'a pas besoins d'être réévaluée à l'emplacement du voisin sélectionné puisque sa valeur est supposée ne pas avoir changé depuis sa première évaluation.
Ce processus est répété jusqu'à ce qu'un minimum local soit trouvé. Le point p_{min} est considéré comme étant un minimum local lorsque sa distance à un de ses voisins est inférieure à ε_{dist}.

3.3 Recherche semi-locale

L'objectif de la recherche semi-locale est de trouver un minimum global en parcourant la surface définie par l'ensemble des minimums locaux déjà découverts. On considère que cette surface donne la tendance générale de la fonction objectif et peut donc être approximée par des fonctions linéaires sans engendrer trop d'erreurs.
Afin de calculer la composante x du prochain point $p(t+1)$ à partir d'approximations linéaires de la fonction, deux points doivent être sélectionnés : le dernier minimum local $p_{min1} = (x_{min1}, o_{min1})$ trouvé par le processus de recherche locale et un de ses voisins. Les **voisins** d'un minimum local sont les autres minimums locaux adjacents. À l'instar des points décrits dans la section 3.2, les minimums locaux possèdent un maximum de deux voisins.
La table 1 décrit quel voisin est sélectionné en fonction de la situation explorée, où $p_l = (x_l, o_l)$ (resp. $p_h = (x_h, o_h)$) est le voisin de p_{min1} avec une composante x plus basse (resp. plus élevée).
Dans les situations 1, 2, 3 et 4, le prochain point $x(t+1)$ est calculé à partir de l'équation 2 en remplaçant p_{min} par p_{min1} et $p(t-1)$ par le voisin sélectionné. La phase 4 de la figure 1 illustre ce processus pour la situation 1. Dans ce diagramme, il y a deux minimums locaux connus, p_{min1} et p_{min2}, ce dernier étant celui découvert en dernier. Le prochain point $p(t+1)$ est estimé par approximation linéaire de la fonction entre ces deux minimums locaux. À l'instar de la recherche locale, $p(t+1)$ ne doit pas être à une distance supérieure à $k_{dist}|x_{min1} - x_n|$. Si tel est le cas, la même opération que celle décrite en section 3.2 est appliquée pour ramener le point sous cette distance.
Dans la situation 5, puisque p_l et p_h ont tous deux une valeur objectif supérieure à celle de p_{min1}, un optimum global ce trouve probablement entre p_l et p_h. L'équation 3 est utilisée pour déterminer le prochain point.
Une fois que $x(t+1)$ a été calculé, le processus de recherche locale reprend à partir de ce point avec une nouvelle chaine.

	Situation	Voisin sélectionné
1	1 voisin p_n	p_n
2	2 voisins, $o_l < o_{min1} < o_h$	p_l
3	2 voisins, $o_l > o_{min1} > o_h$	p_h
4	2 voisins, $o_l < o_{min1} > o_h$	p_l si $o_l < o_h$, sinon p_h
5	2 voisins, $o_l > o_{min1} < o_h$	p_l si $o_l < o_h$, sinon p_h

TABLE 1 – Voisin de p_{min1} sélectionné en fonction de la situation

3.4 Mécanismes de coopération

Les sections 3.2 et 3.3 ont décrit le comportement nominal de CoBOpti. Au cours des processus de recherche locale et semi-locale, le système peut rencontrer un certains nombre de situations particulières. Cette section présente des règles de coopération pour les détecter et les résoudres.
Cas 1. Durant la recherche locale, lorsqu'une chaine est créée, soit parce qu'il s'agit de la première itération ou parce que la recherche semi-locale vient d'en créer une nouvelle, elle ne contient qu'un seul point. Ce point n'a donc aucun voisin à sa disposition pour calculer le prochain point. $x(t+1)$ est alors choisi aléatoirement parmi $\{x_{min} - \delta, x_{min} + \delta\}$ où $\delta = \frac{1}{k_{prop}}|x_{low} - x_{high}|$ et x_{low} (resp. x_{high}) est la borne inférieure (resp. supérieure) du domaine de définition de x. Dans nos expérimentations, $k_{prop} = 100$ a été choisi.
Cas 2. Durant la recherche semi-locale, une situation similaire peut survenir, dans laquelle il n'existe qu'un seul minimum local connu. Dans ce cas-ci, puisqu'aucune approximation linéaire ne peut être calculée, un processus de **hill-climbing** est enclanché pour sortir du minimum local calculé. Ce processus repose sur les deux points de la dernière chaine qui possèdent la valeur x la plus élevée et la plus faible. Ces points sont appelés les extremums. L'objectif est donc de remonter les pentes de la fonction autour du minimum local pour trouver une autre pente de direction opposée (figure 2).
La recherche se focalise sur la pente où se situe l'extremum avec la plus basse valeur objectif. Le prochain point est cal-

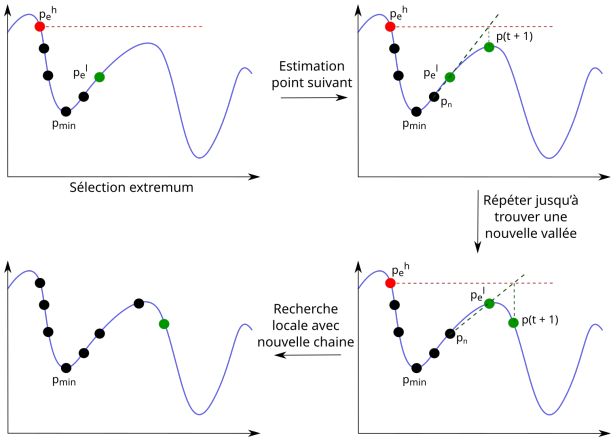


FIGURE 2 – Processus de hill-climbing

culé à partir de l'équation 4 où $p_e^l = (x_e^l, o_e^l)$ est l'extremum avec la plus basse valeur objectif et $p_e^h = (x_e^h, o_e^h)$ est l'autre extremum. $p_n = (x_n, o_n)$ est l'unique voisin de p_e^l. Cette équation calcule la composante x du prochain point qui aurait une valeur objectif égale à celle de l'autre extremum d'après l'approximation linéaire de la fonction entre p_e^l et son voisin p_n.

$$x(t+1) = x_e^l + \frac{(o_e^h - o_e^l)(x_n - x_e^l)}{o_n - o_e^l} \tag{4}$$

À l'itération suivante, si la valeur objectif réelle est supérieur à o_e^h, la recherche change de côté ; si ce n'est pas le cas, elle continue sur le même côté. Ce processus est répété jusqu'à ce que la valeur objectif réelle soit inférieure à o_e^l. Le processus de recherche locale reprend alors avec une nouvelle chaine.

Durant la phase de hill-climbing, la distance $|x(t+1) - x_e^l|$ ne doit pas être inférieure à un seuil δ_{min} afin d'éviter que le processus ne ralentisse trop.

Cas 3. Il est possible que le processus de recherche locale trouve un minimum local qui a déjà été découvert au cours des itérations précédentes. Afin d'éviter que le système ne boucle indéfiniment, deux décisions peuvent être prises. Si un processus de hill-climbing a déjà été initié précédemment à cet endroit, le prochain point $x(t+1)$ est calculé par approximation linéaire avec un facteur 2 afin d'explorer une nouvelle région un peu plus éloignée et d'éviter de revenir sur ce même minimum local. Si aucune phase de hill-climbing n'a déjà été entreprise, elle est démarrée pour explorer une nouvelle pente.

Deux minimums locaux sont considérés identiques si leur distance selon l'axe x est inférieure à un seuil ε_{same}.

Cette section a présenté notre approche. Elle s'appuie sur la notion de chaines de points. Nous avons d'abord présenté le processus de recherche locale sur une chaine qui permet de trouver des optimums locaux. Lorsqu'un optimum local est trouvé, un processus de recherche semi-locale permet de trouver de nouvelles régions à explorer dans l'espace de recherche. Des mécanismes de coopération ont été introduits pour répondre à quelques situations particulières, diversifier les solutions et créer de nouvelles chaines.

Dans la section suivante, nous évaluons les performances de notre méthode. Nous la comparons à quatre autres métaheuristiques sur des problèmes d'optimisation mono-objectifs sans contraintes.

4 Expérimentations et résultats

Cette section compare les performances de CoBOpti avec quatre autres méthodes citées dans la section 2 : le recuit simulé (RS), le *Generalized Simulated Annealing* (GSA), l'évolution différentielle (ED) et la *Particle Swarm Optimization* (PSO).

La section 4.1 présente les différentes fonctions utilisées pour tester les performances. La section 4.2 décrit le protocole pour comparer les performances de CoBOpti avec les autres méthodes. La section 4.3 présente les résultats des expériences. Enfin, les résultats sont discutés dans la section 5.

4.1 Fonctions de test

Quatres fonctions ont été sélectionnées pour les expériences de comparaison des performances :

1. Gramacy et Lee (domaine : $[0.5, 2.5]$) ;
2. Ackley (paramètres : $d = 1$, $a = 20$, $b = 0.2$, $c = 2\pi$; domaine : $[-32, 32]$) ;
3. Rastrigin (paramètres : $d = 1$; domaine : $[-5.12, 5.12]$) ;
4. Levy (paramètre : $d = 1$; domaine : $[-10, 10]$).

Ces fonctions ont été choisies car elles présentent de nombreux minimums locaux, un unique minimum global et un seul paramètre [1, 7, 10, 13].

4.2 Comparaison des méthodes

Les performances de chaque approche (RS, GSA, ED et PSO) sont comparées avec celle de CoBOpti. Chaque méthode a été implémentée en Python 3.8. GSA et ED ont été implémentées en utilisant `scipy.optimize`, PSO a été implémentée avec le package `pyswarm.pso` et RS a été implémenté par nous-même. Les paramètres optionnels de GSA, ED et PSO sauf ceux liés aux bornes, à l'état initial et au nombre maximal d'itérations ont été laissés à leur valeur par défaut.

Les variables de controle de CoBOpti sont définies comme suit : $\varepsilon_{dist} = 10^{-4}$ (seuil de détection d'un minimum local), $\varepsilon_{same} = 0.01$ (distance minimale entre deux minimums locaux), $\delta_{min} = 10^{-4}$ (le pas minimum lors de la phase de hill climbing) et $\varepsilon_{obj} = 5 \cdot 10^{-3}$ (seuil de détection du minimum global).

Pour chaque méthode, sauf PSO, la valeur initiale v_{init} de la variable de décision pour chaque exécution est sélectionnée par une séquence de Sobol. Sachant que les valeurs générées par cette séquence sont dans l'intervalle $[0, 1]$, elles sont ajustées au domaine de la variable de décision par la formule $v_{init} = s \cdot (d_{max} - d_{min}) + d_{min}$ où s est la valeur générée par la séquence. Aucune valeur initiale n'a pu

être spécifiée pour PSO car l'implémentation utilisée ne le permettait pas.

Trois métriques sont définies : le **taux de succès** (proportion d'exécutions qui ont abouti au minimum global), le **nombre d'itérations** nécessaire pour atteindre le minimum global et le **nombre d'évaluations de la fonction objectif**. Si une exécution ne parvient pas à atteindre le minimum global, le nombre d'itérations effectuées est fixé à la valeur maximale autorisée, 1000 dans les résultats suivants.

4.3 Résultats

La table 2 montre le taux de succès, le nombre moyen d'itérations et d'évaluations de la fonction objectif sur 200 exécutions pour chaque méthode et fonction, avec un maximum de 1000 itérations.

CoBOpti s'est montré capable de trouver le minimum global pour chacune des quatre fonctions. Le nombre moyen d'itérations se situe entre 35 et 100 ; le nombre d'évaluations de la fonction objectif est similaire.

Les 1000 itérations constantes pour RS et GSA s'expliquent par le critère d'arrêt de ces méthodes. Elles s'appuient sur le nombre d'itérations écoulé pour calculer des distributions de probabilités : plus le nombre d'itérations écoulé est grand, moins il y a de chance que l'algorithme sélectionne des actions qui n'améliorent pas la solution courante. Une fois que le nombre maximal d'itérations est atteint, aucune action qui dégraderait la solution ne peut être sélectionnée et l'algorithme s'arrête. La solution avec la valeur objectif la plus basse est ensuite retournée.

RS n'a pas donné de bons résultats, sauf dans le cas de la fonction de Gramacy et Lee avec un taux de succès de près de 100 %. Il a donné de très mauvais résultats pour la fonction d'Ackley avec seulement 2 % de réussite. Ces résultats sont cohérents avec ce qui a été décrit dans l'état de l'art (section 2).

GSA a donné de très bons résultats avec 100 % de réussite pour toutes les fonctions. Le nombre d'évaluations est cependant deux fois supérieur à RS, autour de 2000.

Le taux de succès de l'ED est un peu plus bas que les autres méthodes, sauf RS. Cependant, le nombre moyen d'itération est plutôt bas, entre 8 et 50 itérations sont nécessaires pour trouver le minimum global.

PSO a été en mesure de trouver le minimum global pour les quatre fonctions avec un faible nombre d'itérations, entre 20 et 50. Par contre, le nombre d'évaluations de la fonction est plus élevé que les autres méthodes, de 2000 à plus de 4500.

Méthode	Fonction	Succès	Nb. itér.	Nb. éval.
CoBOpti	G. & L.	100 %	49.31	50.31
	Ackley	100 %	95.94	96.94
	Rastrigin	100 %	80.69	81.69
	Levy	100 %	35.3	36.3
SA	G. & L.	99.5 %	1000	1000
	Ackley	2 %	1000	1000
	Rastrigin	10.5 %	1000	1000
	Levy	30 %	1000	1000
GSA	G. & L.	100 %	1000	2035.58
	Ackley	100 %	1000	2124.43
	Rastrigin	100 %	1000	2039.97
	Levy	100 %	1000	2019.60
DE	G. & L.	97.5 %	8.71	154.54
	Ackley	100 %	49.62	801.63
	Rastrigin	94 %	30.91	481.06
	Levy	100 %	50.45	773.75
PSO	G. & L.	100 %	20.61	2008.70
	Ackley	100 %	46.92	4638.51
	Rastrigin	100 %	25.57	2505.57
	Levy	100 %	20.02	1951.32

TABLE 2 – Taux de succès, nombre moyen d'itérations et d'évaluations de la fonction pour les méthodes testées

5 Analyse et discussion

L'hypothèse initiale de continuité de la dynamique de la fonction objectif a été validée par les expérimentations sur plusieurs fonctions standards. CoBOpti a montré des taux de succès plus élevés que RS et ED, et presque aussi bons que GSA et PSO. Même si le nombre d'itérations nécessaires à CoBOpti est comparable à celui de ED ou PSO, le nombre d'évaluations de la fonction est bien plus faible pour CoBOpti.

Ce faible nombre d'évaluations peut être attribué au fait que la fonction objectif n'est évaluée qu'une seule fois par point visité. Ce comportement découle de l'hypothèse initiale qui stipule que la dynamique de la fonction objectif ne change pas significativement entre deux points proches.

Les temps d'exécutions n'ont pas été montrés car les différences n'étaient pas significatives. Ceci est très certainement du à la faible complexité des fonctions sélectionnées.

Un analyse de sensibilité devrait être menée pour tester l'influence de k_{dist} et k_{prop} sur les performances de CoBOpti.

Notre méthode a été testée uniquement sur des problèmes d'optimisation mono-objectif sans contrainte avec une seule variable de décision. De plus amples recherches sont nécessaires pour généraliser cette approche aux problèmes multi-objectifs avec plusieurs variables de décision. Le principe central de la méthode que nous explorons reste très similaire à ce qui a été présenté ici. De nouveaux mécanismes de coopération seront ajoutés pour la sélection de l'objectif à minimiser et des variables de décision à ajuster à chaque itération. Quelques expériences ont été menées et semblent indiquer que l'augmentation du nombre de fonctions objectif (optimisation de problèmes multi-objectifs) n'impacte que peu les performances de CoBOpti.

D'autres expérimentations sont également à mener avec des fonctions plus complexes. Les cas d'applications réels étant sujets à des données bruitées, la résilience au bruit doit être testée.

6 Conclusion

Dans ce papier, CoBOpti, une nouvelle métaheuristique pour l'optimisation globale, a été présentée. Elle se fonde

sur une hypothèse de continuité locale de la dynamique de la fonction objectif. CoBOpti explore l'espace de recherche en s'appuyant sur la coopération des solutions visitées, elle-même basée sur l'hypothèse mentionnée plus tôt.
Ce papier se focalise sur des problèmes d'optimisation globale mono-objectifs non contraints avec une seule variable de décision. Les expérimentations montrent que CoBOpti nécessite moins d'évaluations de la fonction objectif par rapport à d'autres métaheuristiques communes tout en maintenant des taux de succès similaires voire meilleurs sur des fonctions unidimensionnelles.
CoBOpti est une proposition prometteuse pour la calibration en temps réel. En effet, son faible nombre d'évaluations de la fonction objectif pourrait être utile dans le context de la calibration en ligne de modèles de simulation complexes avec des fonctions objectif couteuses en temps de calcul. Cette propriété pourrait aider à réduire le temps nécessaire pour calibrer ce type de modèle.

Références

[1] Md. Alauddin. Mosquito flying optimization (MFO). In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pages 79–84, March 2016.

[2] Richard Arsenault, Annie Poulin, Pascal Côté, and François Brissette. Comparison of Stochastic Optimization Algorithms in Hydrological Model Calibration. *Journal of Hydrologic Engineering*, 19(7) :1374–1384, July 2014. Publisher : American Society of Civil Engineers.

[3] Jin-Hee Cho, Yating Wang, Ing-Ray Chen, Kevin S. Chan, and Ananthram Swami. A Survey on Modeling and Optimizing Multi-Objective Systems. *IEEE Communications Surveys Tutorials*, 19(3) :1867–1901, 2017. Conference Name : IEEE Communications Surveys Tutorials.

[4] Chao Fan, Cheng Zhang, Alex Yahja, and Ali Mostafavi. Disaster City Digital Twin : A vision for integrating artificial and human intelligence for disaster management. *International Journal of Information Management*, 56 :102049, February 2021.

[5] Michel Gendreau and Jean-Yves Potvin. Tabu Search. In Edmund K. Burke and Graham Kendall, editors, *Search Methodologies : Introductory Tutorials in Optimization and Decision Support Techniques*, pages 165–186. Springer US, Boston, MA, 2005.

[6] Michel Gendreau and Jean-Yves Potvin, editors. *Handbook of Metaheuristics*, volume 146 of *International Series in Operations Research & Management Science*. Springer US, Boston, MA, 2010.

[7] Robert B. Gramacy and Herbert K. H. Lee. Cases for the nugget in modeling computer experiments. *Statistics and Computing*, 22(3) :713–722, May 2012.

[8] Jingtao Ma, Hu Dong, and H. Michael Zhang. Calibration of Microsimulation with Heuristic Optimization Methods. *Transportation Research Record*, 1999(1) :208–217, January 2007. Publisher : SAGE Publications Inc.

[9] Karol R. Opara and Jarosław Arabas. Differential Evolution : A survey of theoretical analyses. *Swarm and Evolutionary Computation*, 44 :546–558, February 2019.

[10] Mitchell A. Potter and Kenneth A. De Jong. A cooperative coevolutionary approach to function optimization. In Yuval Davidor, Hans-Paul Schwefel, and Reinhard Männer, editors, *Parallel Problem Solving from Nature — PPSN III*, Lecture Notes in Computer Science, pages 249–257, Berlin, Heidelberg, 1994. Springer.

[11] Rainer Storn and Kenneth Price. Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11(4) :341–359, December 1997.

[12] Kenneth Sörensen, Marc Sevaux, and Fred Glover. A History of Metaheuristics. *Handbook of Heuristics*, to appear, January 2017.

[13] Fevrier Valdez and Patricia Melin. Parallel Evolutionary Computing using a cluster for Mathematical Function Optimization. In *NAFIPS 2007 - 2007 Annual Meeting of the North American Fuzzy Information Processing Society*, pages 598–603, June 2007.

[14] Yudong Zhang, Shuihua Wang, and Genlin Ji. A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications. *Mathematical Problems in Engineering*, 2015 :e931256, October 2015. Publisher : Hindawi.

Session "IA & humain"

L'Humain dans l'Apprentissage Automatique Interactif : aperçu de l'état de l'art

Kevin Delcourt, Jean-Paul Arcangeli, Sylvie Trouilhet, Françoise Adreit

Université de Toulouse, IRIT, UT3, UT2J, Toulouse, France

{Prenom.Nom}@irit.fr

Résumé

Les systèmes d'Apprentissage Automatique Interactif, ou IML (Interactive Machine Learning), font coopérer un utilisateur humain et une machine apprenante afin d'accomplir les tâches souhaitées par l'humain. Nous proposons dans cet article un aperçu de l'état de l'art de l'intégration de l'humain dans ces systèmes. Pour cela nous réalisons une étude de la littérature récente sur les solutions d'apprentissage automatique avec l'humain dans la boucle, et identifions les moyens mis en oeuvre pour prendre en compte et assister l'humain.

Mots-clés

Apprentissage Automatique, Apprentissage Interactif, Humain dans la boucle, Apprentissage Supervisé, Apprentissage par Renforcement, Interaction Humain-IA, Système Multi-Agent

Abstract

Interactive Machine Learning (IML) systems involve the cooperation of a human user and a machine learner in order to accomplish the tasks desired by the human. We propose in this paper an overview of the state of the art of human integration in these systems. To do so, we conduct a survey of recent literature on machine learning solutions with the human in the loop, and identify the means implemented to take into account and assist the human.

Keywords

Machine Learning, Interactive Learning, Human-in-the-loop, Supervised Learning, Reinforcement Learning, Human-AI Interaction, Multi-Agent System

1 Contexte

Dans le cas de problèmes complexes pour lesquels la programmation d'une solution n'est pas faisable à des coûts raisonnables, l'apprentissage automatique est la méthode de référence pour construire une solution. Pour cela, il est nécessaire de faire participer un expert en apprentissage au processus de développement de ces solutions. Or, d'après Amershi et al. [3], la demande en applications d'apprentissage automatique est bien supérieure à l'offre en compétences dans ce domaine.

Pour pallier ce manque, plusieurs approches proposent de permettre à des utilisateurs humains souvent sans connaissances en apprentissage automatique de répondre par eux-mêmes à leurs besoins. Ces approches sont regroupées sous le terme d'Apprentissage Automatique Interactif, ou IML (*Interactive Machine Learning*).

1.1 Apprentissage Automatique Interactif

Définition. Fails et Olsen Jr [16] définissent l'IML comme le domaine de l'apprentissage automatique dans lequel tout ou une partie des données exploitées par l'apprentissage provient des interactions avec un utilisateur humain du système : un humain, le plus souvent dépourvu de compétences en apprentissage automatique, interagit avec un système d'IML pour produire une solution répondant à ses besoins.

