



Conférence Nationale d'Intelligence Artificielle Année 2025

Nathalie Abadie, Ghislain Ateazing, Grégory Bonnet, Tristan Cazenave, Antoine Cornuéjols, Vincent Guigue, Jean-Guy Mailly, Fleur Mougin, Pascal Préa, François Schwarzentruher, et al.

► To cite this version:

Nathalie Abadie, Ghislain Ateazing, Grégory Bonnet, Tristan Cazenave, Antoine Cornuéjols, et al.. Conférence Nationale d'Intelligence Artificielle Année 2025. Association Française pour l'Intelligence Artificielle, 2025. hal-05409313

HAL Id: hal-05409313

<https://hal.science/hal-05409313v1>

Submitted on 10 Dec 2025

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

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



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



Conférence Nationale d'Intelligence Artificielle Année 2025

Sections Spéciales

Nathalie Abadie
Ghislain Ateamezing
Grégory Bonnet
Tristan Cazenave
Antoine Cornuéjols
Vincent Guigue
Jean-Guy Mailly
Fleur Mougin
Pascal Préa
François Schwarzentruher
Danai Symeonidou
Hélène Verhaeghe
Anaëlle Wilczynski

Coordination AFIA

Grégory Bonnet
Thomas Guyet
Benoit Le Blanc
Dominique Longin
Fatiha Saïs
Ahmed Samet



Conférence Nationale
d'Intelligence Artificielle
Année 2025

Actes CNIA 2025

Conférence Nationale d'Intelligence Artificielle

Éditeurs : Président(e)s des Comités de Programme de PFIA 2025

Tables des matières

Dijon, au cœur de l'Intelligence Artificielle	
de demain	3
Comité de programmation	5
Comités de programme	5
Comité d'organisation	10
Logo partenaires PFIA 2025	11
Romain Barrault, Cédric Pralet, Gauthier Picard, Eric Sawyer (CNIA)	
<i>Hybridizing machine learning and optimization for planning satellite observations</i>	13
Swann Bessa, Darius Dabert, Max Bourgeat, Louis-Martin Rousseau, Quentin Cappart (JFPC)	
Apprentissage de bornes duales valides en programmation par contrainte : décomposition lagrangienne amplifiée avec apprentissage auto-supervisé (article résumé)	23
Victor Charpenay (IC)	
Génération et validation de données structurées	27
Didier Dubois, Henri Prade (JIAF-JFPDA)	
40 ans de recherche en logique possibiliste - Une vue d'ensemble	35
Lilian Hollard, Lucas Mohimont, Nathalie Gaveau, Luiz Angelo Steffenel (APIA)	
LeYOLO, nouvelle architecture embarquée pour la détection d'objets	45
Noor Khalal, Abdallah Alaa-Eddine Djamai, Imed Keraghel, Mohamed Nadif (SFC)	
Vers un <i>clustering</i> plus équilibré : augmentation de données par GMM et LLM	55
Christophe Lecoutre (JFPC)	
Search is not Dead!	61
Maxime Morge (JFSMA)	
Systèmes multiagents et agents autonomes génératifs : état de l'art des connaissances et des ressources	65
Sena Ozgunay, Louise Travé-Massuyès, Jean-Michel Loubes, Raul Sena Ferreira (RJCIA)	
<i>Detecting anomalies using graph neural networks: a review</i>	75
Julien Soulé, Jean-Paul Jamont, Michel Occello, Louis-Marie Traonouez, Paul Théron (JFSMA)	
Une approche organisationnelle pour améliorer l'explicabilité et le contrôle dans l'apprentissage par renforcement multi-agent	83
Hénoïk Willot, Jean-Louis Farges, Gauthier Picard, Philippe Pavero (JFPC)	
Gestion des ressources de communication pour satellites dans une fédération de constellations d'observation de la Terre (article résumé)	93
Zhifei Zheng, Sami Cherif, Rui Sá Shibasaki, Chu-Min Li (JFPC)	
Approches exactes pour le problème de satisfiabilité diversifiée	95

Dijon, au cœur de l'Intelligence Artificielle de demain

Lylia Abrouk

Professeur, LIB, Université Bourgogne-Franche-Comté
Président du Comité d'Organisation de la Plate-Forme IA 2025

Thomas Guyet, Fatiha Saïs, Ahmed Samet

Chercheur INRIA Lyon, Professeur Université de Paris-Saclay, Maître de Conférence INSA Strasbourg,
Membres du Conseil d'Administration de l'AFIA,
Co-Présidents du Comité de Programmation de la Plate-Forme IA 2025

L'AFIA et le laboratoire LIB s'associent pour organiser du 30 juin au 4 juillet 2025 la 18^e Plate-Forme Intelligence Artificielle – [PFIA 2025](#). Les précédentes éditions se sont tenues à Palaiseau (1999), Grenoble (2001), Laval (2003), Nice (2005), Grenoble (2007), Hammamet (2009), Chambéry (2011), Lille ([2013](#)), Rennes ([2015](#)), Caen ([2017](#)), Nancy ([2018](#)), Toulouse ([2019](#)), Angers ([2020](#)), Bordeaux ([2021](#)), Saint-Étienne ([2022](#)), Strasbourg ([2023](#)) et La Rochelle ([2024](#)).

La plate-forme IA constitue un point de rencontre unique pour la communauté IA permettant de rapprocher les différentes disciplines qui la composent et d'établir des passerelles entre elles. À cette intention, la Plate-forme IA s'adresse à l'ensemble de la communauté francophone en IA pour aborder des problématiques communes.

Dijon compte des acteurs de premier plan dans le domaine de l'Intelligence Artificielle, avec de réelles compétences tant dans son cœur scientifique et technique que dans son intégration à d'autres technologies ou dans ses applications dans différents domaines.

Pour son édition 2025, la Plate-Forme IA héberge les 9 conférences suivantes :

APIA	11 ^e conférence nationale sur les Applications Pratiques de l'IA	G. Ateamezing et N. Abadie
CAp	27 ^e Conférence sur l'Apprentissage automatique	A. Cornuéjols et V. Guigue
CNIA	28 ^e Conférence Nationale d'Intelligence Artificielle	T. Cazenave
IC	36 ^{es} journées francophones d'Ingénierie des Connaissances	F. Mougin
JFPC	20 ^{es} Journées Francophones de Programmation par Contraintes	H. Verhaeghe
JFSMA	33 ^{es} Journées Francophones sur les Systèmes Multi-Agents	G. Bonnet
JIAF	19 ^{es} Journées d'Intelligence Artificielle Fondamentale	J.-G. Mailly, F. Schwarzentru- ber et A. Wilczynski
RJCIA	23 ^{es} Rencontres des Jeunes Chercheurs en Intelligence Artificielle	D. Symeonidou
SFC	32 ^{es} rencontres de la Société Francophone de Classification	P. Préa

les 4 journées suivantes :

Défense & IA	journée « Défense & IA », en partenariat avec l' INRIA et l' ONERA
Humanité & IA	journée « Humanité & IA », en partenariat avec l'action Sources aux Données Historiques en Humanités Numériques du GdR MADICS
Société & IA	journée « Société & IA », en partenariat avec le collège Humanités, Société et Intelligence Artificielle de l' AFIA et le cluster IA MIAI de Grenoble
IA en BFC	journée « IA en Bourgogne-Franche-Comté », en partenariat avec l' Université Bourgogne Europe et SATT Sayens

6 ateliers thématiques ([ACAI](#), [GDR RADIA](#), [IN-OVIVE](#), [Jeux & IA](#), [MAFTEC](#), [Santé & IA](#)),

ainsi que plusieurs [tutoriels hébergés](#) sur les thèmes :

- Comprendre et Maîtriser l'Incertain dans les Réseaux de Neurones : Approches Bayésiennes, Ensemblistes et Applications Pratiques (G. Franchi, O. Laurent, A. Lafage) ;
- Introduction à la modélisation en programmation par contraintes (C. Lecoutre) ;
- Coordination multi-agents : approches coopératives par optimisation distribuée et enchères (G. Picard) ;
- Apprentissage continu appliqué à la classification d'images (E. Feillet, C. Hudelot, A. Popescu) ;
- Explicabilité pour le NLP : théorie et pratique (W. Ouerdane, A. Poché, J.P. Poli) ;
- Apprendre et enseigner à l'aune de l'IA générative (N. Abchiche Mimouni, S. Tronçon).

Dix collègues ont également été invités pour des conférences en début de chaque demi-journée :

Vaishak Belle	University of Edinburgh	UK
Thomas Fel	Université Harvard	USA
Rémi Flamary	École Polytechnique	France
Luis Galárraga	IRISA	France
Christian Hennig	University of Bologna	Italie
Emiliano Lorini	CNRS, Université de Toulouse	France
Simon Lucas	University of London	UK
Nardine Z. Osman	Spanish National Research Council	Espagne
Louis-Martin Rousseau	Polytechnique Montréal	Canada
Marieke van Erp	Royal Netherlands Academy of Arts and Sciences Humanities Cluster	Pays-Bas

plus les 3 lauréats du prix de thèse de l'AFIA :

Thomas Fel	(1 ^{er} prix)	Université de Toulouse	France
Edwige Cyffers	(Accessit <i>ex-aequo</i>)	Université de Lille	France
Marc Jourdan	(Accessit <i>ex-aequo</i>)	Université de Lille	France

Nous remercions les présidents et membres de comités de programme des conférences et journées hébergées, les orateurs, les membres des comités de programmation et d'organisation, nos partenaires institutionnels et industriels, ainsi que tous les participants, pour leurs contributions précieuses à la réussite de cette plate-forme.

L'ensemble des informations sur ces événements est disponible à partir du site de la Plate-Forme IA 2025 (<https://pfia2025.u-bourgogne.fr>).

Comité de programmation

Conseil d'Administration de l'Association Française pour l'Intelligence Artificielle

Président

- Thomas GUYET, AIstroSight, Inria, Centre de Lyon.

Membres

- Davy Monticolo, Université de Lorraine, ERPI ;
- Ahmed Samet, INSA Strasbourg, ICube.

Comités de programme

Conférence nationale sur les Applications Pratiques de l'IA (APIA)

Présidence

- Ghislain Ateazing (European Union Agency for Railways) ;
- Nathalie Abadie (LASTIG, Université Gustave Eiffel, IGN-ENSG).

Membres

- Florence Amardeilh (Elzeard) ;
- Fabien Amarger (Digitanie) ;
- Nicolas Audebert (LASTIG, Université Gustave Eiffel, IGN-ENSG) ;
- Nathalie Aussenac (IRIT, CNRS) ;
- Alain Berger (Ardans) ;
- Sandra Bringay (LIRMM) ;
- Xavier Briottet (Office National d'Etudes et de Recherches Aérospatiales) ;
- Stephan Brunessaux (Sensei Consult) ;
- Davide Buscaldi (LIPN, Université Paris 13, Sorbonne Paris Cité) ;
- Gaëtan Caillaut (Lingua Custodia) ;
- Bruno Carron (Airbus) ;
- Laurent Cervoni (Talan) ;
- Gaël de Chalendar (CEA LIST) ;
- Nicolas Chauvat (Logilab) ;
- Caroline Chopinaud (Hub France IA) ;
- Franck Cotton (Making Sense) ;
- Yves Demazeau (CNRS - LIG) ;
- Sylvie Despres (Laboratoire d'Informatique Médicale et de BIOinformatique) ;
- Gayo Diallo (ISPED & LABRI, Université de Bordeaux) ;
- Valentina Dragos (Office National d'Etudes et de Recherches Aérospatiales) ;
- Guillaume Dubuisson Duplessis (EDF) ;
- Françoise Fogelman-Soulié (Hub France IA) ;
- Nicolas Gonthier (LASTIG, Université Gustave Eiffel, IGN-ENSG) ;

- Céline Hudelot (Ecole Centrale Paris) ;
- Arnaud Lallouet (Huawei) ;
- Christine Largouët (Irisa - Agrocampus Ouest) ;
- Christelle Launois (Société Générale) ;
- Dino Lenco (INRAE) ;
- Dominique Lenne (Heudiasyc, Université de Technologie de Compiègne) ;
- Mustapha Lebbah (Université Paris-Saclay - UVSQ Versailles Campus) ;
- Sylvain Mahé (EDF R & D) ;
- Juliette Mattioli (Thales) ;
- Mathieu Roche (CIRAD, TETIS) ;
- Catherine Roussey (INRAE) ;
- Céline Rouveirol (LIPN, Université Paris 13) ;
- Brigitte Trousse (INRIA).

Conférence Nationale en Intelligence Artificielle (CNIA)

Présidence

- Tristan Cazenave.

Membres

- Ines Alaya - LAMSADE, Université Paris-Dauphine ;
- Jérôme Azé - LIRMM - CNRS UMR 5506 ;
- Nicolas Beldiceanu - IMT Atlantique (LS2N) ;
- Armelle Brun - LORIA - Université de Lorraine ;
- Olivier Boissier - LIMOS UMR 6158 CNRS ;
- Grégory Bonnet - Université de Caen Normandie ;
- Sandra Bringay - LIRMM ;
- Benjamin Dalmas - Computer Research Institute of Montreal ;
- Yves Demazeau - CNRS - LIG ;
- Arnaud Doniec - IMT Nord Europe ;
- Louis Falissard - LIASD, Paris 8 ;
- Olivier Goudet - LERIA ;
- Jean-Gabriel Ganascia - LIP6, Sorbonne Université ;
- Christophe Guettier - SAFRAN ;
- Nicolas Jouandeau - Université Paris 8 ;
- Camille Kurtz - Université de Paris ;
- Jérôme Lang - LAMSADE, Université Paris-Dauphine ;
- Nicolas Lachiche - University of Strasbourg ;
- Florence Le Ber - ICube ;
- Marie-Jeanne Lesot - LIP6 - Sorbonne Université ;
- Nicolas Maudet - LIP6, Sorbonne Université ;
- Denis Pallez - Université Côte d'Azur ;
- Damien Pellier - Laboratoire d'Informatique de Grenoble ;
- Ana Roxin - Université Bourgogne Europe, LIB EA7534 ;

- Pascale Sébillot - Université de Rennes, CNRS, Inria / IRISA ;
- Laurent Vercouter - LITIS lab, INSA de Rouen ;
- Bruno Zanuttini - GREYC, Normandie Univ. ; UNICAEN, CNRS, ENSICAEN.

Journées francophones d'Ingénierie des Connaissances (IC)

Présidence

- Fleur Mougin, BPH - Inserm U1219, Université de Bordeaux.

Membres

- Xavier Aimé, Cogsonomy ;
- Nathalie Abadie, LaSTIG, IGN France ;
- Marie-Hélène Abel, Heudiasyc, Université de Technologie de Compiègne ;
- Yamine Ait Ameer, IRIT, Toulouse INP ;
- Nathalie Aussenac-Gilles, IRIT, CNRS ;
- Bruno Bachimont, Heudiasyc, University de Technologie de Compiègne ;
- Nacéra Bennacer-Seghouani, LRI, Centrale Supélec ;
- Nathalie Bricon-Souf, IRIT, Université de Toulouse ;
- Sandra Bringay, LIRMM, Université Paul-Valéry Montpellier ;
- Davide Buscaldi, LIPN, Université Sorbonne Paris Nord ;
- Sylvie Calabretto, LIRIS, INSA de Lyon ;
- Pierre-Antoine Champin, LIRIS, Université Claude Bernard Lyon 1 ;
- Jean Charlet, AP-HP & Inserm U1142 ;
- Victor Charpenay, LIMOS, MINES Saint-Etienne ;
- Adrien Coulet, HeKA - Inserm & Inria, Université Paris Cité ;
- Jérôme David, mOeX - LIG & Inria, Université Grenoble Alpes ;
- Sylvie Despres, LIMICS, Université Sorbonne Paris Nord ;
- Gayo Diallo, BPH - Inserm U1219, Université de Bordeaux ;
- Catherine Faron, I3S, Université Côte d'Azur ;
- Béatrice Fuchs, LIRIS, Université Jean Moulin Lyon III ;
- Frédéric Fürst, MIS, Université de Picardie Jules Verne ;
- Alban Gaignard, Institut du Thorax, Nantes Université ;
- Mounira Harzallah, LS2N, Nantes Université ;
- Nathalie Hernandez, IRIT, Université Toulouse Jean Jaurès ;
- Liliana Ibanescu, INRAE, AgroParisTech ;
- Sébastien Iksal, LIUM, Le Mans Université ;
- Antoine Isaac, Europeana & Vrije Universiteit Amsterdam ;
- Clément Jonquet, INRAE & LIRMM ;
- Mouna Kamel, Université de Perpignan Via Domitia ;
- Gilles Kassel, Université de Picardie Jules Verne ;
- Maxime Lefrançois, LIMOS, MINES Saint-Etienne ;
- Dominique Lenne, Heudiasyc, Université de Technologie de Compiègne ;
- Pascal Molli, LS2N, Nantes Université ;
- Jérôme Nobécourt, LIMICS, Université Sorbonne Paris Nord ;

- Nathalie Pernelle, LIPN, Université Sorbonne Paris Nord ;
- Cédric Pruski, Luxembourg Institute of Science and Technology ;
- Joe Raad, LISN, Université Paris Saclay ;
- Sylvie Ranwez, EuroMov DHM, Ecole des Mines d’Alès ;
- Catherine Roussey, INRAE ;
- Fatiha Saïs, LISN, Université Paris Saclay ;
- Karim Sehaba, LIRIS, Université Lumière Lyon 2 ;
- Lina F. Soualmia, LITIS & LIMICS, Normandie Universités ;
- Konstantin Todorov, LIRMM, Université de Montpellier ;
- Cassia Trojahn, mOeX - LIG & Inria, Université Grenoble Alpes ;
- Raphaël Troncy, EURECOM ;
- Danai Symeonidou, INRAE ;
- Haïfa Zargayouna, LIPN, Université Sorbonne Paris Nord.

Journées d’Intelligence Artificielle Fondamentale & Journées Francophones sur la Planification, la Décision et l’Apprentissage (JIAF-JFPDA)

Présidence

- Jean-Guy Mailly (IRIT, Université de Toulouse, UT Capitole) ;
- François Schwarzentruher (IRISA, ENS Rennes) ;
- Anaëlle Wilczynski (MICS, CentraleSupélec, Université Paris-Saclay).

Membres

- Francesco Belardinelli (Imperial College London) ;
- Aurélie Beynier (LIP6, Sorbonne Université) ;
- Élise Bonzon (LIPADE, Université Paris Cité) ;
- Nadjat Bourdache (GREYC, Université de Normandie) ;
- Olivier Buffet (INRIA / LORIA) ;
- Martin C. Cooper (IRIT - Université Paul Sabatier) ;
- Célia da Costa Pereira (Université Côte d’Azur) ;
- Tiago de Lima (Université d’Artois, CRIL CNRS) ;
- Sylvie Doutre (Université Toulouse Capitole - IRIT) ;
- Florence Dupin de Saint-Cyr (Université Paul Sabatier - IRIT) ;
- Alain Dutech (Loria - Inria) ;
- Sabine Frittella (INSA Centre Val de Loire) ;
- Hugo Gilbert (Lamsade - Université Paris Dauphine) ;
- Raïda Ktari (Aix-Marseille Université) ;
- Jérôme Lang (CNRS, LAMSADE, Université Paris-Dauphine) ;
- Jean Lieber (LORIA - INRIA Lorraine) ;
- Emiliano Lorini (CNRS, IRIT) ;
- Pierre Marquis (CRIL, U. Artois & CNRS - Institut Universitaire de France) ;
- Amedeo Napoli (LORIA Nancy, CNRS - Inria - Université de Lorraine) ;
- Wassila Ouerdane (MICS, CentraleSupélec, Université Paris-Saclay) ;

- Damien Pellier (Laboratoire d’Informatique de Grenoble) ;
- Julien Rossit (Université Paris Cité, LIPADE) ;
- Stéphanie Roussel (ONERA) ;
- Régis Sabbadin (INRA-UBIA) ;
- Vincent Thomas (LORIA) ;
- Srdjan Vesic (CRIL - CNRS) ;
- Thierry Vidal (LGP, UTTOP Tarbes) ;
- Bruno Zanuttini (GREYC, Université de Normandie).

Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA)

Présidence

- Danai Symeonidou - INRAE.

Membres

- Lylia Abrouk - LIB ;
- Vincent Armant - Institut de Recherche pour le Développement ;
- Sandra Bringay - Université de Montpellier 2 ;
- Francesca Bugiotti Paris-Saclay University ;
- Davide Di Pierro - Université de Montpellier ;
- Luis Galárraga - Inria ;
- Nathalie Hernandez - Université de Toulouse ;
- Liliana Ibanescu - Université Paris-Saclay ;
- Pierre Larmande - IRD ;
- Melanie Munch - INRAE ;
- Pierre-Henri Paris - Paris-Saclay University ;
- Nathalie Pernelle - Université Sorbonne Paris Nord ;
- Joe Raad - Vrije Universiteit, Amsterdam, The Netherlands ;
- Catherine Roussey - INRAE ;
- Danai Symeonidou - INRAE ;
- Cassia Trojahn - Université Grenoble Alpes ;
- Katerina Tzompanaki - CY Cergy Paris University ;
- Felipe Vargas-Rojas - IRD.

Comité d'organisation

ICube, Université de Strasbourg

Présidence

- Lylia Abrouk (UBE).

Pilotage

- Hamid Ahaggach (Syartec) ;
- Angélique Bardollet (UBE) ;
- Claire Bourgeois-République (UBE) ;
- Nadine Cullot (UBE) ;
- Alexandrine Delbano (UBE) ;
- Alexis Guyot (Université de Montpellier) ;
- Marinette Savonnet (UBE).

Membres

- Pauline Armary ;
- Olivier Bailleux ;
- Maëlle Beuret ;
- David Camarazo ;
- Ibrahim Diarra ;
- Davide Di Pierro ;
- Lucie Druoton ;
- Laurence Dujourdy ;
- Mahya Faraji ;
- Irène Foucherot ;
- Christian Gentil ;
- Hussam Ghanem ;
- Sarah Ghidalia ;
- Sébastien Guillemin ;
- Matéo Iori ;
- Mahdi Madani ;
- Cyrille Migniot ;
- Clément Poull ;
- Céline Roudet ;
- Ana Roxin ;
- Joël Savelli ;
- Olivier Togni.

Partenaires « Platine »



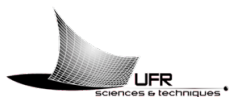
Partenaires « Or »



Partenaires « Bronze »



Ils soutiennent PFIA 2025



Hybridizing Machine Learning and Optimization for Planning Satellite Observations*

Romain Barrault¹, Cédric Pralet¹, Gauthier Picard¹, Eric Sawyer²

¹ DTIS, ONERA, Université de Toulouse, Toulouse, France

² CNES, Toulouse, France

{romain.barrault, gauthier.picard, cedric.pralet}@onera.fr
eric.sawyer@cnes.fr

Résumé

La planification des tâches d'un satellite d'observation terrestre est un problème hautement combinatoire. Cela consiste à calculer régulièrement la séquence d'observations à effectuer par un satellite pour collecter des photographies de points d'intérêt tout en tenant compte des manœuvres temporelles nécessaires pour diriger le satellite vers les points d'intérêts successifs. Pour résoudre un tel problème d'optimisation, nous proposons une nouvelle approche qui exploite des techniques d'apprentissage hors ligne pour approximer la faisabilité de l'ordonnancement pour des ensembles de tâches d'observation au sein d'un algorithme d'optimisation.

Mots-clés

IA hybride, Réseaux de Neurones Artificiels, Recherche à Voisinage Large, Planification, Satellite d'Observation Terrestre.

Abstract

Planning the activities of an Earth observation satellite is a highly combinatorial task. It consists in regularly computing the sequence of observations to be performed by a satellite to collect images of candidate points of interest (POIs), while taking into account the time-dependent maneuvers required to point the satellite to the successive POIs. To solve such a recurrent optimization problem, we propose a novel approach that exploits offline learning techniques to approximate scheduling feasibility for sets of observation tasks within an optimization algorithm.

Keywords

Hybrid AI, Neural Networks, Large Neighbourhood Search, Scheduling, Earth Observation Satellite.

1 Introduction

Earth observation satellites (EOSs) are vital for monitoring the planet, supporting diverse applications from climate studies to emergency response. Agile EOSs enhance responsiveness through advanced attitude control, enabling swift instrument reorientation towards targets of interest.

However, planning the activities of agile EOSs presents significant challenges in terms of optimization. This explains why this problem has received attention over the last 25 years [24], leading to the use of various optimization techniques. The latter include, for example, greedy algorithms that iteratively insert observations one by one to build feasible plans, and several types of metaheuristics such as Large Neighborhood Search (LNS), where a current plan is iteratively optimized using destroy and repair operations. Among the state-of-the-art methods, we can cite an Adaptive LNS (ALNS) where six removal operators and three insertion operators are exploited [12]. Improvements of this algorithm using tabu search mechanisms have been proposed [10], as well as an extension that deals with the multi-satellite case by adding decisions related to the assignment of observations to satellites [9]. Another metaheuristic called GRILS for Greedy Randomized Iterated Local Search has also been recently developed and outperforms ALNS in the multi-satellite context [15]. In another direction, dynamic programming techniques have been studied to deal with cases where the profits collected depend on the time at which the observations are performed [16]. All of these approaches include a critical feasibility constraint that assesses the temporal viability of observation schedules based on satellite agility models. For example, operational satellites may use detailed models that take into account the peculiarities of their on-board momentum control gyroscopes. Frequent evaluation of this constraint during the optimization process can significantly slow down solvers. Consequently, many approaches simplify the agility model by making assumptions such as constant maneuver times or constant angular velocity/acceleration to reduce computational complexity. However, these simplifications can lead to less optimal or even impractical schedules. There are also approaches that first search for solution plans using constant maneuver times, and then use a more complex time-dependent agility model [22].

But a key point is that none of the previous contributions exploit the fact that observation scheduling for EOSs is a highly recurrent optimization problem that is solved every day during several years of operation. Therefore, to go beyond these existing approaches, we propose a set of techniques

*Cet article a été accepté à CPAIOR'25.

that allow us to *learn* some aspects of the EOS scheduling problems to be solved regularly. The models learned during an offline phase are then used during an online phase, where EOS scheduling problems have to be solved for precise sets of candidate observations. In the literature, the use of learning methods for satellite scheduling problems has only been tested by a few authors, with attempts to build a decision policy that returns the next observation to insert in the current plan given the features of the candidate requests and the observations already selected [2, 11], and attempts to train a neural network that predicts the probability that an observation can be inserted in the plan of a given satellite [5]. The approach we propose is original compared to existing works because instead of learning an insertion heuristic or an observation insertion success estimator, we try to learn a so-called *scheduling feasibility function*. The latter is a Boolean function that takes as an input a set of POIs I to be observed and returns the value 1 if and only if the execution of all observations in I is considered as feasible from the point of view of the temporal constraints. More specifically, for the offline phase, the idea we promote is to learn a surrogate model that provides a good approximation of the temporal feasibility constraints, which is one of the main bottlenecks of the EOS schedulers. To do this, we first map the temporal feasibility problem to a binary classification problem for a neural networks (NN). We also approximate capacity constraints over the number of POIs that can be observed within orbit sections of different sizes. Since we are trying to infer constraints corresponding to lower bounds on the set of feasible solution plans, obtaining these capacity constraints can be seen as a form of approximate knowledge compilation [21].

In the end, we adopt a hybrid AI and combinatorial optimization approach combining ML-based constraint modeling and classical decision algorithms [13, 1]. Globally, this paper brings several contributions with regard to the existing works:

- (i) We show how a complex routing/scheduling problem (the visit of a set of POIs during some time windows) can be approximated as a simpler binary classification problem, by exploiting the geographical nature of the problem;
- (ii) We propose an NN-based feasibility classifier that has a surprisingly low average error rate on average in terms of scheduling feasibility;
- (iii) We propose an approximation of capacity constraints to help the optimization process, to tackle some challenging POI distributions that the NN struggles with;
- (iv) We develop a hybrid AI approach, HySSEO (*Hybrid Selection and Scheduling for Earth Observation*), where the feasibility constraints learned offline are exploited within an online optimization model that searches for an optimal selection of POIs.

Section 2 provides background on the EOS scheduling problem. Section 3 presents the offline learning methods we propose to approximate the scheduling feasibility constraints.

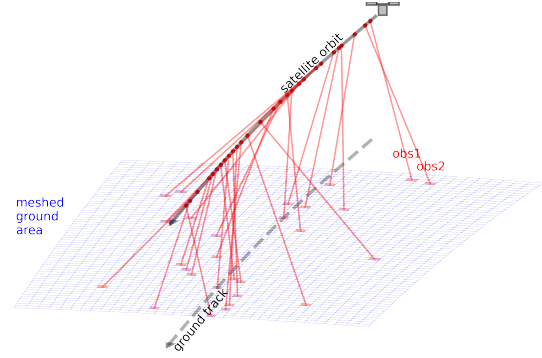


Figure 1: Satellite-centered discretization of the field of view (in blue) and illustration of a set of selected meshes (in red) that can be successively observed by the satellite over its orbit.

Section 4 details the online algorithms used to schedule EOS activities, including the hybrid AI approach proposed. Section 5 provides experimental results concerning the precision of the learned feasibility model and the quality of the schedules obtained. Section 6 concludes and provides perspectives on multi-satellite scenarios.

2 Problem Formalization

We consider a satellite in low Earth orbit that needs to collect acquisitions over points of interest on the ground during one pass over a specific area (during one orbit). Our goal is to select and schedule observations so as to maximize a total reward. The problem is overconstrained in the sense that capturing all candidate acquisitions is usually infeasible.

2.1 Space Discretization and Candidates POIs

For the sake of the learning approach mentioned in the introduction, we define the candidate observations from the point of view of the satellite (satellite-centered representation of the candidate observations), instead of using a fixed meshing of the Earth surface as usually done in the literature. This choice allows us to learn a unique scheduling feasibility model that can be reused for all the satellite revolutions over the Earth, whatever the precise longitude at which the satellite crosses the equatorial plane.

Formally, as illustrated in Figure 1, we build a grid of meshes between two latitudes $LatMin$, $LatMax$. To compute this grid for a North-to-South pass over all potential observation targets, we follow the process described below (the case of a South-to-North pass is symmetric). First, we consider different rolling angles (right-left rotations of the satellite allowing to capture areas on the left and right of its ground track). More precisely, we consider an angular step δ_r , derived from the width of the field of view of the satellite sensor, and a maximum number of steps N_r related to the maximum observation angle usable to capture images. The set of candidate rolling angles is then $R = \{k \cdot \delta_r \mid k \in [-N_r, N_r]\}$. To define different positions of the satellite on its orbit, we consider an initial time t_0 at which the satellite

reaches latitude $LatMax$ and a time-step δ_t derived from the length of the field of view of the sensor and the speed of the satellite on its orbit. We also define the number of time-steps N_t required to reach latitude $LatMin$. The set of candidate positions of the satellite on its orbit is then derived from the set of times $T = \{t_0 + k \cdot \delta_t \mid k \in [0, N_t - 1]\}$ and the space mechanic equations. From the previous sets of candidate rolling angles and candidate satellite positions, we build a grid of meshes by projecting, for each configuration, the field of view of the sensor on the Earth ellipsoid. This grid contains H rows and W columns, where $H = N_t$ and $W = 2 \cdot N_r + 1$.

In the following, we denote by \mathcal{M} the set of meshes obtained from this discretization process, and each potential POI of the problem is mapped to the center of a mesh in \mathcal{M} .

2.2 Optimization Problem

As an input, we consider a set of candidate observations $M \subseteq \mathcal{M}$. This set is strictly included in \mathcal{M} when the end-users require images only on specific points on Earth. For each candidate observation $i \in M$, we have: (i) a position $P_i \in [1..H] \times [1..W]$ in the grid; (ii) a time window $[S_i, E_i]$, during which observation i can be performed, derived from a fixed maximum observation angle and from the ability of the satellite to point towards right-left and/or forward-backward directions; (iii) a reward Rw_i depending on the user requirements and the weather forecast (the higher the cloud coverage prevision, the lower the reward).

Additionally, the satellite must maneuver between successive observations in order to point its sensor towards the right direction at each step. In the following, we denote by tt the transition function such that $tt(i, j, t)$ returns the duration required by a maneuver from a configuration where the satellite is pointed to observation $i \in M$ to a configuration where the satellite is pointed to observation $j \in M$, when the maneuver starts at time t . This transition function is *time-dependent* (it depends on t) mainly due to the motion of the satellite on its orbit along time.

Definition 1 (EOSP). *The Earth Observation Scheduling Problem (or EOSP) consists in finding a sequence of observations $\sigma = [\sigma_1, \dots, \sigma_K]$ such that: (i) each candidate observation in M appears at most once in σ ; (ii) the successive observations can be performed during the allowed time windows; formally, the earliest start time of the first observation is $s_{\sigma_1} = S_{\sigma_1}$, the earliest start time of the k th observation is given by $s_{\sigma_k} = \max(S_{\sigma_k}, s_{\sigma_{k-1}} + tt(\sigma_{k-1}, \sigma_k, s_{\sigma_{k-1}}))$, and condition $s_{\sigma_k} \leq E_{\sigma_k}$ must be satisfied for every observation σ_k involved in σ ; (iii) the total reward collected ($\sum_{i \in \sigma} Rw_i$) is maximized.*

In terms of Operations Research, the problem obtained is a Time-Dependent Orienteering Problem with Time Windows (TD-OPTW) [20]. This problem is known to be NP-hard [8]. Therefore, it is usually addressed using incomplete methods, such as ant colony optimization [23], iterated local search [7], or large neighborhood search (LNS) [20]. The mapping between EOSP and TD-OPTW has been studied in previous works and attacked using large neighborhood

search specifically [12]. Most of these local search or metaheuristic methods start searching from good quality solutions found by a heuristic greedy algorithm. We also use this approach as a baseline in our investigations.

3 Learning the Schedule Feasibility

For an EOS, schedules for successive orbits must be optimized several times a day over several years of operation. To solve this highly recurrent problem, we propose to learn, during an offline phase, a set of constraints called *feasibility constraints* that should be satisfied by any selection of observations.

3.1 Global Approach

To learn feasibility constraints during an offline phase, we combine two distinct learning approaches: (1) based on machine learning methods, we learn a globally feasibility function *feasibilityNN* evaluating whether a set of POIs can be observed during a single satellite pass; (2) based on operations research methods, we learn hard capacity constraints limiting the number of POIs that can be observed within specific areas. More specifically, for the second point, we consider a set of rectangular areas A in the grid of meshes, and for each area $a \in A$ we try to compute the maximum number of POIs that can be observed within a during a single satellite pass, referred to as *Capacity_a*.

During the online phase where a set of candidate observations M is available, the idea is to exploit the previous feasibility constraints to compute in a coarse-grain fashion an optimal selection of observations, before working on detailed scheduling decisions. Equation (1), Equation (2), Equation (3) and Equation (4) give the corresponding coarse-grain optimization model, where Boolean variable x_i takes value 1 if the i th candidate mesh is selected, and value 0 otherwise. In Equation (2), function *feasibilityNN* exploits the global feasibility function expressed as a neural network classifier to determine whether an assignment of the x_i variables is feasible. In Equation (3), the hard capacity constraints enforce upper bounds about the selection. As a result, the hybrid approach proposed combines on the one hand a constraint-based optimization model that must be solved online, several times per day, to compute a plan for the satellite over the next decision horizon, and on the other hand learning techniques through the offline computation of function *feasibilityNN* and bounds *Capacity_a*.

$$\text{maximize } \sum_{i \in M} Rw_i \cdot x_i \quad (1)$$

$$feasibilityNN(x_i, \dots, x_{|M|}) = 1 \quad (2)$$

$$\forall a \in A, \sum_{i \in M \mid P_i \in a} x_i \leq Capacity_a \quad (3)$$

$$\forall i \in M, x_i \in \{0, 1\} \quad (4)$$

One of the main advantages here is that the optimization model proposed is focused on the selection aspect and does not directly use the time-dependent transition function tt that is associated with costly computations. As the calls to the transition function are the main bottleneck of the

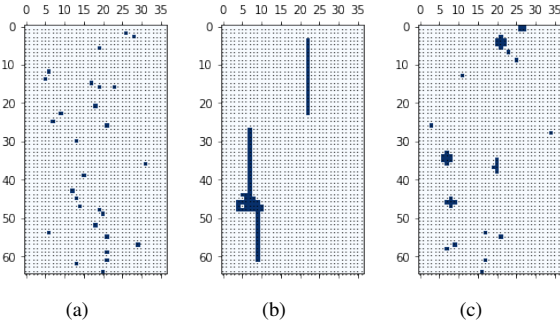


Figure 2: Examples of maximum sets of meshes observable by the satellite during a single pass (mesh indices on the x-axis and y-axis): (a) sparse instance, (b) dense instance, (c) mixed instance

search methods in practice, building a surrogate model of the feasibility function is highly relevant. In the following we successively detail how the global feasibility function *feasibilityNN* and the capacities $Capacity_a$ are learned.

3.2 Learning a global feasibility function

3.2.1 From a Scheduling Feasibility Function to a Classification Problem.

From a machine learning point of view, if we consider the grid of candidate meshes as an image where each pixel is black if the corresponding mesh is activated and white otherwise, our goal is to classify the images depending on whether they correspond to feasible sets of observations. On such an image, the satellite ground track would consist in a straight vertical line crossing the middle of the x axis. We will see later how we consider other local and global features, derived from this initial image.

To illustrate the approach, Figure 2 provides images corresponding to sets of meshes for which it is possible to find feasible observation schedules given a realistic maneuver model. More specifically, Figure 2(a) gives the image associated with the set of meshes activated in Figure 1. In this case, the set of activated meshes is sparse and only a reduced number of meshes are observed due to the time consumed by the satellite maneuvers between the corresponding acquisition tasks. On the other side, Figure 2(b) gives an example of an image where the set of activated meshes is dense. In this case, many contiguous meshes are activated and many small maneuvers are used, which is why the satellite is able to observe many meshes during a single pass. Figure 2(b) also shows standard patterns where the satellite observes strips of contiguous meshes that are parallel to its ground track. Last, Figure 2(c) corresponds to a mixed instance involving both clusters of meshes and individual meshes spread over the observation area.

3.2.2 Input features and NN architecture.

In our study, three kinds of features are considered to classify the images according to the feasibility of the mesh selection:

- *Image*: raw features corresponding to the activation of individual pixels in the mesh selection image;

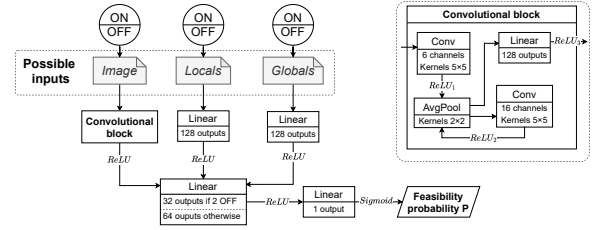


Figure 3: Common architecture of designed neural networks

- *Locals*: features of a set of predefined slices of the grid. Globally, we consider successive slices of height h separated by r rows, h and r being two parameters of the method. These slices are defined as $Slice_{k \cdot r + 1, k \cdot r + h} = [k \cdot r + 1, k \cdot r + h] \times [1..W]$ for different values of k , so as to cover the whole grid. For each slice $Slice_{k \cdot r + 1, k \cdot r + h}$, we compute two features: $nMeshesInSlice_k$ that represents the number of meshes activated in the slice, and $dispersionInSlice_k$ that approximates the total maneuver time required to successively visit the columns where meshes are activated in the slice. On the last point, if we denote by $Columns_k = [c_1, \dots, c_Q]$ the indices of the successive columns where meshes are activated, we have $dispersionInSlice_k = \sum_{i \in [1..Q-1]} TI_ttCols(c_{i+1} - c_i)$ where $TI_ttCols(\delta)$ stands for a Time-Independent approximation on the minimum transition time required by any maneuver traversing δ columns; such a transition time is non-linear in δ , especially for small maneuvers requiring an acceleration phase for the satellite;
- *Globals*: features representing global metrics over the set of selected meshes. Two global features are exploited: $nActive$, the total number of meshes activated in the grid, and $costMST$, the cost of a minimum spanning tree covering all the activated meshes, given that the cost of an edge between two activated meshes i and j is defined from a Time-Independent approximation $TI_tt(i, j)$ of the (time-dependent) transition time between i and j . Note that the cost of a minimum spanning tree is also used as an efficient lower bound in works on traveling salesman problems with time windows [6].

Figure 3 displays the architecture shared by all neural networks designed in this work. The first feature to point out is that as a binary classifier, it returns a single number between 0 and 1 (thanks to the sigmoid activation function) embodying probability; in this case, it is a feasibility probability for a given set of meshes. For the inputs, the three kinds of features listed before can be activated or deactivated, which gives us eight configurations ranging from a case where the neural network is empty to a case where all the features are used.