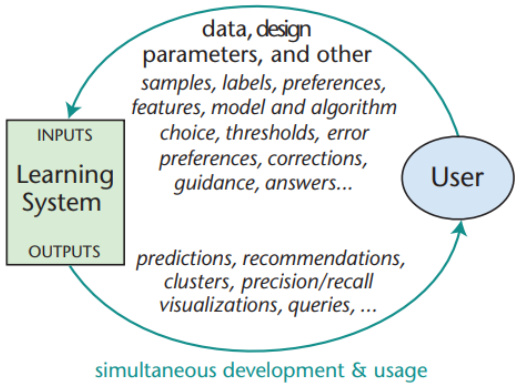


FIGURE 1 – Processus d'un système d'IML, extraite de [3]

Processus itératif. Le fonctionnement généralisé d'un système d'IML, décrit dans la figure 1, est itératif : l'humain fournit les divers paramètres, préférences ou plus généralement toute donnée demandée par l'apprenant (*i.e.* la machine) pour son fonctionnement. Ce dernier présente en retour le résultat de son apprentissage à l'humain, par exemple des recommandations ou des prédictions. L'humain doit alors déterminer si les résultats présentés par l'apprenant sont satisfaisants par rapport à l'objectif à atteindre. Le processus se répète jusqu'à ce que l'humain soit satisfait.

Applications. Les applications d'IML permettent par exemple de produire un classifieur d'images [7] ou de texte

[24], d'entraîner un robot à réaliser des tâches de manutention [8] ou de proposer des recommandations sur la base des actions de l'utilisateur [1].

1.2 Composition Logicielle Opportuniste

Nous avons été amenés à nous intéresser à l'IML par le biais de notre problématique de fond : la composition logicielle opportuniste dans le cadre de l'intelligence ambiante.

Intelligence Ambiante. L'intelligence ambiante a pour objectif d'offrir à un utilisateur humain un environnement physique et logiciel personnalisé et adapté à sa situation et à ses besoins [25]. Les principales difficultés à surmonter proviennent de la dynamique, de l'ouverture et de l'imprévisibilité des systèmes ambiants, induites par la mobilité des appareils et des utilisateurs. Le nombre de composants présents dans l'environnement est une autre source de complexité. Pour proposer des applications utiles et utilisables, les solutions doivent donc être capables de prendre en compte le contexte opérationnel et ses changements, notamment les préférences et les besoins de l'utilisateur ainsi que leurs évolutions.

Composition Opportuniste. Notre projet de recherche vise à concevoir et à développer une solution qui intègre l'utilisateur humain dans la boucle et fait émerger de nouvelles fonctionnalités dans un environnement ambiant. Il s'agit de construire, pour un humain, les bonnes applications au bon moment sans qu'il doive explicitement les demander. Notre solution utilise un apprentissage par renforcement [28], distribué au sein d'un système multi-agent (SMA), qui tire parti de retours de l'utilisateur pour construire une connaissance sur ses choix préférentiels en fonction des composants présents dans l'environnement.

Prototype. Un modèle de composition opportuniste a été conçu et un prototype logiciel a été développé [32, 22]. Une démonstration de ce dernier sur la base d'un cas d'utilisation est présentée dans [12]. Le fonctionnement itératif du moteur de composition opportuniste OCE (*Opportunistic Composition Engine*) est décrit dans la figure 2 : OCE sonde l'environnement de l'humain ; son SMA assemble ensuite une application qui est proposée à l'humain pour validation ; enfin, l'interface de visualisation de l'assemblage, ICE (*Interactive Control Environment*) [22], permet à l'humain d'accepter, de modifier ou de rejeter la proposition d'OCE. Cette action constitue le retour d'information qui est la source d'apprentissage pour les agents du SMA d'OCE.

1.3 Problématique

Notre solution de composition logicielle opportuniste est un exemple de système d'Apprentissage Automatique Interactif. Nous y retrouvons en effet un processus itératif avec, dans la boucle, un humain et une machine apprenante. L'humain occupe ici une place essentielle : en interaction avec le système apprenant, il est la principale source de l'apprentissage et c'est pour lui que l'apprenant opère.

Alors qu'une bonne intégration de l'humain est indispensable à une solution d'IML de qualité [3], la question de la place de l'humain dans notre système n'a pas été approfondie jusqu'à présent.

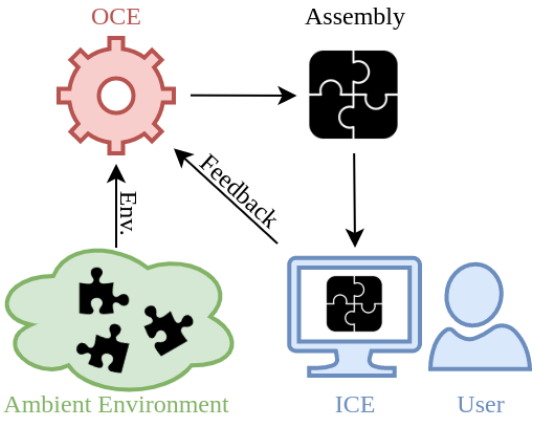


FIGURE 2 – Composition Logicielle Opportuniste

Plusieurs questions se posent quant à l'intégration de l'humain dans la composition logicielle opportuniste :

- Comment informer l'utilisateur de manière appropriée par rapport à ses capacités sur les applications que fait émerger le moteur de composition ?
- Comment faire contribuer l'utilisateur au système de manière à améliorer l'apprentissage ?
- Comment faire en sorte que ces interactions ne sollicitent pas trop l'utilisateur ?

Des réponses partielles existent dans OCE, comme les options de visualisation de ICE. Notre objectif est de proposer d'autres éléments de réponse. Pour cela, les questions de recherche définies dans la section suivante vont guider notre lecture des contributions récentes du domaine de l'IML.

1.4 Questions de recherche

Nous cherchons à répondre aux questions suivantes :

QR1-Humain. Quel sont le rôle et les responsabilités de l'utilisateur dans la boucle ?

- **QR11-Tâches.** Quelles tâches l'humain doit-il accomplir ? Pour accomplir ces tâches, quelle est la part de compétences en apprentissage automatique ?
- **QR12-Charge.** Quelle charge est imposée à l'humain ? Quel niveau d'engagement ou d'implication est attendu ?

QR2-Apprenant. Comment l'humain est-il pris en compte par la machine apprenante ?

- **QR21-Information.** Quelles sont les informations fournies par l'humain ? Comment et quand sont-elles considérées dans le processus ?
- **QR22-Assistance.** Comment le système accompagne-t-il et assiste-t-il l'humain ?

La section 2 présente notre lecture de la littérature récente par rapport à ces questions. La section 3 synthétise et analyse les réponses à ces questions.

2 Relations entre humain et machine apprenante

Afin de répondre aux questions de recherche élicitées dans la section 1.4, nous avons identifié des contributions scientifiques pertinentes dans la littérature récente du domaine l'IML. Sur la base de 32 articles traitant de différents sujets relatifs à des applications d'IML avec l'humain dans la boucle, nous avons sélectionné 12 contributions pertinentes par rapport à la question de la co-construction humain-machine d'applications. D'autres travaux d'état de l'art sur l'IML ou des domaines connexes ont été publiés [9, 10, 14, 20, 23, 33] mais ils ne ciblent pas la place de l'humain dans la boucle.

Dans cette section, nous présentons les contributions sélectionnées. Nous résumons chacune d'entre elles puis répondons aux quatre questions de recherche. Nous présentons d'abord les travaux d'apprentissage supervisé interactif, qui est à notre connaissance le plus étudié dans la littérature, puis les travaux d'apprentissage par renforcement interactif, plus proches de notre problématique. Nous terminons par les contributions traitant d'autres types d'apprentissage.

2.1 Apprentissage Supervisé

Les travaux sélectionnés sur l'apprentissage supervisé interactif concernent la production par l'humain de classifieurs d'image, de texte, ou de mouvements. La première contribution présentée concerne une approche particulière d'IML, la deuxième une étude, et les suivantes des outils.

2.1.1 Enseignement Automatique Interactif

Défini par Ramos et al. [24], l'Enseignement Automatique Interactif (*Interactive Machine Teaching*) est une approche d'IML dans laquelle l'humain joue le rôle d'un professeur, et doit enseigner une tâche à la machine. La notion d'enseignement regroupe ici le choix des informations à la source de l'apprentissage, et l'évaluation des performances de l'apprenant. L'hypothèse de fond est que l'humain acquiert plus facilement des compétences en pédagogie qu'en apprentissage automatique, ces compétences étant plus répandues dans le grand public [30].

Simard et al. [27] font un parallèle entre l'Enseignement Automatique Interactif et la programmation en général. Les deux activités partagent par exemple le fait de produire un artefact (un modèle appris ou un programme) répondant aux besoins d'un ou plusieurs humains. Cette comparaison amène les auteurs à penser que, à l'instar de la programmation qui a bénéficié d'outils et de langages de haut niveau, l'IML a besoin de ses propres outils et abstractions pour faciliter le travail des humains.

Les concepts et processus étudiés dans le cadre de l'Enseignement Automatique Interactif sont applicables à tout paradigme d'apprentissage. Néanmoins, les auteurs présentent une application de démonstration, appelée PICL [24] (anciennement MATE [30]), qui applique les principes de l'Apprentissage Automatique Interactif à l'apprentissage supervisé en permettant à des utilisateurs d'enseigner des tâches de classification de texte. Nous répondons aux questions de recherche pour cette application.

QR11-Tâches. L'humain, jouant ici le rôle d'un enseignant, doit planifier un curriculum, puis le mettre à jour en fonction des résultats de l'apprenant. Un curriculum désigne les données (exemples, étiquettes) exploitées par l'apprenant. Les compétences demandées sont plus de l'ordre de la pédagogie que de l'apprentissage automatique.

QR12-Charge. Demander à l'humain de déterminer quels sont les exemples les plus pertinents ou quels concepts enseigner l'implique fortement dans le processus. De plus, il doit être capable de comprendre si les résultats de l'apprenant sont satisfaisants ou non.

QR21-Information. Les labels et les concepts fournis par l'humain sont la source de l'apprentissage supervisé.

QR22-Assistance. L'interface des outils présentés [24] assistent l'humain en lui permettant de visualiser efficacement les résultats de l'apprenant. Une étude du comportement d'experts en apprentissage supervisé [30] a également permis d'identifier de bonnes pratiques d'enseignement automatique, implantées sous la forme de notifications.

2.1.2 Étude sur la confiance dans l'IML

L'étude menée par Honeycutt et al. [19] porte sur la question de la confiance de l'humain dans les applications d'apprentissage automatique. Elle se base sur des résultats en psychologie [29] qui montrent que la confiance d'un individu envers un groupe décisionnel humain augmente si un avis émis par l'individu, est pris en compte par le groupe. Inversement, l'individu aura moins confiance si son avis est ignoré. L'objectif de l'étude est de retrouver ces résultats en remplaçant le groupe décisionnel humain par un apprenant automatique, ici une application d'apprentissage supervisé de reconnaissance de visages dans des images.

L'expérimentation menée en ligne a mesuré la confiance d'utilisateurs humains envers cette application avec ou sans interaction d'une part, et avec une performance croissante, constante ou décroissante de l'apprenant d'autre part. Concrètement, la moitié des participants devait corriger les erreurs de l'apprenant, et la fréquence de ces erreurs était croissante pour un tiers, constante pour un autre tiers et décroissante pour le dernier.

Les résultats montrent qu'en général le groupe en interaction a moins confiance dans le système que le groupe qui n'est pas en interaction. L'explication avancée par les auteurs est que le groupe en interaction a passé plus de temps focalisé sur les erreurs du système pour les corriger.

Nous répondons aux questions de recherche pour l'application de reconnaissance de visages exploitée dans le cadre de l'expérimentation.

QR11-Tâches. Les compétences attendues pour les utilisateurs sont uniquement relatives à la tâche concernée : la reconnaissance de visage.

QR12-Charge. En dehors de la tâche de vérification des sorties de l'apprenant, l'utilisateur est faiblement impliqué.

QR21-Information. L'application simulée prend en compte les retours des utilisateurs sur ses erreurs afin d'affiner son modèle.

QR22-Assistance. Les utilisateurs sont volontairement peu accompagnés afin de pouvoir mesurer leur appréciation subjective de la performance de l'application, qui est une mesure proportionnelle à la confiance ressentie [31].

2.1.3 Classication d'image, son et postures

Carney et al. [7] présentent l'outil Teachable Machine de Google [1] permettant à des utilisateurs finaux de construire un classifieur d'images, de sons, ou de postures du corps humain. L'outil s'adresse à des utilisateurs novices en ML, avec possiblement des compétences en codage, puisque les modèles peuvent être exportés dans le langage JavaScript.
Afin de faciliter la tâche des utilisateurs, un transfert d'apprentissage a été mis en place : l'entraînement d'un nouveau modèle par l'humain est fait sur la base d'un modèle de base polyvalent, entraîné en amont par les développeurs de l'application.

QR11-Tâches. Ici l'humain a la charge de sélectionner les concepts à enseigner, et de recueillir les différents exemples (*i.e.* les images) associés. Il doit observer et juger lui-même les résultats de l'apprenant, donc fournir les données de test.

QR12-Charge. Il intervient à chaque étape de la production du modèle appris, du choix des données d'entraînement à l'évaluation de l'apprenant. Cela l'implique fortement.

QR21-Information. Les entrées fournies par l'humain, sous la forme d'images étiquetées, permettent de personnaliser le modèle de base fourni par l'application en fonction des besoins. Il est également possible de modifier les paramètres avancés de l'apprenant.

QR22-Assistance. L'interface guide l'utilisateur dans les différentes étapes de production du modèle. De plus, elle cache dans un premier temps les réglages avancés de l'apprenant, afin que l'investissement pour produire un modèle soit le plus faible possible.

2.1.4 Reconnaissance de formes

Fails et Olsen Jr [16] proposent à un humain de produire un système de reconnaissance de formes, au moyen d'une application d'IML appelée Crayons. Via une interface, l'humain colorie (*i.e.* étiquette) des portions de l'image qu'il souhaite que l'apprenant classifie. Ce dernier propose ensuite une classification de l'ensemble de l'image à l'humain, qui peut affiner son étiquetage pour corriger les éventuelles erreurs de Crayons.
Les auteurs discutent les différentes exigences liées à l'IML, en particulier le fait que l'apprentissage doit se faire le plus rapidement possible pour ne pas décourager l'humain. Par rapport à ces exigences, ils identifient les arbres de décision comme étant la meilleure approche, étant donné un temps d'apprentissage plus court malgré de moins bonnes performances que d'autres méthodes comme les réseaux de neurones. Notons cependant que cette conclusion n'est peut-être plus d'actualité étant donné l'ancienneté de cet article (2003).

QR11-Tâches. L'humain doit savoir déterminer sur l'image les classes à identifier (par exemple la peau) et en colorier une partie.

1. https://teachablemachine.withgoogle.com/

QR12-Charge. L'humain n'a pas à colorier avec exactitude les zones à classifier, ce qui évite donc de le surcharger mentalement. Il doit cependant affiner cet étiquetage jusqu'à ce que les résultats lui conviennent.

QR21-Information. L'image fournie par l'humain et les étiquettes qu'il appose en coloriant sont la source de l'apprentissage supervisé.

QR22-Assistance. L'étiquetage par coloriage permet de simplifier la tâche de l'humain. En dehors de cela, l'humain n'est pas particulièrement accompagné.

2.1.5 Classification d'images médicales

Berg et al. [5] présentent Ilastik, un outil d'Apprentissage Automatique Interactif pour la classification d'images dans le domaine médical. Ilastik permet à un expert du domaine médical de produire un classifieur d'images, de manière similaire à Crayons (section 2.1.4). Il est cependant plus complet, prenant en compte des données jusqu'à 5 dimensions (3 pour l'espace, 1 pour le temps, et 1 pour la multiplicité des points de vue). L'outil propose également à l'humain un choix parmi sept modes de classification d'images, qui impliquent différentes interactions avec l'humain. Par exemple, la manière d'étiqueter les exemples fournis par l'humain change en fonction de l'algorithme choisi (coloriage, clic sur des formes à détecter, etc.).

QR11-Tâches. Outre des compétences dans le domaine médical, l'humain doit savoir choisir, en fonction de son objectif, quel mode de classification utiliser. Il doit ensuite fournir et annoter des exemples représentatifs.

QR12-Charge. La multitude des modes de classification proposés, qui impliquent chacun des actions différentes de la part de l'humain, font que ce dernier doit être fortement investi dans l'application afin de la maîtriser.

QR21-Information. Les exemples annotés fournis par l'utilisateur sont la seule source d'entraînement.

QR22-Assistance. Si ce n'est la documentation disponible en ligne, l'humain est peu accompagné pour accomplir sa tâche. Une bibliothèque de modèles pré-entraînés est proposée, sans pouvoir toutefois modifier ces modèles pour les adapter à une tâche particulière.

2.1.6 Détection de l'activité

Flutura et al. [17] ont développé Drinkwatch, une application embarquée dans une montre connectée permettant à des humains de suivre leur consommation de boissons. Pour cela, l'application apprend les habitudes de consommation de l'humain via les capteurs de la montre.

QR11-Tâches. Tout le long de son activité, l'humain est sollicité pour valider ou invalider certaines activités détectées par la montre, c'est-à-dire qu'il collabore pour détecter les faux positifs. Il doit également signaler quand une activité n'a pas été détectée, c'est-à-dire les faux négatifs.

QR12-Charge. L'humain est sollicité uniquement quand l'application a un degré de confiance faible dans son classement, afin d'éviter de surcharger l'humain. Il reste cependant impliqué étant donné qu'il doit être vigilant aux faux négatifs et positifs de l'application.

QR21-Information. L'apprentissage se base sur les mouvements détectés par la montre et les retours de l'humain.

QR22-Assistance. Outre le fait que l'application sollicite le moins possible l'humain, des notifications sonores signalent la détection d'une activité par Drinkwatch afin de faciliter le travail de l'utilisateur.

2.2 Apprentissage par Renforcement

Les différents projets de recherche sélectionnés ici proposent des approches indépendantes de tout domaine d'application : tous permettent à un humain d'entraîner un agent (robot virtuel ou réel) à effectuer une tâche.

2.2.1 Apprendre d'humains inattentifs

Kessler Faulkner et Thomaz [21] s'intéressent au problème de l'inattention des utilisateurs humains. Pour cela, ils considèrent un cas d'utilisation où un humain doit enseigner des actions à un robot. L'humain observe le robot agir et peut récompenser positivement les actions du robot qui lui conviennent. Le robot a alors le choix entre l'exploitation d'actions qu'il sait satisfaisantes pour l'humain, et l'exploration de nouvelles actions à la récompense incertaine.
Ainsi, en faisant l'hypothèse que l'agent apprenant sache détecter l'absence d'attention de l'humain, les auteurs proposent que l'agent favorise l'exploitation quand l'humain est inattentif, et l'exploration dans le cas contraire. Plutôt que de demander l'attention de l'humain, ce qui peut conduire à une expérience utilisateur négative, le robot tire parti au maximum de l'attention disponible.

QR11-Tâches. L'humain doit surveiller les actions de l'apprenant et récompenser via une interface simple celles qui lui conviennent.

QR12-Charge. Bien que la surveillance de l'apprenant a tendance à fortement mobiliser l'humain, le fait qu'une baisse d'attention n'ait pas de conséquence néfaste pour l'apprenant est bénéfique pour l'expérience utilisateur.

QR21-Information. Les récompenses données par l'utilisateur sont celles de l'apprentissage par renforcement que l'apprenant cherche à maximiser.

QR22-Assistance. Ici, l'humain n'est pas accompagné.

2.2.2 Apprendre malgré des retours erronés

Akrour et al. [2] présentent une approche de l'apprentissage par renforcement interactif nommée programmation par feedback. Toujours dans un contexte d'humain récompensant les actions d'un apprenant, une particularité de cette approche est que l'apprenant estime le taux d'erreur de l'humain, c'est-à-dire une estimation de la probabilité que le retour donné par l'humain soit faux. Ainsi les avis de l'humain qui semblent être à l'opposé d'avis précédents sont ignorés.

QR11-Tâches. Après chaque séquence d'actions de l'apprenant, l'humain doit comparer la dernière séquence avec la meilleure séquence parmi les précédentes, et choisir celle qui correspond le mieux à la tâche à réaliser. Il doit donc uniquement savoir évaluer le comportement de l'apprenant dans sa globalité.

QR12-Charge. L'humain doit étudier chaque séquence d'actions de l'apprenant, ce qui demande une implication forte.

QR21-Information. Les comparaisons produites par l'humain permettent à l'apprenant d'estimer une fonction de récompense qu'il essaye de maximiser par ses actions.

QR22-Assistance. Ici, l'humain est peu accompagné dans son évaluation des séquences d'actions.

2.2.3 Apprentissage profond par renforcement

Christiano et al. [11] ont conçu un algorithme d'apprentissage profond par renforcement à partir de retours humains. Pour modeler un réseau de neurones sur la base des retours de l'humain, l'apprenant soumet à l'humain de courtes séquences d'actions à comparer. Cette comparaison est la source d'apprentissage d'une fonction de récompense que l'algorithme cherche en permanence à optimiser.
Les auteurs ont expérimenté leur solution sur plusieurs cas d'utilisation via la plateforme OpenAI Gym [6]. Dans les cas où les fonctions de récompenses sont connues, la solution a pu être comparée avec un algorithme d'apprentissage par renforcement classique et dans certains cas elle atteint de meilleurs résultats. Nous notons cependant qu'un seul utilisateur humain a participé aux expérimentations, ce qui limite la portée des conclusions.

QR11-Tâches. Ici l'utilisateur doit comparer deux séquences d'actions et décider de la meilleure par rapport à son objectif.

QR12-Charge. Le fonctionnement asynchrone apprenant-humain fait que moins de 1% des actions de l'apprenant sont jugées par l'humain. La charge qui lui incombe est donc réduite.

QR21-Information. Les retours de l'humain permettent de mettre à jour la fonction de récompense de l'algorithme.

QR22-Assistance. L'interface offre des raccourcis clavier facilitant l'interaction de l'humain.

2.3 Autres travaux

2.3.1 Apprentissage interactif pour l'optimisation

Holzinger et al. [18] présentent une expérimentation comparant des approches d'apprentissage avec et sans assistance humaine dans le cadre de résolution de problèmes du voyageur de commerce.
Sur la base d'un système multi-agent de colonie de fourmis (*Ant Colony Optimization*) [13], deux solutions, faisant participer ou non l'humain ; sont comparées. Dans l'étude présentée, l'ajout de l'humain en assistance du système multi-agent améliore les résultats de l'algorithme d'optimisation.

QR11-Tâches. L'utilisateur agit ici de manière implicite sur l'apprentissage : l'interface utilisée prend la forme d'un jeu. On lui demande donc simplement de jouer sans l'informer du système multi-agent qu'il influence.