3.2.3 Feasibility Classifier and Generation of Training Instances.

To train the NN classifier, we first build a set of feasible mesh selections \mathcal{P}_{max} that are maximal in terms of inclusion. This means that for each maximal set $P \in \mathcal{P}_{max}$, each mesh selection $P' \subseteq P$ is feasible and the scheduling algorithm estimates that it is not possible to activate one more mesh in P while keeping the feasibility of the selection. The examples provided in Figures 2(a) à 2(c) are maximal positive instances obtained from a realistic time-dependent maneuver model. On this point, the maximal instances built are actually *approximately maximal* in the sense that the feasibility of a mesh selection is tested based on an approximate greedy algorithm that inserts the observations one by one in the current sequence of visits, each time at a position that is considered as the best one. Other efficient scheduling algorithms could be used [18], but it is worth mentioning that determining whether a single mesh can be added to a mesh selection while preserving the scheduling feasibility is NP-hard, due to the NP-hardness of the Traveling Salesman Problem with Time Windows [19]. This is why we only use approximate scheduling algorithms to estimate whether a set of mesh activations is maximal.

Then, for each maximum instance $P \in \mathcal{P}_{max}$, we generate N positive (resp.) instances by randomly deactivating (resp. activating) meshes. To better approximate the feasibility-infeasibility frontier, the number of positive (resp. negative) instances generated is higher for small numbers of meshes deactivations (resp. activations). Details about the instance generation protocol are given in the experimental section. All the positive and negative instances generated are used to train the neural network, together with all the maximum instances.

3.3 Learning capacity constraints

We now describe how the areas A and the bounds $\{Capacity_a, a \in A\}$ used in Equation (3) are defined. We recall that H and W respectively denote the height and width of the grid of meshes. The areas manipulated correspond to slices of meshes of various heights. More precisely, for each $h \in [1..H]$, we consider a slice $Slice_{1,h} = [1..h] \times [1..W]$ covering h successive rows and the entire grid width. Then, we solve an EOSP as defined in Definition 1 for a set M containing one candidate observation i per mesh in $[1..h] \times [1..W]$, with a unit reward $Rw_i = 1$. The best total reward $TotalRw_h^*$ found for area $a = Slice_{1,h}$ is then used to define capacity $Capacity_a$, that is $Capacity_a = TotalRw_h^*$. Due to the invariance of the problem along the grid rows, reward $TotalRw_h^*$ is exploited to define a maximum capacity not only over slice $Slice_{1,h}$, but also over all slices $Slice_{k+1,k+h} = [k+1..k+h] \times [1..W]$ of height h for $k \in [0..H-h]$. Note that for $h = H$, total reward $TotalRw_H^*$ gives the maximum number of meshes that can be selected over the whole grid.

From a theoretical point of view, the previous process guarantees that any solution of an EOSP problem defined over the grid must satisfy the capacity bounds, otherwise the optimal rewards found above would not be optimal. However,

one difficulty is that solving the previous EOSP problem in NP-hard, because it contains the standard Traveling Salesman Problem with Time Windows as a special case [19]. As a result, finding the optimal total reward $TotalRw_h^*$ is not necessarily easy for large grid slices.

To overcome this difficulty, we exploit two main ideas. First, we use an incomplete LNS solver instead of a complete search engine to optimize the total reward. This LNS solver is run during an arbitrary CPU time set to h minutes. This implies that the capacities computed are not guaranteed to correspond to actual hard constraints, but it is worth mentioning that LNS is a state-of-the-art incomplete method for Time-Dependent Orienteering Problems with Time Windows [20]. Second, when optimizing the total reward for slice $Slice_{1,h+1}$, we reuse the solution found for slice $Slice_{1,h}$. Basically, the best solution found for a slice $Slice_{1,h}$ selects customers within column range $[cmin_h, cmax_h]$. In practice, these customers are always placed around the ground track of the satellite because such customers have time windows that are large and contain the windows of customers placed near the borders of the grid. To simplify the resolution for height $h+1$, we consider only the customers located in area $[1..h+1] \times [cmin_h, cmax_h]$. For high values of h , this heuristic approach strongly simplifies the problem because the number of columns to take into account is quickly reduced to only 3 or 4.

4 Integrating the Learned Model into a Scheduler

This section describes how to exploit such trained models within an EOSP solver. We first recall conventional methods.

4.1 Greedy Scheduler

In order to solve a planning and scheduling problem such as EOSP or TD-OPTW, greedy algorithms are candidates of choice. Basically, greedy algorithms tackle problems by making the choice that seems the best at each decision step. While this usually fails to lead to the globally optimal solutions, it often creates solutions that have a quite good quality [3]. In the Space domain, for operational satellites, greedy algorithms are the baseline when it comes to scheduling tasks, because (i) they are fast (polynomial time in general), (ii) they can be guided by many efficient heuristics exploiting information about the observation tasks and the available time windows.

In the case of EOSP, Algorithm 1 sketches a greedy algorithm, looping over candidate observations. At each step, the idea is to select the best observation that can be added to the current schedule and to insert this observation at its best possible position in the sequence of observations. The process continues until all the observations are scheduled or there is no more place left in the current solution.

Selecting the next observation to schedule and its insertion position is a decision that strongly impacts the quality of the final solution. For this, the algorithm relies on two subfunctions that instantiate specific greedy heurist-

Algorithm 1 : GreedySch**Input :** An EOSP p **Output :** An admissible solution σ $\sigma \leftarrow []$ $continue \leftarrow true$ **while** $|\sigma| \neq |M|$ and $continue$ **do** $i \leftarrow \text{selectObs}(\sigma, M)$ **if** $i \neq \emptyset$ **then** $\sigma \leftarrow \text{scheduleObs}(i, \sigma, M)$ **else** $continue \leftarrow false$ **return** σ **Algorithm 2 : Generic LNS****Input :** A problem p **Input :** An admissible solution s **Output :** An admissible solution s^* $s^* \leftarrow s$ **while** stop criterion is not met **do** $s' \leftarrow \text{repair}(\text{destroy}(s))$ **if** $\text{accept}(s', s)$ **then** $s \leftarrow s'$ **if** $\sum_{i \in s'} R w_i > \sum_{i \in s^*} R w_i$ **then** $s^* \leftarrow s'$ **return** s^*

ics. Function $\text{selectObs}()$ first selects an observation from set M given the current solution σ . Here, we consider that the best observation is the observation $i \in M$ that is not already contained in σ and maximizes the ratio between the reward $R w_i$ and the additional time Δ_i required to maneuver between its predecessors and successors σ_k and σ_{k+1} at the best feasible insertion position in σ . By denoting as s_{σ_k} the earliest start time of observation σ_k in the current schedule and by $s_i = \max(S_i, s_{\sigma_k} + tt(\sigma_k, i, s_{\sigma_k}))$ the earliest time at which observation i can be performed after σ_k , this additional maneuver time is $\Delta_i = tt(\sigma_k, i, s_k) + tt(i, \sigma_{k+1}, s_i) - tt(\sigma_k, \sigma_{k+1}, s_k)$. The observation selection heuristic then chooses an observation in $\arg \max \{ \frac{R w_i}{\Delta_i} \mid i \in M, i \notin \sigma \}$, the idea being to favor observations that have the best "reward over maneuver time" ratio given the current schedule, or in other words the best yield. Function $\text{scheduleObs}()$ positions the observation selected at the right place in the schedule. Note that $\text{selectObs}()$ and $\text{scheduleObs}()$ can be optimized by sharing some computations and data, but we omit these details for sake of conciseness.

4.2 Large Neighborhood Search (LNS)

LNS is a metaheuristic method employed in optimization problems, particularly those involving combinatorial optimization [17]. It is an advanced version of the simpler local search technique. Local search begins with an initial solution to a problem, followed by an exploration of "neighboring" solutions that are generated by making minor adjustments to the current solution. The aim is to discover a solution that optimizes a specific objective function (e.g., minimizing expenses or maximizing productivity). However, local search can become trapped in local optima, where

all surrounding solutions are dominated by the current one. LNS addresses this limitation by examining a much larger set of neighboring solutions. In this extended neighborhood, the solution modifications can be more significant, potentially involving larger alterations of the current solution. This wider search allows LNS to potentially escape from local optima and discover superior solutions overall. An LNS algorithm for the EOSP is sketched in Algorithm 2. It mainly alternates between $\text{destroy}()$ and $\text{repair}()$ operations to find a better solution. The $\text{destroy}()$ operation consists in deactivating some meshes from the current solution, and the $\text{repair}()$ operation consists in refilling the solution plan through mesh activations. To decide whether the next destroy and repair operations have to work on the new solution obtained, we use function $\text{accept}()$ that can be implemented in different ways. The simplest choice is to accept any solution after the repair process. In our investigations we consider two LNS solvers:

LNS for Selecting Observations (LNSSel): in this case, solutions are subsets of M , function $\text{destroy}()$ removes a portion of observations from the current solution with a priority for the deactivation of the observations that have a small ratio between the reward and the distance to the set of activated meshes, and function $\text{repair}()$ adds as many observations as possible in a greedy manner while checking the feasibility with feasibilityNN . Since checking the feasibility using the surrogate is very fast, we can also afford multiple restarts until reaching a given time limit. A restart is performed after a fixed number of LNS iterations, and to perform a restart we generate a new initial solution containing 15 randomly chosen observations, enriched with as many observations as possible, in a greedy manner.

LNS for solving an EOSP (LNSSch): in this case, solutions are sequences of observations from M . Function $\text{destroy}()$ randomly removes observations from the current sequence with a priority to observations close to the ones that just got removed, while function $\text{repair}()$ adds and schedules as many observations as possible in a greedy manner, by calling the transition function. Here, each observation activation involves more computations than in LNSSel.

4.3 Using the Trained Models when Solving EOSP

Figure 4 illustrates our hybrid approach. It is composed of an offline phase and an online phase. During the offline phase, feasibility classifier is trained, and capacity constraints are determined, as explained in Section 3. The online phase first selects a subset of good candidate observations M' using an LNS based on our feasibleNN model and capacity constraints. Then, this subset is used as an input by an LNS-based scheduler, whose initial point is obtained by a greedy algorithm that attempts to schedule as many observations from M' as possible. This algorithm is developed to answer two objectives. (i) The first objective is to take advantage of

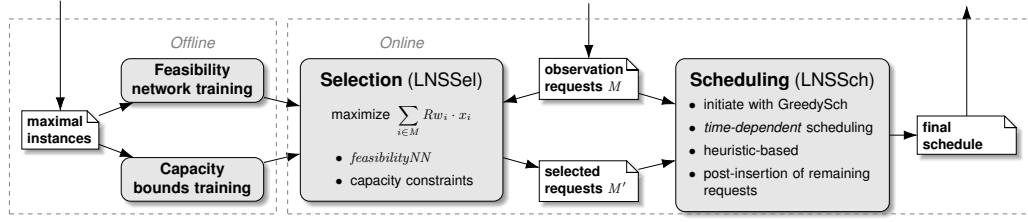


Figure 4: HysSEO components: (i) training on maximal instances is performed offline; (ii) from the initial set of observation requests M , the selection component selects a subset $M' \subseteq M$ by using the trained feasibility model and capacity constraints; (iii) finally M' is used in priority by an LNS-based scheduler, that could add more requests from M if there is room for.

the surrogate feasibility function *feasibilityNN* and capacity constraints to quickly compute good-quality solutions by exploring many observation selection strategies, instead of directly using complex time-dependent scheduling operations. (ii) The second objective is to challenge the robustness of the surrogate model *feasibilityNN*, by first searching for an optimal set of selected meshes $M' \subseteq M$ according to this model and then checking, based on the standard LNS algorithm for EOSP, whether all meshes in M' can actually be simultaneously observed. From a machine learning point of view, this objective is more challenging than just classifying some randomly generated instances.

5 Experimental Evaluation

Now, we provide an experimental analysis on both the precision of the learned classifier and its capacity to select good sets of candidate observations within an EOSP solving process. The EOSP solvers are implemented in Java and executed on 20-core Intel(R) Xeon(R) CPU E5-2660 v3 @ 2.60GHz, 62GB RAM, Ubuntu 18.04.5 LTS, with an OpenJDK 11.0.9 JVM. The machine learning models have been developed and pre-trained using pytorch library in Python 3.11 [14]. These models have been serialized in ONNX and then loaded and called by our solvers using the ONNX Runtime for Java [4].

5.1 Generation of the positive and negative instances

From each maximal set of meshes $P^* \in \mathcal{P}_{max}$, we generate 35 *positive instances* by deactivating m meshes belonging to P^* (random choice of the meshes deactivated). To better approximate the feasibility-infeasibility frontier, we generate more positive instances for small values of m : we generate $K_1 = 6$ random positive instances for $m = 1$, and for $m \in [2, 10]$, we generate $K_m = \text{round}(5/6 \cdot K_{m-1})$ positive instances. Additionally, we generate $K_{>10} = 6$ positive instances where more than 10 meshes are deactivated. Similarly, for each maximal instance $P^* \in \mathcal{P}_{max}$, we generate 35 *negative instances* by randomly activating at least one mesh that does not belong to P^* (random choice of the meshes activated). Again, to better approximate the feasibility-infeasibility frontier, we generate more negative instances for small values of m , using the values of K_m introduced before. For last, each instance belongs to one out of five archetypes: *dense* instances show a huge amount

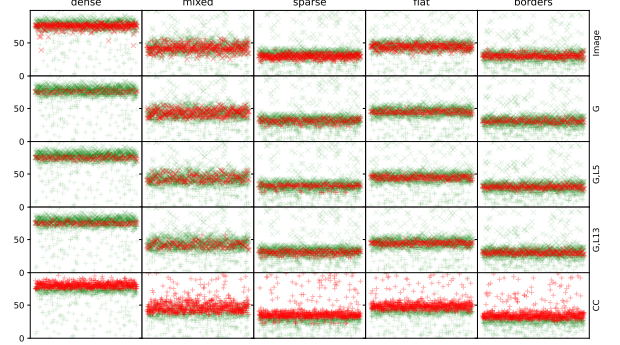


Figure 5: Classification results for five classifiers (Image, G, {G,L5}, {G,L13}, and CC) on five types of instances (*dense*, *mixed*, *sparse*, *flat* and *borders*). On each graph, each dot corresponds to a test instance classified by the model, green if correctly classified and red otherwise. Negative instances are represented by crosses and positive instances by pluses. The x -axis is for the instance number while the y -axis is the amount of activated meshes in the instance.

of activated meshes that are close to each other, implying small maneuvers; *mixed* instances have many isolated activated meshes with a few clusters of close meshes; *sparse* instances only have isolated meshes, and therefore a smaller amount of observable meshes due to higher maneuver durations; *flat* instances show meshes displayed mostly horizontally, which is less easy to capture for the EOS; *borders* instances are characterized by a high amount of meshes near the borders of the meshing, which have smaller visibility time windows. We aim to cover realistic cases and distinguish the ease of observing a whole set of POIs based on the geometry of the instance.

5.2 Precision of the learned classifier

As explained in Section 3.2.2, several NN models have been designed for different input features. Figure 5 shows the results of several such models. Models Image and G take as input only the corresponding features described in Section 3.2.2. {G,L5} and {G,L13} are based on global and local features computed according to the description of Section 3.2.2. {G,L5} considers successive slices of height $h = 5$ separated by $r = 1$ row, while {G,L13} considers

Table 1: Error rates for the classifiers and each instance type: *dense* (D), *mixed* (M), *sparse* (S), *flat* (F), *borders* (B).

NN model	Image					G					{G,L5}					{G,L13}					CC				
Instance type	D	M	S	F	B	D	M	S	F	B	D	M	S	F	B	D	M	S	F	B	D	M	S	F	B
Error rate (%)	25	24	25	21	20	4.6	19	11	11	9.8	6.6	11	12	13	13	6.6	9.1	12	12	13	42	45	46	45	46

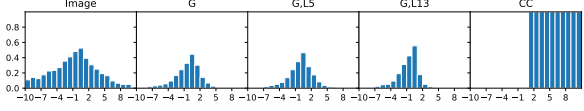


Figure 6: Error rates of designed neural networks for different numbers of meshes added (positive value) or removed (negative values) from maximal instances.

successive slices of height $h = 13$ separated by $r = 6$ rows, implying much less NN inputs. Finally, a classifier based only on the capacity constraints (CC) claims that an instance is feasible if and only if it satisfies these constraints. The error rates of all these models are shown in Table 1, which indicates that models with global features perform best. Models with the image input do not show good performance in the current architecture, which is why our study is focused on the three other NN models. In addition, since capacity constraints are necessary feasibility conditions, they classify instances poorly on their own. On this point, only a minor part of the negative instances in the dataset are well classified by the CC approach, since activating less than 10 additional meshes rarely leads to a violation of the capacity constraints. Figure 6 provides the error rates of the models as a function of the number of meshes activated and deactivated in the positive and negative instances respectively. It shows that the G and {G,L5} models appear to be slightly more robust while {G,L13} gives fewer false positives. Finally, the CC classifier returns no false negative. This makes sense since it classifies the instances based on necessary feasibility conditions, and it returns only false positives. Consequently, only the three most promising models are kept for testing in our HySSEO method.

5.3 Performances of HySSEO

Table 2 shows the results of the runs performed to evaluate the online algorithm. The learned model used in the LNS-Sel part of HySSEO varies between G, {G,L5}, {G,L13}, and all of the feasibility checks used in this part also verify the capacity constraints learned offline. As preliminary results have shown that neural networks alone are not sufficiently constrained on dense instances and may return large sets of observations whose cardinality exceeds the capacity constraints, models G, {G,L5}, and {G,L13} are combined with CC in all of our tests. A last HySSEO algorithm is used to be compared with and does not include any NN: it only checks the capacity constraints. The algorithms are run for 1 and 5 minutes in order to compare their ability to converge faster or produce better results with enough time. For a fixed timespan, in the HySSEO algorithms, 16% of that

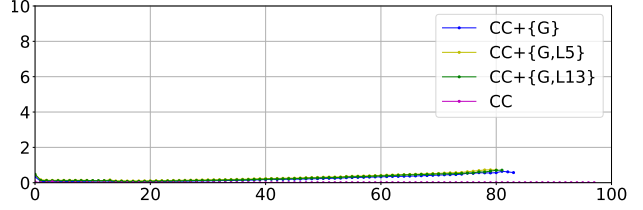


Figure 7: Average inference time (in ms) for each number of activated meshes in the current solution.

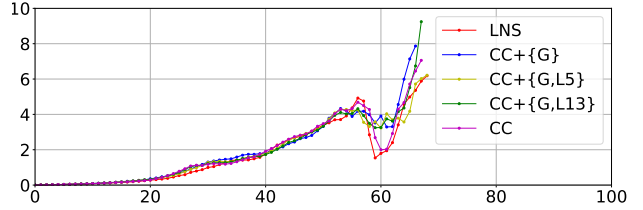


Figure 8: Average insertion time (in ms) of an observation in the current schedule as a function of its current size.

timespan is used by LNSSel, 34% is used by LNSSched on the set of observations returned by LNSSel, and the remaining 50% of the computation time is used to optimize the insertion of other remaining requests. Table 2 shows the performance of the algorithms with $mean \pm std$ on 100 instances equally composed of dense, mixed and sparse instances, with 200 observation requests. The optimality gap is obtained by comparing to an LNSSched algorithm working on the full set of requested observations during a longer computation time of 1 hour. In the table, $nSites$ is the number of observations involved in the final schedule, and LNS_1 (resp. LNS_2) refers to the first (resp. the second) use of LNSSched in HySSEO. The results show that the baseline LNS is beaten on average by all of our HySSEO algorithms, regardless of the allowed computation time. In addition, except for dense instances where $CC + \{ \}$ is highly efficient, using one of our three neural networks to compute feasibility checks improves the solution quality. In the experiments, the capacity constraint checks tend to return much too large request sets on mixed and sparse instances since it is blind with regards to the actual maneuver times required within a slice of meshes. However, we can note that configuration CC returns the best solutions after the first LNSSched, which is quite natural since it is easier to obtain a good solution when requests are less filtered. On the other hand, for the three NN-based HySSEO algorithms, the post-insertion process has a higher impact since the initial LNSSel phase is more constrained in this case. These three NN-based approaches show similar results in terms of solution quality. Figures 7

Table 2: Scheduling performance metrics.

Instance type		1 minute					5 minutes				
		LNS	CC+{G}	CC+{G,L5}	CC+{G,L13}	CC+{ }	LNS	CC+{G}	CC+{G,L5}	CC+{G,L13}	CC+{ }
Dense	n_{Active}	n/a	61.6 \pm 13.6	59.6 \pm 11.3	60.6 \pm 12.5	62.0 \pm 14.2	n/a	61.5 \pm 13.5	60.1 \pm 11.8	60.6 \pm 12.6	62.1 \pm 14.2
	$n_{SitesLNS_1}$	n/a	42.0 \pm 9.0	41.8 \pm 8.7	42.1 \pm 9.0	41.5 \pm 8.8	n/a	42.4 \pm 8.9	42.4 \pm 9.0	42.4 \pm 9.4	41.9 \pm 8.9
	$gapLNS_1(\%)$	n/a	7.6 \pm 3.9	8.3 \pm 3.6	7.7 \pm 4.2	7.2 \pm 4.0	n/a	5.9 \pm 2.7	6.6 \pm 2.7	6.6 \pm 3.5	6.1 \pm 3.1
	$n_{SitesLNS_2}$	47.2 \pm 9.1	45.6 \pm 9.1	45.9 \pm 8.6	45.7 \pm 9.2	45.6 \pm 8.9	47.8 \pm 9.0	46.8 \pm 9.1	46.9 \pm 9.0	47.0 \pm 9.1	46.7 \pm 8.7
	$gapLNS_2(\%)$	5.5 \pm 3.6	3.6 \pm 2.7	3.9 \pm 2.0	3.5 \pm 2.2	3.3 \pm 2.0	2.8 \pm 2.2	2.0 \pm 1.4	1.9 \pm 1.5	2.0 \pm 1.3	1.8 \pm 1.5
Mixed	n_{Active}	n/a	66.5 \pm 4.9	59.3 \pm 3.5	56.8 \pm 4.0	97.5 \pm 0.7	n/a	67.4 \pm 4.0	60.2 \pm 3.4	58.1 \pm 4.2	97.5 \pm 0.7
	$n_{SitesLNS_1}$	n/a	46.4 \pm 5.3	47.5 \pm 3.8	47.4 \pm 5.2	46.9 \pm 2.7	n/a	46.5 \pm 5.5	47.9 \pm 4.0	48.3 \pm 4.5	48.1 \pm 2.9
	$gapLNS_1(\%)$	n/a	18.8 \pm 6.8	17.3 \pm 6.7	17.8 \pm 6.9	10.0 \pm 3.9	n/a	17.6 \pm 6.7	14.8 \pm 5.7	15.3 \pm 5.7	7.0 \pm 3.2
	$n_{SitesLNS_2}$	52.1 \pm 3.3	54.0 \pm 2.9	53.9 \pm 3.0	54.1 \pm 3.0	51.8 \pm 2.8	53.4 \pm 2.7	54.8 \pm 2.6	54.3 \pm 2.9	54.7 \pm 2.7	53.5 \pm 2.4
	$gapLNS_2(\%)$	7.4 \pm 3.9	4.8 \pm 2.5	4.8 \pm 2.4	4.4 \pm 2.3	5.4 \pm 3.2	4.0 \pm 3.4	1.7 \pm 1.5	2.2 \pm 1.6	2.0 \pm 1.6	2.4 \pm 2.2
Sparse	n_{Active}	n/a	49.2 \pm 1.7	48.9 \pm 2.1	46.3 \pm 2.1	97.8 \pm 0.4	n/a	49.1 \pm 1.3	49.5 \pm 2.2	46.7 \pm 2.6	97.7 \pm 0.5
	$n_{SitesLNS_1}$	n/a	39.3 \pm 4.2	43.1 \pm 3.0	42.6 \pm 2.8	43.5 \pm 1.6	n/a	39.6 \pm 3.8	43.9 \pm 3.0	43.6 \pm 2.9	44.7 \pm 1.9
	$gapLNS_1(\%)$	n/a	18.1 \pm 7.6	13.1 \pm 4.6	16.1 \pm 4.6	7.7 \pm 3.0	n/a	16.2 \pm 7.1	11.1 \pm 4.3	13.3 \pm 3.8	5.2 \pm 2.9
	$n_{SitesLNS_2}$	46.8 \pm 1.6	47.2 \pm 1.7	47.5 \pm 1.7	47.9 \pm 1.9	46.5 \pm 1.5	47.5 \pm 1.4	47.7 \pm 1.6	48.0 \pm 1.3	48.2 \pm 1.5	47.4 \pm 1.8
	$gapLNS_2(\%)$	6.6 \pm 2.7	3.1 \pm 2.0	4.3 \pm 2.0	5.0 \pm 2.5	5.0 \pm 2.4	3.7 \pm 2.0	2.1 \pm 1.3	2.0 \pm 1.7	2.5 \pm 1.9	2.6 \pm 2.1

et 8 display the average inference time for feasibility checks in `LNSSel` and the average insertion time of a request into the current schedule in `LNSSched`, respectively. As expected, our learned model allows us to check the feasibility of a set of requests much faster than the feasibility of an actual observation insertion into a schedule. Note that the first kind of checks slightly increases with the number of activated meshes due to the operations required to compute features such as $costMST$, while the second kind of checks is much slower since it requires testing each insertion position in the worst case. In Figure 8, the behavior around 60 requests is due to the different kinds of instances involved in our dataset. More precisely, for many instances, the maximum number of selectable observations is around 60, and in this case testing all the possible insertion positions is long, while for other instances (typically the dense ones), the maximum number of selectable observations is much higher, and in this case finding a feasible insertion position is not that hard.

6 Conclusion

This paper proposes a novel approach to approximate, using a neural network, a feasibility constraint for a set of observation tasks by a satellite. The main idea is to map the feasibility check to a classification task, based on local and global features of the set of POIs. We evaluated the classification performance of several combinations of features, and used them to select good subsets of candidate observation tasks upstream of a conventional scheduling process based on greedy search and large neighborhood search. Combined with this classifier we also propose approximate capacity constraints, to help the NN discriminate unfeasible observation sets. The resulting feasibility classifiers have a small error rate, even on very hard instances, and `HysSEO` is better than baseline methods in terms of solution quality. Computing local features for feasibility does not seem to improve the solution quality, and combining global features and capacity constraints is sufficient to provide results among the best.

One of the next steps is to apply this hybrid AI approach

to plan the observations of several satellites. In this case, the problem to be solved is similar to a team orienteering problem. The latter is much harder to solve since it involves decisions on the allocation of observation tasks to the satellites, instead of just observation selection decisions. On this line, we believe that the techniques proposed and tested in a mono-satellite context can be highly relevant in multi-satellite scenarios, to explore large sets of observation dispatching decisions contrarily to traditional EOS planning methods that are slowed down by the computation of candidate maneuvers for many candidate observation insertion positions across all the satellites of the constellation. In another direction, we are also considering using our feasibility model as a constraint in constraint programming or linear programming frameworks, as in the Empirical Model Learning approach, which consists in encoding neural networks into linear constraints, so as to approximate non-linear or costly constraints [13].

References

- [1] Yoshua Bengio, Andrea Lodi, and Antoine Prouvost. Machine learning for combinatorial optimization: A methodological tour d’horizon. *European Journal of Operational Research*, 290(2):405–421, 2021.
- [2] Eric Bensana, G Verfaillie, C Michelon-Edery, and Nicolas Bataille. Dealing with uncertainty when managing an earth observation satellite. In *Fifth International Symposium on Artificial Intelligence, Robotic and Automation in Space (i-SAIRAS’99)*, pages 120–124, 01 1999.
- [3] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. The MIT Press, 2nd edition, 2001.
- [4] ONNX Runtime developers. Onnx runtime. <https://www.onnxruntime.ai>, 2024. Version 1.17.3.
- [5] Yonghao Du, Tao Wang, Bin Xin, Ling Wang, Yingguo Chen, and Lining Xing. A data-driven parallel

- scheduling approach for multiple agile earth observation satellites. *IEEE Transactions on Evolutionary Computation*, 24(4):679–693, 2020.
- [6] Sylvain Ducomman, Hadrien Cambazard, and Bernard Penz. Alternative filtering for the weighted circuit constraint: Comparing lower bounds for the tsp and solving tsptw. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), Mar. 2016.
 - [7] Ander Garcia, Olatz Arbelaitz, Pieter Vansteenwegen, Wouter Souffriau, and Maria Teresa Linaza. Hybrid approach for the public transportation time dependent orienteering problem with time windows. In Emilio Corchado, Manuel Graña Romy, and Alexandre Manhaes Savio, editors, *Hybrid Artificial Intelligence Systems*, pages 151–158, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
 - [8] Bruce L Golden, Larry Levy, and Rakesh Vohra. The orienteering problem. *Naval Research Logistics (NRL)*, 34(3):307–318, 1987.
 - [9] Lei He, Xiaolu Liu, Gilbert Laporte, Yingwu Chen, and Yingguo Chen. An improved adaptive large neighborhood search algorithm for multiple agile satellites scheduling. *Computers & Operations Research*, 100:12–25, 2018.
 - [10] Lei He, Mathijs Weerd, and Neil Yorke-Smith. Time/sequence-dependent scheduling: the design and evaluation of a general purpose tabu-based adaptive large neighbourhood search algorithm. *Journal of Intelligent Manufacturing*, 31(4):1051–1078, 2020.
 - [11] Jason T. Lam, François Rivest, and Jean Berger. Deep reinforcement learning for multi-satellite collection scheduling. In Carlos Martín-Vide, Geoffrey Pond, and Miguel A. Vega-Rodríguez, editors, *Theory and Practice of Natural Computing*, pages 184–196, Cham, 2019. Springer International Publishing.
 - [12] Xiaolu Liu, Gilbert Laporte, Yingwu Chen, and Renjie He. An adaptive large neighborhood search metaheuristic for agile satellite scheduling with time-dependent transition time. *Computers & Operations Research*, 86:41–53, 2017.
 - [13] Michele Lombardi, Michela Milano, and Andrea Bartolini. Empirical decision model learning. *Artificial Intelligence*, 244, 01 2016.
 - [14] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.
 - [15] Guansheng Peng, Guopeng Song, Yongming He, Jing Yu, Shang Xiang, Lining Xing, and Pieter Vansteenwegen. Solving the agile earth observation satellite scheduling problem with time-dependent transition times. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(3):1614–1625, 2022.
 - [16] Guansheng Peng, Guopeng Song, Lining Xing, Aldy Gunawan, and Pieter Vansteenwegen. An exact algorithm for agile earth observation satellite scheduling with time-dependent profits. *Computers & Operations Research*, 120:104946, 03 2020.
 - [17] David Pisinger and Stefan Ropke. Large neighborhood search. In Michel Gendreau and Jean-Yves Potvin, editors, *Handbook of Metaheuristics*, pages 99–127, Cham, 2019. Springer International Publishing.
 - [18] Cédric Pralet. Iterated maximum large neighborhood search for the traveling salesman problem with time windows and its time-dependent version. *Computers & Operations Research*, 150:106078, 2023.
 - [19] M.W.P. Savelsbergh. Local search in routing problems with time windows. *Annals of Operations Research*, 4:285–305, 1985.
 - [20] Verena Schmid and Jan Fabian Ehmke. An effective large neighborhood search for the team orienteering problem with time windows. In Tolga Bektaş, Stefano Coniglio, Antonio Martinez-Sykora, and Stefan Voß, editors, *Computational Logistics*, pages 3–18, Cham, 2017. Springer International Publishing.
 - [21] Bart Selman and Henry Kautz. Knowledge compilation and theory approximation. *J. ACM*, 43(2):193–224, March 1996.
 - [22] Samuel Squillaci, Cédric Pralet, and Stéphanie Rousel. Comparison of time-dependent and time-independent scheduling approaches for a constellation of earth observing satellites. In *13th International Workshop on Planning and Scheduling for Space (IWSS)*, 2023.
 - [23] Cédric Verbeeck, Pieter Vansteenwegen, and El-Houssaine Aghezzaf. The time-dependent orienteering problem with time windows: a fast ant colony system. *Annals of Operations Research*, 254(1):481–505, July 2017.
 - [24] Xinwei Wang, Guohua Wu, Lining Xing, and Witold Pedrycz. Agile earth observation satellite scheduling over 20 years: Formulations, methods, and future directions. *IEEE Systems Journal*, 15(3):3881–3892, 2021.

Apprentissage de bornes duales valides en programmation par contrainte : décomposition lagrangienne amplifiée avec apprentissage auto-supervisé

Swann Bessa^{1,2}, Darius Dabert^{1,2}, Max Bourgeat¹, Louis-Martin Rousseau¹, Quentin Cappart¹

¹ Polytechnique Montréal, Montréal, Canada

² École Polytechnique, Palaiseau, France

12 mai 2025

Résumé

Ce papier est un résumé de l'article "Learning Valid Dual Bounds in Constraint Programming : Boosted Lagrangian Decomposition with Self-Supervised Learning" publié à AAAI 2025. La décomposition lagrangienne relaxe les problèmes d'optimisation en sous-problèmes plus simples pour améliorer les algorithmes de séparation et d'évaluation en permettant le calcul d'une borne duale. Toutefois, en programmation par contraintes, l'optimisation des multiplicateurs de Lagrange est coûteuse en raison de la complexité de résolution des sous-problèmes. Nous proposons une approche d'apprentissage auto-supervisé utilisant des réseaux de neurones pour générer ces multiplicateurs et obtenir des bornes plus serrées. Cela réduit le nombre d'itérations nécessaires et accélère les solveurs. Notre méthode démontre une bonne généralisation sur des problèmes comme le sac à dos multidimensionnel.

1 Introduction

L'optimisation combinatoire est utilisée dans divers domaines comme l'aérospatial, la planification des transports, l'ordonnancement et l'économie. Un défi majeur est l'explosion combinatoire : le nombre de solutions possibles croît exponentiellement avec la taille du problème, rendant la résolution de problèmes à grande échelle difficile. La programmation par contraintes (CP) est un outil versatile qui permet de traiter des problèmes combinatoires complexes, y compris ceux avec des contraintes non linéaires, grâce à des techniques de recherche et de propagation [7]. Cependant, contrairement à la programmation en nombres entiers, la programmation par contraintes ne dispose pas d'un mécanisme efficace pour obtenir des bornes duales, limitant ainsi sa compétitivité sur de nombreux problèmes [2, 1, 8, 3]. Cet article propose une approche pour combler cette lacune en développant une méthode générale et efficace de bornage dual pour la programmation par contraintes.

2 Décomposition lagrangienne en CP

Un problème d'optimisation sous contraintes (COP) est défini par un ensemble de variables discrètes X , leurs do-

maines de valeurs $D(X)$, un ensemble de contraintes C limitant les affectations possibles, et une fonction objectif $f(X)$ à optimiser. Une solution faisable est une affectation des valeurs de $D(X)$ à X respectant toutes les contraintes. Une solution optimale est une solution faisable qui maximise (ou minimise) la fonction objectif. La formulation mathématique d'un COP inclut généralement m contraintes qui définissent l'espace des solutions admissibles.

$$\max_{X \in D(X)} \left\{ f(X) \mid \bigwedge_{i=1}^m C_i(X) \right\} \quad (1)$$

La notation $C_i(X)$ indique que la contrainte C_i s'applique aux variables X . La décomposition lagrangienne (LD) consiste à relaxer le problème afin d'obtenir une borne supérieure valide et, idéalement, serrée. Pour ce faire, les variables impliquées dans chaque contrainte sont dupliquées, à l'exception de celles de la première contrainte. On obtient alors la borne duale suivante :

$$\mathcal{B}(\mu) = \max \left\{ f(X_1) + \sum_{i=2}^m \mu_i \cdot (X_1 - X_i) \mid \bigwedge_{i=1}^m C_i(X_i) \right\} \quad (2)$$

où les μ_i sont les multiplicateurs de Lagrange. Le problème résultant est une relaxation du problème initial, fournissant une borne duale $\mathcal{B}(\mu)$, qui correspond à une borne supérieure dans un problème de maximisation. Chaque contrainte C_i ayant un ensemble de variables distinct, les sous-problèmes peuvent être résolus indépendamment, simplifiant ainsi la résolution. Ainsi, Une borne duale valide est obtenue en résolvant m sous-problèmes plus simples, mais la qualité de cette borne dépend fortement des multiplicateurs de Lagrange utilisés. Trouver ces multiplicateurs optimaux est le principal défi de la décomposition lagrangienne, car l'équation associée n'est pas différentiable. La méthode la plus courante pour les ajuster est l'optimisation par sous-gradient, qui les met à jour de manière itérative à partir d'une valeur initiale arbitraire [9].

$$\mu_i^{t+1} = \mu_i^t + \alpha(X_1^t - X_i^t) \quad \forall i \in \{2, \dots, m\} \quad (3)$$

Hà et al. (2015) [4] ont proposé d'intégrer cette procédure dans la phase de recherche d'un solveur de programmation par contraintes (CP), en appliquant la décomposition lagrangienne (LD) et l'optimisation par sous-gradient à chaque nœud de l'arbre de recherche. Cependant, cette approche est très coûteuse en temps, car chaque itération du sous-gradient nécessite de résoudre chaque sous-problème, qui peuvent être NP-difficile. Nous proposons d'accélérer considérablement ce processus en utilisant de l'apprentissage automatique.

3 Notre Approche

La borne calculée dans l'Équation (2) possède deux propriétés importantes : (1) elle peut être paramétrée à l'aide des multiplicateurs de Lagrange μ et (2) elle est toujours valide, c'est-à-dire qu'elle ne sous-estimera jamais le profit réel. Ces propriétés ouvrent la voie à une approche basée sur l'apprentissage pour calculer cette borne. Inspirés par Parjadis et al. (2024) [6], nous proposons d'entraîner un modèle $\Omega_\theta : G(V, E) \rightarrow R^{|V|}$ capable de prédire directement tous les multiplicateurs μ pour un problème d'optimisation combinatoire (COP) donné sous forme de graphe G . Ce graphe est constitué d'un ensemble de nœuds V (un par multiplicateur) et d'arêtes E reliant les nœuds partageant la même variable ou la même contrainte. Le modèle Ω_θ est calibré par un ensemble de paramètres différentiables θ . L'objectif est d'éliminer les itérations du sous-gradient en apprenant directement les multiplicateurs qui produisent une borne serrée, ce qui permet d'accélérer l'exécution et d'améliorer le filtrage. La différentiabilité du modèle est essentielle, car la borne est calculée via une optimisation basée sur le gradient. L'objectif est de trouver des paramètres θ minimisant la borne. Grâce à la deuxième propriété, la borne obtenue a la garantie d'être valide, indépendamment de la précision du modèle. Ce travail introduit ainsi une méthode générique d'apprentissage de bornes duales valides pour tout type de COP discret, ce qui est un atout majeur étant donné les difficultés à obtenir des garanties en optimisation combinatoire avec l'apprentissage automatique [5]. La minimisation de la borne en fonction des multiplicateurs μ et son gradient sont ainsi formulés :

$$\min_{\theta} \mathcal{B}(\mu) \mapsto \nabla_{\theta} \mathcal{B}(\mu) \text{ avec } \mu = \Omega_{\theta}(G) \quad (4)$$

En appliquant la règle du chaînage on obtient alors :

$$\nabla_{\theta} \mathcal{B}(\mu) = \left\langle (X_1 - X_i) \cdot \frac{\partial \mu_i}{\partial \theta}, \dots \right\rangle \forall i \in \{2, \dots, m\} \quad (5)$$

Nous proposons d'utiliser une approche d'apprentissage auto-supervisé pour paramétrer le modèle Ω_θ . La procédure d'entraînement est formalisée comme suit. En entrée, elle reçoit un ensemble de données D composé de graphes $G(V, E)$ représentant des instances d'un problème d'optimisation combinatoire (COP). Ces instances peuvent provenir de données historiques ou être générées synthétiquement, et leurs caractéristiques dépendent du problème

traité. À chaque itération, une étape d'optimisation L_D est réalisée, et tous les sous-problèmes sont résolus via une procédure dédiée afin d'obtenir les valeurs optimales de X_i pour les multiplicateurs μ actuels. Ensuite, la borne et son gradient sont calculés, suivis d'une mise à jour par descente de gradient. Cela ajuste les valeurs des multiplicateurs pour l'itération suivante, modifiant ainsi les solutions optimales des variables X_i . Finalement, les paramètres θ du réseau de neurones entraîné Ω_θ sont retournés.

4 Expérimentation

La méthode est testée sur deux problèmes d'optimisation combinatoire difficiles : le problème du sac à dos multidimensionnel et le problème de planification des horaires. Nous testons différentes méthodes :

- **CP** : Une approche pure de programmation par contraintes sans décomposition lagrangienne ni apprentissage.
- **CP+SG** : Le même modèle CP, amélioré avec la décomposition lagrangienne et l'optimisation par sous-gradient, comme proposé par Hà et al. (2015) [4].
- **CP+Learning(all)** : Le modèle CP avec décomposition lagrangienne, utilisant les bornes apprises au lieu du sous-gradient, avec l'apprentissage appliqué à chaque nœud de l'arbre de recherche.
- **CP+Learning(all)+SG** : Similaire à l'approche précédente, mais les bornes apprises sont encore améliorées par l'optimisation sous-gradient. Le modèle entraîné Ω_θ est appelé à chaque nœud de l'arbre de recherche pour obtenir les bornes.
- **CP+Learning(root)+SG** : Le modèle Ω_θ est utilisé uniquement au nœud racine, la borne résultante servant de valeur initiale pour amorcer l'optimisation sous-gradient dans les autres nœuds.