QR12-Charge. Il est faiblement impliqué, car il n'est pas informé de l'apprentissage, ni de ses résultats.

QR21-Information. Les actions de l'humain dans le jeu ont pour effet d'aider le système multi-agent.

QR22-Assistance. Ici, l'humain n'est pas accompagné mais il n'a pas à l'être puisque sa contribution n'est pas explicite : il n'est pas informé de l'algorithme en arrière-plan.

2.3.2 Recommandation de lecture

Schnabel et al. [26] font une étude de plusieurs interfaces pour une même application de recommandation d'articles à lire. Ils s'intéressent à la visibilité qu'a un utilisateur sur l'effet de son action (*i.e.* sélectionner ou non un article à lire) sur l'apprentissage et les actions futures de l'apprenant (*i.e.* les futures recommandations). L'objectif est de mesurer l'impact de cette visibilité sur l'efficacité des utilisateurs mesurée en nombre de bonnes recommandations acceptées par l'humain en un temps donné. L'étude s'intéresse également à l'expérience utilisateur ressentie.
Pour répondre aux questions de recherche nous nous intéressons à la version de l'application préférée des utilisateurs lors de l'étude. Dans cette dernière les conséquences de la sélection d'un article à lire sont mises en valeur pour l'humain avant même que l'action soit effectuée. C'est-à-dire qu'au survol d'un article par la souris, les modifications que son ajout apporterait à la liste de des recommandations sont affichées en surbrillance.

QR11-Tâches. L'utilisateur doit sélectionner des articles à lire parmi des recommandations. Il doit juger de la pertinence de ces articles en fonction de ses goûts.

QR12-Charge. Le niveau d'implication de l'humain est faible compte tenu de la tâche à effectuer. Dans le cadre de l'expérimentation, une contrainte de temps a été instaurée, qui ajoute une certaine charge mentale aux humains.

QR21-Information. Les actions de l'utilisateur, d'ajouter un article à leur liste de lecture sont la base de la personnalisation du système de recommandation.

QR22-Assistance. La pré-visualisation de l'impact des actions de l'humain est une assistance qui a été fortement appréciée lors de l'étude. Elle a permis au groupe concerné d'être plus efficace dans le choix d'articles.

2.3.3 Lignes directrices pour tout type d'application

Amershi et al. [4] proposent 18 lignes directrices pour la conception des modalités d'interaction des applications d'Apprentissage Automatique Interactif. L'objectif est d'aider les concepteurs à éviter les expériences utilisateurs négatives dues à une interface mal conçue.
Afin de produire ces lignes directrices, les auteurs sont partis d'une étude de la littérature des 20 dernières années sur le sujet. Cette étude a fait ressortir 168 recommandations spécifiques, qui ont été regroupées, filtrées et raffinées pour arriver à 18 propositions. Ce processus de raffinage a consisté en une étude auprès de praticiens et d'experts en IHM, qui ont apprécié la pertinence de ces lignes directrices dans diverses applications d'Apprentissage Automatique Interactif disponibles dans le commerce.
Ces lignes directrices générales constituent une aide pertinente aux professionnels de l'IHM. On y retrouve par exemple le fait d'expliquer clairement les conséquences des actions de l'humain sur l'apprenant, ou inversement d'expliquer comment une décision a été prise par la machine.

QR22-Assistance. Les lignes directrices aident à mettre au point un accompagnement qui allège l'utilisateur.
Étant donné l'aspect généraliste de ces recommandations, nous ne pouvons pas répondre aux autres questions de recherche pour cette contribution.

3 Analyse et conclusion

3.1 Réponses aux questions de recherche

La table 1 synthétise les niveaux de réponses aux quatre questions élicitées section 1.4. Pour chacune, nous discutons des éléments importants ou originaux trouvés dans la littérature.

Contribution	QR11	QR12	QR21	QR22
Ramos et al. [24]	+	-	+	++
Honeycutt et al. [19]	++	++	-	-
Carney et al. [7]	+	+	++	+
Fails et Olsen Jr [16]	+	+	-	-
Berg et al. [5]	-	-	++	-
Flutura et al. [17]	+	+	+	+
Kessler et al. [21]	++	+	+	-
Akrour et al. [2]	+	-	-	-
Christiano et al. [11]	+	+	-	+
Holzinger et al. [18]	++	++	-	-
Schnabel et al. [26]	+	+	+	++
Amershi et al. [4]	N/A	N/A	N/A	++

TABLE 1 – Synthèse des niveaux de réponses aux différentes questions de recherche. L'échelle "-" à "++" indique le niveau de réponse à une question. N/A = Non Applicable.

QR11-Tâches. Les applications d'IML sont variées, par conséquent les tâches demandées à l'humain le sont aussi : enseigner, superviser, reconnaître une image, choisir parmi une liste de propositions... La colonne QR11 indique en quoi ces tâches sont propres au métier et non à l'apprentissage automatique.
De manière générale, peu de compétences en apprentissage sont requises dans les contributions présentées (+ ou ++).
Malgré tout, les contributions notées + demandent à l'humain de manipuler des concepts propres à l'apprentissage automatique, par exemple des étiquettes dans l'apprentissage supervisé [24, 7, 16, 17] et des séquences d'actions dans l'apprentissage par renforcement [2, 11].
Enfin, Ilastik [5] demande à l'humain une familiarité avec différents algorithmes de reconnaissance d'images. Cette tâche relève fortement du domaine de l'apprentissage.

QR12-Charge. La colonne QR12 donne une évaluation de la charge et de l'implication demandées à l'humain : elle est modérée (notée +) dans la moitié des applications. Les applications où l'humain est le moins surchargé sont celles où la tâche demandée à l'utilisateur est la plus simple : pointer les erreurs d'un apprenant [19] ou alimenter indirectement l'apprentissage [18].

QR21-Information. L'apprenant exploite de diverses manières les informations fournies par l'humain. La colonne QR21 indique le niveau de complexité de l'interven-

tion de l'humain. Les contributions notées - ne proposent qu'une seule manière pour l'humain d'interagir, comme par exemple l'avis sur les actions de l'apprenant dans [11].
À l'inverse, les contributions notées ++ proposent des interactions plus riches donnant plus de possibilités pour l'humain d'influer sur l'apprentissage machine. Il est par exemple possible dans [7] d'agir plus en profondeur dans le paramétrage de l'apprenant, ce qui nécessite plus de compétences en apprentissage de la part de l'humain. Certaines sources d'interaction implicites, comme le survol de la souris [26] ou la détection de l'attention [21] permettent également à l'humain d'agir plus indirectement sur l'apprenant.

QR22-Assistance. La colonne QR22 donne le niveau d'assistance qu'apporte l'application d'IML à l'humain : il est faible pour la majorité des applications. À l'exception des travaux de Honeycutt et al. [19] (assistance volontairement minimale) et de Holzinger et al. [18] (assistance non nécessaire), une meilleure assistance pourrait être profitable.

3.2 Positionnement et contributions

Pour terminer, nous discutons quelques principes particulièrement intéressants pour la conception d'interactions humain-IA en général. Ils constituent également pour nous des futures pistes de réflexion pour la composition logicielle opportuniste.

Utiliser des lignes directrices. Les lignes directrices proposées par Amershi et al. [4], bien que générales, permettent d'identifier des pistes pour faciliter le travail de l'humain. Par exemple, le fait de permettre à l'humain de pré-visualiser les conséquences de ses actions [26].

Entraîner l'apprenant avec des experts en apprentissage. Une approche également intéressante est d'observer des utilisateurs experts en apprentissage afin d'en tirer des recommandations, implantées par la suite sous forme de notifications [30]. Cela participe à former les humains inexpérimentés aux particularités d'une application avec un apprenant dans la boucle.

Opérer des transferts de connaissances. Il est aussi possible de permettre aux humains de partager leurs modèles, comme ce qui est présenté dans [5]. Récupérer un modèle existant répondant à une tâche proche de celle que l'on souhaite résoudre, puis le spécialiser (ce que proposent Carney et al. [7]) semble constituer un moyen intéressant de tirer parti d'un grand nombre d'utilisateurs. Ce principe pourrait permettre de prendre en compte l'apparition de composants jusque là inconnus dans l'environnement ambiant, ainsi que la personnalisation face à un nouvel utilisateur.

Prendre en compte les changements d'objectifs, de préférences de l'humain. Dans la majeure partie des contributions, l'humain interagit avec un apprenant automatique pour une tâche clairement définie et la possibilité qu'elle change avec le temps n'est pas envisagée.
Par exemple, dans les approches proposées dans [2] et [21], un changement d'objectif de l'humain entraînera des données possiblement contradictoires pour l'apprentissage, qui seront traitées comme des erreurs.
Dans les approches basées sur la construction de classifieurs d'images ou de texte [5, 7, 24, 16], l'humain doit construire un nouveau modèle si son objectif change, éventuellement en modifiant les données d'entraînement d'un modèle existant. Les solutions d'IML pourraient pourtant être sensibles à la dynamique pour continuer à travailler correctement en cas de changements.

Utiliser la participation implicite comme source d'apprentissage. La participation implicite [15], telle que la prise en compte des mouvements de la souris [21], peut enrichir l'apprentissage sans surcharger l'humain ou le contraindre à acquérir de nouvelles compétences (par exemple en pédagogie pour [24]).

Références

[1] M. Aamir and M. Bhusry. Recommendation system : state of the art approach. *International Journal of Computer Applications*, 120(12), 2015.

[2] R. Akrour, M. Schoenauer, M. Sebag, and J.-C. Souplet. Programming by feedback. In *Int. Conf. on Machine Learning*, volume 32, pages 1503–1511. JMLR. org, 2014.

[3] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza. Power to the people : The role of humans in interactive machine learning. *Ai Magazine*, 35(4) :105–120, 2014.

[4] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, and K. Inkpen. Guidelines for human-AI interaction. In *Proc. of the 2019 CHI conf. on human factors in computing systems*, pages 1–13, 2019.

[5] S. Berg, D. Kutra, T. Kroeger, C. N. Straehle, B. X. Kausler, C. Haubold, M. Schiegg, J. Ales, T. Beier, and M. Rudy. Ilastik : interactive machine learning for (bio) image analysis. *Nature Methods*, 16(12) : 1226–1232, 2019.

[6] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. OpenAI Gym. *arXiv preprint arXiv :1606.01540*, 2016.

[7] M. Carney, B. Webster, I. Alvarado, K. Phillips, N. Howell, J. Griffith, J. Jongejan, A. Pitaru, and A. Chen. Teachable machine : Approachable Web-based tool for exploring machine learning classification. In *Extended abstracts of the 2020 CHI Conf. on human factors in computing systems*, pages 1–8, 2020.

[8] C. Celemin and J. Ruiz-del Solar. An interactive framework for learning continuous actions policies based on corrective feedback. *Journal of Intelligent & Robotic Systems*, 95(1) :77–97, 2019.

[9] A. Chatzimparmpas, R. M. Martins, I. Jusufi, and A. Kerren. A survey of surveys on the use of visualization for interpreting machine learning models. *Information Visualization*, 19(3) :207–233, 2020.

[10] A. Chatzimparmpas, R. M. Martins, I. Jusufi, K. Kucher, F. Rossi, and A. Kerren. The state of the art in enhancing trust in machine learning models with the use of visualizations. In *Computer Graphics Forum*, volume 39, pages 713–756. Wiley Online Library, 2020.

[11] P. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. *arXiv preprint arXiv :1706.03741*, 2017.

[12] K. Delcourt, F. Adreit, J.-P. Arcangeli, K. Hacid, S. Trouilhet, and W. Younes. Automatic and Intelligent Composition of Pervasive Applications - Demonstration. In *19th IEEE Int. Conf. on Pervasive Computing and Communications (PerCom 2021)*, Kassel (virtual), Germany, March 2021.

[13] M. Dorigo, M. Birattari, and T. Stutzle. Ant colony optimization. *IEEE computational intelligence magazine*, 1(4) :28–39, 2006.

[14] J. J. Dudley and P. O. Kristensson. A review of user interface design for interactive machine learning. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 8(2) :1–37, 2018.

[15] C. Evers, R. Kniewel, K. Geihs, and L. Schmidt. The user in the loop : Enabling user participation for self-adaptive applications. *Future Generation Computer Systems*, 34 :110–123, May 2014.

[16] J. A. Fails and D. R. Olsen Jr. Interactive machine learning. In *Proceedings of the 8th Int. Conf. on Intelligent user interfaces*, pages 39–45, 2003.

[17] S. Flutura, A. Seiderer, I. Aslan, C.-T. Dang, R. Schwarz, D. Schiller, and E. André. Drinkwatch : A mobile wellbeing application based on interactive and cooperative machine learning. In *Proceedings of the 2018 Int. Conf. on Digital Health*, pages 65–74, 2018.

[18] A. Holzinger, M. Plass, M. Kickmeier-Rust, K. Holzinger, G. C. Crişan, C.-M. Pintea, and V. Palade. Interactive machine learning : experimental evidence for the human in the algorithmic loop. *Applied Intelligence*, 49(7) :2401–2414, 2019.

[19] D. Honeycutt, M. Nourani, and E. Ragan. Soliciting human-in-the-loop user feedback for interactive machine learning reduces user trust and impressions of model accuracy. In *Proceedings of the AAAI Conf. on Human Computation and Crowdsourcing*, volume 8, pages 63–72, 2020.

[20] L. Jiang, S. Liu, and C. Chen. Recent research advances on interactive machine learning. *Journal of Visualization*, 22(2) :401–417, 2019.

[21] T. A. Kessler Faulkner and A. Thomaz. Interactive Reinforcement Learning from Imperfect Teachers. In *Companion of the 2021 ACM/IEEE Int. Conf. on Human-Robot Interaction*, pages 577–579, 2021.

[22] M. Koussaifi. *Modélisation centrée utilisateur pour la configuration logicielle en environnement ambiant*. Thèse de doctorat, Université Paul Sabatier-Toulouse III, 2020.

[23] G. Li, R. Gomez, K. Nakamura, and B. He. Human-centered reinforcement learning : A survey. *IEEE Transactions on Human-Machine Systems*, 49(4) : 337–349, 2019.

[24] G. Ramos, C. Meek, P. Simard, J. Suh, and S. Ghorashi. Interactive machine teaching : a human-centered approach to building machine-learned models. *Human–Computer Interaction*, 35(5-6) :413–451, 2020.

[25] F. Sadri. Ambient intelligence : A survey. *ACM Computing Surveys*, 43(4) :1–66, 2011.

[26] T. Schnabel, S. Amershi, P. N. Bennett, P. Bailey, and T. Joachims. The Impact of More Transparent Interfaces on Behavior in Personalized Recommendation. In *Proceedings of the 43rd Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 991–1000, 2020.

[27] P. Y. Simard, S. Amershi, D. M. Chickering, A. E. Pelton, S. Ghorashi, C. Meek, G. Ramos, J. Suh, J. Verwey, and M. Wang. Machine teaching : A new paradigm for building machine learning systems. *arXiv preprint arXiv :1707.06742*, 2017.

[28] R. Sutton and A. Barto. *Reinforcement learning : An introduction*. MIT press, 2018.

[29] K. Van den Bos, R. Vermunt, and H. A. Wilke. The consistency rule and the voice effect : The influence of expectations on procedural fairness judgements and performance. *European Journal of Social Psychology*, 26(3) :411–428, 1996.

[30] E. Wall, S. Ghorashi, and G. Ramos. Using expert patterns in assisted interactive machine learning : A study in machine teaching. In *IFIP Conf. on Human-Computer Interaction*, pages 578–599. Springer, 2019.

[31] M. Yin, J. Wortman Vaughan, and H. Wallach. Understanding the effect of accuracy on trust in machine learning models. In *Proceedings of the 2019 chi conference on human factors in computing systems*, pages 1–12, 2019.

[32] W. Younes. *Un système multi-agent pour la composition logicielle opportuniste en environnement ambiant et dynamique*. These de doctorat, Université Paul Sabatier - Toulouse III, June 2021.

[33] R. Zhang, F. Torabi, G. Warnell, and P. Stone. Recent advances in leveraging human guidance for sequential decision-making tasks. *Autonomous Agents and Multi-Agent Systems*, 35(2) :1–39, 2021.

Le droit aux prises des incertitudes de l'intelligence artificielle

Pouzet. C
Université Jean Moulin Lyon III, CEE
clementinepouzet@orange.fr

Résumé

Face aux incertitudes inhérentes aux systèmes d'intelligence artificielle, se pose la question de l'encadrement juridique adéquat. Alors que ce dernier se fait principalement à travers des instruments non contraignants (soft law), l'Union travaille à l'édiction d'un règlement contraignant (hard law). Outre la problématique de la force normative, la finalité de tels instruments est interrogée. Tandis que, l'approche fondée sur les droits peut être prônée, le Conseil de l'Europe et l'Union optent pour une approche fondée sur les risques.

Mots-clés

Intelligence artificielle – incertitude – risque – droit européen – Union européenne – Conseil de l'Europe – droits de l'homme.

Abstract

Faced with the uncertainties inherent in artificial intelligence systems, the question arises as to the appropriate legal framework. While this is mainly done through non-binding instruments (soft law), the European Union is working on the enactment of a binding regulation (hard law). In addition to the problem of normative force, the purpose of such instruments is questioned. While the rights-based approach can be advocated, the Council of Europe and the Union opt for a risk-based approach.

Keywords

Artificial intelligence – uncertainty – risk – European law – European Union – Council of Europe – human rights.

Introduction

« La technologie n'est ni bonne ni mauvaise, pas plus qu'elle n'est neutre »[7]. L'absence de neutralité de l'intelligence artificielle est un des éléments concourant à l'incertitude qui entoure cette technologie. Afin de maitriser cette incertitude, le droit s'empare actuellement de l'encadrement des systèmes d'intelligence artificielle. Toutefois les travaux juridiques, bien que foisonnants, se heurtent à plusieurs obstacles.

Tout d'abord, l'incertitude, pouvant communément être entendue comme ce qui est incertain[24], imprécis[23], ou encore imprévisible[23], ne dispose pas de définition juridique et ne peut, en tant que telle, être aisément appréhendée par le droit.

La notion d'incertitude peut être rapprochée de ce qui est incertain, c'est-à-dire, au sens juridique, ce qui est « indéterminé et indéterminable »[9]. La notion d'intelligence artificielle semble, à l'heure actuelle, indéterminée à deux titres.

D'une part, la définition technique de cette technologie est incertaine et ne fait pas consensus au sein même de la communauté scientifique[16]. Cette incertitude rejaillit sur l'appréhension juridique de l'intelligence artificielle puisque cette dernière ne dispose pas encore de définition juridique communément admise et systématisée, ce qui rend son encadrement complexe. Les termes d'intelligence artificielle ou de systèmes d'intelligence artificielle sont parfois utilisés sans qu'une distinction entre les deux ne soit faite ou que l'assimilation des deux notions ne soit établie.

D'autre part, outre son caractère indéterminé, se pose la question de savoir si l'intelligence artificielle est ou non déterminable. Cette technologie évoluant constamment, il est loisible de se demander si une définition figée et précise de cette dernière est possible et pertinente. A ce caractère déterminable ou non de l'intelligence artificielle, peut être rattachée la notion d'(im)précision. En effet, si l'intelligence artificielle est indéterminable cette notion fera l'objet de définitions imprécises, ce qui aura des répercussions sur son encadrement juridique et *in fine* sur les utilisateurs, les individus. Si la définition de l'intelligence artificielle est imprécise son encadrement risque de manquer de clarté, précision, intelligibilité et prévisibilité. La sécurité juridique, recouvrant les notions précitées, sera entachée, au détriment des individus. S'opposent les tenants d'un encadrement non contraignant de l'intelligence artificielle au travers du droit dit « souple » ou « *soft law* » (recommandations, incitations non obligatoires) à ceux prônant un encadrement strict et contraignant de l'intelligence artificielle par le biais du droit dit « dur » ou « *hard law* »[19]. Ces éléments pris en considération, il est loisible de constater que la notion d'intelligence artificielle semble incertaine et mère d'incertitudes.

Enfin, l'intelligence artificielle paraît pouvoir être qualifiée d'imprévisible[9], et ce au regard des résultats qu'elle génère. De tels propos peuvent être illustrés par les systèmes de *machine learning* dont une perte de mainmise de la part de concepteurs est possible[13]. Parmi les exemples de résultats biaisés induits par des systèmes d'intelligence artificielle peuvent notamment être cités l'algorithme de recrutement d'Amazon[10], les algorithmes de reconnaissance faciale[11], les effets de bulles de filtres[5]...

La complexité des systèmes d'intelligence artificielle et les nombreuses incertitudes inhérentes à ces derniers impactent leur encadrement. Afin d'avoir un encadrement cohérent et efficace le niveau national a cédé le pas à l'échelon européen[19]. C'est pourquoi l'étude juridique de l'intelligence artificielle a lieu au sein de l'Union européenne et du Conseil de l'Europe. Ces deux organisations désirent être pionnières en matière de réglementation des systèmes d'intelligence artificielle[1][8]. A ce titre le niveau européen semble pertinent puisqu'à la fois le niveau national s'avère insuffisant et le niveau international inadéquat[19]. Les données, élément indispensable au fonctionnement des systèmes d'intelligence artificielle, sont transfrontières, un encadrement au seul niveau national semble donc peu pertinent[19]. De même, les valeurs et objectifs de l'Europe, de la Chine, des États-Unis ou de la Russie étant trop éloignées, un encadrement international de l'intelligence artificielle est peu probable voire utopique[19].

Par conséquent, pour les besoins de la démonstration, la définition retenue de l'intelligence artificielle sera celle proposée par la Commission européenne dans sa proposition de règlement sur l'intelligence artificielle préférant l'expression « systèmes d'intelligence artificielle »[2], tandis que l'incertitude sera rattachée à la notion de risque, tel que cela est proposé en droit européen[2].

Ainsi, l'encadrement des systèmes d'intelligence artificielle s'avère essentiel mais est confronté à de nombreux défis. La réglementation de ces systèmes doit prendre en compte les différents enjeux précités, à savoir celui de leur incertitude au niveau terminologique, technique et de leurs conséquences. En effet, les systèmes d'intelligence artificielle peuvent notamment être décriés pour leur potentiel d'atteinte aux droits de l'homme. Ladite incertitude génère également des problématiques relatives à l'opacité desdits systèmes et la défense de ses droits en cas de dommage. Cela est propice à générer une incertitude et une insécurité juridique chez l'utilisateur[16]. Nonobstant cela, l'enjeu économique issu des systèmes d'intelligence artificielle et la volonté de ne pas brider l'économie sont également à considérer lors de l'élaboration d'un cadre réglementaire des systèmes d'intelligence artificielle, avec pour objectif d'édicter l'encadrement le plus efficace et effectif.