Les résultats obtenus confirment que l'on obtient en moyenne des bornes de meilleures qualités et que notre modèle a une bonne capacité à s'adapter à des instances dont la distribution n'a pas été rencontrée durant la phase d'entraînement.

5 Conclusion

L'article propose une méthode innovante pour améliorer l'efficacité de la décomposition lagrangienne dans la programmation par contraintes. Il utilise un réseau de neurones pour apprendre automatiquement des multiplicateurs lagrangiens, permettant de produire des bornes duales serrées sans nécessiter de bornes étiquetées. Cette approche auto-supervisée permet soit de remplacer entièrement les itérations de sous-gradient, soit de les amorcer afin d'accélérer la convergence. Les expériences montrent que la méthode réduit le temps d'exécution tout en conservant des bornes de qualité. De plus, un ajustement fin permet une meilleure généralisation à des instances hors distribution. Ces résultats ouvrent des perspectives prometteuses pour appliquer cette technique à des problèmes combinatoires où la programmation par contraintes est performante.

Références

- [1] Pascal Benchimol, Willem-Jan van Hoeve, Jean-Charles Régin, Louis-Martin Rousseau, and Michel Rueher. Improved filtering for weighted circuit constraints. *Constraints*, 17 :205–233, 2012.
- [2] Filippo Focacci, Andrea Lodi, and Michela Milano. Cost-based domain filtering. *Principles and Practice of Constraint Programming–CP’99 : 5th International Conference, CP’99, Alexandria, VA, USA, October 11-14, 1999. Proceedings 5*, 1999.
- [3] Monique Guignard and Siwhan Kim. Lagrangean decomposition : A model yielding stronger lagrangean bounds. *Mathematical programming*, 39(2) :215–228, 1987.
- [4] Minh Hoàng Hà, Claude-Guy Quimper, and Louis-Martin Rousseau. General bounding mechanism for constraint programs. In *Principles and Practice of Constraint Programming : 21st International Conference, CP 2015, Cork, Ireland, August 31–September 4, 2015, Proceedings 21*, pages 158–172. Springer, 2015.
- [5] James Kotary, Ferdinando Fioretto, Pascal van Hentenryck, and Bryan Wilder. End-to-end constrained optimization learning : A survey. In *30th International Joint Conference on Artificial Intelligence, IJCAI 2021*, pages 4475–4482. International Joint Conferences on Artificial Intelligence, 2021.
- [6] Augustin Parjadis, Quentin Cappart, Bistra Dilkina, Aaron Ferber, and Louis-Martin Rousseau. Learning Lagrangian Multipliers for the Travelling Salesman Problem. In Paul Shaw, editor, *30th International Conference on Principles and Practice of Constraint Programming (CP 2024)*, volume 307 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 22 :1–22 :18, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [7] Francesca Rossi, Peter Van Beek, and Toby Walsh. *Handbook of constraint programming*. Elsevier, 2006.
- [8] Meinolf Sellmann and Torsten Fahle. CP-based lagrangian relaxation for a multimedia application. In *Third International Workshop on the Integration of AI and OR Techniques (CPAIOR 2001)*, 2001.
- [9] Naum Zuselevich Shor. *Minimization methods for non-differentiable functions*, volume 3. Springer Science & Business Media, 2012.

Génération et validation de données structurées

Victor Charpenay
Mines Saint-Étienne, LIMOS

victor.charpenay@emse.fr

Résumé

Les Transformers permettent de générer rapidement toute forme de données, structurées ou non. Si ces données sont toujours plausibles, elles sont parfois incohérentes. Or, la nature séquentielle du processus de génération amène à la perpétuation ou l'accumulation d'erreurs facilement détectables lorsqu'un schéma de données est disponible. Cet article présente une méthode de génération et de validation conjointe reposant sur des grammaires à base de clauses de Horn (Definite Clause Grammars) issues de la programmation logique.

Mots-clés

Transformer, encodeur-décodeur, langue contrôlée, Prolog.

Abstract

Transformers can quickly generate any form of data, structured or not. These data are always plausible but sometimes inconsistent. The sequential nature of the generation procedure leads to the accumulation of errors that could easily be detected if a data schema is available. This article introduces a joint generation and validation method relying on Definite Clause Grammars, closely related to logic programming.

Keywords

Transformer, Encoder-Decoder, Controlled Language, Prolog.

1 Introduction

Les grands modèles de langage (*Large Language Models*, LLM) permettent un rapprochement sans précédent entre langue naturelle et données structurées, manipulables par une machine. Leur architecture est ajustée pour générer, par exemple, du code Python, des requêtes SQL ou des données au format JSON. Les LLM à la base des chatbots d'OpenAI, Google ou Mistral peuvent aussi transcrire des connaissances en un graphe RDF syntaxiquement correct, au format Turtle. Cependant, il est rare que ces modèles respectent des contraintes usuelles associées au langage, comme le respect d'un vocabulaire pré-existant pour la génération de prédicats.

On cherche par exemple à transformer le paragraphe suivant (tiré de la fiche Wikipédia de la ville de Saint-Étienne) en triplets RDF.

Saint-Étienne, appelée « Sainté » en langage familier, renommée Armeville à la Révolution française, est une commune française située au sud-ouest de Lyon (60 km environ), au sud-est de Clermont-Ferrand (90 km environ) et dans le quart-sud-est de la France, en région Auvergne-Rhône-Alpes. C'est le chef-lieu du département de la Loire.

En réponse au prompt 'Traduis ce paragraphe en triplets RDF', le chatbot de Mistral génère les triplets suivants :

```
@prefix ex: <http://example.org/> .

ex:Saint-Étienne a ex:Commune ;
  ex:nomFamilier "Sainté" ;
  ex:ancienNom "Armeville" ;
  ex:situéeÀ ex:Sud-Ouest_de_Lyon,
            ex:Sud-Est_de_Clermont-Ferrand,
            ex:Quart-Sud-Est_de_la_France ;
  ex:région ex:Auvergne-Rhône-Alpes ;
  ex:chefLieuDe ex:Département_de_la_Loire .

ex:Sud-Ouest_de_Lyon ex:distance "60 km" .
ex:Sud-Est_de_Clermont-Ferrand ex:distance "90 km" .
```

Le résultat paraît satisfaisant mais il ne réutilise aucun vocabulaire connu. Si l'on veut contraindre la génération en indiquant la liste des propriétés de schema.org comme seules propriétés autorisées, on obtient le résultat suivant :

```
<Saint-Étienne> a <Commune> ;
  <alternateName> "Sainté" ;
  <alternateName> "Armeville" ;
  <addressLocality> "Saint-Étienne" ;
  <addressRegion> "Auvergne-Rhône-Alpes" ;
  <addressCountry> "France" ;
  <geo> <Point> .
<Point> <latitude> 45.4397 ;
  <longitude> 4.3871 .
<Saint-Étienne> <distance> <Distance_Lyon> .
<Distance_Lyon> <value> 60 ;
  <unitCode> "km" .
<Saint-Étienne>
  <distance> <Distance_Clermont-Ferrand> .
<Distance_Clermont-Ferrand> <value> 90 ;
  <unitCode> "km" .
<Saint-Étienne> <isPartOf> <Loire> .
<Loire> a <Department> .
<Saint-Étienne> <isPartOf> <France> .
<France> a <Country> .
```

Le prompt utilisé dans ce cas était 'Traduis ce paragraphe en triplets RDF en utilisant uniquement les propriétés suivantes : liste des propriétés de schema.org¹'. Ce graphe

1. <https://schema.org/version/latest#propaz>

obtenu sous contrainte est de nettement moins bonne qualité. Il ajoute des informations qui ne sont pas dans le texte original (coordonnées géographiques) et ne respecte pas la sémantique des propriétés de `schema.org`. La propriété `distance`, par exemple, doit être suivie d'un littéral associant un nombre à une unité de mesure (comme `"60 km"`). Il est relativement facile de vérifier ces contraintes a posteriori, une fois les triplets disponibles, mais étant donné la structure formelle de `schema.org`, la génération par le LLM pourrait aussi être guidée par une validation à la volée des triplets générés, permettant au LLM de générer des données structurées de meilleure qualité. C'est ce que propose cet article.

On s'intéresse en fait à n'importe quel modèle d'apprentissage générant une séquence de symboles (*tokens*) à partir d'un autre ensemble de symboles; des modèles désignés comme *sequence-to-sequence*, principalement basés sur l'architecture Transformer [18]. Les symboles d'entrée peuvent être du texte, comme dans l'exemple ci-dessus mais ils peuvent aussi être la discrétisation d'une série temporelle ou d'un signal audio. Les modèles de transcription (*speech-to-text*) comme Whisper [13] peuvent par exemple être utilisés dans un contexte précis, dans lequel la langue est contrôlée, comme lors de la transcription de commandes vocales. Les résultats présentés en fin d'article montrent que la validation à la volée de la transcription textuelle d'une commande permet de diminuer la taille du modèle génératif, sans en réduire les performances.

Les modèles génératifs produisant des séquences sont itératifs : ils produisent les symboles l'un après l'autre. La méthode proposée ici est simple, elle consiste à valider chaque symbole généré selon une grammaire pré-définie et à appliquer un algorithme de retour sur trace (*backtracking*) lorsqu'un symbole n'est pas accepté par la grammaire. La section suivante (section 2) présente un formalisme de grammaire lui-même construit sur l'idée de retour sur trace : les grammaires à base de clauses de Horn. La section 3 présente ensuite les détails de l'approche de génération et validation conjointe, symbole par symbole. La section 4 donne les résultats d'une évaluation de l'approche dans deux cas de figure : la transcription de commandes simples et la reformulation de descriptions issues de Wikipédia dans une langue contrôlée. Un rapide aperçu de l'état de l'art sur la génération contrainte termine l'article (section 5).

2 Schémas et grammaires

Toute donnée structurée s'exprime dans un langage formalisé par une grammaire. Lorsqu'un schéma est disponible pour cette donnée, on peut y associer une autre grammaire plus contraignante. On s'intéresse donc ici à la spécification et l'utilisation de grammaires pour la génération séquentielle de texte.

Il existe de nombreux formalismes pour spécifier une grammaire. Le plus connu est certainement la forme de Backus-Naur mais le formalisme qui nous intéresse ici est celui qui s'inscrit dans l'histoire de Prolog : les grammaires à base de clauses de Horn (*Definite Clause Grammars*, DCG). Dans

leur ouvrage de référence *The Art of Prolog*, Sterling et Shapiro en donne l'exemple suivant [17, p. 257] :

```
% grammar rules

sentence --> noun_phrase, verb_phrase.

noun_phrase --> determiner, noun_phrase2.
noun_phrase --> noun_phrase2.

noun_phrase2 --> adjective, noun_phrase.
noun_phrase2 --> noun.

verb_phrase --> verb.
verb_phrase --> verb, noun_phrase.

% vocabulary

determiner --> [the]; [a].
adjective --> [decorated].
noun --> [pieplate]; [surprise].
verb --> [contains].
```

Les grammaires DCG ont leur propre syntaxe mais elles peuvent être traduites simplement en Prolog. La première règle de cette grammaire correspond par exemple à la règle Prolog suivante :

```
sentence(S, S0) :-
    noun_phrase(S, S1),
    verb_phrase(S1, S0).
```

où les deux arguments de chaque prédicat sont des listes. La seconde liste est toujours le suffixe de la première, de façon à ce que les deux ensemble représentent une liste par différence, un concept couramment utilisé en programmation logique.

Une fois la grammaire transformée en programme logique, Prolog peut être utilisé pour la validation d'une séquence de symboles. Le fait suivant, qui inclut une phrase complète, sera vrai car la phrase est reconnue par la grammaire :

```
sentence(
    [the, pieplate, contains, a, surprise],
    []).
```

et toute phrase qui n'est pas dans le langage conduirait à un échec de la résolution Prolog. La résolution Prolog permet aussi de décider si une séquence donnée est le début d'une phrase reconnue par la grammaire, comme ci-dessous :

```
sentence(
    [the, pieplate | Tail],
    []).
```

On peut ainsi utiliser Prolog pour générer l'ensemble des phrases du langage bien que dans cet exemple, la résolution ne terminerait pas du fait de la récursivité des règles pour `noun_phrase` et `noun_phrase2`. L'exemple est représentatif de nombreuses grammaires qui reconnaissent un nombre infini de phrases — celles formalisant les langues naturelles, par exemple.

Les modèles de génération *sequence-to-sequence* par Transformer peuvent eux aussi générer des phrases d'un langage-cible, avec une probabilité associée, sans pour autant garantir que toutes les phrases générées appartiendront

bien au langage. En combinant la génération par Transformer avec une validation Prolog, on peut garantir que la génération est contrôlée et espérer que les phrases générées soient parmi les plus probables dans le langage-cible.

3 Détail de l'approche

L'approche présentée dans cet article suppose l'existence d'un modèle *sequence-to-sequence* pré-entraîné et d'une grammaire DCG, que l'on peut respectivement décrire à travers les fonctions `GENERATE` et `VALIDATE`. Dans une configuration où une séquence d'entrée I et une séquence de sortie O sont utilisées par le modèle², la fonction `GENERATE` produit à partir de ces deux séquences un vecteur donnant la probabilité qu'un symbole t soit le suivant dans la séquence O , pour chacun des symboles du langage. La signature de la fonction est donc `GENERATE : $T^l \times T^l \rightarrow \mathbb{R}^{|T|}$` , où T est l'ensemble des symboles définis dans le langage et l est la longueur maximale d'une séquence. On suppose qu'une séquence incomplète est complétée par un symbole spécial (*padding*). La fonction `VALIDATE` retourne simplement vrai ou faux selon si une séquence quelconque est un préfixe valide. Sa signature est `VALIDATE : $T^l \rightarrow \{\text{true}, \text{false}\}$` .

La fonction `GENERATE` est itérative : elle ne génère qu'un seul symbole à la fois. Elle permet une validation à chaque symbole et un retour sur trace si la validation échoue. On peut donc combiner génération et validation avec retour sur trace, dans l'algorithme ci-dessous :

```
function GENERATE_AND_VALIDATE( $I, O$ )
   $P = \text{GENERATE}(I, O)$ 
   $\text{tokens} = \text{FILTER}(O, P)$ 
   $\text{sorted} = \text{SORT}(\text{tokens}, P)$ 
  for all  $t \in \text{sorted}$  do
     $O' = O + t$ 
    if VALIDATE( $O'$ ) then
       $O'' = \text{GENERATE\_AND\_VALIDATE}(I, O')$ 
      if  $O''$  is complete then
        return  $O''$ 
      end if
    end if
  end for
  return  $\emptyset$ 
end function
```

La fonction `GENERATE_AND_VALIDATE` appelle d'abord `GENERATE`, puis trie les symboles du plus probables au moins probables. Dans cet ordre, chacun des symboles t est testé avec `VALIDATE`, qui vérifie que la séquence $O + t$ est reconnue par la grammaire. Si ce n'est pas le cas, on teste le symbole suivant, moins probable. Si, au contraire, la séquence est valide, la procédure continue récursivement, jusqu'à qu'un symbole spécial de fin de génération soit atteint ou que l'ensemble des symboles aient été testés. Dans ce dernier cas, un retour sur trace s'effectue.

Sans condition d'arrêt particulière, cet algorithme génère

l'ensemble des phrases possibles du langage, en nombre potentiellement infini. Les algorithmes classiques de génération ont des conditions d'arrêt supplémentaires, comme une taille maximum pour la séquence de sortie O . Ici, on considère une forme générique de contrôle de la terminaison de l'algorithme, en y ajoutant une fonction `FILTER`. Cette fonction réduit l'ensemble des symboles à explorer sur la base de la séquence disponible O et des probabilités retournées par `GENERATE`. On peut par exemple définir `FILTER(O, P) = \emptyset` lorsque la longueur de O atteint une valeur maximum. On peut aussi écarter les symboles avec une faible probabilité. Une pratique courante lors de la génération de séquences consiste à sélectionner le plus petit ensemble de symboles T_p dont la somme des probabilités dépasse la valeur p [6]. Une autre approche, évaluée dans la section suivante, consiste à écarter les symboles dont la probabilité ne serait pas supérieure à celle d'un tirage aléatoire :

$$\text{FILTER}(O, P) = \{t \mid p_t > \frac{1}{|T|}\} \quad (1)$$

Le principal défaut de l'approche par retour sur trace est qu'il nécessite de générer et valider l'ensemble des séquences possibles avant d'échouer complètement. Lorsque le modèle génératif est sous-optimal et génère des séquences quasi-aléatoires, le temps de calcul peut alors être prohibitif, voire infini si aucun filtrage des symboles n'est effectué et que la grammaire reconnaît une infinité de séquences. Ce semi-déterminisme est une propriété héritée de Prolog et de la méthode de résolution pour le raisonnement. Cependant, lorsque la distribution de probabilité de P est obtenu par application de la fonction `SOFTMAX`, un filtrage basé sur les valeurs de probabilité permet d'élaguer efficacement la majorité des branches à explorer.

L'enjeu du filtrage de symbole est donc plutôt de ne pas filtrer trop de symboles, pour permettre au modèle génératif de générer des séquences peu probables mais correctes du point de vue de la grammaire. L'évaluation de la section suivante montre que le filtrage défini par l'équation 1 est suffisamment permissif.

4 Évaluation

On évalue l'approche dans deux configurations. On évalue d'abord la transcription d'un signal audio en texte avec le modèle Whisper. Cette évaluation permettra de démontrer l'intérêt d'une génération contrôlée lorsque le modèle est imprécis. Whisper, en effet, est un modèle multilingue qui a été entraîné avec plus ou moins de paramètres sur des enregistrements fortement dominés par l'anglais. Si un enregistrement est en français, les variantes de Whisper avec de nombreux paramètres sont relativement précises mais elles sont trop coûteuses pour une transcription en temps réel sur un CPU. À l'inverse, les variantes avec peu de paramètres s'exécutent rapidement mais elles produisent souvent des termes qui ne sont même pas dans le vocabulaire français. Lorsque le langage est contrôlé, une validation symbole par symbole devrait permettre de générer des séquences correctes, même pour les variantes de Whisper avec peu de paramètres.

2. dans cette configuration, les modèles ont deux parties : un encodeur et un décodeur. On trouve aussi des modèles qui n'incluent qu'un décodeur, comme les modèles de la famille des *Generative Pretrained Transformers* (GPT) [14, 1], pour lesquels on définit alors $I = \emptyset$.

La deuxième configuration évaluée dans l'article est celle d'une génération de texte à partir de texte, dans le but de reformuler (et simplifier) le texte d'origine. On suppose ici que le texte-cible est dans une langue contrôlée qui sert d'intermédiaire à une représentation formelle, comme *Attempto Controlled English* (ACE), dont toutes les phrases valides peuvent être réécrites en axiomes OWL ou en règles SWRL [8]. La reformulation est une tâche que peuvent exécuter des modèles comme *Text-to-Text Transfer Transformer* (T5) [15] ou BART [9]. Ces modèles se distinguent des GPT —famille à laquelle appartiennent les modèles Mistral [7]— par le fait que la séquence d'entrée est traitée séparément de la séquence de sortie (dans une architecture de type encodeur-décodeur). De nombreuses expériences autour de ces modèles montrent qu'ils sont capables d'atteindre de bonnes performances avec peu de paramètres pour différentes tâches, dont la reformulation. Comme avec Whisper, on cherche donc à démontrer que la validation symbole par symbole permet une génération rapide sur CPU de données structurées (par le biais d'une langue contrôlée).

D'autres configurations auraient pu être testées, notamment avec un GPT, pour évaluer l'importance de passer par une langue contrôlée dans la génération. Des tests préliminaires suggèrent en effet que T5 et BART ne peuvent pas générer efficacement une structure JSON ou des triplets RDF, la probabilité d'occurrence de symboles comme `""` ou `<` étant trop faible quelque soit la séquence d'entrée. Un GPT, en revanche, est typiquement entraîné avec un corpus de texte plus diversifié et peut donc théoriquement générer directement des données structurées, au prix d'une génération plus lente. Ce type d'évaluation n'est pas considéré ici mais fera l'objet de travaux futurs.