Enfin, l'incertitude entourant lesdits systèmes aura un impact sur la manière plus ou moins souple de les encadrer. Il s'agit ici de l'enjeu relevant au droit optimal à adopter entre un encadrement juridique strict, par la *hard law*, et visant la protection des droits de l'homme *versus* un encadrement plus souple, par la *soft law*, permettant une innovation plus aisée. L'enjeu principal de la réponse juridique est de limiter les conséquences néfastes de cette incertitude. Il s'agit d'adopter un encadrement adapté, protecteur des droits de l'homme et de l'innovation tout en étant efficace et effectif.

Dès lors, comment le droit européen s'empare-t-il de l'incertitude propre aux systèmes d'intelligence artificielle pour en assurer un développement respectueux des droits de l'homme ?

Le droit européen s'est d'abord vu confronté à la difficulté de réguler et réglementer un objet incertain et méconnu, mais un encadrement émerge visant à s'accommoder et diminuer l'incertitude tout en préservant l'innovation (1). Dès lors, plusieurs approches et finalités sont poursuivies dont une approche fondée sur les risques émanant des systèmes d'intelligence artificielle ou encore sur les droits devant prioritairement être protégés (2).

1 L'émergence d'un encadrement juridique de plus en plus contraignant de l'intelligence artificielle

Face à l'absence de définition systématisée des systèmes d'intelligence artificielle au niveau technique, mais aussi juridique, le droit a dû faire montre d'adaptabilité, par le recours initial à la *soft law*. Le recours à la *soft law,* offre une certaine flexibilité et a permis d'appréhender au mieux les systèmes d'intelligence artificielle et leurs incertitudes (1.1). Mais face aux faiblesses d'un tel instrument et à la nécessité d'endiguer au maximum les effets potentiellement dommageables des systèmes d'intelligence artificielle, des travaux de *hard law* émergent (1.2).

1.1 Le recours initial à la *soft law* : palliatif à la difficile adaptation du droit européen à un objet incertain

Les systèmes d'intelligence artificielle présentent plusieurs incertitudes et interrogent sur plusieurs points, ce qui influe, par conséquent, sur leur encadrement juridique. En effet, au niveau technique ces derniers ne font pas consensus quant à ce qu'ils incluent en leur champ. Cette incertitude terminologique a des conséquences sur le volet juridique. Se pose, en effet, la question de savoir comment encadrer un objet dont la nature est non définie et débattue. Afin de pallier cette difficulté l'Union européenne, le Conseil de l'Europe et d'autres organisations[22] ont élaboré des instruments juridiques souples, c'est-à-dire non assortis de sanctions en cas de non-respect. Un tel phénomène s'explique à plusieurs titres.

Tout d'abord, comme énoncé précédemment les systèmes d'intelligence artificielle peuvent générer des atteintes aux droits des utilisateurs, aux droits de l'homme qu'il convient de limiter et de réparer. Le recours à un encadrement juridique s'avère ainsi pertinent.

Ensuite, l'encadrement des systèmes d'intelligence artificielle ne fait pour l'heure pas consensus à l'échelle mondiale. Aucune instance mondiale n'existe en matière d'intelligence artificielle et d'encadrement, régulation de ses systèmes, les pays ayant des

intérêts radicalement divergents en la matière. En outre, les valeurs mises en avant et entourant cette technologie diffèrent largement avec la promotion de systèmes respectant les droits de l'homme en Europe, la volonté de la Russie de développer un leadership en la matière et la promotion d'une technologie de plus en plus prégnante et invasive en Chine, le système de crédit social illustrant ce propos[21].

La combinaison de ces deux facteurs a conduit à l'adoption de normes de *soft law*, dont les avantages sont connus, et ont pu apparaître comme particulièrement adaptés aux systèmes d'intelligence artificielle : rapidité d'élaboration, adaptabilité et flexibilité de ces règles, facilement modifiables et donc plus à même de répondre aux incertitudes engendrées par l'intelligence artificielle. Le recours à la *soft law* a ainsi été privilégié, se traduisant par l'adoption de lignes directrices, ou par le recours à des méthodes d'autorégulation ou encore de corégulation[19]. A titre d'exemple peuvent être citées les Lignes directrices en matière d'éthique pour une IA digne de confiance adoptées par le Conseil de l'Europe en 2019 qui visent à établir un cadre éthique non contraignant des systèmes d'intelligence artificielle.

Cela ne signifie pas que le droit contraignant (ou *hard law*), soit totalement absent. Certains textes juridiques obligatoires existent dans des domaines circonscrits. Tel est le cas en matière de protection des données avec le RGPD[20] par exemple ou encore la Convention 108[4]. Les dispositions juridiques nationales en matière de responsabilité s'appliquent également[19]. La combinaison *soft law/hard law* permet ainsi de définir un cadre en vue de limiter les incertitudes issues des systèmes d'intelligence artificielle.

Toutefois, le recours à la *soft law,* bien que palliatif nécessaire à l'absence de *hard law* spécifiquement consacrée aux systèmes d'intelligence artificielle, participe aussi de l'incertitude de cette technologie dans la mesure où règne une certaine insécurité juridique. En effet, la *soft law* a longtemps été critiquée pour son incertitude, notamment au regard de l'impératif de sécurité juridique[3]. Cela pose à nouveau la question de l'efficacité des moyens juridiques à disposition et ceux adoptés. De même, aux limites inhérentes à la *soft law* s'ajoutent celles relatives au fait que la *hard law* existante ne s'avère pas toujours adaptée aux systèmes d'intelligence artificielle. Cela engendre des incertitudes notamment dans le chef de l'utilisateur ou de l'individu concerné par les systèmes et leurs résultats. A titre d'exemple, il peut être compliqué de déclencher une action en responsabilité civile en vue d'obtenir réparation d'un dommage causé par un système d'intelligence artificielle du fait de la difficulté de prouver ce dommage (en lien avec l'opacité des systèmes d'intelligence artificielle) [13], ou de l'imputation de la responsabilité. La complexité de cette technologie rend ainsi particulièrement difficile la mise en œuvre du régime juridique traditionnel de responsabilité.

Enfin, les instruments de *soft law* indiquent des principes à suivre lors de la conception et de l'activation de systèmes d'intelligence artificielle. Ont ainsi été mis en avant les principes de bienveillance, non-malfaisance, justice ou encore dignité[15]. Toutefois, une fois encore la *soft law* et le droit en général présentent ici des limites. Il est en effet difficile, voire impossible, d'implémenter de tels principes dans un algorithme[19], ce qui par voie de conséquence amoindrit l'effectivité des instruments de *soft law*.

Ainsi, face aux critiques à l'égard de ce type d'encadrement alliant *soft law* et *hard law* et aux risques des systèmes d'intelligence artificielle, la *hard law* semble devoir s'imposer ou du moins édicter un cadre minimal. Un tel encadrement est en discussion sur le continent européen.

1.2 L'adoption progressive d'un encadrement juridique des systèmes d'intelligence artificielle fondé sur de la *hard law*

Le recours à la *hard law* pour encadrer les systèmes d'intelligence artificielle est récent, et ce, pour plusieurs raisons. Parmi celles-ci, l'impératif économique a motivé l'absence de règles juridiques contraignantes propres aux systèmes d'intelligence artificielle. Cela justifie d'ailleurs aujourd'hui les finalités de l'encadrement proposé par la Commission européenne. Le potentiel d'essor économique et d'innovation des systèmes d'intelligence artificielle justifie les réticences d'un recours à la *hard law* considérée comme pouvant brider l'économie[19] en limitant les possibilités d'innovations par l'adoption de règles trop restrictives. Une telle raison doit être mise en parallèle avec la crainte d'une perte de compétitivité et d'une concurrence normative[19]. En effet, réglementer les systèmes d'intelligence artificielle pourrait inciter les opérateurs économiques à s'installer dans des États n'ayant pas encadré ces systèmes afin de ne pas être soumis à un cadre trop contraignant[19].

Néanmoins, face aux potentielles dérives de l'intelligence artificielle et aux lacunes de la *soft law*, un encadrement juridique contraignant a, tout de même, été jugé nécessaire. L'Union européenne travaille actuellement à l'édiction d'un cadre juridique contraignant de l'intelligence artificielle. En effet, la Commission a émis le 21 avril 2021 sa proposition de règlement sur l'intelligence artificielle[2]. Cette proposition présente un encadrement *ex ante*, c'est-à-dire imposant des règles en amont de l'activation du système d'intelligence artificielle et de l'éventuelle réalisation d'un dommage. L'objectif est d'éviter l'apparition de tout dommage issu d'un système d'intelligence artificielle. Un tel encadrement *ex ante* permet de gérer l'incertitude de ces systèmes via la notion de risque ; risque à déterminer et prévenir en amont de leur fonctionnement. En effet, lorsqu'un événement est prévisible alors cela semble plus aisé de légiférer efficacement. L'enjeu réside donc ici dans le fait d'envisager et d'encadrer ce qui parait, au premier abord, difficilement prévisible au regard des connaissances actuelles. Ainsi, il est intéressant de noter que la notion d'incertitude n'est pas reprise telle qu'elle dans la proposition de règlement mais semble bel et bien être transposée en la notion de « risque ». Le risque semble principalement envisagé comme le risque d'atteinte aux droits des individus faisant l'objet des systèmes d'intelligence artificielle et des droits de l'homme. Il s'agit de parer le risque

de biais dès la confection du système et jusqu'au résultat. Le risque doit ainsi être évité par tout maillon de la chaine de conception d'un système ainsi que par l'utilisateur.

En parallèle de l'Union européenne, le Conseil de l'Europe travaille également à un encadrement des systèmes d'intelligence artificielle prenant en compte la notion de risque mais la nature de l'encadrement n'est pas encore déterminée. Une étude de faisabilité a été édictée le 17 décembre 2020 dans laquelle le recours à des instruments contraignants est proposé, tout comme celui à des instruments non-contraignants dans la continuité de ce qui est fait aujourd'hui.

L'Union européenne et le Conseil de l'Europe travaillent à un encadrement des systèmes d'intelligence artificielle fondé sur les risques engendrés par ces derniers. Toutefois, d'autres finalités peuvent être poursuivies et proposées telles que la finalité de protection et préservation des droits de l'homme et des individus concernés par des systèmes d'intelligence artificielle.

2 L'émergence de diverses finalités, fondement d'encadrements juridiques des systèmes d'intelligence artificielle

L'Union européenne et le Conseil de l'Europe travaillent actuellement à l'édiction de règles, fondées sur le risque engendré par les systèmes d'intelligence artificielle et visant à l'encadrer dans le but de limiter ses incertitudes et ses potentielles néfastes conséquences (2.1). En parallèle de ces travaux, des propositions d'encadrements alternatifs fondés sur la protection des droits émergent (2.2).

2.1 L'adoption progressive d'un encadrement fondé sur les risques

La proposition de règlement sur l'intelligence artificielle de la Commission européenne détermine un encadrement de ladite technologie fondée sur les risques. Cela signifie, en d'autres termes, que plus un système d'intelligence artificielle présentera des risques et des incertitudes quant à ses résultats, plus son encadrement sera strict. *A contrario*, les systèmes qui ne présentent que peu de risques seront plus légèrement encadrés. Un tel constat étant fait, se pose la question de savoir ce qu'est le risque, la manière dont il est défini et appréhendé par le droit européen.

La proposition de règlement sur l'intelligence artificielle propose une échelle des risques. Ainsi, quatre niveaux peuvent être déterminés : les systèmes interdits tels que ceux de crédit social, les systèmes présentant de hauts risques (étant au cœur de cette proposition de règlement), les systèmes à faibles risques, tels que les chatbots, et les systèmes à risques minimum, tels que les systèmes dans les jeux vidéo ou les filtres de spam.

Afin de déterminer si un système d'intelligence artificielle présente un haut risque sont pris en compte ses conséquences sur « la santé, la sécurité et la sécurité ou les droits fondamentaux des personnes »[2]. Plus précisément deux conditions cumulatives permettent de déterminer si un système d'intelligence artificielle est à haut risque ou non. Ainsi, le système d'intelligence artificielle étudié doit « être utilisé comme composant de sécurité » du produit fini et le produit fini ou le système d'intelligence artificielle doivent faire l'objet d'un contrôle par un tiers avant qu'il ne soit commercialisé. Il y a donc un critère tenant à la sécurité et un au contrôle du produit fini.

Certains systèmes sont par nature à haut risque tels que ceux utilisés dans le domaine de la justice, de la police comme les algorithmes de justice préventive, dans le domaine de l'éducation etc[2].

Ainsi, par l'interdiction de certains systèmes ou au contraire leur développement, le législateur s'empare de l'incertitude et la diminue en en interdisant certains ou l'accepte quand ils ne présentent pas de risques jugés trop importants.

L'objectif d'un encadrement fondé sur le haut risque semble multiple. Tout d'abord, l'idée est de permettre un développement des systèmes d'intelligence artificielle. En effet, seuls certains systèmes sont interdits, les autres peuvent se développer sous certaines conditions. Par ailleurs, un tel encadrement vise à obtenir la confiance des utilisateurs de tels systèmes d'intelligence artificielle, ce qui aura pour effet de maintenir l'innovation et l'économie découlant de cette technologie.

La notion de risque est également présente dans les travaux du Conseil de l'Europe. Bien que le risque soit également pris en compte, le prisme adopté par le Conseil de l'Europe diffère de celui de l'Union et du fait même de leur objet. En effet, le Conseil de l'Europe, ayant vocation à protéger les droits de l'homme, adopte un prisme tourné vers leur préservation lors du fonctionnement de systèmes d'intelligence artificielle. L'Union européenne, quant à elle, adopte un angle plus économique en vue du développement de l'intelligence artificielle et de son innovation[14].

Enfin, il est intéressant de noter que le recours à la notion de risque semble récent. Il n'a pas toujours été fait référence à ce dernier mais aussi aux notions d'incidences, de menaces, de défis ou encore d'inquiétudes face aux systèmes d'intelligence artificielle.

Les travaux relatifs à l'encadrement des systèmes d'intelligence artificielle sont encore à leurs débuts et d'autres voies se dessinent dans la recherche d'un encadrement juridique optimal des systèmes d'intelligence artificielle.

2.2 L'émergence de propositions d'encadrements fondés sur les droits

D'autres positions quant à l'encadrement des systèmes d'intelligence artificielle voient le jour et visent principalement une approche fondée sur les droits.

Ainsi, peut être mise en avant la nécessité de s'inspirer de matières confrontées à des difficultés similaires, telles que le droit de l'environnement. En effet, cette matière génère des incertitudes et des inconnus et comporte des enjeux économiques similaires, à savoir, ne pas brider l'innovation. Dès lors, la *soft law* a été privilégiée aux prémices du droit de l'environnement, et notamment du droit à un environnement sain. Peuvent s'appliquer au fonctionnement des systèmes d'intelligence artificielle, tout comme à la matière environnementale, les principes de précaution et de prévention. De même, le recours aux études/analyses/évaluations d'impact dans le cadre de la gestion des risques issus de ces matières peut aussi avoir lieu à l'égard des systèmes d'intelligence artificielle.

Également en lien avec le droit de l'environnement, et notamment le droit à un environnement sain, pourrait être étendue la théorie du droit des générations futures au numérique. Selon la théorie des droits transgénérationnels, l'humain est envisagé comme étant l'être actuel mais également, pour certains, l'être en devenir. Ainsi, les générations actuelles auraient des droits, mais également des devoirs fondamentaux envers les générations futures. En effet, une relation asymétrique entre les individus d'aujourd'hui et ceux de demain existe. Cela impliquerait concrètement, selon la vision d'Hans Jonas, de préserver « un destin authentiquement humain »[12] pour l'humanité, et ce, à l'encontre de potentielles dérives des systèmes d'intelligence artificielle. Il s'agit de préserver la capacité de penser et le savoir-faire des Hommes. Une telle vision s'oppose aux thèses transhumanistes. Ces derniers prônent, par exemple, le droit à l'innovation, au progrès technique mais aussi à l'augmentation de soi.

Les droits transgénérationnels pourraient être édictés selon deux manières : élargir des droits préexistants, ou créer de nouveaux droits comme cela a été le cas au Chili avec les neurodroits et le droit à l'identité humaine[12]. Il s'agirait par exemple d'étendre le droit à la vie à celui d'avoir un destin humain[12].

De plus, et en approfondissant le parallèle avec le droit à un environnement sain, mais également le développement durable, pourrait être mise en avant la conceptualisation d'une intelligence artificielle saine, de systèmes d'intelligence artificielle sains.

Il s'agirait pour les systèmes d'intelligence artificielle de satisfaire certains impératifs tels que le respect des droits de l'homme, le respect de ce qui caractérise l'identité même de l'être humain, la vocation à augmenter le bien-être des individus (ou à ne pas le compromettre), le respect du modèle démocratique, de l'état de droit, de l'ordre public et la durabilité.

Enfin, l'édiction d'une quatrième génération de droits de l'homme dédiée aux droits du numérique est également envisagée afin d'assurer au mieux la protection des individus et de leurs droits. Une telle idée semble pouvoir rejoindre l'illustration du Chili présentée au titre des droits transgénérationnels.

En tout état de cause, le droit semble devoir s'adapter aux systèmes d'intelligence artificielle et, pour ce faire, pouvoir s'inspirer d'autres domaines présentant des enjeux similaires. Ainsi, il est souhaité de passer d'un droit de réaction à un droit d'anticipation. L'enjeu majeur est celui de l'effectivité du droit adopté, de sa bonne application et de son caractère réparateur en cas de préjudice.

Cela pose notamment la question de la responsabilité en cas de dommage lié au fonctionnement d'un système d'intelligence artificielle, interrogation d'actualité et encore en débat.

3 Références

[1] Commission européenne, « Livre Blanc Intelligence artificielle Une approche européenne axée sur l'excellence et la confiance », COM (2020) 65 final, Bruxelles, 19 février 2020, 30 p.

[2] Commission européenne, « Proposal for a regulation of the european parliament and of the council laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union legislative acts », COM(2021) 206 final, Brussels, 21 avril 2021, 107 p.

[3] Conseil d'État, « Le droit souple », Étude annuelle, *La Documentation française*, 2013.

[4] Conseil de l'Europe, « Convention pour la protection des personnes à l'égard du traitement automatisé des données à caractère personnel », Strasbourg, 28 janvier 1981, *S.T.E.*, n°108, 9 p.

[5] Conseil de l'Europe, Ad Hoc Committee On Artificial Intelligence, « Vers une régulation des systèmes d'IA », DGI (2020) 16, Décembre 2020, 203 p.

[6] Conseil de l'Europe, Ad Hoc Committee On Artificial Intelligence, « Feasability Study », CAHAI (2020)23, Strasbourg, 17 décembre 2020, 56 p.

[7] Conseil de l'Europe, Commission de la culture, de la science, de l'éducation et des médias, « Rapport La convergence technologique, l'intelligence artificielle et les droits de l'homme », Doc. 14288, Strasbourg, 10 avril 2017, 19 p.

[8] Conseil de l'Europe, Direction générale Droits de l'homme

et État de droit, Comité consultatif de la Convention pour la protection des personnes à l'égard du traitement automatisé des données à caractère personnel, Convention 108, « Rapport sur l'intelligence artificielle. Intelligence artificielle et protection des données : enjeux et solutions possibles », T-PD(2018)09Rev, Strasbourg, 3 décembre 2018.

[9] G. CORNU, *Vocabulaire juridique,* Jouve, 13è ed., PUF, janvier 2020, 1091 p.
[10] J. DASTIN, « Amazon scraps secret AI recruiting tool that showed bias against women », *Reuters*, 10 octobre 2018, [reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool- that-showed-bias-against-women-idUSKCN1MK08G - _blank], consulté le 9 mars 2020.

[11] FRA, « Facial recognition technology: fundamental rights considerations in the context of law enforcement», European union agency for fundamental rights, 2020, 34 p.
[12] Intervention d'E. GAILLARD, « La dimension environnementale de l'IA au regard du droit des générations futures», Colloque organisé dans le cadre de la Chaire Normandie pour la paix et, du Pôle Risques Qualité et Environnement Durable de la MRSH (Université de CAEN), en ligne – Facebook, 8 octobre 2021.

[13] GROZDANOVSKI L., « In search of effectiveness and fairness in proving algorithmic discrimination in EU law », *Common Market Law Review*, Janvier 2021, n°58, pp. 99-136.
[14] Intervention de F. JAULT-SESEK et C. POUZET, « Regards croisés sur l'intelligence artificielle : le Conseil de l'Europe et l'UE », Journée Patrick Daillier « Stratégies juridiques européennes sur la scène internationale : entre projection et négociation », CEDIN Université Paris Nanterre, Webinaire, 12 mars 2021.

[15] Y. MENECEUR, *L'intelligence artificielle en procès Plaidoyer pour une réglementation internationale et européenne,* A. Garapon (Préface), J. Kleijssen (Postface), Bruylant Macro Droit Micro Droit, 2020, 434p.

[16] S. MERABET, *Vers un droit de l'intelligence artificielle*, Thèse de doctorat, Université Jean Moulin Lyon 3, 2018, 556 p.
[17] Parlement européen, « Résolution du Parlement européen du 16 février 2017 contenant des recommandations à la Commission concernant des règles de droit civil sur la robotique », 16 février 2017.

[18] Parlement européen, « Résolution du Parlement européen du 12 février 2019 sur une politique industrielle européenne globale sur l'intelligence artificielle et la robotique» *(2018/2088(INI)),* 12 février 2019.

[19] C. POUZET, *Les modes de régulation de l'intelligence artificielle par le droit européen : entre droit souple et droit dur,* Mémoire de Master 2 Droit européen des droits de l'homme / sous la direction de Gaëlle Marti, Professeur, Lyon, Équipe de droit international, européen et comparé, 2020, 103 p.

[20] Règlement (UE) 2016/679 du parlement européen et du conseil du 27 avril 2016 relatif à la protection des personnes physiques à l'égard du traitement des données à caractère personnel et à la libre circulation de ces données, et abrogeant la directive 95/46/CE (règlement général sur la protection des données).

[21] STEFAN V. "L'intelligence artificielle et son impact sur les jeunes », Centre européen de la jeunesse, Conseil de l'Europe, Rapport du séminaire du 4-6 décembre 2019, 138 p.
[22] UNESCO ou encore OCDE

[23] https://www.cnrtl.fr/definition/incertitude.
[24]https://www.larousse.fr/dictionnaires/francais/incertitude/42222.

Session "IA & décision"

Futurs possibles d'un système d'acteurs : formalisation et génération automatique de scénarios

C. Blanchard[1], C.Saurel[1], C.Tessier[1]

[1] ONERA/DTIS, Université de Toulouse, France

camille.blanchard@onera.fr

Résumé

L'étude des futurs possibles d'un système d'acteurs par la génération de scénarios est un moyen d'anticipation et d'aide à la décision pour les organisations. Les méthodes existantes, le plus souvent participatives, produisent en général un nombre restreint de scénarios. Une génération systématique et automatisée de scénarios permettrait à la fois d'élargir l'ensemble des résultats produits et de limiter les biais inhérents à la réflexion des participants. Nous présentons ici un modèle formel permettant de définir un système et ses composantes (acteurs, principes moraux, variables...) et de générer et d'analyser des scénarios, en particulier lorsque des conflits se produisent pour un acteur (conflit moral) ou entre plusieurs acteurs (conflit logique).

Mots-clés

Aide à la décision, modélisation formelle des connaissances et du raisonnement, principes moraux, scénarios prospectifs, avenir du transport aérien

Abstract

Scenarios-based future studies on a system of actors are anticipatory and decision aid means for organizations. Existing methods, mostly participatory ones, generally produce a limited number of scenarios. A systematic and automated generation of scenarios could both enlarge the set of the results and limit the biases due to participants thoughts. This paper focuses on a formal model aiming at defining a system and its components (actors, moral principles, variables, etc.). This allows scenarios to be generated and analysed especially when conflicts occur in an actor (moral conflict) or between several actors (logical conflict).

Keywords

Decision aid, knowledge and reasoning formal modelling, moral principles, prospective scenarios, future of aviation

1 Introduction

Un système, qu'il soit industriel, économique, qu'il représente une institution ou une population, est l'objet d'enjeux complexes et les acteurs qui le composent rencontrent de nombreuses incertitudes qui peuvent freiner leurs prises de décision. La compréhension d'un système et l'exploration de ses futurs possibles sont essentielles pour envisager les changements qui s'imposent.

Pour limiter les biais liés aux calculs de probabilités et accorder une plus grande importance aux changements radicaux possibles dans l'évolution d'un système, les méthodes de scénarios sont les plus indiquées.

Dans l'abondante littérature de ce domaine (voir section 2.2), on retrouve en majorité des méthodes s'appuyant sur des modèles quantitatifs ou sur des analyses qualitatives en groupe de travail. Cependant, la formalisation des différentes composantes de ces méthodes pourrait permettre de faciliter la génération d'un nombre plus important de scénarios : en effet, le nombre de scénarios produits est limité par les biais inhérents aux domaines d'intérêt des participants aux réflexions, par leurs opinions ainsi que par la difficulté de traiter manuellement un problème très combinatoire.

L'objectif de cet article est donc de proposer une méthodologie et des outils formels génériques de construction et d'analyse de scénarios afin de systématiser la réflexion préalable à la prise de décision au sein d'un système d'acteurs, notamment par la prise en compte de principes moraux.

Après avoir donné, dans la section 2, un aperçu des méthodes de scénarios et des définitions existantes pour la notion de système, nous proposons un modèle formel permettant de générer des scénarios dans la section 3. Dans un premier temps nous définirons la notion de système d'acteurs puis nous formaliserons les éléments dynamiques de ce système. Nous présenterons ensuite la formalisation de la définition de scénarios pour permettre leur génération systématique. La mise en œuvre de ce modèle est ensuite présentée en section 4.

2 État de l'art

K. Muiderman [16] classe les approches pour envisager les différents futurs d'un système en quatre catégories [1] : la prédiction, la génération de scénarios, les approches « expérimentales » et enfin les approches « critiques ». Les approches « expérimentales » sont fondées sur une vision du futur collective sur l'expérience et la création ; elles ne revendiquent aucune méthodologie formelle. Les approches « critiques » interrogent les conséquences sur le présent des études concernant le futur, et notamment les implications politiques potentielles ; elles ne concernent pas la généra-

1. [15] précise que ces catégories peuvent se recouper.

tion de scénarios. Ces deux approches ne feront donc pas l'objet de développement dans cet article.

2.1 La prédiction

Cette approche considère que le futur est en partie connu. On utilise dans ce cas des outils de planification ou des modèles pour déterminer les probabilités d'un certain nombre de futurs jugés intéressants. Ces probabilités reposent notamment sur une analyse du passé.

La planification classique se retrouve majoritairement dans cette catégorie. Elle cherche notamment à optimiser le chemin permettant d'atteindre un objectif donné, le plus souvent en minimisant les risques. Elle ne vise pas en général à élargir la réflexion pour considérer les changements.

2.2 Méthodes de scénarios

Cette approche est davantage fondée sur l'exploration et le fait de pouvoir répondre à l'incertitude qu'est le futur en se préparant à différentes éventualités. On explore ces futurs possibles souvent grâce à des méthodes participatives mais aussi par des modèles quantitatifs. Les méthodes de génération de scénarios [1] constituent la majorité des techniques adoptées.

«Un scénario est un outil , utilisé par la réflexion prospective, pour appréhender le futur. Il décrit une image d'une société dans un espace géographique et dans un avenir donnés et des chemins liant l'état actuel de la société à celui décrit par l'image.» [4]

Définir la notion de scénario n'est pas aisé au vu de l'abondance des méthodes et interprétations existantes. Les nombreuses tentatives de définitions sont à l'origine de l'utilisation du terme « chaos méthodologique » dans la littérature de ce domaine [18]. Toutefois, [19] en propose une définition qui correspond principalement au point de vue proposé par l'école "*Intuitive Logic*" : un scénario est orienté vers le futur sur un sujet global. Il comprend une description rédigée, est possible voire plausible et fait partie d'un ensemble de scénarios différents générés systématiquement.

Les trois méthodes de scénarios principales sont :

- l'école de l'*Intuitive Logic* ;
- l'école "*Probabilistic Modified Trend*" ;
- l'école française de La Prospective.

2.2.1 Les méthodes américaines

La méthode *Intuitive Logic* repose essentiellement sur des travaux participatifs en groupe de travail et une analyse qualitative du système. Un groupe de travail est composé des membres d'une organisation à l'initiative de l'étude et d'une équipe d'animation qualifiée. On y retrouve parfois également des participants experts extérieurs à cette organisation. Son utilisation dans le milieu industriel a été plébiscitée notamment par le groupe Shell [22].

La méthode *Probabilistic Modified Trend* est davantage fondée sur des modèles quantitatifs. L'extrapolation des tendances est utilisée puis modifiée par l'ajout de facteurs qualitatifs et d'éléments de rupture moins probables pour enrichir l'analyse [12].

On retrouve dans ces deux méthodes l'utilisation le plus souvent implicite de variables clés sur lesquelles se fondent les scénarios, mais pas de délimitation ni de définition du système d'intérêt.

2.2.2 La Prospective

Il s'agit d'une démarche pluridisciplinaire de discussion pour dégager les enjeux majeurs d'un système (un secteur d'activité par exemple). C'est aussi une stratégie proactive grâce à laquelle un acteur peut envisager différents futurs possibles qui mèneront à un objectif préalablement défini (voir par exemple [17]).
L'école de la Prospective [2], première méthode de génération de scénarios française, propose la définition suivante de système : *« Un système est un ensemble d'éléments en interaction dynamique, organisés en fonction d'un but »* [10]. La représentation de ce système passe par la détermination de variables, d'acteurs et d'objectifs. Les éléments du système à étudier en priorité sont qualifiés de variables clés. Les acteurs sont *« ceux qui jouent un rôle important dans le système par l'intermédiaire des variables qui caractérisent leurs projets et qu'ils contrôlent plus ou moins »* [11]. On peut donc leur attribuer des stratégies d'action sur les variables, en particulier les variables clés qui sont le reflet de leurs objectifs, et ainsi mettre en lumière des jeux de pouvoir au sein du système. À cette approche sont associés des outils tels que MICMAC (MultiplIcation Matricielle Appliquée à un classement) ou MACTOR (Matrice des alliances, Conflits, Tactiques et Objectifs entre acteurs et Recommandations associées) [2] pour accompagner un acteur dans sa démarche d'analyse du système.

L'analyse morphologique [14] permet de faire une analyse détaillée du système en le décomposant en dimensions. Contrairement aux autres méthodes de génération de scénarios, elle permet de prendre en compte plus facilement des éléments de rupture. Proposée à l'origine par le physicien Fritz Zwicky (1898-1974), son objectif est d'accorder une attention particulière à la formulation du problème : définition des limites du système et des questions auxquelles on veut répondre. Ce problème est exprimé en paramètres ou variables. Certaines parties du système sont ainsi précisées avec des variables plus détaillées. Des variables internes et des variables externes au système sont aussi distinguées.

Le recensement des différents éléments est fait lors de sessions de travail réalisé « à la main », en groupes composés de membres de l'organisation à l'origine de l'étude, d'animateurs formés à la prospective et parfois d'experts du domaine. Bien que clairement structurée, la méthode nécessite de limiter le nombre de variables prises en compte. En effet, un trop grand nombre d'acteurs ou de variables rend le travail de groupe long, complexe et fastidieux.

2. Ces outils, qui font partie de la suite Scenaring Tools, ont été développés par le LIRSA-CNAM (Laboratoire Interdisciplinaire de Recherches en Sciences de l'Action) anciennement LIPSOR.

2.3 Critères de définition d'une méthode de génération de scénarios

La typologie proposée par [6] présente des critères permettant de classifier les différentes méthodes fondées sur la génération de scénarios. Nous retenons ci-dessous uniquement les critères que nous jugeons pertinents dans le cadre de notre travail. Cela nous permettra de préciser la nature des scénarios que nous voulons générer.

- le critère *Value/Reality* : évalue la « désirabilité » des scénarios :
  - scénario descriptif : chemin conduisant à un futur possible non fixé au préalable (hypothétique si exploratoire ; plausible si la notion de probabilité entre en jeu) ;
  - scénario normatif : chemin permettant d'atteindre un objectif spécifique (actif s'il simule une stratégie d'actions ou passif s'il permet d'observer le déroulement potentiel d'un processus ou l'évolution de normes.
- le point de départ du scénario [9] :
  - *forward-casting* lorsque le point de départ est le présent (*likely-futures* : scénarios tendanciels ou *what-if scenario* : scénarios exploratoires) ;
  - *backcasting* lorsque le raisonnement est abductif (à partir d'un idéal ou d'une situation redoutée).
- l'horizon temporel des scénarios ;
- la façon de prendre en compte le temps : temps continu ou temps discret ;
- l'échelle des variables : internes ou externes au système ;
- le nombre de scénarios : inférieur ou supérieur à deux ;
- les participants à l'étude : les membres de l'organisation à l'initiative de l'étude, les actionnaires et décideurs (appelés dans la suite les « utilisateurs ») ;
- le contrôle de l'organisation à l'initiative de l'étude sur son environnement (interne ou externe au système, actrice ou spectatrice, etc.). Ce critère doit notamment permettre de délimiter le système d'étude.

2.4 Positionnement de notre approche

On travaillera dans le cas de notre modèle sur des scénarios *descriptifs*, *hypothétiques* pour limiter les biais venant de l'utilisation de probabilités. Cependant, l'objectif d'aide à la décision pourra entraîner la simulation d'actions précises ce qui placera le travail dans une perspective *normative* et *active*. La génération de scénarios exploratoires nécessite de se positionner dans une logique de *forward-casting* mais ne doit pas exclure de travailler plus en détail sur un futur choisi par avance.

Concernant l'horizon temporel de nos scénarios, ils seront limités par des critères d'arrêt définis dans la section 3.5. Le temps ne sera pas modélisé explicitement, les scénarios étant construits comme une suite d'états et d'événements formant l'histoire d'un futur possible.

Enfin, si dans la typologie de [6] la distinction est faite entre un nombre de scénarios générés inférieur ou supérieur à deux, nous ferons ici la distinction entre la majorité des méthodes qui aboutissent à un nombre de scénarios construits « à la main », compris entre quatre et six et la génération « massive » de scénarios qui a été proposée par [7]. Comme dans [7] nous souhaitons générer un grand nombre de scénarios. Toutefois, nous nous démarquons de ce travail par le fait que la nature et l'analyse de ces scénarios ne seront pas quantitatives.

Remarque
Considérer un système d'acteurs pourrait appeler une modélisation par une approche multiagent [20]. Or dans notre cadre, la génération de scénarios se distingue de la résolution de problème multiagent par les points suivants :

- l'absence d'interactions entre acteurs ;
- l'absence d'intentions et de buts pour les acteurs ;
- l'absence de résolutions des conflits : nous considérons en effet qu'il est plus intéressant pour l'utilisateur de relever et caractériser les conflits que de les résoudre (voir section 3.4).

3 Modélisation

3.1 Objectifs

L'objectif de ce travail consiste à proposer une méthodologie et un outil générique de construction et d'analyse de scénarios afin de systématiser la réflexion préalable à la prise de décision d'un acteur du système, utilisateur de l'outil.

L'utilisation d'un langage formel mathématique permet d'exprimer les connaissances requises en limitant certaines ambiguïtés, de systématiser la génération de scénarios et de les calculer automatiquement.

Un système d'acteurs est défini à partir des différentes composantes utilisées en prospective, à savoir les acteurs et les variables. En outre sont ajoutés les principes moraux [3] auxquels adhèrent les acteurs, ce qui n'est généralement pas proposé avec les méthodes de scénarios existantes.

Dans la suite, nous illustrons la démarche grâce à un exemple fondé sur des questionnements liés au secteur du transport aérien.

3.2 Définition du système

Définition 1 *(*Principe - *Ensemble* $\mathbf{\Pi}$*)* $\mathbf{\Pi}$ *est un ensemble d'éléments* π *appelés principes.*

Exemple

$$\mathbf{\Pi} = \{CreationRichesse, SatisfactionPopulation, ProtectionEnvironnement\}$$

3. « *Les valeurs ont un rôle important en tant qu'idéaux motivant l'orientation des mesures politiques et des normes juridiques. Alors que l'ensemble des valeurs [...] inspirent des comportements souhaitables et constituent les fondements des principes, les principes quant à eux explicitent les valeurs de manière plus concrète, de façon à faciliter l'application de ces dernières dans les déclarations et actions politiques.* » [21]

Définition 2 *(*Variable - *Ensemble $\mathcal{V}$) $\mathcal{V}$ est un ensemble d'éléments v appelés variables prenant chacune leurs valeurs dans un ensemble discret noté $\mathcal{W}_v$. Ces valeurs sont instanciées dans l'état courant du système.*
On note $\mathcal{W}_\mathcal{V} = \bigcup_{v \in \mathcal{V}} \mathcal{W}_v$.

On fait ici les hypothèses simplificatrices suivantes :
- il n'y a pas de variable sur laquelle aucun acteur ne peut influer ;
- les variables sont indépendantes entre elles : pour que la valeur d'une variable change, il faut l'action directe d'un acteur.

Exemple

$\mathcal{V} = \{OffreAvion, TaxeGouv, OffreKero\}$
Avec
$\mathcal{W}_{OffreAvion} = \{Faible, Stable, Elevee\}$
$\mathcal{W}_{TaxeGouv} = \{Oui, Non\}$
$\mathcal{W}_{OffreKero} = \{Faible, Stable, Elevee\}$

Définition 3 *(*Lois du domaine - *Ensemble $\mathcal{C}$) Expression des contraintes sur le système.*

Parmi les lois du domaine figurent des couples valeur/variable qui sont incompatibles entre eux : la fonction $incompatible$ renvoie « Vrai » si un ensemble de couples $(variable, valeur)$ sont incompatibles.

Définition 4

$$incompatible : P(\mathcal{V} \times \mathcal{W}_\mathcal{V}) \rightarrow \{Vrai, Faux\} \quad (1)$$

avec P l'ensemble des parties de $\mathcal{V} \times \mathcal{W}_\mathcal{V}$

Exemple

$$\mathcal{C} = \{incompatible((OffreKero, Faible), (OffreAvion, Elevee)) = Vrai\}$$

Définition 5 *(*Acteur - *Ensemble $\mathcal{A}$) Un acteur a est défini par son identifiant i_a et par l'ensemble $\mathcal{V}_a$ des variables sur lesquelles il peut influer (notamment par des prises de décision).*

$$\forall a \in \mathcal{A}, a =< i_a, \mathcal{V}_a > \quad (2)$$

On distingue les acteurs intérieurs et les acteurs extérieurs au système.

Définition 6 *(*Acteur extérieur - *Ensemble $\mathcal{A}_X$) Un acteur extérieur peut créer des perturbations en agissant sur des variables du système.*

Exemple $\mathcal{A}_X = \{SARS\text{-}CoV\text{-}2\}$

Définition 7 *(*Acteur intérieur - *Ensemble $\mathcal{A}_I$) Un acteur intérieur a_I est un acteur qui fait partie du système. En plus de pouvoir réaliser des actions, il a la capacité de prendre des décisions.*

Exemple

$$\mathcal{A}_I = \{VolFacile, GouvPaysO, SuperKero\}$$

avec *VolFacile* une compagnie aérienne, *GouvPaysO* le gouvernement des Pays-O et *SuperKero* un fournisseur de carburant conventionnel type kérosène.

On fait ici les hypothèses simplificatrices suivantes :
- un acteur est limité à une décision par variable dans chaque état du système ;
- il connaît l'état de toutes les variables sur lesquelles il peut agir ;
- dans chaque état, il doit prendre des décisions sur toutes les variables sur lesquelles il peut influer.

Un acteur intérieur est caractérisé avec les fonctions suivantes :

Définition 8 *La fonction $position$ indique la position d'un acteur intérieur a_I sur un des principes du système dans un état donné e (voir définition 12). L'acteur peut adhérer $(+)$, être neutre $(=)$ ou être défavorable $(-)$ vis-à-vis d'un principe.*

$$position_{a,e} : \mathbf{\Pi} \rightarrow \{+, =, -\} \quad (3)$$

L'ensemble des positions des acteurs intérieurs sur les principes, données par la fonction $position$ dans un état e, est noté $\mathcal{P}_e$ et l'ensemble des positions d'un unique acteur a dans un état e est noté $\mathcal{P}_{a,e}$. Tous les acteurs intérieurs doivent avoir une position (même neutre) sur chacun des principes.

Définition 9 *La fonction $opinion$ renvoie l'opinion affichée d'un acteur intérieur a sur la manière dont la valeur d'une variable se positionne vis-à-vis d'un principe dans un état donné. L'acteur peut estimer que la valeur de la variable va dans le sens du principe (1), qu'elle n'est pas liée au principe (0) ou qu'elle est en contradiction avec le principe (-1).*

$$opinion_{a,e} : \mathcal{V} \times \mathcal{W}_\mathcal{V} \times \mathbf{\Pi} \rightarrow \{1, 0, -1\} \quad (4)$$

L'ensemble des opinions affichées des acteurs intérieurs, générées par les fonctions $opinion_a$ dans un état e, est noté $\mathcal{O}_e$ et l'ensemble des opinions d'un unique acteur a dans un état e est noté $\mathcal{O}_{a,e}$.

Remarque

Un acteur ne peut pas prendre de décisions à l'encontre des principes auxquels il est favorable. Il peut en revanche changer ses positions et ses opinions en cours de scénario.

Exemple

Soit e un état donné,

$$position_{GouvPaysO,e}(SatisfactionPopulation) = +$$
$$opinion_{GouvPaysO,e}((TaxeGouv, Oui), SatisfactionPopulation) = -1$$

Dans l'état e, le gouvernement *GouvPaysO* est favorable au principe *SatisfactionPopulation* et considère que si la valeur de la variable *TaxeGouv* est *Oui*, cette dernière ne respecte pas le principe *SatisfactionPopulation*.

Cas particulier de l'utilisateur
On suppose que l'acteur qui est à l'initiative de l'étude – l'utilisateur – peut prendre des décisions qui vont à l'encontre de ses propres principes. En effet, l'utilisateur, en plus d'agir selon des principes, peut être guidé par des objectifs (voir définition 10 ci-dessous) :

Définition 10 *(*Objectif $\mathcal{G}_u \subset (\mathcal{V} \times \mathcal{W}_\mathcal{V})$*) Ensemble fixé par l'utilisateur des instanciations de variables qu'il souhaite obtenir.*

On peut donc, avec ces éléments, poser la définition suivante du système :

Définition 11 *(*Système $\mathbf{\Sigma}$*) Un système est un quadruplet composé d'un ensemble $\mathcal{A}_I$ d'acteurs intérieurs, d'un ensemble $\mathbf{\Pi}$ de principes, d'un ensemble $\mathcal{V}$ de variables et d'un ensemble de lois du domaine $\mathcal{C}$. Il est caractérisé par son état courant e.*

$$\mathbf{\Sigma} =< \mathcal{A}_I, \mathbf{\Pi}, \mathcal{V}, \mathcal{C} > \tag{5}$$

On fait ici les hypothèses simplificatrices suivantes :
- le système est fermé [4] ;
- il n'y a pas de dynamique propre au système : une valeur de variable ne change que sous l'action d'un acteur (voir définition 15).

Un état du système est défini formellement de la manière suivante :

Définition 12 *(*État du système - *Ensemble* $\mathbf{E}$*) Un état du système, noté e, est composé d'un ensemble $\mathcal{P}_e$ des positions des acteurs intérieurs sur les principes, d'un ensemble $\mathcal{O}_e$ des opinions des acteurs ainsi que d'un ensemble $\mathcal{I}_e$ des variables instanciées. Une variable ne peut pas avoir deux valeurs différentes dans un état donné.*

$$e =< \mathcal{P}_e, \mathcal{O}_e, \mathcal{I}_e > \tag{6}$$

Avec

$$\mathcal{I}_e = \{\mathcal{I}_e \subset (\mathcal{V} \times \mathcal{W}_\mathcal{V}), (v, w_v) \in \mathcal{I}_e \; et \; (v, w'_v) \in \mathcal{I}_e, \\ w_v \neq w'_v \Rightarrow incompatible_e((v, w_v), (v, w'_v)) = Vrai\} \tag{7}$$

L'état initial du système est donné. Comme dans tous les états, toutes les variables y sont instanciées.

Exemple

Soit e_0 l'état initial donné,

$$\mathcal{I}_{e_0} = \{(TaxeGouv, Non), \\ (OffreKero, Stable), (OffreAvion, Stable)\}$$

3.3 Décision et action

Les acteurs intérieurs au système peuvent décider de modifier (ou non) la valeur des variables sur lesquelles ils peuvent influer.

4. Les composantes du système sont figées.

Définition 13 *(*Décision - *Ensemble $\mathcal{D}$) Une décision $d_{a,v,e}$ est le choix d'un acteur intérieur a_i de faire quelque chose vis-à-vis d'une variable v dans un état e.*

Une décision peut être :
- une volonté d'action (changer ou maintenir la valeur de la variable) ;
- ne rien faire vis-à-vis de cette variable (c'est-à-dire laisser faire les autres acteurs). Dans le cas où un acteur est le seul à pouvoir agir sur une variable, la décision de ne rien faire équivaut à la décision de maintenir l'état de la variable.

On note $\mathcal{D}_e$ l'ensemble des décisions envisagées dans l'état e, $\mathcal{D}_{a,e}$, l'ensemble des décisions envisagées par l'acteur a dans l'état e et $\mathcal{D}_{a,v,e}$, l'ensemble des décisions envisagées par l'acteur a sur une variable v dans l'état e.