4.1 Transcription

Pour évaluer Whisper, on prend comme cas d'usage l'envoi de commandes vocales en français à un robot mobile autonome. Les commandes envoyées sont par exemple 'à gauche' ou 'va vers la porte et arrête-toi'. On évalue deux variantes de Whisper, *tiny* et *small*, sur un jeu de données composé de 46 commandes vocales de 3s chacune, enregistrées par un micro d'ordinateur. Elles ont été enregistrées par quatre personnes différentes (trois femmes, un homme). Dans cette expérience, la grammaire utilisée est la plus simple possible. Elle ne reconnaît que quatre phrases : 'en avant', 'en arrière', 'à gauche', 'à droite'. Parmi les 46 commandes enregistrées, 16 sont conforme à cette grammaire.

Pour mesurer la qualité des différents modèles considérés, on interprète la tâche comme une tâche de classification, ce qui permet de calculer une précision et un rappel. Lorsque les deux variantes de Whisper génèrent du texte sans validation, le résultat est un vrai positif lorsque la séquence de sortie est exactement celle attendue, à la ponctuation près, et qu'elle est conforme à la grammaire. Lorsque la génération est accompagnée d'une validation, on considère une génération en deux étapes : si la séquence générée est reconnue par la grammaire, elle est prise telle quelle ; si elle n'est

	Params	Précision	Rappel	F1
tiny	39M	0.28	0.29	0.28
small	244M	0.70	0.69	0.69
tiny + cmd	39M	0.48	0.93	0.63

TABLE 1 – Évaluation de variantes de Whisper avec une grammaire simple de commandes vocales (cmd)

pas reconnue par la grammaire, est prise la séquence obtenue par génération gloutonne (sans validation) par le modèle d'origine. Par exemple, lorsque la séquence attendue est 'avance tout droit', le modèle *tiny* avec validation génère 'à d' puis renvoie \emptyset . Par défaut, on prend alors la séquence générée par le modèle *tiny* seul (à savoir 'Avons-tu droit ?', qui est un vrai négatif).

Le tableau de résultats (table 1) montre qu'une validation par Prolog bénéficie nettement au modèle *tiny*. Sans augmenter le nombre de paramètres du modèle et sans réentraînement, sa précision passe de 28% à 48% et son score F1 approche celui du modèle *small*, avec 6 fois moins de paramètres.

4.2 Reformulation

Pour évaluer les capacités de reformulation d'un modèle *sequence-to-sequence*, un autre protocole expérimental est nécessaire. La transcription est une opération fonctionnelle : il n'existe qu'une seule transcription possible pour un enregistrement audio donné, à la ponctuation près. Ce n'est pas le cas lorsqu'on reformule du texte. De nombreuses structures de phrases différentes peuvent avoir la même sémantique et donc être considérées comme des reformulations l'une de l'autre, même dans une langue contrôlée. La procédure d'évaluation la plus évidente consisterait donc à éliminer les variations de formulation en transformant en données structurées la séquence de sortie du modèle et à comparer sémantiquement les structures de données. Si une ontologie existe pour ces données, des équivalences sémantiques peuvent être inférées automatiquement. Le travail d'ingénierie pour aboutir à ce résultat étant conséquent, l'évaluation faite ici a été simplifiée. Elle ne présente que des résultats préliminaires.

Le langage ACE est un bon candidat pour la génération de données structurées. Au-delà du fait qu'il existe une transformation d'ACE vers les langages usuels du Web sémantique, l'outil d'analyse syntaxique développé par ses concepteurs se base en effet sur une grammaire DCG, qui pourrait être utilisée telle quelle pour la validation. Une évaluation complète avec ACE n'a pas encore été faite mais les résultats préliminaires présentés ci-dessous suggèrent que cette langue contrôlée est suffisamment proche de la langue naturelle pour l'envisager comme intermédiaire.

On choisit comme modèle le successeur de T5, FLAN-T5, entraîné sur une plus grande variété de tâches de génération (comme l'explication ou le raisonnement) [3]. Comme séquence d'entrée, cinq descriptions en anglais issues de Wikipédia ont été sélectionnées, parmi celles incluses dans

le jeu de données T-Rex. T-Rex a été construit dans le but d’entraîner des modèles de générations de triplets RDF à partir de texte [4]. Les cinq descriptions ont été choisies au hasard, en vérifiant qu’elle décrivent des entités de type différent : une personne, une organisation, un pays, un nom commun et un produit manufacturé. Ces descriptions ont été reformulées manuellement dans la langue ACE puis validées par son outil d’analyse syntaxique³. Elles sont reportées dans un tableau, table 2. Les descriptions d’origine incluses dans T-Rex, elles, ne respectent pas la syntaxe d’ACE.

À partir d’une séquence d’entrée issue d’une description Wikipédia et d’une séquence de sortie en ACE, on peut estimer la difficulté pour FLAN-T5 à générer la séquence de sortie. Pour ce faire, on collecte le rang de chaque symbole de la séquence selon les probabilités P estimées par le modèle. Il existe différentes variantes de FLAN-T5. On évalue ses deux versions les plus petites, *small* (80M paramètres) et *base* (250M paramètres). Étant donnée une séquence décrivant David Oliver Huffman, par exemple, FLAN-T5 *small* donne la probabilité la plus élevée au symbole ‘David’ pour commencer la génération (rang 0, voir table 2). À l’inverse, au dixième symbole (toujours ‘David’ mais en début de seconde phrase), le modèle donne une probabilité trop faible pour être traitée par l’algorithme selon l’équation 1 (rang -1). La séquence complète ne pourrait donc pas être générée. C’est aussi le cas pour la deuxième séquence, décrivant BMW, mais pas pour les trois suivantes (Kazakhstan, moine et Thalia). La variante FLAN-T5 *base*, elle, peut générer les cinq séquences.

Il apparaît aussi dans ces résultats préliminaires que le modèle génère difficilement certains termes, comme ‘comes’, ‘year’ ou ‘transportation’. Si l’on compare à une génération directe en triplets RDF, ces termes correspondent à la génération d’un prédicat étant donné un sujet ($\langle s, ?, ? \rangle$) et d’un objet étant donné un sujet et un prédicat ($\langle s, p, ? \rangle$). Or, dans un graphe RDF, il existe rarement une solution unique à ces requêtes. Il n’est donc pas étonnant que le rang de ces symboles particulièrement soit élevé. En revanche, il est intéressant de noter que la méthode de filtrage décrite par l’équation 1 permet de conserver ces symboles importants pour la qualité des données générées, malgré leur rang élevé.

5 État de l’art

La génération de texte avec des modèles pré-entraînés peut se faire selon différentes approches. Étant donné que la fonction GENERATE renvoie une distribution de probabilités plutôt qu’un unique symbole, un choix doit être fait à chaque étape pour générer une séquence entière. L’approche la plus évidente est gloutonne : elle consiste à sélectionner le symbole avec la plus haute probabilité. Il a cependant été démontré qu’elle ne permet pas de reproduire fidèlement la langue naturelle, beaucoup plus variée [6]. Pour pallier ce problème, il est possible de générer non pas une mais plusieurs séquences entières en parallèle et d’en choi-

sir une après génération (par exemple, celle avec la plus grande probabilité cumulée). Pour cela, on peut choisir les k symboles les plus probables à chaque étape [5] ou les symboles dont la somme des probabilités dépasse un seuil p [6]. Ces différentes approches peuvent être combinées à un tirage aléatoire de symboles pour plus de diversité dans la génération. À titre d’exemple, la bibliothèque logicielle d’HuggingFace implémente huit stratégies différentes pour la génération de texte⁴. Elles ont toutes en commun de ne pas s’appuyer sur des connaissances ou un schéma connus a priori, contrairement à l’approche présentée dans cet article.

D’autres approches cherchant à contraindre syntaxiquement ou sémantiquement la sortie d’un modèle ont été proposées. Ces approches sont basées sur un ré-entraînement du modèle, soit en fixant le format des séquence d’entrée pendant l’apprentissage [10], soit en modifiant les paramètres d’apprentissage [16]. Il est aussi courant de régulariser la fonction de coût de l’apprentissage du modèle, approche qui a démontré son efficacité dans la génération directe de données structurées [11]. REBEL, un modèle de génération de triplets RDF à partir de texte, est le produit d’un ré-entraînement de BART sur des données issues de T-Rex [2]. Contrairement à ces approches par ré-entraînement, l’approche par génération et validation peut intégrer des modèles pré-entraînés « sur étagère ».

Pour finir, certains travaux dans le domaine de l’intégration neuro-symbolique ont un lien direct avec la question traitée dans cet article. DeepProbLog, notamment, est un outil qui combine modèles d’apprentissage et programmation logique [12]. Le programme Prolog suivant, incluant le prédicat `nn` issu de DeepProbLog, permet théoriquement de générer et valider des séquences symbole par symbole, avec retour de trace :

```
seq2seq(I, O, N) :- sentence(O, []).
seq2seq(I, O, N) :-
    % generation
    nn(I, O, Token, N),
    % validation
    append(O, [Token|Tail], Op),
    sentence(Op, []),
    % recursive call
    Np is N+1, seq2seq(I, Op, Np).
```

Le prédicat `nn` permet de faire appel à un modèle pré-entraîné, dont on suppose ici qu’il énumère les symboles possibles du plus probable au moins probable, étant donné une séquence d’entrée I et une séquence (partielle) de sortie O . Cependant, les détails d’implémentation de DeepProbLog font qu’un tel programme ne pourrait pas être exécuté en pratique, du fait que le moteur d’inférence de DeepProbLog cherchera à énumérer toutes les instanciations possibles de `seq2seq` avant de calculer leur probabilité — instanciations qui peuvent être en nombre infini.

3. <https://github.com/Attempto/APE/>

4. https://huggingface.co/docs/transformers/generation_strategies

Entité	Rang [<i>small</i> , <i>base</i>]
David Oliver Huffman	<code>_David</code> [0,1], <code>_Oliver</code> [0,0], <code>_H</code> [0,0], <code>_uff</code> [0,0], <code>_man</code> [0,0], <code>_is</code> [2,7], <code>_an</code> [3,0], <code>_actor</code> [2,1], <code>_</code> [1,5], <code>_David</code> [-1,19], <code>_Oliver</code> [0,0], <code>_H</code> [0,0], <code>_uff</code> [0,0], <code>_man</code> [0,0], <code>_comes</code> [89,123], <code>_from</code> [0,0], <code>_the</code> [0,1], <code>_USA</code> [31,314], <code>_</code> [0,0]
BMW	<code>_BMW</code> [4,0], <code>_is</code> [1,3], <code>_</code> [0,0], <code>_a</code> [0,0], <code>_company</code> [11,2], <code>_</code> [40,12], <code>_BMW</code> [-1,3], <code>_produces</code> [3,3], <code>_luxury</code> [10,11], <code>_vehicles</code> [0,0], <code>_</code> [0,1], <code>_The</code> [2,2], <code>_year</code> [232,17], <code>_of</code> [1,1], <code>_creation</code> [13,4], <code>_of</code> [2,2], <code>_BMW</code> [6,0], <code>_is</code> [0,0], <code>_1916</code> [0,4], <code>_</code> [0,0]
Kazakhstan	<code>_Kazakhstan</code> [0,0], <code>_is</code> [0,0], <code>_</code> [3,2], <code>_a</code> [0,0], <code>_country</code> [0,0], <code>_</code> [26,17], <code>_It</code> [1,1], <code>_is</code> [2,0], <code>_located</code> [0,1], <code>_in</code> [0,0], <code>_Central</code> [0,0], <code>_Asia</code> [0,0], <code>_and</code> [0,0], <code>_Eastern</code> [0,0], <code>_Europe</code> [0,0], <code>_</code> [0,0]
Monk	<code>_A</code> [12,1], <code>_mon</code> [0,0], <code>_k</code> [0,0], <code>_is</code> [0,0], <code>_</code> [0,0], <code>_a</code> [0,0], <code>_religious</code> [2,2], <code>_person</code> [0,1], <code>_</code> [1,1]
Thalia	<code>_Th</code> [175,518], <code>_alia</code> [0,0], <code>_is</code> [1,16], <code>_an</code> [2,2], <code>_aircraft</code> [0,15], <code>_for</code> [15,24], <code>_transportation</code> [93,8], <code>_</code> [2,2], <code>_It</code> [1,1], <code>_is</code> [1,1], <code>_used</code> [1,6], <code>_in</code> [0,0], <code>_Japan</code> [5,1], <code>_during</code> [2,13], <code>_World</code> [0,0], <code>_War</code> [0,0], <code>_II</code> [0,0], <code>_</code> [0,0]

TABLE 2 – Rang de chaque symbole lors de la génération d’une séquence pré-définie dans la langue contrôlée ACE avec les variantes *small* et *base* du modèle FLAN-T5 (le caractère `_` encode un espace)

6 Conclusion

Les résultats préliminaires présentés dans cet article sont prometteurs. Ils indiquent que la génération de texte avec des modèles *sequence-to-sequence* de petite taille, combinée à une validation symbole par symbole, permettrait d’obtenir des données structurées de qualité, une perspective intéressante pour le domaine de l’ingénierie des connaissances.

Il reste à confirmer ces résultats dans des expériences plus poussées, dans lesquelles seraient produits des triplets RDF selon un vocabulaire connu. Par ailleurs, l’approche devrait être comparée à une génération par GPT qui produirait directement des données structurées.

Références

- [1] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33 : Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.
- [2] Pere-Lluís Huguet Cabot and Roberto Navigli. REBEL : relation extraction by end-to-end language generation. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics : EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 2370–2381. Association for Computational Linguistics, 2021.
- [3] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. *J. Mach. Learn. Res.*, 25 :70 :1–70 :53, 2024.
- [4] Hady ElSahar, Pavlos Vougiouklis, Arslan Remaci, Christophe Gravier, Jonathon S. Hare, Frédérique Laforest, and Elena Simperl. T-rex : A large scale alignment of natural language with knowledge base triples. In Nicoletta Calzolari, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Kôiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings*

- of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018. European Language Resources Association (ELRA), 2018.
- [5] Angela Fan, Mike Lewis, and Yann N. Dauphin. Hierarchical neural story generation. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1 : Long Papers*, pages 889–898. Association for Computational Linguistics, 2018.
 - [6] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.
 - [7] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L  lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. Mistral 7b. *CoRR*, abs/2310.06825, 2023.
 - [8] Tobias Kuhn. The understandability of OWL statements in controlled english. *Semantic Web*, 4(1):101–115, 2013.
 - [9] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART : denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics, 2020.
 - [10] Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. Commongen : A constrained text generation challenge for generative commonsense reasoning. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics : EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 1823–1840. Association for Computational Linguistics, 2020.
 - [11] Tengfei Ma, Jie Chen, and Cao Xiao. Constrained generation of semantically valid graphs via regularizing variational autoencoders. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicol   Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31 : Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montr  al, Canada*, pages 7113–7124, 2018.
 - [12] Robin Manhaeve, Sebastijan Dumancic, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. Neural probabilistic logic programming in deepproblog. *Artif. Intell.*, 298 :103504, 2021.
 - [13] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 28492–28518. PMLR, 2023.
 - [14] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
 - [15] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21 :140 :1–140 :67, 2020.
 - [16] Lei Sha. Gradient-guided unsupervised lexically constrained text generation. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8692–8703. Association for Computational Linguistics, 2020.
 - [17] Leon Sterling and Ehud Y Shapiro. *The art of Prolog : advanced programming techniques*. MIT press, 1994.
 - [18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

40 ans de recherche en logique possibiliste - Une vue d'ensemble

Didier Dubois¹, Henri Prade¹

¹ IRIT, CNRS & Université Toulouse III - Paul Sabatier, 118 route de Narbonne,
31062 Toulouse cedex 9, France

{didier.dubois, henri.prade}@irit.fr

Résumé

Les premiers éléments de logique possibiliste datent de 40 ans. Cette logique manipule des formules logiques classiques associées à des pondérations prenant des valeurs dans un ensemble linéairement ordonné ou plus généralement dans un treillis. Au cours des décennies, la logique possibiliste a connu de nombreux développements tant au niveau théorique qu'au niveau appliqué. L'ambition de cet article est de passer en revue ces développements tout en exposant les idées principales qui les sous-tendent.

Mots-clés

théorie des possibilités, logique, incertitude, préférence.

Abstract

The first elements of possibilistic logic date back 40 years. This logic handles classical logic formulas associated with weights taking values in a linearly ordered set or more generally in a lattice. Over the decades, possibilistic logic has undergone numerous developments at both theoretical and applied levels. The goal of this article is to review all these developments while exposing the main ideas behind them.

Keywords

possibility theory, classical logic, uncertainty, preference.

1 Introduction

La logique possibiliste est issue de la théorie des possibilités. Cette théorie offre un cadre pour la représentation de l'incertitude épistémique due à une information incomplète. Cette théorie a été d'abord proposée par un économiste, G. L. S. Shackle [80], qui a introduit un calcul de degrés de surprise potentielle (qui sont des degrés d'impossibilité); elle a été redécouverte indépendamment par L. A. Zadeh [84] qui s'est concentré sur l'idée de possibilité graduelle en relation avec la modélisation de l'information linguistique, et finalement développée dans [45] en utilisant conjointement la double paire de mesures de possibilité et de nécessité associées à une distribution de possibilité.

La logique possibiliste [41] (dans sa forme de base) manipule des formules logiques classiques associées à des bornes inférieures de mesures de nécessité entendues comme des niveaux de certitude. La règle du modus ponens prend alors, sémantiquement, la forme :

$$N(p) \geq \alpha, N(p \rightarrow q) \geq \beta \Rightarrow N(q) \geq \min(\alpha, \beta),$$

où N est une mesure de nécessité, p et q sont des formules logiques, et $\alpha, \beta \in [0, 1]$. Cela correspond à l'intuition (remontant à Théophraste [77]) selon laquelle la force d'une conclusion reflète la force de la prémisse la plus faible. Cette règle d'inférence pondérée apparaît pour la première fois en 1982.¹ Cependant, ce n'est qu'au milieu des années 1980 que les premiers éléments d'une logique possibiliste à part entière ont commencé à être développés [59, 39].

Les informations incomplètes sont présentes partout et il est important de gérer correctement l'incertitude épistémique. Comme nous le verrons, la logique possibiliste, en stratifiant la connaissance en niveaux de certitude, offre un cadre simple, proche de la logique classique, pour traiter l'incertitude et l'incohérence, mais la logique possibiliste peut aussi prendre d'autres formes, comme les réseaux ou les matrices possibilistes. De plus, la logique possibiliste hérite sa polyvalence de la puissance de représentation de la théorie des possibilités.

Cet article propose une vue d'ensemble - aussi complète que possible en 10 pages - des travaux sur la logique possibiliste depuis 40 ans. Il y a déjà eu plusieurs synthèses sur le sujet depuis [47] qui sont maintenant en partie dépassées. Certaines se concentrent sur les relations avec la logique modale [51], d'autres offrent une perspective plus appliquée [52]. En outre, il existe aussi des introductions plus longues et plus détaillées (mais désormais incomplètes) [41, 48]. Le présent article, avec une structure renouvelée, offre un nouveau regard sur la logique possibiliste.

L'exposé est organisé en deux parties principales. La première partie présente les principaux aspects théoriques et insiste sur les questions de représentation. La seconde partie passe en revue une série de domaines de recherche en IA auxquels la logique possibiliste a été appliquée et peut encore contribuer. Plus précisément, la première partie, après un rappel sur les mesures de possibilité et de nécessité, présente la syntaxe, la sémantique et la théorie de la preuve de la logique possibiliste de base où seules des contraintes de la forme $N(p) \geq \alpha$ sont traitées. Puis, les principales caractéristiques du calcul matriciel possibiliste et des réseaux possibilistes (de type bayésien) sont présentées. Ensuite, divers types d'extensions de la logique possi-

1. [Prade, Thèse d'Etat, 1982]; [Prade, IJCAI'1983, 130-136] (éq. 56).

biliste sont passés en revue : i) pour traiter l'incohérence ; ii) pour prendre en compte des niveaux de certitude symboliques (dont la valeur précise reste inconnue) ; iii) pour introduire de nouveaux types de poids afin de traiter le temps, le multi-sources, les agents, les raisons argumentatives, ou les niveaux de certitude mal connus, grâce à l'utilisation de fonctions de possibilité et de nécessité généralisées prenant leurs valeurs sur un treillis booléen ou sur un treillis distributif pseudo-complémenté plutôt que sur une échelle linéaire ; iv) pour faire face à l'information bipolaire (c'est-à-dire ayant des composantes positives et négatives) grâce à la notion de possibilité garantie, une autre fonction d'ensemble de la théorie des possibilités ; v) pour gérer non seulement les conjonctions, mais aussi les négations et les disjonctions des contraintes présentes en logique possibiliste de base. La première partie se termine par une brève discussion sur le lien avec deux calculs proches : les fonctions de classement de Spohn et la logique de Markov. La deuxième partie passe en revue l'utilisation de la logique possibiliste dans le raisonnement par défaut, la révision des croyances, la fusion d'informations, les logiques de description, la programmation logique, la modélisation des préférences et la décision, l'argumentation et l'apprentissage automatique. Une sous-section est également consacrée à des applications en base de données, en raison de leurs liens étroits avec la représentation des connaissances.