Exemple

$$\mathcal{D}_{GouvPaysO,TaxeGouv,e_0} = \{InstaurerTaxe, \\ NeRienFaireTaxe\}$$

Définition 14 *La fonction h exprime le résultat de la décision $d_{a,v,e}$ de changer la valeur w_v d'une variable v en la valeur w'_v.*

$$h : \mathcal{D} \times \mathcal{V} \times \mathcal{W}_\mathcal{V} \rightarrow \mathcal{V} \times \mathcal{W}_\mathcal{V} \tag{8}$$

Définition 15 *(*Action - *Ensemble $\mathcal{A}_c$) Une action est la réalisation d'une décision. Elle permet de passer d'une instanciation (v, w_v) à une instanciation (v, w'_v).*

Les actions des acteurs viennent modifier l'état du système.

Définition 16 *(*Événement - *Ensemble $\mathcal{E}$) Un événement est une modification de l'état du système par un changement de valeur d'une ou de plusieurs variables suite à une action, ou bien suite à un changement de la position des acteurs sur les principes ou de leur opinion affichée sur la valeurs de variables.*

3.4 Conflits

Suite à leurs décisions dans un état donné, des acteurs peuvent se retrouver dans des situations de conflit, conflit entre acteurs (conflit logique) ou conflit pour un acteur (conflit moral).

3.4.1 Conflit logique

Le conflit logique intervient dans deux cas :
- des acteurs cherchent à instancier une même variable avec des valeurs différentes ;
- des acteurs cherchent à instancier des variables d'une manière définie comme incompatible dans les lois du domaine du système.

Définition 17 *(*Conflit logique*) :*

$$\forall e, \forall \mathcal{D}_e, conflitlogique(\mathcal{D}_e, e) = vrai \iff \\ \left[\begin{array}{l} \exists (v^1, {w_v}^1), ..., (v^n, {w_v}^n) \in \mathcal{H}_e, \exists 0 < n \leq |\mathcal{V}| \\ incompatible((v^1, {w_v}^1), ..., (v^n, {w_v}^n)) = Vrai \end{array} \right. \tag{9}$$

avec $\mathcal{H}_e$ l'état partiel du système résultant des décisions des acteurs dans l'état e.

$$\mathcal{H}_e = \{h(d_{a,v,e}, v, w_v), v \in \mathcal{V}, w_v \in \mathcal{W}_v, d_{a,v,e} \in \mathcal{D}_e\} \tag{10}$$

Exemple Considérons la décision de la compagnie *VolFacile* d'*AugmenterOffreAvion* dans l'état initial e_0 où l'*OffreAvion* est *Stable* :

$$d_{VolFacile,OffreAvion,e_0} = AugmenterOffreAvion$$

$$h(AugmenterOffreAvion, OffreAvion, Stable) = (OffreAvion, Elevee)$$

Si l'acteur *VolFacile* prend cette décision, la valeur de la variable *OffreAvion* passera de *Stable* à *Elevee*.

Considérons la décision du fournisseur *SuperKero* dans l'état initial où l'*OffreKero* est *Stable* :

$$d_{SuperKero,OffreKero,e_0} = DiminuerOffreKero$$

$$h(DiminuerOffreKero, OffreKero, Stable) = (OffreKero, Faible)$$

Si l'acteur *SuperKero* prend cette décision, la valeur de la variable *OffreKero* passera de *Stable* à *Faible*

La loi du domaine :

$$\mathcal{C} = \{incompatible((OffreKero, Faible), (OffreAvion, Elevee)) = Vrai\}$$

indique que ces deux couples sont incompatibles, on se trouve dans une situation de conflit logique.

3.4.2 Conflit moral

La définition de conflit moral est fondée sur la définition proposée par [5]. Contrairement au conflit logique, un conflit moral porte sur les principes et les opinions d'un seul acteur.

La fonction $ContrairePrincipe$ renvoie le booléen « Vrai » lorsqu'une décision d'un acteur a est contraire aux principes auxquels adhère cet acteur dans un état e :

Définition 18

$$ContrairePrincipe_{a,e} : \mathcal{D}_{a,e} \times \mathbf{\Pi} \rightarrow \{Vrai, Faux\} \tag{11}$$

Le fait qu'une décision d'un acteur soit contraire à un principe est spécifié dans les lois du domaine.

Exemple $ContrairePrincipe_{GouvPaysO,e_0}($
$DesinformerSurRechauffementClimatique,$
$Honnetete) = Vrai$
L'acteur *GouvPaysO* considère que la décision *DesinformerSurRechauffementClimatique* est contraire au principe *Honnetete* dans l'état initial e_0.

La fonction $ConsequenceNegative$ renvoie le booléen « Vrai » si l'action résultant d'une décision aurait des conséquences négatives pour un acteur a dans un état e. On entend ici par conséquences un état partiel $\mathcal{H}_a$ dont des instanciations de variables vont à l'encontre d'au moins un principe auquel l'acteur adhère.

Définition 19

$$ConsequenceNegative_{a,e} : \mathcal{D}_{a,v,e} \times \mathcal{P}_{a,e} \times \mathcal{O}_{a,e} \rightarrow \{Vrai, Faux\} \tag{12}$$

Exemple

$h_1 = h(InstaurerTaxe, TaxeGouv, Non) =$
$(TaxeGouv, Oui)$
Avec la position et l'opinion spécifiées après la définition 9
$ConsequenceNegative_{GouvPaysO,e_0}($
$InstaurerTaxe, +, -1) = Vrai$

Dans l'état initial e_0, l'acteur *GouvPaysO* est favorable au principe *SatisfactionPopulation* (+). Or, il a une opinion négative (-1) sur la manière dont il envisage le respect de ce principe par la valeur *Oui* de la variable *TaxeGouv*. La décision d'*InstaurerTaxe* qui instancierait la variable par cette valeur a donc des conséquences négatives.

Un acteur est dans une situation de conflit moral dans un état e lorsque chacune des décisions qu'il pourrait prendre sur une variable est soit contraire à ses principes soit pourrait avoir des conséquences négatives.

Définition 20 *(*Conflit moral*) :*

$$\begin{aligned} &conflitmoral(a, \mathcal{P}_{a,e}, \mathcal{O}_{a,e}, e) = Vrai \iff \\ &\exists v \in \mathcal{V}, \forall d_{a,v,e} \in \mathcal{D}_{a,v,e}, \exists \pi \in \mathbf{\Pi},\ position_{a,e}(\pi) = + \\ &\left[\begin{array}{l} ContrairePrincipe(d_{a,v,e}, \pi) = Vrai \vee \\ ConsequenceNegative(d_{a,v,e}, position_{a,e}(\pi), \\ \qquad opinion_{a,e}(h(d_{a,v,e}, v, w_v), \pi) = Vrai \end{array} \right. \end{aligned} \tag{13}$$

3.5 Scénario

Un scénario est défini par l'état initial du système e_0, un état final du système e_f, et un chemin c allant de l'état initial à cet état final.

Définition 21 *(*Scénario - *Ensemble $\mathcal{S}$)*

$$\forall s \in \mathcal{S}, s =< e_0, e_f, c > \tag{14}$$

avec $e_0, e_f \in \mathbf{E}$ et c le chemin.

c est une liste qui comprend :

- les différents états du système ;
- les événements à l'origine des modifications d'états ;
- des informations telles que : l'impossibilité de transformer une décision en action et la raison de cette impossibilité, les décisions de ne rien faire qui ne modifient pas l'ensemble des variables instanciées d'un état du système ou encore la justification de l'arrêt d'un scénario.

Les critères d'arrêt d'un scénario sont :

- la convergence du scénario vers un état, caractérisée par :

- une stabilisation (deux états successifs identiques) ;
- un conflit logique ou un conflit moral ;
- l'atteinte de l'objectif de l'utilisateur ;
- l'atteinte d'un état d'intérêt prédéfini.

— la convergence du scénario vers une boucle limite.

Les acteurs, variables et principes initialement définis ne pouvant pas changer au cours de la génération des scénarios, le nombre de scénarios possibles est fini.

4 Mise en œuvre et analyse

4.1 Algorithme

La génération d'un scénario se fait sous la forme de la génération d'une succession d'états. Le passage d'un état à un autre résulte de l'agrégation des décisions prises par les différents acteurs. En effet, individuellement, les acteurs comparent leurs opinions à la valeur des variables dans l'état courant du système. Les décisions contraires aux principes ou menant à des conséquences négatives sont éliminées. On relève à ce niveau les conflits moraux. On compare ensuite les décisions possibles des acteurs, ce qui peut donner lieu à des conflits logiques. Une fois l'agrégation des décisions restantes terminée, on réalise les actions correspondantes et on obtient un nouvel état du système.

Algorithme Génération d'un scénario

Requiert : données du système, état initial
fonction SCENARIO(données du système, état précédent) :
Pour tout Acteur **Faire**
Pour tout Variable sur laquelle l'acteur peut agir **Faire**
Liste des décisions possibles dans cet état
Éliminer les décisions qui sont contraires aux principes ou qui ont des conséquences négatives
Si Conflit moral (Liste de décisions possibles vide) **alors**
renvoyer Scénario
Sinon
Choix d'une décision possible
Fin si
Fin pour
Choix d'une combinaison de décisions possibles
Fin pour
Agrégation des décisions de tous les acteurs
Si Conflit logique **alors**
Arrêt du scénario
Sinon
Réaliser les actions correspondant aux décisions : état suivant
Appliquer la fonction SCENARIO à l'état suivant
Fin si
renvoyer Scénario
Fin fonction

4.2 Mise en œuvre

Nous avons mis en œuvre ce modèle sous Python v3.8. L'exécution du code se divise en trois étapes :

- la récupération des données du système : elles sont fournies par l'utilisateur et peuvent résulter d'un travail tel qu'une analyse morphologique (voir section 2.2.2) ;
- la génération des scénarios ;
- l'analyse des résultats.

```
[1, {}, False]
Etat 1 = {'OffreAvion' : 'Stable', 'TaxeGouv' : 'Non', 'OffreKero' : 'Stable'}
[2, {'VolFacile': 'AugmenterOffreAvion', 'GouvPaysO': 'InstaurerTaxe',
'SuperKero': 'AugmenterOffreKero'}, False]
Etat 2 : {'OffreAvion' : 'Elevee', 'TaxeGouv' : 'Oui', 'OffreKero' : 'Elevee'}
[3, {'VolFacile': 'ReduireOffreAvion', 'GouvPaysO': 'LaisserTaxe',
'SuperKero': 'ReduireOffreKero'}, False]
Etat 3 : {'OffreAvion' : 'Stable', 'TaxeGouv' : 'Oui', 'OffreKero' : 'Stable'}
[6, {'VolFacile': 'AugmenterOffreAvion', 'GouvPaysO': 'LaisserTaxe',
'SuperKero': 'NeRienFaireOffreKero'}, False]
Etat 6 : {'OffreAvion' : 'Elevee', 'TaxeGouv' : 'Oui', 'OffreKero' : 'Stable'}
[6, ['logique', ['NeRienFaireOffreEleveeAvion', 'DiminuerOffreKero'],
['VolFacile', 'SuperKero']], True]
```

FIGURE 1 – Exemple de scénario

Dans cet exemple le scénario est une succession de quatre états et se termine par un conflit logique suite à la décision *NeRienFaireOffreEleveeAvion* de l'acteur *VolFacile* et à la décision *DiminuerOffreKero* de l'acteur *SuperKero*.

La génération automatique des scénarios est produite par récurrence sous forme d'un arbre d'états du système. Nous avons appliqué les critères d'arrêt (voir section 3.5) mais également un critère d'arrêt arbitraire sur la profondeur de l'arbre pour limiter le nombre de scénarios générés. En effet, la complexité du problème est exponentielle – en fonction du nombre de composantes du système – à cause de la présence d'une récurrence dans une boucle *for*.

4.3 Analyse

Ce type de génération peut répondre à deux types de requêtes relatives à un scénario :

- celles qui portent sur la fin du scénario ;
  Exemples : Quels sont les acteurs le plus à l'origine de conflits ? Quelles sont les raisons des conflits ? Quels scénarios conduisent à des états prédéfinis ? Quels états ne peuvent pas être atteints avec ces données initiales ?
- celles qui portent sur le chemin du scénario ;
  Exemples : À quels principes l'utilisateur doit-il renoncer pour atteindre ses objectifs ? Quelles sont les décisions qui ont peu voire pas d'impact sur les changements d'états ?

5 Conclusion et perspectives

Nous avons proposé un outil formel permettant de générer automatiquement et systématiquement des scénarios. Cet

outil peut être utilisé pour aider à la prise de décision au sein d'un système d'acteurs comme ceux considérés dans [3], [13], [8]. De plus, la modélisation prend en compte des principes moraux qui guident les décisions des acteurs.
Cet outil devrait permettre de s'affranchir de certaines limites inhérentes aux méthodes manuelles en élargissant la réflexion des parties prenantes grâce à un plus grand nombre de scénarios considérés.
La validation et l'analyse des scénarios est en cours de définition. Nous pourrons faire référence à des critères existants comme ceux proposés dans la définition du paramètre *Validation* de la typologie de Crawford [6] : pertinence, transparence, nouveauté, etc.
En outre, pour analyser de manière pertinente et sans perdre d'information le grand nombre de scénarios générés, on pourrait s'intéresser à la proximité de deux scénarios différents et définir une notion de similarité entre scénarios.
Enfin, un cas d'étude fondé sur le scénario *Stripping Down* de l'EREA [13] est actuellement en cours d'élaboration.

Remerciements

Les autrices remercient Thomas Chaboud, Brieuc Danet, Isabelle Laplace et Claire Sarrat pour leur participation aux travaux ayant permis l'élaboration de cet article.

Références

[1] M. Amer, T. U. Daim, and A. Jetter. A review of scenario planning. *Futures*, 46, 2013.

[2] G. Berger. L'attitude prospective. In P. Durance, editor, *De la prospective. Textes fondamentaux de la prospective française 1955-1966*. L'Harmattan, 2e edition, 2007.

[3] R. Berghof, A. Schmitt, J. Middel, C. Eyers, R. Hancox, A. Gruebler, and Hepting M. Constrained Scenarios on Aviation and Emissions 2050. Technical report, European Commission, 2005.

[4] J.C. Bluet and J. Zemor. Schéma général d'aménagement de la France. In *Travaux et recherches de prospective*. La documentation française, 1971.

[5] V. Bonnemains. *Formal ethical reasoning and dilemma identification in a human-artificial agent system*. PhD thesis, ONERA-ISAE-SUPAERO, 2019.

[6] M.M. Crawford. A comprehensive scenario intervention typology. *Technological forecasting and social change*, 149, 2019.

[7] P.K. Davis, S.C. Bankes, and M. Egner. Enhancing strategic planning with massive scenario generation. Technical report, The RAND Corporation, National security research division, 2007.

[8] Agence de l'environnement et de la maîtrise de l'énergie. Transition(s) 2050 Choisir maintenant-Agir pour le climat. Technical report, novembre 2021.

[9] G. Ducot and G.J. Lubben. A typology for scenarios. *Futures*, 12, 1980.

[10] M. Godet. *Manuel de prospective stratégique*, volume 1. Dunod, 2007.

[11] M. Godet. *Manuel de prospective stratégique*, volume 2. Dunod, 2007.

[12] T.J. Gordon. Trend impact analysis. In *Futures Research Methodology*, pages 1–19.

[13] European Research Establishments in Aeronautics. EREA Vision Study - The Future of Aviation in 2050. Technical report, December 2020.

[14] I. Johansen. Scenario modelling with morphological analysis. *Technological forecasting and social change*, 126, 2018.

[15] A.C. Mangnus, J. Oomen, J.M. Vervoort, and M.A. Hajer. Futures literacy and the diversity of the future. *Futures*, 132, 2021.

[16] K. Muiderman, J. M. Vervoort, A. Gupta, and F. Biermann. Identifying four approaches to anticipatory climate governance : Varying conceptions of the future and their implications for the present. *Wiley Interdisciplinary Reviews : Climate Change*, 11(6), 2020.

[17] Association NégaWatt. La transition énergétique au cœur d'une transition sociétale. Technical report, octobre 2021.

[18] M.J. Spaniol and N.J. Rowland. The scenario planning paradox. *Futures*, 95, 2018.

[19] M.J. Spaniol and N.J. Rowland. Defining scenario. *Futures foresight science*, 1(1), 2019.

[20] J-P. Treuil, A. Drogoul, and J-D. Zucker. *Modélisation et simulation à base d'agents*. Dunod, 2008.

[21] UNESCO. Recommandation sur l'éthique de l'Intelligence Artificielle, 2021.

[22] P. Wack. Scenarios : Uncharted waters ahead. *Harvard Business Review*, 85516, 1985.

De l'équivalence entre les modèles structurels causaux et les systèmes abstraits d'argumentation

Y. Munro[1,2], I. Bloch[1], M. Chetouani[2], M-J. Lesot[1], C. Pelachaud[3]

[1] Sorbonne Université, CNRS, LIP6, Paris, France
[2] Sorbonne Université, CNRS, ISIR, Paris, France
[3] CNRS, Sorbonne Université, ISIR, Paris, France

prenom.nom@sorbonne-universite.fr

Résumé

Les modèles structurels causaux et les systèmes abstraits d'argumentation sont deux approches s'inscrivant dans les problématiques de l'intelligence artificielle explicable. Dans cet article, nous mettons en évidence une équivalence entre un cas particulier de ces modèles causaux, que nous appelons graphes causaux argumentatifs, et les systèmes abstraits d'argumentation. Nous proposons également une transformation permettant de passer d'une représentation à l'autre.

Mots-clés

Modèles structurels causaux, Systèmes abstraits d'argumentation, Explications en intelligence artificielle (XAI).

Abstract

In the field of explainable artificial intelligence, causal models and abstract argumentation frameworks are two formal approaches that provide a definition of an explanation. In this paper, we show the equivalence between a particular type of causal models, that we call argumentative causal graphs, and abstract argumentation frameworks. We also propose a transformation between these two systems.

Keywords

Causal models, Abstract argumentation frameworks, Explainable artificial intelligence (XAI).

1 Introduction

Il existe de nombreuses méthodes permettant de contribuer à l'interprétabilité et à l'explicabilité des systèmes d'intelligence artificielle (XAI) [5]. Des approches numériques ont pour objectif de fournir une explication en cherchant par exemple les corrélations entre les attributs d'entrée et de sortie. Des approches symboliques reposent sur des formalismes logiques pour raisonner par abduction ou rechercher des causalités, à partir de la modélisation formelle d'un problème ou d'une situation. C'est à ce type d'approche que nous nous intéressons dans cet article.

Pour améliorer la qualité de ces méthodes, une approche consiste à s'inspirer des mécanismes cognitifs et sociaux humains notamment ceux liés au processus d'explications et en déduire des propriétés et comportements intéressants. Dans [13], Tim Miller, en s'appuyant sur des travaux en sciences sociales et cognitives, dégage des caractéristiques essentielles qu'il conviendra de retrouver lors du développement de méthodes d'intelligence artificielle explicable.

Un premier cadre formel provient des travaux de Joseph Halpern et Judea Pearl [9] sur la causalité et notamment sur ce qu'ils appellent des modèles structurels causaux. Cette notion est en effet très intimement liée à celle d'explication : expliquer un fait revient souvent à fournir une cause et on retrouve donc logiquement dans leurs travaux une définition de cette notion [10]. Ce cadre a par exemple été mis en œuvre par Prashan Madumal et al. pour générer des explications pour un agent jouant à Starcraft II [12], un jeu de stratégie en temps réel. Cet article s'intéresse à un cas particulier de ces modèles que nous proposons d'appeler des graphes causaux argumentatifs.

Un autre cadre proposant une définition de la notion d'explication est celui de l'argumentation. Introduits par Phan Minh Dung en 1995 [6], les systèmes abstraits d'argumentation (*abstract argumentation framework*, AAF) permettent de modéliser les interactions entre des arguments provenant de plusieurs entités ou agents. De nombreuses méthodes de XAI ont déjà été développées dans ce cadre [17], que ce soit pour des problèmes modélisés initialement par des graphes argumentatifs ou bien pour des modèles qui à l'origine ne l'étaient pas.

Après avoir présenté brièvement ces deux cadres dans les sections 2 et 3, nous mettons en évidence une équivalence entre eux par l'intermédiaire d'une transformation permettant de passer des graphes argumentatifs aux graphes causaux argumentatifs et inversement (sections 4 et 5). A notre connaissance, il n'y a pas eu de travaux dans ce sens et c'est pourquoi nous proposons des transformations permettant de lier les deux champs, ce qui constitue la contribution principale de l'article. L'objectif n'est donc pas de présenter une nouvelle méthode ou un nouveau cadre mais bien de pouvoir passer de l'un à l'autre et donc permettre d'exploiter les propriétés intéressantes de chacun.

L'article illustre les principes proposés sur un exemple inspiré des assistants de régulation médicaux dans le cadre de la situation sanitaire liée au COVID-19 : il considère un

agent, humain ou autonome, dont l'objectif est de conseiller sur la nécessité de réaliser un test PCR [1]. Il s'agit évidemment d'un modèle très simplifié de la réalité dont l'unique but est d'illustrer nos contributions et qui n'a pas vocation à remplacer les consignes sanitaires existantes.

2 Modèles structurels causaux

Cette section rappelle les concepts définis par J. Halpern [8] qui conduisent à la définition de la notion d'explication dans le cadre des modèles structurels causaux.

2.1 Définition

Un modèle structurel causal tel qu'introduit par J. Halpern [8] est un triplet $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ tel que :

- $\mathcal{U}$ est un ensemble des variables exogènes, c'est-à-dire un ensemble de variables dont les valeurs sont indépendantes du modèle ;
- $\mathcal{V}$ est un ensemble des variables endogènes ;
- $\mathcal{F}$ est l'ensemble des équations structurelles du modèle (une pour chaque variable de $\mathcal{V}$). Elles permettent d'associer une valeur à chacune des variables endogènes en fonction des valeurs des variables exogènes.

En associant à chaque variable un nœud et en traçant des arcs entre ces nœuds pour indiquer les dépendances fonctionnelles de $\mathcal{F}$, on obtient une représentation d'un modèle structurel M sous la forme d'un graphe.
L'équivalence discutée dans les sections 4 et 5 s'intéresse à un cas particulier de modèles structurels causaux que nous proposons d'appeler **graphes causaux argumentatifs** (GCA). Ce sont des triplets $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ pour lesquels :

1. Les variables sont à valeur binaire. Les équations structurelles s'écrivent donc comme des formules logiques.
2. Ces formules ne contiennent pas de disjonction.
3. Le graphe associé est acyclique.

Dans la suite, les notations supplémentaires suivantes sont utilisées :

- Soit $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$, un modèle structurel causal. On appelle un **contexte**, noté $\mathbf{u}$, une affectation des variables de $\mathcal{U}$. La paire $(M, \mathbf{u})$ est appelée **monde**.
- Soit $\mathbf{X}$ un ensemble de variables de $\mathcal{V}$, on note $\mathbf{X} = \mathbf{x}$ une affectation des variables de $\mathbf{X}$ avec les valeurs de $\mathbf{x}$.
- Soit $\mathcal{K}$ un ensemble de contextes et $\mathbf{u} \in \mathcal{K}$, on note $(M, \mathbf{u}) \models \mathbf{X} = \mathbf{x}$ si $\mathbf{X} = \mathbf{x}$ est l'unique solution aux équations de $\mathcal{F}$ dans $\mathbf{u}$.
- Soit $\mathcal{K}$ un ensemble de contextes. Soit $\mathbf{X} \in \mathcal{V}$ et $\mathbf{x}$ des valeurs de $\mathbf{X}$. On note $\mathcal{K}_{\mathbf{X}=\mathbf{x}}$, l'ensemble des contextes $\mathbf{u}'$ de $\mathcal{K}$ tel que $(M, \mathbf{u}') \models \mathbf{X} = \mathbf{x}$.
- Soit $\mathcal{K}$ un ensemble de contextes et $\mathbf{u} \in \mathcal{K}$, la notation $(M, \mathbf{u}) \models [\mathbf{X} = \mathbf{x}](\mathbf{Y} = \mathbf{y})$ signifie que l'on se place dans le monde $(M, \mathbf{u})$ dans lequel les équations de $\mathcal{F}$ portant sur les variables de $\mathbf{X}$ sont remplacées par l'équation $\mathbf{X} = \mathbf{x}$.

1. *Reverse Transcriptase-Polymerase Chain Reaction*

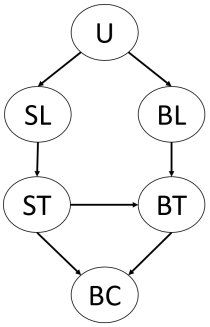


FIGURE 1 – Scénario causal, inspiré de [16].

Exemple 1. *Nous reprenons ici un exemple classique tiré de [16]. Suzy et Billy lancent tous les deux une pierre en direction d'une bouteille de verre. Ils sont tous les deux parfaitement précis et sont donc sûrs de toucher la bouteille s'ils lancent effectivement la pierre. Si l'une des pierres atteint la bouteille alors celle-ci se casse. La pierre de Suzy touche toujours la première.*
Pour modéliser la situation, on introduit les variables suivantes : SL (respectivement BL) et ST (resp. BT) représentent «Suzy (resp. Billy) lance» et «Suzy (resp. Billy) touche». Enfin, BC renvoie à «la bouteille se casse».
On introduit également un ensemble $\mathcal{U}$ de variables exogènes qui représentent des facteurs extérieurs au problème qui influencent le fait que Billy ou Suzy lancent la pierre.
Les fonctions de $\mathcal{F}$ complètent la modélisation du problème. Par exemple, le fait que Billy touche la bouteille dans le cas (et uniquement dans ce cas) où il a lancé une pierre et Suzy n'a pas touché la bouteille est représenté par la fonction suivante : $BT = BL \wedge \neg ST$.
On obtient le modèle structurel causal suivant, illustré figure 1 :
$\mathcal{U} = \{U\}$
$\mathcal{V} = \{SL, BL, SL, BT, BC\}$
$\mathcal{F} = \{(ST = SL), (BT = BL \wedge \neg ST), (BC = ST \vee BT)\}$

2.2 Cause effective

Dans le formalisme précédent des modèles structurels causaux, J. Halpern [8] propose ensuite de définir la notion de cause de la façon suivante.
L'affectation $\mathbf{X} = \mathbf{x}$ est une **cause effective** de φ dans le monde $(M, \mathbf{u})$ si les trois conditions suivantes sont vérifiées :

AC1 $(M, \mathbf{u}) \models (\mathbf{X} = \mathbf{x}) \wedge \varphi$, c'est-à-dire la cause et la conséquence sont toutes les deux vraies dans le monde considéré.

AC2 Il existe un ensemble $\mathbf{W}$ de variables endogènes avec des valeurs $\mathbf{w}$ et une configuration $\mathbf{x}'$ pour la variable $\mathbf{X}$ tels que si $(M, \mathbf{u}) \models (\mathbf{W} = \mathbf{w})$ alors :

$$(M, \mathbf{u}) \models [\mathbf{X} = \mathbf{x}', \mathbf{W} = \mathbf{w}]\neg\varphi$$

.

AC3 $\mathbf{X}$ est minimal : il n'existe pas de sous-ensemble de $\mathbf{X}$ qui satisfasse **AC1** et **AC2**. Cette dernière condition vise à éviter d'avoir des variables inutiles dans la cause.

Pour savoir qu'une chose est une conséquence d'une autre, il est possible de raisonner en se demandant : si la cause présumée ne s'était pas produite, est-ce que la conséquence se serait tout de même produite ? C'est ce que l'on appelle un scénario contrefactuel ou hypothétique. Si la réponse à la question précédente est « non » alors la cause présumée devient une cause effective.
La condition **AC2** renvoie à ce raisonnement sur les contrefactuels. Plus précisément, cette condition impose que s'il existe un scénario contrefactuel, c'est-à-dire un scénario dans lequel la cause présumée ne s'est pas produite ($\mathbf{X} = \mathbf{x}'$) **et** éventuellement d'autres évènements se sont quand même produits ($\mathbf{W} = \mathbf{w}$), tel que la conséquence à expliquer ne se produise pas, alors la cause présumée est bien une cause.

Exemple 1. *(suite) – Intuitivement, une cause de la bouteille qui se casse est le fait que Suzy ait lancé la pierre. En effet, c'est sa pierre qui a touché la bouteille et l'a donc cassée. Cependant, si on se pose la question : si Suzy n'avait pas lancé sa pierre, la bouteille se serait-elle cassée ? La réponse est oui car Billy aurait alors touché la bouteille ($BT = BL \wedge \neg ST$)).*
Il faut donc envisager le contrefactuel suivant : si Suzy n'avait pas lancé sa pierre ***en sachant que*** *Billy n'a pas touché la bouteille, la bouteille se serait-elle cassée ? Dans ce cas la réponse est effectivement non, c'est-à-dire que le fait que Suzy ait lancé sa pierre est bien une cause effective du fait que la bouteille se casse.*

2.3 Cause suffisante

Soit $\mathcal{K}$ un ensemble de contextes et $\mathbf{u} \in \mathcal{K}$. L'affectation $\mathbf{X} = \mathbf{x}$ est une **cause suffisante** de φ dans le monde $(M, \mathbf{u})$ si les quatre conditions suivantes sont vérifiées :

SC1 $(M, \mathbf{u}) \models (\mathbf{X} = \mathbf{x}) \wedge \varphi$.

SC2 Il existe une partie de $\mathbf{X}, X = x$, et une autre conjonction $(\mathbf{Y} = \mathbf{y})$ (éventuellement vide) telles que $(X = x) \wedge (\mathbf{Y} = \mathbf{y})$ est une cause effective de φ dans $(M, \mathbf{u})$, c'est-à-dire une partie de $\mathbf{X}$ est une partie d'une cause effective dans le monde considéré.

SC3 $(M, \mathbf{u}') \models [\mathbf{X} = \mathbf{x}]\varphi$ pour tous les contextes $\mathbf{u}' \in \mathcal{K}$, c'est-à-dire si l'on a $\mathbf{X} = \mathbf{x}$ alors on a φ quel que soit le contexte considéré.

SC4 $\mathbf{X}$ est minimal.

Remarque 1. *Il existe une deuxième version de la définition de cause suffisante proposée par T. Miller dans [14]. Il définit cette notion comme une cause effective non minimale, c'est-à-dire qui ne vérifie que* **AC1** *et* **AC2***. La différence majeure se situe dans* **SC3***. Le point de vue de T. Miller se concentre uniquement sur le contexte en cours, contrairement à J. Halpern qui définit une cause suffisante sur un ensemble de contextes donnés. On fait le choix ici de considérer plutôt la définition de J. Halpern notamment car en affaiblissant* **SC3** *on peut définir une notion de pouvoir explicatif utile pour comparer les explications générées.*

2.4 Explication

Lorsque l'on fournit une explication, il est important de tenir compte de la personne à qui est fournie cette explication. On appelle cette personne le destinataire de l'explication ou en anglais l'*explainee*. Pour cette raison, la recherche de cause effective et de cause suffisante va être contrainte à un ensemble de contextes $\mathcal{K}$ déterminé par ce que l'*explainee* considère comme possible.
L'affectation $\mathbf{X} = \mathbf{x}$ est une **explication** de φ relative à l'ensemble de contextes $\mathcal{K}$ si les trois conditions suivantes sont vérifiées :

EX1 $\mathbf{X} = \mathbf{x}$ est une cause suffisante pour tous les contextes $\mathbf{u}$ dans $\mathcal{K}$ qui vérifient $(\mathbf{X} = \mathbf{x}) \wedge \varphi$.

EX2 $\mathbf{X}$ est minimal.

EX3 $\mathcal{K}_{(\mathbf{X}=\mathbf{x})\wedge\varphi} \neq \emptyset$, c'est-à-dire les contextes considérés comme possibles par l'*explainee* sont compatibles avec l'explication.

L'explication est dite non triviale si elle vérifie en plus

EX4 $(M, \mathbf{u}') \models \neg(\mathbf{X} = \mathbf{x})$ pour certains contextes $\mathbf{u}' \in K_\varphi$.

L'ensemble de contextes $\mathcal{K}$ est déterminé par l'*explainee*. Ainsi, il est possible que cet ensemble soit trop restrictif, c'est-à-dire que les contextes considérés ne soient pas compatibles avec les explications. En effet, s'il n'existe pas de causes suffisantes dans au moins un contexte de $\mathcal{K}$ (c'est-à-dire $\mathcal{K}_{(\mathbf{X}=\mathbf{x})\wedge\varphi} = \emptyset$) alors, il n'y a pas d'explication possible.
Il existe une définition plus générale de la notion d'explication proposée par J. Halpern [8]. Elle permet notamment de remédier au problème mentionné ci-dessus. En effet, dans celle-ci, on prend également en compte le fait que le destinataire de l'explication n'a pas une connaissance parfaite du modèle, et donc l'explication doit apporter une connaissance supplémentaire. Pour cela, on renvoie non seulement une affectation mais également des formules permettant à ce destinataire de mieux comprendre le modèle. Ainsi, si aucune cause suffisante n'existe dans l'ensemble de contextes $\mathcal{K}$ considéré par l'*explainee*, alors renvoyer une formule en plus peut permettre à ce dernier d'élargir l'ensemble $\mathcal{K}$ des contextes possibles.

3 Système abstrait d'argumentation

Cette section rappelle brièvement les principes des systèmes abstraits d'argumentation de P.M. Dung [6] ainsi qu'une définition d'explication [7] pour ce cadre.

3.1 Définition

Un système abstrait d'argumentation est un couple $AF = (A, R)$ telle que :

— A est un ensemble d'arguments,

— R est une relation binaire sur $A \times A$.

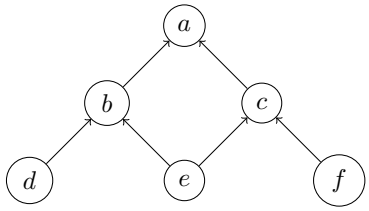


FIGURE 2 – Scénario argumentatif.

On appelle R la relation d'attaque et on dit qu'un argument $a \in A$ attaque $b \in A$ si $(a, b) \in R$ et on écrit $R(a, b)$.
Comme R est une relation binaire à support fini, on peut naturellement représenter un système abstrait d'argumentation sous la forme d'un graphe associé.
Ce formalisme n'impose rien quant à la structure interne d'un argument, ni sur la nature d'une attaque. Ainsi, un argument peut simplement être un énoncé en langage naturel. Il peut également s'agir d'une formule définie dans un certain langage selon des règles, comme dans le cas du système ASPIC+ [15].

Exemple 2. *Cet exemple considère un scénario simple d'aide au dépistage du COVID-19 par un agent. Imaginons que l'utilisateur n'a pas l'impression d'avoir de symptômes particuliers, il s'est juste réveillé avec quelques courbatures. Il décide alors de consulter cet agent. Celui-ci pose un certain nombre de questions à l'utilisateur sur son état de santé. En effet, avoir des courbatures n'est pas suffisant pour justifier d'aller faire un test PCR, un auto-test pourrait par exemple suffire. L'agent lui demande de goûter un condiment au goût prononcé (sel,sucre,vinaigre ...) afin de tester son goût. Enfin, il faut également vérifier s'il est cas contact.*
Leur conversation peut être représentée par le système abstrait d'argumentation suivant, illustré figure 2 :

- $A = \{a$ *: «Test PCR nécessaire»,* b *: «Aucun symptôme»,* c *: «Parcours vaccinal complet»,* d *: «Courbatures»,* e *: «Perte du goût»,* f *: «Je suis cas contact»}*
- $R = \{(b, a), (c, a), (d, b), (e, b), (e, c), (f, c)\}$

Dans le cas où l'utilisateur n'a pas l'impression d'avoir de symptômes particuliers et est vacciné, un test PCR n'est pas nécessaire. Cela est représenté par les deux premières relations d'attaque (b, a) *et* (c, a)*. Toutefois, s'il a des courbatures ou une perte de goût, il n'est plus possible de dire qu'il n'a plus de symptômes (*$(d, b), (e, b)$*). De même, s'il est cas contact ou bien s'il n'a plus de goût, le fait d'avoir un parcours vaccinal complet ne justifie plus de ne pas aller faire de test PCR (*$(e, c), (f, c)$*). En particulier, être vacciné n'empêche pas d'attraper le COVID-19.*
Le graphe présenté en figure 2 représente le cas où l'utilisateur a des courbatures, perdu le goût et est cas contact (d, e, f)*. D'après ce graphe,* a *n'est attaqué que par des arguments non acceptés (car attaqué par des arguments non attaqués) et peut donc être accepté. Ainsi, il faut réaliser un test PCR.*

3.2 Quelques définitions supplémentaires

- On note Att_a^R **l'ensemble des attaquants directs** de a pour la relation R :

$$Att_a^R = \{b \in A \mid R(b, a)\}$$

  Quand une seule relation d'attaque est définie, on notera simplement Att_a.
- Un ensemble S est **sans conflit** s'il n'y pas d'arguments $(a, b) \in S^2$ tel que $(a, b) \in R$:

$$\forall (a, b) \in S^2, (a, b) \notin R$$

- Un argument $a \in A$ est **acceptable** par un ensemble S si S attaque tous les attaquants de a :

$$\forall b \in Att_a, \exists c \in S \cap Att_b$$

- Un ensemble S sans conflit et tel que tous ses éléments sont acceptables par S est dit **admissible** :

$$\forall (a, b) \in S^2, (a, b) \notin R$$
$$\text{et } \forall a \in S, \forall b \in Att_a, \exists c \in S \cap Att_b$$

- Un ensemble S est dit **admissible lié** s'il est admissible et si au moins un de ses arguments est attaqué :

$$S \text{ est admissible et } \exists x \in S \text{ tel que } Att_x \neq \emptyset.$$

  Un tel x est alors appelé un **sujet** de S.

Exemple 2. *(suite) – Dans le cas de l'*AF *défini précédemment, cherchons un ensemble admissible lié* S_{ex} *de sujet* a*. Comme* a *est attaqué par* b*, il faut que* S_{ex} *contienne un attaquant de* b*. Prenons* d *par exemple. Ensuite* d *n'est pas attaqué donc il est acceptable par* S_{ex}*. De plus,* a *est également attaqué par* c*. Il faut donc ajouter un attaquant de* c *à* S_{ex}*. Ajoutons par exemple* e*. L'argument* e *est non attaqué, il est donc lui aussi acceptable par* S_{ex}*. Enfin, tous les attaquants de* a *sont attaqués par un élément de* S_{ex}*,* a *est donc acceptable par* S_{ex}*. On a ainsi construit* $S_{ex} = \{a, d, e\}$*.*
*De la même manière, l'ensemble des ensembles admissibles (**liés** de sujet* a*) est :*
$S_{adm} = \{\{d\}, \{e\}, \{f\}, \{d, e\}, \{d, f\}, \{e, f\}, \{d, e, f\},$
{a,d,e,f}, ***{a,d,f}***, ***{a,e,f}***, ***{a,d,e}***, ***{a,e}***$\}$.

3.3 Explications

Dans cette section, nous reprenons la définition d'explication donnée par X. Fan et F. Toni dans [7].
Soit $x \in A$, une **explication** S de x est un ensemble admissible lié de sujet x.
Une explication de x est dite **compacte** si elle est *minimale* au sens de l'inclusion.
Une explication de x est dite **verbeuse** si elle est *maximale* au sens de l'inclusion.

Exemple 2. *(suite) – L'argument* a *possède ici deux explications compactes : «un test PCR est nécessaire» car l'humain a «une perte de goût», soit* $\{a, e\}$*, ou car il a «des courbatures» et est «cas contact», soit* $\{a, d, f\}$*.*
Il y a également une explication verbeuse : «perte de goût, des courbatures et cas contact» ($\{a, d, e, f\}$*).*

Il existe d'autres définitions de la notion d'explication pour les systèmes d'argumentation. Cependant, dans la plupart des cas, celles-ci nécessitent des notions supplémentaires [2] et sortent du cadre des AAF défini par P.M. Dung [6]. Pour cette raison, nous ne les considérons pas dans le cadre de cet article.

Remarque 2. *Dans le cadre de l'argumentation abstraite, l'objectif n'est pas de modéliser le destinataire de l'explication. Il s'agit plutôt d'une retranscription d'un échange d'arguments entre une ou plusieurs entités. L'explication sert ainsi à justifier pourquoi un argument peut être accepté en renvoyant les différents arguments qui sont intervenus pour défendre ce dernier. Ici, il n'est pas question de contexte. En particulier, il est supposé que chaque entité connaît l'intégralité des arguments et comment ils interagissent.*

Les deux sections suivantes présentent la contribution principale de l'article, à savoir l'équivalence entre les GCA et les systèmes abstraits d'argumentation.

4 Passage des AAF aux GCA

Cette section présente une transformation des graphes argumentatifs en GCA. Nous nous intéresserons aussi à comment la notion d'explication se transporte des AAF aux GCA.

4.1 Transformation proposée

On considère un couple $AF = (A, R)$ et son graphe associé que l'on suppose acyclique.
On associe à chaque argument a une variable booléenne X_a telle que $X_a = 1$ se lit comme « l'argument a est accepté ». Ces variables constituent l'ensemble des variables endogènes.
De plus, pour tous les arguments a non attaqués, on crée une variable booléenne supplémentaire $\tilde{X_a}$. Ces variables forment l'ensemble des variables exogènes.
Formellement, posons :

- $\mathcal{V} = \{X_a \mid a \in A\}$,
- $\mathcal{U} = \{\tilde{X_a} \mid (a \in A) \wedge (Att_a = \emptyset)\}$,
- $\mathcal{F} = \{F_{X_a} \mid X_a \in \mathcal{V}\}$ avec :
  - $\diamond$ $\forall a \in A$ tel que $Att_a \neq \emptyset$, $F_{X_a} = \bigwedge_{b \in Att_a} \neg X_b$,
  - $\diamond$ $\forall a \in A$ tel que $Att_a = \emptyset$, $F_{X_a} = \tilde{X_a}$.

Le triplet $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ est un modèle structurel causal, acyclique et dont les équations structurelles $\mathcal{F}$ n'utilisent pas de disjonctions. Ce modèle M est donc bien un GCA.
Pour chaque argument non attaqué, nous avons proposé de créer deux variables, une endogène et une exogène. Ce dédoublement permet de choisir si un argument non attaqué est accepté ou non par l'intermédiaire de son représentant exogène $\tilde{X_a}$ en l'initialisant à 0 ou à 1. De plus, dans le cadre défini par J. Halpern et J. Pearl [9], seules les variables endogènes peuvent être des causes et donc des explications. Ainsi, avec son représentant endogène, un argument non attaqué pourra lui aussi être une cause.

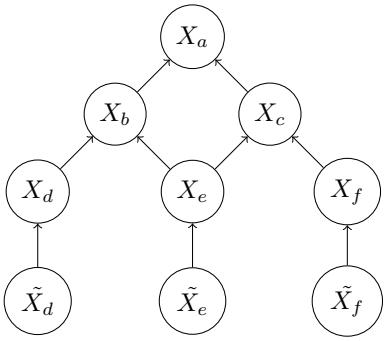


FIGURE 3 – Graphe causal argumentatif issu de la transformation du graphe argumentatif présenté en figure 2.

Exemple 2. *(suite) – L'application de la transformation décrite dans cette section à l'exemple 2 conduit à la construction de six variables endogènes : $\mathcal{V} = \{X_a, X_b, X_c, X_d, X_e, X_f\}$, et de trois variables exogènes, correspondant aux trois arguments non attaqués (d, e, f) : $\mathcal{U} = \{\tilde{X_d}, \tilde{X_e}, \tilde{X_f}\}$.*
Enfin, on transforme les relations d'attaque en équations structurelles. Par exemple, a est attaqué par b et c donc $F_{X_a} = \neg X_b \wedge \neg X_c$.
Avec ces transformations, on obtient le graphe causal argumentatif présenté en figure 3.

On appelle **contexte par défaut** de l'argumentation l'unique contexte $\mathbf{u}^*$ tel que toutes les variables exogènes valent 1. Il représente la situation décrite par le graphe argumentatif dans lequel tous les arguments non attaqués sont acceptés.

4.2 Retour sur les explications

Ces deux formalismes possèdent chacun leur propre définition de la notion d'explication. Avec la transformation que nous avons proposée, il est intéressant de voir si ces définitions sont compatibles ou non.

Proposition 1. *Soit $AF = (A, R)$ un système abstrait d'argumentation, dont le graphe est supposé acyclique. Soit $a^* \in A$ tel qu'il existe un ensemble admissible dont il est le sujet. Soit S une explication compacte de a^*. Soit $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ le graphe causal argumentatif issu de la transformation décrite ci-dessus.*
On définit :

- $\varphi = (X_{a^*} = 1)$,
- $X_{arg} = S \setminus \{a^*\}$ *et* $\mathbf{X} = \{X_a \mid a \in X_{arg}\}$,
- $\mathcal{K}$ *l'ensemble des contextes considérés comme possibles par l'*explainee*. On fait l'hypothèse que le contexte par défaut* $\mathbf{u}^* \in \mathcal{K}$.

Alors $\mathbf{X} = \mathbf{1}$ est une explication, au sens causal, non minimale de φ relative à $\mathcal{K}$, c'est-à-dire $\mathbf{X} = \mathbf{1}$ vérifie **EX1** *et* **EX3** *dans $\mathcal{K}$.*

Dans cette proposition, nous avons réintroduit la notion de destinataire de l'explication. En effet, $\mathcal{K}$ représente l'ensemble des contextes considérés par l'*explainee*. Nous im-

posons seulement que $\mathbf{u}^*$ est inclus dans $\mathcal{K}$. Cette hypothèse semble raisonnable car il s'agit de l'unique contexte considéré lorsque l'on travaille d'un point de vue purement argumentatif.