2 Questions théoriques et représentationnelles

Cette première partie traite des bases de la logique possibiliste et des cadres de représentation associés, avant de présenter diverses extensions de la logique possibiliste, et enfin de discuter des relations avec d'autres cadres.

2.1 Théorie des possibilités

En théorie des possibilités, les informations disponibles sont représentées par des distributions de possibilité. Une distribution de possibilité est une application π d'un ensemble U , compris comme un ensemble d'états, de valeurs ou d'alternatives, mutuellement exclusifs (dont l'un est le monde réel, si U est exhaustif), dans une échelle totalement ordonnée \mathcal{S} bornée, dont le plus grand élément est noté 1 et le plus petit 0. Différents types d'échelles peuvent être utilisés depuis une échelle finie $\mathcal{S} = \{1 = \lambda_1 > \dots \lambda_n > \lambda_{n+1} = 0\}$ dans le cas qualitatif, jusqu'à l'intervalle unitaire $\mathcal{S} = [0, 1]$ dans le cas quantitatif, voir [50] pour d'autres options. $\pi(u) = 0$ signifie que u est rejeté comme impossible ; $\pi(u) = 1$ signifie que l'état u est entièrement possible. Plus $\pi(u)$ est grand, plus u est possible. La cohérence de l'état épistémique décrit par π est exprimée par la condition de normalisation $\exists u, \pi(u) = 1$ qui garantit qu'au moins un u est entièrement possible. Si l'information est sans incertitude, mais peut-être imprécise, π est la fonction caractéristique d'un sous-ensemble E de U et $\pi(u) \in \{0, 1\}$. L'information complète correspond aux situations où E est un singleton. L'échelle \mathcal{S} est supposée équipée d'une bijection inversant l'ordre $\lambda \in \mathcal{S} \mapsto 1 - \lambda \in \mathcal{S}$.

Une mesure de possibilité Π et une mesure de nécessité duale N sont associées à une distribution de possibilité π : $\forall A \subseteq U$,

$\Pi(A) = \sup_{u \in A} \pi(u)$; $N(A) = 1 - \Pi(A^c) = \inf_{u \notin A} 1 - \pi(u)$ avec $A^c = U \setminus A$. Lorsque la distribution de possibilité se réduit à un sous-ensemble classique $E \subseteq U$, on a : i) $\Pi(A) = 1$ si $A \cap E \neq \emptyset$, et 0 sinon ; ii) $N(A) = 1$ si $E \subseteq A$, et 0 sinon. $\Pi(A)$ (resp. $N(A)$) évalue à quel point l'événement A est cohérent avec π (resp. est impliqué par π). Par normalisation, $\Pi(U) = N(U) = 1$ et $\Pi(\emptyset) = N(\emptyset) = 0$.

Les mesures de possibilité sont caractérisées par la propriété de « maxitivité » $\Pi(A \cup B) = \max(\Pi(A), \Pi(B))$, et les mesures de nécessité sont « minitives » : $N(A \cap B) = \min(N(A), N(B))$. En raison de la normalisation de π , $\min(N(A), N(A^c)) = 0$ et $\max(\Pi(A), \Pi(A^c)) = 1$, ou de manière équivalente $\Pi(A) = 1$ lorsque $N(A) > 0$, à savoir que quelque chose de quelque peu certain doit être entièrement possible, c'est-à-dire cohérent avec l'information disponible. De plus, on ne peut pas être quelque peu certain à la fois de A et de A^c , sans être incohérent. Nous n'avons que $N(A \cup B) \geq \max(N(A), N(B))$, ce qui va bien avec l'idée qu'on peut être certain de l'événement $A \cup B$, sans être vraiment certain d'événements plus spécifiques comme A ou comme B . La possibilité et la nécessité se différencient d'une probabilité P , qui est auto-duale, et telle que $P(A^c) = 0 \Rightarrow P(A) = 1$, tandis que $N(A^c) = 0 \not\Rightarrow N(A) = 1$ (mais $\Pi(A^c) = 0 \Rightarrow \Pi(A) = 1$).

Les énoncés qualifiés en termes de certitude de la forme « A est certain au degré α » sont représentés par la contrainte $N(A) \geq \alpha$. La plus grande distribution de possibilité π , donc la moins restrictive, qui obéit à cette contrainte est donnée par [45] : $\pi_{(A, \alpha)}(u) = 1$ si $u \in A$, $\pi_{(A, \alpha)}(u) = 1 - \alpha$ sinon. Si $\alpha = 1$ on obtient la fonction caractéristique de A . Si $\alpha = 0$, on obtient l'ignorance totale. C'est un élément clé de la sémantique de la logique possibiliste.

2.2 Logique possibiliste de base

Une formule de logique possibiliste de base (LPB en abrégé) est une paire (p, α) où p est une formule en logique classique et α un niveau de certitude dans $\mathcal{S} \setminus \{0\}$, considéré comme une borne inférieure d'une mesure de nécessité : (p, α) signifie sémantiquement $N(p) \geq \alpha$. En raison de la minitivité des mesures de nécessité, une base en LPB, c'est-à-dire un ensemble de formules de LPB, peut être mise sous une forme clausale équivalente.

Aspects syntaxiques Nous nous intéressons ici au cas où p dans (p, α) est une proposition ; pour la logique possibiliste du premier ordre (de base), voir [41].

Axiomes et règles d'inférence. Les axiomes de LPB [41] sont ceux de la logique propositionnelle, où chaque schéma d'axiomes a une certitude 1. Ses règles d'inférence sont :

- si $\beta \leq \alpha$ alors $(p, \alpha) \vdash (p, \beta)$ (diminution de la certitude)
- $(\neg p \vee q, \alpha), (p, \alpha) \vdash (q, \alpha), \forall \alpha \in (0, 1]$ (modus ponens).

De plus, la règle d'inférence suivante est valide :

- $(\neg p \vee q, \alpha), (p \vee r, \beta) \vdash (q \vee r, \min(\alpha, \beta))$ (résolution)

La règle d'inférence suivante d'affaiblissement de la formule, est aussi valide, en conséquence de la α - β -résolution :

- si $p \vdash q$ alors $(p, \alpha) \vdash (q, \alpha), \forall \alpha \in (0, 1]$.

Inférence et cohérence. Soit $K = \{(p_i, \alpha_i), i = 1, \dots, m\}$ un ensemble de formules en LPB. Prouver $K \vdash (p, \alpha)$ revient alors à prouver $K, (\neg p, 1) \vdash (\perp, \alpha)$ par application répétée de la règle de résolution. De plus, notons que $K \vdash (p, \alpha)$ ssi $K_\alpha \vdash (p, \alpha)$ ssi $(K_\alpha)^* \vdash p$, où $K_\alpha = \{(p_i, \alpha_i) \in K, \alpha_i \geq \alpha\}$ et $K^* = \{p_i \mid (p_i, \alpha_i) \in K\}$. Les niveaux de certitude stratifient la base de connaissances K en coupes de niveaux imbriquées K_α , c'est-à-dire $K_\alpha \subseteq K_\beta$ si $\beta \leq \alpha$. Une conséquence (p, α) de K ne peut être obtenue qu'à partir de formules dans K_α .

Le *niveau d'incohérence* de K est défini par $inc(K) = \max\{\alpha \mid K \vdash (\perp, \alpha)\}$. Les formules (p_i, α_i) dans K telles que $\alpha_i > inc(K)$ sont à l'abri de l'incohérence. En effet, si $\alpha > inc(K)$, $(K_\alpha)^*$ est cohérent, et K^* cohérent $\Leftrightarrow inc(K) = 0$.

La complexité de l'inférence en LPB reste similaire à celle de la logique classique [68].

Aspects sémantiques. La sémantique du LPB [41] s'exprime en termes de distributions de possibilité, et de mesures de nécessité sur l'ensemble Ω des interprétations ω du langage. La base K est sémantiquement associée à la distribution de possibilité, qui est un ensemble *flo* d'interprétations :

$$\pi_K(\omega) = \min_{i=1}^m \max([p_i](\omega), 1 - \alpha_i)$$

où $[p_i]$ est la fonction caractéristique des modèles de p_i , à savoir $[p_i](\omega) = 1$ si $\omega \models p_i$ et $[p_i](\omega) = 0$ sinon. Ceci est en accord avec la qualification en termes de certitude : Intuitivement, cela signifie que toute interprétation qui est un contre-modèle de p_i , est d'autant moins possible que p_i est plus certain ; π_K est obtenu comme la conjonction basée sur min des distributions de possibilité représentant chaque formule. Comme attendu, $N_K(p_i) \geq \alpha_i$ pour $i=1, \dots, m$, où N_K est défini à partir de π_K . L'implication sémantique est définie par $K \models (p, \alpha)$ ssi $\forall \omega, \pi_K(\omega) \leq \pi_{\{(p, \alpha)\}}(\omega)$. LPB est sain et complet [41] par rapport à cette sémantique :

$$K \vdash (p, \alpha) \text{ ssi } K \models (p, \alpha).$$

De plus, nous avons $inc(K) = 1 - \max_{\omega \in \Omega} \pi_K(\omega)$, qui reconnaît le fait que la normalisation de π_K est équivalente à la cohérence classique de K^* .

En probabilités, la seule utilisation de la règle de résolution (localement optimale) $Prob(\neg p \vee q) \geq \alpha, Prob(p \vee r) \geq \beta \vdash Prob(q \vee r) \geq \max(0, \alpha + \beta - 1)$, ne peut pas assurer la complétude d'une contrepartie probabiliste de LPB.

2.3 Forme matricielle

Une règle « si p alors q » est représentée plus naturellement en termes de conditionnement plutôt qu'en utilisant l'implication matérielle de la logique qui permet la contraposition. Le conditionnement en théorie des possibilités obéit à :

$$\Pi(p \wedge q) = \Pi(q \mid p) \star \Pi(p)$$

où \star est min ou le produit, selon que l'on choisit d'être dans un cadre qualitatif ou quantitatif.² Pour $\star = \min$, la solution la plus grande et la moins restrictive de l'équation ci-dessus est $\Pi(q \mid p) = \Pi(p \wedge q)$ si $\Pi(p \wedge q) < \Pi(p)$, $\Pi(q \mid p) = 1$ sinon. Pour $\star = \text{produit}$, le conditionnement (quantitatif) ressemble à un conditionnement probabiliste :

2. Dans ce dernier cas, la possibilité et la nécessité peuvent être interprétées comme une probabilité supérieure et inférieure, voir, e.g., [53].

$\Pi(q \mid p) = \frac{\Pi(p \wedge q)}{\Pi(p)}$ pour $\Pi(p) \neq 0$ et correspond à la règle de conditionnement de Dempster dans la théorie de Shafer [81]. La nécessité conditionnelle est définie par dualité : $N(q \mid p) = 1 - \Pi(\neg q \mid p)$.

En utilisant la possibilité conditionnelle qualitative, un calcul matriciel (voir [44][53] pour des études approfondies) peut être développé en utilisant le produit matriciel max-min \otimes (en notant que $\Pi(q) = \max(\Pi(p \wedge q), \Pi(\neg p \wedge q))$) : $\begin{bmatrix} \Pi(q) \\ \Pi(\neg q) \end{bmatrix} = \begin{bmatrix} \Pi(q \mid p) & \Pi(q \mid \neg p) \\ \Pi(\neg q \mid p) & \Pi(\neg q \mid \neg p) \end{bmatrix} \otimes \begin{bmatrix} \Pi(p) \\ \Pi(\neg p) \end{bmatrix}$. \otimes préserve la normalisation : $\forall r, \max(\Pi(r), \Pi(\neg r)) = 1$.

Un tel produit matriciel peut être appliqué à un ensemble de m règles incertaines en parallèle de la forme « si $a_i^1(x)$ est P_i^1 et \dots et $a_i^k(x)$ est P_i^k alors $b_i(x)$ est Q_i » ($i = 1, \dots, m$) qui relie des variables appartenant aux valeurs d'attributs d'un élément x , et où les P_i^j et Q_i sont des sous-ensembles classiques dans les domaines d'attributs correspondants. Il a été montré que le résultat de leur application conjointe (incluant la fusion des résultats obtenus à partir de chaque règle) peut être mis sous la forme d'un produit matriciel min-max [53]; voir [5] pour le cas général. Le résultat de ce produit min-max est une distribution de possibilité sur une collection d'alternatives mutuellement exclusives (induite par des conclusions pondérées sur les Q_i). De plus, la vision conditionnelle peut être étroitement liée à la LPB, puisque $N(q \mid p) = N(\neg p \vee q)$ si $N(q \mid p) > 0$.

2.4 Réseaux possibilistes

Comme pour les distributions de probabilité conjointes, une distribution de possibilité conjointe associée à des variables ordonnées X_1, \dots, X_n peut être décomposée en termes de distributions de possibilité conditionnelle à l'aide d'une règle de chaînage, en utilisant $\star = \min$, ou $\star = \text{produit}$: $\pi(X_1, \dots, X_n) = \pi(X_n \mid X_1, \dots, X_{n-1}) \star \dots \star \pi(X_2 \mid X_1) \star \pi(X_1)$. De la même manière que pour les réseaux bayésiens, une forme d'indépendance permet de simplifier la décomposition. Cependant, il existe plusieurs définitions de l'indépendance possibiliste conditionnelle entre variables en théorie des possibilités qualitatives, l'une étant symétrique : $\Pi(x, y \mid z) = \min(\Pi(x \mid z), \Pi(y \mid z))$ et une autre, plus forte, étant asymétrique : $\Pi(x \mid z) = \Pi(x \mid z, y)$. Dans le cadre quantitatif, l'indépendance basée sur le produit entre variables ($\forall x, y, z, \Pi(x \mid y, z) = \Pi(x \mid z)$ où $\Pi(y, z) > 0$) est symétrique car elle est équivalente à $\forall x, y, z, \Pi(x, y \mid z) = \Pi(x \mid z) \cdot \Pi(y \mid z)$. Il existe des algorithmes efficaces pour l'inférence dans les réseaux possibilistes. [9], [69].

Les réseaux possibilistes et les bases LPB sont des représentations compactes des distributions de possibilité. Une caractéristique remarquable de ce cadre est que les réseaux possibilistes peuvent être directement traduits en bases LPB et vice-versa, que le conditionnement soit basé sur le minimum ou sur le produit [14].

Des formats de représentation hybrides ont été introduits où des bases en LPB sont associées localement aux nœuds d'une structure graphique plutôt que des tables de possibilités conditionnelles [31].

Ainsi, le cadre de la LPB offre de multiples formats de représentation équivalents : ensemble de formules logiques

priorisées, réseaux possibilistes, mais aussi ensemble de conditionnels de la forme $\Pi(p \wedge q) > \Pi(p \wedge \neg q)$ ($\Leftrightarrow N(q|p) > 0$), tous sémantiquement équivalents à des pré-ordres sur les interprétations (c'est-à-dire à des distributions de possibilité). Il existe des algorithmes permettant de traduire un format dans un autre [14].

Par ailleurs, les réseaux possibilistes ont été étudiés du point de vue du raisonnement causal, en utilisant le concept d'*intervention*, qui revient à imposer les valeurs de certaines variables afin de révéler leur influence sur d'autres [11].

2.5 Gestion des incohérences

Le niveau d'incohérence $inc(K)$ d'une base K en LPB fournit un outil pour gérer les incohérences. Cependant, la LPB souffre d'un « effet de noyade » puisque toutes les formules en dessous de $inc(K)$ sont perdues même si elles ne participent pas à une sous-base incohérente minimale. Il existe différentes manières d'élargir l'ensemble des conséquences qui peuvent être déduites de K [21].

Une façon de le faire tout en préservant un ensemble cohérent de conséquences est la suivante. Étant donnée une base K en LPB, on construit sa complétion paraconsistante K° constituée de formules bi-pondérées : pour chaque formule (p, α) de K , on calcule un triplet (p, β, γ) où β (resp. γ) est le degré le plus élevé avec lequel p (resp. $\neg p$) est soutenu dans K (p est dit *soutenu* dans K au moins au degré β s'il existe une sous-base cohérente de $(K_\beta)^*$ qui prouve p).

Le sous-ensemble des formules de la forme $(p, \beta, 0)$ dans K° ne sont pas paraconsistantes et conduisent à des conclusions sûres. Nous pouvons toujours obtenir un ensemble plus large de conclusions cohérentes à partir de K° comme suit. Nous avons besoin de deux évaluations : i) le niveau d'*infaillibilité* d'un ensemble cohérent S de formules : $UD(S) = \min\{\beta \mid (p, \beta, \gamma) \in K^\circ \text{ et } p \in S\}$; ii) le niveau d'*insécurité* d'un ensemble cohérent S de formules : $US(S) = \max\{\gamma \mid (p, \beta, \gamma) \in K^\circ \text{ et } p \in S\}$. Alors une inférence \vdash_{SS} , nommée relation de conséquence *soutenue de manière sûre*, est définie par $K^\circ \vdash_{SS} q$ si et seulement \exists un sous-ensemble minimal cohérent S qui implique classiquement q tel que $UD(S) > US(S)$. On peut montrer que l'ensemble $\{q \mid K^\circ \vdash_{SS} q\}$ est classiquement cohérent. Voir [49] pour les détails, les discussions et d'autres approches de la gestion de l'incohérence dans le cadre de la LPB, y compris la logique quasi-possibiliste où l'utilisation de la résolution après l'introduction d'une disjonction est interdite (pour éviter le *sequitur ex falso quodlibet*).

2.6 Logique possibiliste symbolique

Il peut y avoir plusieurs raisons pour gérer les niveaux de certitude des formules en LPB de manière *symbolique* : notamment pour garder une trace de l'impact de certains niveaux dans le calcul, ou parce que leur valeur est inconnue. Dans ce dernier cas, les valeurs des niveaux de certitude associés aux formules (toujours supposées appartenir à une échelle totalement ordonnée) sont inconnues, mais l'ordre relatif entre certaines d'entre elles peut être partiellement connu. Dans [29], cela est codé au moyen d'une logique propositionnelle typée, où les formules possibilistes

sont des clauses avec des littéraux spéciaux qui font référence aux niveaux. Les contraintes sur l'ordre de certains des niveaux se traduisent en formules logiques du type correspondant et sont rassemblées dans une base de connaissances auxiliaire distincte. Le processus d'inférence est caractérisé par l'utilisation de « variables d'oubli » pour gérer les niveaux symboliques, et ainsi un processus d'inférence est obtenu au moyen d'une compilation en DNF des deux bases de connaissance [29].

Lorsque l'ordre des poids est complètement connu, ce codage offre un moyen de compiler une base de connaissance possibiliste afin de pouvoir en traiter l'inférence en temps polynomial [30].

Dans une approche [35] qui rejoint la précédente pour traiter les connaissances partielles sur la valeur relative des niveaux de certitude, deux méthodes d'inférence syntaxique sont proposées : l'une calcule le degré de nécessité d'une formule possibiliste en utilisant la notion de sous-base minimale incohérente, tandis que l'autre s'inspire des ATMS, en utilisant les nogoods et les labels.

2.7 Extensions de la logique possibiliste basées sur des treillis

Il existe plusieurs extensions de la logique possibiliste où les poids sont des niveaux de certitude combinés avec des ensembles tels que des périodes de temps [40], des ensembles de sources, ou des groupes d'agents [8, 54] qui conduisent à utiliser des structures de treillis distributif pseudo-complémenté. Lorsque les ensembles sont remplacés par un singleton unique (c'est-à-dire que nous considérons un instant, une source ou un agent), la logique possibiliste de base est retrouvée.

Nous prenons l'exemple de la logique possibiliste multi-agents pour expliquer l'idée. Les formules (propositionnelles) sont désormais associées à un sous-ensemble d'agents : chaque formule (p, A) signifie que *au moins tous* les agents de A croient que p est vrai. Une telle pondération booléenne introduit une différence notable : le supremum de deux sous-ensembles propres peut être l'univers entier (tandis que le supremum de deux niveaux non maximum n'est jamais le niveau maximum dans une échelle totalement ordonnée). C'est pourquoi la règle explicite de renforcement $(p, A), (p, B) \vdash (p, A \cup B)$ est nécessaire, au niveau syntaxique, à côté de règles d'inférence sur l'affaiblissement du sous-ensemble, le modus ponens et la résolution. Les théorèmes de correction et de complétude sont valides par rapport à une sémantique en termes de fonctions de possibilité et de nécessité à valeurs ensemblistes : $\Pi(p) = \bigcup_{w \models p} \pi(w)$ où $\pi(w)$ est le sous-ensemble *maximal* des agents qui trouvent l'interprétation w possible, et $N(p) = [\Pi(\neg p)]^c = \bigcap_{w \models \neg p} [\pi(w)]^c$ (où c désigne la complémentation d'ensemble).