Démonstration. Montrons que $\mathbf{X} = \mathbf{1}$ vérifie **EX1** et **EX3** dans $\mathcal{K}$.
(EX3) Montrons d'abord que u^* est inclus dans $\mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$.
Par hypothèse, $\mathbf{u}^*$ est inclus dans $\mathcal{K}$.
(i) Montrons par l'absurde que $\mathbf{u}^*$ est inclus dans $\mathcal{K}_{(\mathbf{X}=\mathbf{1})}$.
Supposons que $(M, \mathbf{u}^*) \models \neg(\mathbf{X} = \mathbf{1})$. Alors, $\exists X_a \in \mathbf{X}$ tel que $X_a = 0$. Or $Att_a \neq \emptyset$, donc comme $F_{X_a} = (\bigwedge_{b\in Att_a} \neg X_b), \exists b \in Att_{X_a}$ tel que $X_b = 1$.
Or S est admissible donc $\exists c \in Att_b \cap S$.
Si $Att_c = \emptyset$ alors $X_c = 1$ par définition de u^*. C'est impossible car $X_b = 1$. On arrive donc à une contradiction.
Sinon, comme $X_b = 1$ alors $X_c = 0$. Or $Att_c \neq \emptyset$, donc comme $F_{X_c} = (\bigwedge_{d\in Att_c} \neg X_d), \exists d \in Att_{X_c}$ tel que $X_d = 1$.
Avec S admissible, $\exists e \in Att_d \cap S$.
Si $Att_e = \emptyset$ alors $X_e = 1$ par définition de $\mathbf{u}^*$. C'est impossible car $X_d = 1$. On arrive donc à une contradiction.
Sinon, on peut encore une fois répéter ce raisonnement jusqu'à ce que $Att_e = \emptyset$ car le graphe est fini et acyclique.
Donc $\mathbf{u}^*$ est inclus dans $\mathcal{K}_{\mathbf{X}=\mathbf{1}}$.
(ii) Montrons maintenant que $\mathbf{u}^*$ est inclus dans $\mathcal{K}_{X_{a^*}}$.
Comme S est admissible de sujet a^* et que le graphe est acyclique, alors $\forall b \in Att_{a^*}, \exists c \in X \cap Att_b$. Or $\mathbf{X} = \mathbf{1}$, donc $X_c = 1$ et de fait $X_b = 0$. Ainsi, $\forall b \in Att_{a^*}, X_b = 0$ et donc $X_{a^*} = \bigwedge_{b\in Att_{a^*}} \neg 0 = 1$.
Donc $\mathbf{u}^*$ est inclus dans $\mathcal{K}_{X_{a^*}}$.
Ainsi, $\mathbf{u}^*$ est inclus dans $\mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$, et donc cet ensemble est non vide et **EX3** est satisfait.
(**EX1**) Montrons maintenant que $\mathbf{X} = \mathbf{1}$ est une cause suffisante dans $\mathcal{K}$, c'est-à-dire qu'il vérifie **SC1**, **SC2** et **SC3**, pour tout $\mathbf{u} \in \mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$.
SC4 est une condition de minimalité qui porte sur la cause suffisante mais qui dans le cas des explications est équivalente à **EX2** [8]. Pour cette raison, on ne démontre pas que $\mathbf{X} = \mathbf{1}$ vérifie **SC4**.
Soit $\mathbf{u} \in \mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$.
1) SC1 est vérifié par définition de $\mathbf{u}$.
2) Montrons par l'absurde que **SC3** est vérifié. Soit $\mathbf{u}'$ un contexte tel que $(M, \mathbf{u}') \models [\mathbf{X} = \mathbf{1}]\neg\varphi$.
Comme on a $\neg\varphi$ (c'est-à-dire $X_{a^*} = 0$), d'après $\mathcal{F}$ pour les arguments attaqués (a^* est un sujet de S et donc $Att_{a^*} \neq \emptyset$) $\exists X_b \in \mathcal{V}$, tel que $b \in Att_{a^*}$ et $X_b = 1$.
Or S est admissible donc $\exists c \in S$ tel que $R(c, b)$. De plus, le graphe est acyclique donc $c \neq a^*$ et donc $c \in X$. Comme $\mathbf{X} = \mathbf{1}$, on a en particulier $X_c = 1$ et donc $X_b = 0$ d'après F_{X_b}. On arrive à une *contradiction*.
3) Enfin montrons que **SC2** est bien vérifié :

(i) On construit d'abord une cause effective de φ dans $\mathbf{u}$.

(ii) On montre ensuite que cet ensemble contient bien au moins un élément de $\mathbf{X}$.

(i) Soit $b \in Att_{a^*}$, posons
$\mathbf{Z_b} = \bigcup_{c\in Att_b} \{X_c \mid (M, \mathbf{u}) \models (X_c = 1)\}$.
Comme $\mathbf{u} \in \mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$, alors $X_{a^*} = 1$ et donc $X_b = 0$.
Or S est admissible donc en particulier, $\forall\alpha \in S, \forall\beta \in Att_\alpha, \exists\gamma \in S \cap Att_\beta$.
Comme $b \in Att_{a^*}$ et $a^* \in S$, $Att_b \neq \emptyset$. De plus, $X_b = 0$ donc $\exists X_c \in Att_b$ tel que $X_c = 1$. Ainsi, $X_c \in Z_b$ donc $\mathbf{Z_b}$ est non vide.
Posons ensuite $\mathbf{Z} = \bigcup_{b\in Att_a} \mathbf{Z_b}$.
$\mathbf{Z}$ n'est pas le candidat pour être une cause effective. Toutefois, montrons qu'il vérifie **AC1** et **AC2** :

— **AC1** est vérifié par construction de $\mathbf{Z_b}$.

— Par construction de $\mathbf{Z}$, si on impose $\mathbf{Z} = \mathbf{0}$, alors on a $\forall b \in Att_{a^*}, \forall c \in Att_b, X_c = 0$. Or $F_{X_b} = \bigwedge_{c\in Att_b} \neg X_c = \bigwedge_{b\in Att_a} \neg 0 = 1$.
On a donc bien $(M, \mathbf{u}) \models [\mathbf{Z} = \mathbf{0}]\neg\varphi$ et donc **AC2** est vérifié avec $W = \emptyset$.

Notons $\mathbf{Z^m}$ un sous-ensemble minimal de $\mathbf{Z}$ tel que $(\mathbf{Z^m} = \mathbf{1})$ vérifie **AC1** et **AC2**. Il est bien défini et est non vide car $(\mathbf{Z} = \mathbf{1})$ vérifie **AC1** et **AC2**. De plus, $\mathbf{Z^m}$ vérifie **AC3** par définition. On a donc construit $\mathbf{Z^m}$ tel que $(\mathbf{Z^m} = \mathbf{1})$ vérifie **AC1**, **AC2** et **AC3** c'est-à-dire que $(\mathbf{Z^m} = \mathbf{1})$ est une cause effective de φ.
(ii) Prouvons maintenant que l'on peut construire une cause effective $(\mathbf{Z}' = \mathbf{z}')$ de φ telle que $\mathbf{Z}' \cap \mathbf{X} \neq \emptyset$.
Si $\mathbf{Z^m} \cap \mathbf{X} \neq \emptyset$ alors $\mathbf{Z}' = \mathbf{Z^m}$ convient.
Sinon, c'est-à-dire si $\mathbf{Z^m} \cap \mathbf{X} = \emptyset$, soit $b \in Att_{a^*}$:

— $\exists X_c \in \mathbf{Z^m}$ tel que $c \in Att_b$, $(M, \mathbf{u}) \models (X_c = 1)$ et $X_c \notin \mathbf{X}$.

— Comme S est admissible et le graphe est acyclique, $\exists X_{c'} \in \mathbf{X}$ tel que $c' \in Att_b$. De plus, $\mathbf{Z^m} \cap \mathbf{X} = \emptyset$, donc $X_{c'} \notin \mathbf{Z^m}$. Enfin, comme $\mathbf{u} \in \mathcal{K}$ on a $(M, \mathbf{u}) \models (X_{c'} = 1)$.

Posons $\mathbf{Z^{m\prime}} = (\mathbf{Z^m} \setminus \{X_c\}) \cup \{X_{c'}\}$. $\mathbf{Z^{m\prime}}$ vérifie aussi **AC1** et **AC2**. Comme $\mathbf{Z^m}$ est minimal par construction, alors si $\mathbf{Z^{m\prime}}$ ne l'est pas, $\exists \mathbf{Z}' \subseteq \mathbf{Z^{m\prime}}$ tel que $\mathbf{Z}' \nsubseteq \mathbf{Z^m}$. Or $\mathbf{Z^{m\prime}} \setminus \mathbf{Z^m} = \{X_{c'}\}$ donc $X_{c'} \in \mathbf{Z}'$ et on a donc $\mathbf{Z}' \cap \mathbf{X} \neq \emptyset$.
On a donc construit un ensemble $\mathbf{Z}'$ vérifiant **AC1** et **AC2**, minimal pour l'inclusion (**AC3**) et tel que $\mathbf{Z}' \cap \mathbf{X} \neq \emptyset$ on a donc vérifié **SC2**.
On a montré que $\mathbf{X} = \mathbf{1}$ vérifie **EX1** et **EX3**, donc $\mathbf{X} = \mathbf{1}$ est une explication (au sens causal) non minimale de φ. □

Exemple 2. *(suite) – Reprenons l'exemple 2 et sa transformation associée présentée en figure 3.*

(i) *Posons* $a^* = a$, $S = \{a, d, f\}$, $\mathbf{X} = \{X_d, X_f\}$ *et* $\varphi = (X_a = 1)$.
On a $\mathbf{u}^* = (\tilde{X_d} = 1, \tilde{X_e} = 1, \tilde{X_f} = 1)$. *Donc* $(M, \mathbf{u}^*) \models (X_d = 1, X_e = 1, X_f = 1)$. *En particulier, on a bien* $(M, \mathbf{u}^*) \models (\mathbf{X} = \mathbf{1})$.
On a également $X_b = 0$ *et* $X_c = 0$ *et enfin* $X_a = 1$. *On a bien* $(M, \mathbf{u}^*) \models \varphi$.
On a donc bien $\mathbf{u}^* \in \mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$ *et en particulier* ***EX3*** *est bien vérifié.*

(ii) *Soit $\mathcal{K}$ un ensemble de contextes, et $\mathbf{u} \in \mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$ ($\mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$ est non vide car $\mathbf{u}^* \in \mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$).*
On a tout d'abord $(M,\mathbf{u}) \models (\mathbf{X} = \mathbf{1}) \wedge \varphi$ car $\mathbf{u} \in \mathcal{K}_{(\mathbf{X}=\mathbf{1})\wedge\varphi}$.
Soit $\mathbf{u}' \in \mathcal{K}$. On a $(M,\mathbf{u}') \models [\mathbf{X} = \mathbf{1}](X_b = 0 \wedge X_c = 0)$ et donc $(M,\mathbf{u}') \models [\mathbf{X} = \mathbf{1}](X_a = 1)$.
Posons $\mathbf{Y} = \{X_e\}$ et $X = \{X_d\}$. Avec $X = 0 \wedge \mathbf{Y} = \mathbf{0}$, on a $X_b = 1$ et donc $X_a = 0$. Ainsi, on a $(M,\mathbf{u}) \models [X = 0 \wedge \mathbf{Y} = \mathbf{0}]\neg\varphi$.
*On a montré que $\forall \mathbf{u} \in \mathcal{K}$, $\mathbf{X} = \mathbf{1}$ est une cause suffisante de φ, c'est-à-dire que **EX1** est bien vérifié.*

5 Passage des GCA aux AAF

Dans cette section, nous proposons la transformation réciproque ainsi qu'une démonstration de l'équivalence entre ces deux cadres formels.

5.1 Transformation proposée

On considère un graphe causal argumentatif de triplet $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$. On adopte la transformation suivante :

— $A' = \{a \mid X_a \in \mathcal{V}\}$,

— Pour la construction de la relation d'attaque : soit $(X_a, X_b) \in \mathcal{V}^2$, on pose $\mathbf{Y} = \mathcal{V} \setminus \{X_a, X_b\}$. Si on a $(M,\mathbf{u}) \models [X_b = 1, \mathbf{Y} = \mathbf{0}](X_a = 0)$ alors $(b,a) \in R'$.

Exemple 2. *(suite) – Reprenons le graphe causal argumentatif présenté en figure 3.*
On a $\mathcal{V} = \{X_a, X_b, X_c, X_d, X_e, X_f\}$. On pose donc $A' = \{a, b, c, d, e, f\}$.
Soit $\mathcal{K}$ un ensemble de contextes. Soit $\mathbf{u} \in \mathcal{K}$.
On a $F_{X_a} = \neg X_b \wedge \neg X_c$. En particulier, $(M,\mathbf{u}) \models [X_v = 1](X_a = 0)$ avec $X_v \in \{X_b, X_c\}$.
Cela reste vrai en imposant en plus $\mathbf{Y} = \mathbf{0}$ avec $\mathbf{Y}$ construit comme dans la transformation. Donc $(b,a) \in R'$ et $(c,a) \in R'$.
En appliquant le même raisonnement avec toutes les équations structurelles de $\mathcal{F}$ on a $\{(d,b),(e,b),(e,c),(f,c)\} \in R'$.
Posons $\mathbf{Y} = \mathcal{V} \setminus \{X_a, X_v\}$ avec $v \in \{d, e, f\}$.
On a $(M,\mathbf{u}) \models [X_v = 1, \mathbf{Y} = 0](X_a = 1)$. En effet, toutes les équations structurelles ont été remplacées par $F_X = 0$ sauf pour X_a et X_v : $X_v = 1$ et $F_{X_a} = \neg X_b \wedge \neg X_c$ donc $X_a = \neg 0 \wedge \neg 0 = 1$.
Donc $(v,a) \notin R'$.
On a donc $R' = \{(b,a),(c,a),(d,b),(e,b),(e,c),(f,c)\}$.

5.2 Équivalence entre AAF et GCA

Proposition 2. *Soit $AF = (A,R)$ un système abstrait d'argumentation, $M = (\mathcal{U},\mathcal{V},\mathcal{F})$ le graphe causal argumentatif associé à AF par la transformation décrite dans la section 4 et $AF' = (A',R')$ le système d'argumentation associé à M avec la transformation ci-dessus. Alors :*

$$AF = AF'.$$

Démonstration. Soit $AF = (A,R)$ un système abstrait d'argumentation, $M = (\mathcal{U},\mathcal{V},\mathcal{F})$ un graphe causal argumentatif associé à AF et $AF' = (A',R')$ le système d'argumentation associé à M.
Montrons que $AF = AF'$ c'est-à-dire $A = A'$ et $R = R'$.
• On a par construction $A = A'$.
• Montrons que $R = R'$ par double inclusion.

1. Soit $(a,b) \in A^2$ tel que $R(b,a)$.
   On a par définition $X_a = \neg X_b \wedge (\bigwedge_{c \in Att_a \wedge c \neq b} \neg X_c)$.
   En particulier, si $X_b = 1$ alors on a :
   $$X_a = 0 \wedge (\bigwedge_{c \in Att_a \wedge c \neq b} \neg X_c) = 0.$$
   Ainsi, pour tout contexte $\mathbf{u}$, on a $(M,\mathbf{u}) \models [X_b = 1, \mathbf{Y} = \mathbf{0}](X_a = 0)$ avec $\mathbf{Y} = \mathcal{V} \setminus \{X_a, X_b\}$, donc $(b,a) \in R'$ et $R \subseteq R'$.
2. Soit $(a',b') \in A'^2$ tel que $b' \in Att^{R'}_{a'}$.
   Soit $\mathbf{Y} = \mathcal{V} \setminus \{X_{a'}, X_{b'}\}$. On a par définition
   $$(M,\mathbf{u}) \models [X_{b'} = 1, \mathbf{Y} = \mathbf{0}](X_{a'} = 0)$$
   Or $A = A'$, donc $\forall \alpha \in A$, $X_\alpha = X_{\alpha'}$. En particulier $(M,\mathbf{u}) \models [X_{b'} = 1, \mathbf{Y} = \mathbf{0}](X_a = 0)$ et donc $Att^R_a \neq \emptyset$.
   De plus, $F_{X_a} = \bigwedge_{z \in Att^R_a} \neg X_z$. Si $b' \notin Att^R_a$ alors avec $\mathbf{Y} = \mathbf{0}$ on a $F_{X_{a'}} = \bigwedge_{\beta \in Att^R_a} \neg 0 = 1$, ce qui contredit l'hypothèse.
   On a donc $b' \in Att^R_a$ et $R' \subseteq R$.

On a donc montré par double inclusion que $R = R'$. □

6 Conclusion et perspectives

Nous avons mis en évidence dans cet article l'équivalence qui existe entre les graphes causaux argumentatifs et les systèmes abstraits d'argumentation. Nous avons également proposé une transformation permettant de passer de l'un à l'autre. Cela permet de pouvoir utiliser le meilleur des deux mondes.
D'une part, la notion de contexte présent dans les modèles structurels causaux permet de faire varier les valeurs des arguments et offre donc un cadre dynamique. De plus, elle permet de tenir compte des connaissances des agents. Les travaux de J. Pearl et J. Halpern [10] introduisent également la notion de pouvoir explicatif et d'explication partielle, ainsi qu'une définition générale permettant en plus de donner une connaissance du modèle à l'*explainee*. D'autre part, les systèmes d'argumentation proposent un cadre plus naturel pour modéliser les situations d'interaction, pouvant faciliter sa mise en pratique pour des systèmes en interactions avec des humains. Ainsi, une démarche pourrait être de modéliser dynamiquement une interaction avec un AAF, de calculer un résultat ou une action puis d'effectuer la transformation en GCA afin de générer des explications aux propriétés voulues.

Toutefois, se limiter à la notion d'attaque entre arguments peut conduire à des situations un peu étranges dans lesquelles deux arguments n'interagissent pas entre eux alors qu'ils semblent clairement liés. Une première solution consiste à prendre la négation d'un de ces arguments. Une autre, beaucoup moins maladroite, est d'ajouter une relation binaire supplémentaire comme par exemple la relation de support [4]. Il serait alors intéressant de pouvoir intégrer ce genre de relation dans les équations structurelles des GCA. Cela nécessite de choisir un critère de décision en cas d'attaque et de support [4] : si un argument est attaqué par un argument non attaqué et supporté par un argument non attaqué, l'argument est-il accepté, non accepté, indéterminé ?

Les travaux de G. Brewka et al. [3] proposent une généralisation des AAF de P.M. Dung [6] appelée *Abstract Dialectical Framework* (ADF). Ce cadre formel remplace les relations d'attaques par des conditions d'acceptabilités des arguments, souvent sous la forme de formules logiques. On obtient donc une représentation avec des variables à valeurs binaires ou ternaires (arguments acceptés, refusés, ou indécis) dont la valeur est régie par des formules logiques. Cela fait évidemment écho à notre transformation des AAF en GCA mais plus généralement aux modèles structurels causaux de J. Halpern et J. Pearl [9]. Il serait donc intéressant d'explorer ce que chacune des approches propose et ce qu'il est possible de faire dans la continuation de ce que nous proposons avec les GCA.

Il existe également une formulation floue pour ces deux cadres [1, 11] qui apporte des outils permettant une représentation plus humaine des interactions, avec par exemple l'ajout d'un degré d'attaque et de support. L'étude des ces cadres et leur comportement par rapport à la transformation que nous proposons est une piste à venir pour étendre notre approche.

Enfin, l'objectif de tels cadres est de proposer des explications adaptées aux humains afin d'augmenter la confiance de ces derniers envers les systèmes d'IA mais également de faciliter les interactions entre humain et machine. Ainsi, un autre enjeu des travaux à venir consiste à tester ces cadres formels et la transformation proposée sur un exemple plus complet et complexe d'interaction entre humain et machine puis de faire évaluer subjectivement ces modèles par des utilisateurs humains.

Références

[1] Isabelle Bloch and Marie-Jeanne Lesot. Vers une formulation floue des explications par contraste. In *Rencontres Francophones sur la Logique Floue et ses Applications (LFA)*, 2021.

[2] AnneMarie Borg and Floris Bex. A basic framework for explanations in argumentation. *IEEE Intelligent Systems*, 36(2) :25–35, 2021.

[3] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes P Wallner, and Stefan Woltran. Abstract dialectical frameworks. an overview. *IfCoLog Journal of Logics and their Applications*, 4(8) :2263–2317, 2017.

[4] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. On the Acceptability of Arguments in Bipolar Argumentation Frameworks. In *European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 378–389. Springer, 2005.

[5] Christophe Denis and Franck Varenne. Interprétabilité et explicabilité pour l'apprentissage machine : entre modèles descriptifs, modèles prédictifs et modèles causaux. Une nécessaire clarification épistémologique. In *Conférence Nationale en Intelligence Artificielle (CNIA)*, pages 60–68, 2019.

[6] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77 :321–357, 1995.

[7] Xiuyi Fan and Francesca Toni. On Computing Explanations in Abstract Argumentation. In *ECAI*, volume 263, pages 1005–1006, 2014.

[8] Joseph Y. Halpern. *Actual Causality*. MIT Press, 2016.

[9] Joseph Y. Halpern and Judea Pearl. Causes and Explanations : A Structural-Model Approach. Part I : Causes. *The British Journal for the Philosophy of Science*, 56(4) :843–887, 2005.

[10] Joseph Y. Halpern and Judea Pearl. Causes and Explanations : A Structural-Model Approach. Part II : Explanations. *The British Journal for the Philosophy of Science*, 56(4) :889–911, 2005.

[11] Jeroen Janssen, Martine De Cock, and Dirk Vermeir. Fuzzy argumentation frameworks. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU)*, pages 513–520, 2008.

[12] Prashan Madumal, Tim Miller, Liz Sonenberg, and Frank Vetere. Explainable Reinforcement Learning Through a Causal Lens. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

[13] Tim Miller. Explanation in Artificial Intelligence : Insights from the Social Sciences. *Artificial Intelligence*, 267 :1–38, 2019.

[14] Tim Miller. Contrastive explanation : A structural-model approach. *Knowledge Engineering Review*, 36 :E14, 2021.

[15] Sanjay Modgil and Henry Prakken. The ASPIC+ framework for structured argumentation : a tutorial. *Argument & Computation*, 5(1) :31–62, 2014.

[16] Laurie Ann Paul and Ned Hall. *Causation : A User's Guide*. Oxford University Press, 2013.

[17] Kristijonas Čyras, Antonio Rago, Emanuele Albini, Pietro Baroni, and Francesca Toni. Argumentative XAI : A survey. In Zhi-Hua Zhou, editor, *Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4392–4399, 2021.