Nous avons maintenant deux types de normalisation conduisant à une vision plus riche de la / (l'in)cohérence : l'une qui signifie que chaque agent trouve au moins un w possible ($\forall a, \exists w, a \in \pi(w)$, c'est-à-dire qu'aucun agent n'est incohérent. Cette condition est plus faible que la condition $\exists w, \pi(w) = All$ (All est l'ensemble de tous

les agents), ce qui signifie qu'il existe une interprétation que tous les agents croient possible, exprimant une condition de cohérence collective. Par exemple, la base $K = \{(p, A), (\neg p, A^c)\}$ viole la dernière condition, mais pas la première.

Cela s'étend au cas général où les propositions sont à la fois associées à un niveau de certitude et à un ensemble d'agents. Les formules sont alors de la forme $(p, \alpha/A)$ où A est un sous-ensemble d'agents et $\alpha \in (0, 1]$, qui se lit « au moins tous les agents de A sont certains de p au moins au niveau α ». La sémantique est alors en termes de fonctions de possibilité et de nécessité floues : Le poids symbolique α/A représente un ensemble flou d'agents a avec des degrés d'appartenance α si l'agent $a \in A$, et 0 sinon.

Une logique de raisonnement sur les raisons [54] gère des paires de la forme (p, x) où p et x sont deux formules logiques propositionnelles exprimées dans deux langages distincts, p est appelé une affirmation et x une raison. La formule (p, x) se lit donc « x est une raison pour p ». (p, x) est plus faible que $(\neg x \vee p, 1)$ (le premier n'entraîne pas le second). La vérité de (p, x) signifie que toutes les situations où x est vrai sont des raisons de croire p . La sémantique de cette logique est isomorphe à celle de la logique multi-agent précédente ; une extension peut prendre en compte la force des raisons [54]. Cette logique des raisons s'apparente à la logique des “supporters” [67], mais est un peu plus simple. La logique possibiliste basée sur les intervalles [26, 28] est une autre extension basée sur un treillis de la logique possibiliste, où les valeurs possibles de niveaux de certitude mal connus sont restreintes par des intervalles.

Mentionnons enfin une manière de conserver une structure ordonnée linéairement tout en enrichissant l'échelle. En LPB, seul le plus petit poids des formules utilisées dans une preuve est conservé ; aucune différence n'est faite par exemple entre une preuve avec une seule prémisse faible et une preuve avec plusieurs prémisses faibles de même force. Cela peut être capturé en utilisant une nouvelle règle de résolution $(\neg p \vee q, \tilde{\alpha}) ; (p \vee r, \tilde{\beta}) \vdash (q \vee r, \tilde{\alpha}\tilde{\beta})$ où $\tilde{\alpha}$ et $\tilde{\beta}$ sont des listes de poids, et $\tilde{\alpha}\tilde{\beta}$ est la liste obtenue par concaténation. Nous pouvons ensuite classer les preuves en fonction de leur force en utilisant un classement lexicographique des listes (une fois qu'elles ont été complétées par des 1 pour les rendre de longueur égale) ; ceci est décrit dans [52].

2.8 Logique possibiliste bipolaire

En théorie des possibilités, il existe deux autres fonctions d'ensemble : i) une mesure de *possibilité garantie* ou de possibilité *forte* (voir, par exemple, [48]) : $\Delta(A) = \inf_{u \in A} \pi(u)$ qui estime dans quelle mesure *tous* les états dans A sont possibles selon les observations. $\Delta(A)$ peut être utilisé comme un degré de support garanti pour A , et son dual $\nabla(A) = 1 - \Delta(A^c) = \sup_{u \notin A} 1 - \pi(u)$ évalue le degré de nécessité potentielle ou *faible* de A , car il vaut 1 dès qu'un état u hors de A est impossible. Les fonctions Δ et ∇ sont *décroissantes* par rapport à l'inclusion (en complet contraste avec Π et N qui sont croissantes). Elles satisfont les propriétés caractéristiques $\Delta(A \cup B) = \min(\Delta(A), \Delta(B))$ et $\nabla(A \cap B) = \max(\nabla(A), \nabla(B))$.

Ainsi la contrainte $\Delta(p) \geq \gamma$, notée syntaxiquement $[p, \gamma]$, exprime que tout modèle de p est au moins possible au degré γ . Ceci peut être représenté par l'ensemble flou $\delta_{[p, \gamma]}(\omega) = 0$ if $\omega \models \neg p$, and $\delta_{[p, \gamma]}(\omega) = \gamma$ if $\omega \models p$. Un ensemble de contraintes $P = \{[q_j, \gamma_j] | j = 1, k\}$ est alors représenté par la distribution de possibilité $\delta_P(\omega) = \max_{j=1, k} \delta_{[q_j, \gamma_j]}(\omega)$ en cumulant les possibilités garanties. Notons que δ_P est obtenu comme la combinaison *disjonctive* basée sur le max de la représentation de chaque formule dans P . Ceci contraste avec π_K (dans la section 2.2) obtenu comme une combinaison *conjonctive* basée sur le min. Ainsi, une distribution de possibilité peut être représentée “par en haut” au moyen de contraintes basées sur la nécessité, et “par en bas” au moyen de contraintes basées sur la possibilité garantie. Au niveau syntaxique, les contraintes basées sur la nécessité sont naturellement associées à une décomposition en CNF pondérée, tandis que les contraintes basées sur Δ conduisent à une décomposition en DNF pondérée. Cette dernière est régie au niveau de l'inférence par le pendant suivant de la règle de résolution

$$[\neg p \wedge q, \gamma], [p \wedge r, \gamma'] \vdash [q \wedge r, \min(\gamma, \gamma')].$$

Une contrainte $[p, \gamma]$ basée sur Δ correspond naturellement à l'expression d'une information positive, c'est-à-dire que les interprétations qui sont des modèles de p sont possibles au moins au degré γ , tandis qu'une contrainte (p, α) basée sur N correspond à une expression négative indiquant que les contre-modèles de p sont quelque peu impossibles (leur possibilité est au plus $1 - \alpha$). Ainsi, plus d'informations positives augmentent δ_P en rendant plus d'interprétations réellement possibles, tandis que plus d'informations négatives diminuent π_K en restreignant davantage les mondes possibles [17].

On peut utiliser soit des formules basées sur Δ , soit des formules basées sur N pour représenter les informations disponibles, selon ce qui semble le plus pratique. Quand il est judicieux de faire la distinction entre les informations positives et les informations négatives (par exemple, des exemples réels de prix et des prix non-interdits par la réglementation), nous devons conserver séparément la base de connaissance K et la base de connaissance P , chacune sémantiquement associée à leurs distributions respectives (censées satisfaire la condition de cohérence $\delta_P \leq \pi_K$) ; voir [38], [18] pour des cadres de gestion de ce dernier cas.

2.9 Logique possibiliste généralisée

En LPB, seules les conjonctions de formules de logique possibiliste sont autorisées. Mais comme (p, α) est sémantiquement interprété comme $N(p) \geq \alpha$, une formule possibiliste peut être manipulée comme une formule propositionnelle qui est vraie (si $N(p) \geq \alpha$) ou fausse (si $N(p) < \alpha$). Les formules possibilistes peuvent alors être combinées avec tous les connecteurs propositionnels, y compris la disjonction et la négation. Il s'agit de la *logique possibiliste généralisée* (LPG) [57, 51]. La LPG est une logique propositionnelle à deux niveaux, dans laquelle les formules propositionnelles sont encapsulées par des opérateurs modaux pondérés interprétés en termes de mesures de nécessité et de possibilité.

La LPG utilise une échelle finie de degrés de certitude $\Lambda_k = \{0, \frac{1}{k}, \frac{2}{k}, \dots, 1\}$ ($k \in \mathbb{N} \setminus \{0\}$); $\Lambda_k^+ = \Lambda_k \setminus \{0\}$. Le langage de la LPG, $\mathcal{L}_{\mathbf{N}}^k$, est construit sur un langage propositionnel \mathcal{L} comme suit : i) Si $p \in \mathcal{L}$, $\alpha \in \Lambda_k^+$, alors $\mathbf{N}_\alpha(p) \in \mathcal{L}_{\mathbf{N}}^k$; ii) si $\varphi \in \mathcal{L}_{\mathbf{N}}^k, \psi \in \mathcal{L}_{\mathbf{N}}^k$, alors $\neg\varphi$ et $\varphi \wedge \psi$ sont aussi dans $\mathcal{L}_{\mathbf{N}}^k$. Ici, $\mathbf{N}_\alpha(p)$ représente (p, α) , la notation soulignant la proximité avec la logique modale. Ainsi, un agent affirmant $\mathbf{N}_\alpha(p)$ a un état épistémique tel que $N(p) \geq \alpha > 0$. Ainsi, $\neg\mathbf{N}_\alpha(p)$ signifie $N(p) < \alpha$, ce qui est équivalent à $N(p) \leq \alpha - \frac{1}{k}$ et donc $\Pi(\neg p) \geq 1 - \alpha + \frac{1}{k}$. En particulier, $\Pi_1(p) \equiv \neg\mathbf{N}_{\frac{1}{k}}(\neg p)$ si $k > 1$. Ainsi, en LPG, on peut faire la distinction entre l'absence de certitude suffisante que p est vrai ($\neg\mathbf{N}_\alpha(p)$) et l'affirmation plus forte que p est quelque peu certainement faux ($\mathbf{N}_\alpha(\neg p)$).

La sémantique de la LPG est comme en LPB définie en termes de distributions de possibilité normalisées sur des interprétations propositionnelles, où les degrés de possibilité sont dans Λ_k . Toute distribution de possibilité à valeurs sur Λ_k telle que $N(p) \geq \alpha$ est un modèle d'une formule $\mathbf{N}_\alpha(p)$ de la LPG. Mais, l'ensemble des distributions de possibilité satisfaisant une formule de LPG n'a pas toujours un plus grand élément, comme c'était le cas en LPB.

La LPG peut être considérée comme un fragment de la logique modale KD, sans modalités imbriquées, mais les modalités y sont graduées. Voir [57] pour son axiomatique, les résultats de correction et complétude, et l'étude de sa complexité. La LPG est un puissant cadre unificateur pour divers formalismes de représentation des connaissances, y compris la logique possibiliste avec des formules partiellement ordonnées, ou une logique d'assertions conditionnelles. Le raisonnement sur l'ignorance explicite, ou certaines tâches de raisonnement sur plusieurs agents, telles que le problème des "enfants couverts de boue" ("muddy children"), peuvent également être traités en LPG [52].

De même, une logique possibiliste multi-agents généralisée qui permet la disjonction et la négation de ses formules de base a été récemment étudiée [54]. Une construction similaire s'applique également à la logique des raisons.

2.10 Relations avec d'autres cadres

Les fonctions de rang ("ranking functions") de Spohn [82] sont similaires aux mesures de possibilité mais elles sont évaluées sur des entiers positifs. Elles utilisent donc des échelles différentes pour évaluer la (im)plausibilité, ce qui rend leurs pouvoirs expressifs quelque peu différents. En effet, il n'y a pas de côté logique pour les fonctions de classement puisqu'il n'y a pas de contrepartie au modus ponens pondéré, et le conditionnement de Spohn, basé sur l'addition, s'inspire des probabilités infinitésimales, tandis que la logique possibiliste n'utilise que des opérations idempotentes telles que max et min [50].

La logique de Markov [78] utilise des formules pondérées pour encoder de manière compacte une distribution de probabilité, mais les pondérations ne sont pas faciles à interpréter. Cependant, on peut toujours construire une base logique possibiliste qui capture exactement un réseau logique de Markov; voir [65], [57].

3 Applications

La logique possibiliste a trouvé des applications dans de nombreux domaines de recherche en IA. En raison de l'espace limité, nous n'avons pu sélectionner qu'un petit échantillon de références pour chaque application.

3.1 De la gestion de l'incertitude au raisonnement par défaut

La LPB a été conçue à l'origine pour propager l'incertitude dans des moteurs d'inférence pour les systèmes experts, en tirant parti du format matriciel [60].

La capacité de LPB à gérer l'incohérence, en utilisant le niveau d'incohérence d'une base de connaissances, est exploitée dans le raisonnement par défaut, une fois les règles par défaut traduites en formules possibilistes. Une règle par défaut "généralement, si p alors q " est représentée par la condition $\Pi(p \wedge q) > \Pi(p \wedge \neg q) \iff N(q|p) > 0$. Ainsi, $N(q|p) > 0$ exprime que dans le contexte où p est vrai, avoir q vrai est strictement plus possible que q faux. Comme pour les probabilités, ce conditionnement n'est pas monotone. On peut avoir que $N(q|p) > 0$, alors que la conclusion opposée $N(\neg q|p \wedge p') > 0$ est vraie dans le contexte plus restreint $p \wedge p'$.

Ensuite, à partir de la plus grande distribution de possibilité sous-jacente à un ensemble cohérent de défauts $\Pi(p_i \wedge q_i) > \Pi(p_i \wedge \neg q_i)$ pour $i = 1, n$, il est possible de stratifier l'ensemble des défauts selon leur spécificité (les défauts les plus spécifiques reçoivent les niveaux les plus élevés), puis de les coder par des formules de logique possibiliste [20] : chaque défaut est transformé en une clause possibiliste $(\neg a_i \vee b_i, N(\neg a_i \vee b_i))$, où N est calculé à partir de la plus grande distribution de possibilité induite par l'ensemble des contraintes modélisant la base de règles par défaut. Ce codage tire parti du fait que lorsque de nouvelles informations sûres sont reçues, le niveau d'incohérence de la base ne peut pas diminuer, et s'il augmente strictement, certaines inférences qui étaient sûres auparavant sont maintenant noyées dans le nouveau niveau d'incohérence de la base et ne sont donc plus possibles, d'où un mécanisme de conséquence non monotone. Il a été prouvé que cette approche est en plein accord avec une approche basée sur les postulats du raisonnement non monotone [19]. Cela est également équivalent à une modélisation probabiliste des conditionnelles en termes d'un type spécial de distributions de probabilités appelées probabilités à grandes marches [22].

3.2 Révision des croyances

Le raisonnement non monotone et la révision des croyances sont étroitement liés, de sorte que la LPB trouve également une application en révision des croyances. En effet, les relations de nécessité comparative (qui peuvent être codées par des mesures de nécessité) ne sont rien d'autre que les relations d'enracinement épistémique [46] qui sous-tendent les processus de révision des croyances bien conduits [61]. Cela permet au cadre de la LPB de fournir des opérateurs de révision syntaxique qui s'appliquent aux bases de connais-

sances possibilistes, y compris dans le cas d'entrées incertaines [24, 76]. En LPB, l'enracinement épistémique est rendu explicite par les niveaux de certitude des formules. En outre, dans un processus de révision, on s'attend à ce que toutes les formules indépendantes de la validité des informations d'entrée restent dans l'état révisé de croyance ; cette idée peut recevoir une signification précise en utilisant une définition de l'indépendance causale possibiliste entre les événements [37].

3.3 Fusion d'informations

La combinaison des distributions de possibilité peut être effectuée de manière équivalente en termes de bases de LPB : La contrepartie syntaxique de la combinaison point par point de deux distributions de possibilité π_1 et π_2 en une distribution $\pi_1 \otimes \pi_2$ par tout opérateur de combinaison monotone \otimes tel que $1 \otimes 1 = 1$, peut être calculée, suivant une idée proposée pour la première fois dans [33]. A savoir, si la base LPB K_1 est associée à π_1 et la base K_2 à π_2 , une base LPB $K_{1 \otimes 2}$ sémantiquement équivalente à $\pi_1 \otimes \pi_2$ est donnée par : $\{(p_i, 1 - (1 - \alpha_i) \otimes 1) \text{ s.t. } (p_i, \alpha_i) \in K_1\} \cup \{(q_j, 1 - 1 \otimes (1 - \beta_j)) \text{ s.t. } (q_j, \beta_j) \in K_2\} \cup \{(p_i \vee q_j, 1 - (1 - \alpha_i) \otimes (1 - \beta_j)) \text{ s.t. } (p_i, \alpha_i) \in K_1, (q_j, \beta_j) \in K_2\}$. Pour $\otimes = \min$, on obtient $K_{1 \oplus 2} = K_1 \cup K_2$ avec $\pi_{K_1 \cup K_2} = \min(\pi_1, \pi_2)$ comme attendu (combinaison conjonctive). Pour $\otimes = \max$ (combinaison disjonctive), on a $\Gamma_{1 \oplus 2} = \{(p_i \vee q_j, \min(\alpha_i, \beta_j)) \text{ s.t. } (p_i, \alpha_i) \in K_1, \text{ et } (q_j, \beta_j) \in K_2\}$. Avec des opérateurs \oplus non idempotents, certains effets de renforcement peuvent être obtenus. Voir, par exemple [63], pour une étude sur les opérateurs de fusion en LPB. En outre, cette approche peut également être appliquée au codage syntaxique de la fusion de bases logiques classiques basées sur la distance de Hamming (où les distances sont calculées entre chaque interprétation et les différentes bases en logique classique, donnant ainsi naissance à des équivalents de distributions de possibilité) [15]. Dans [27] une représentation basée sur Δ est utilisée ; voir [32] pour un exemple illustratif.

3.4 Logique de description

La gestion possibiliste de l'incertitude en logique de description a été proposée pour la première fois dans [75]. Elle présente des avantages informatiques, en particulier dans le cas de la famille *possibilistic DL-Lite* où l'extension de la puissance expressive de DL-Lite se fait sans coûts de calcul supplémentaires [12] ; il est alors facile d'utiliser l'opération min pour la fusion de bases DL-Lite possibilistes.

Une méthode polynomiale pour calculer une réparation possibiliste unique pour une ABox pondérée partiellement pré-ordonnée qui peut être incohérente par rapport à la TBox a été proposée dans [7].

3.5 Programmation logique

Différentes propositions ont été faites pour une gestion possibiliste de l'incertitude en programmation logique et en programmation par ensembles-réponses [2, 71, 72, 62, 6].

En outre, une application remarquable de la LPG est sa capacité à coder des programmes par ensembles-

réponses (ASP), en utilisant une échelle à 3 valeurs $\Lambda_2 = \{0, 1/2, 1\}$. Ce qui permet de faire la distinction entre les propositions dont on est totalement certain et les propositions qu'on considère seulement comme plausibles, et d'expliciter en LPG la sémantique épistémique de règles avec négation par échec. Par exemple, la règle ASP $r \leftarrow p \wedge \text{not } q$ est codée par $N_1(p) \wedge \Pi_1(\neg q) \rightarrow N_1(r)$ en LPG. Voir [57].

3.6 Bases de données

Le calcul de *provenance*, basé sur deux opérations formant un demi-anneau, combine et propage des annotations associées à des données. Ce calcul, lorsqu'il est basé sur les opérations max et min, correspond exactement à l'évaluation des requêtes lorsque les données sont étiquetées avec des niveaux de certitude, comme en LPB [55].

La LPB présente un intérêt pour la conception de bases de données où la présence de certains tuples dans la base de données peut n'être possible que dans une certaine mesure, et où les dépendances fonctionnelles sont incertaines [70]. Le même genre d'idée a été appliqué au nettoyage possibiliste de données [64].

3.7 Préférences et décision qualitative

Une formule de LPB (p, α) peut représenter un objectif p avec un niveau de priorité α . Des préférences telles que "Je préfère p à q et q à r " (où p, q, r peuvent ne pas être mutuellement exclusifs) peuvent être représentées par la base possibiliste $P = \{(p \vee q \vee r, 1), (p \vee q, 1 - \gamma), (p, 1 - \beta)\}$ avec $\gamma < \beta < 1$, comme un ensemble d'objectifs plus ou moins impératifs. D'autres formats tels que les conditionnels, les réseaux possibilistes, la représentation basée sur Δ sont également intéressants pour représenter les préférences [23]. De plus, l'expression des préférences peut être bipolaire : énoncé de situations qui sont plus ou moins fortement rejetées, et de situations qui sont garanties satisfaisantes à un certain degré [16]. Mentionnons l'équivalence représentationnelle [13] entre la logique de choix qualitatif QCL [34] et la logique des possibilités garanties. De telles préférences sont également intéressantes pour l'expression de requêtes flexibles à une base de données [58].

Les réseaux possibilistes (basés sur le produit) ont été utilisés pour représenter des préférences conditionnelles possibilistes avec des pondérations symboliques [10]. Les interprétations (correspondant aux différentes alternatives) sont alors comparées en termes de vecteurs symboliques exprimant la satisfaction ou la violation des formules associées aux différentes préférences, en utilisant des relations d'ordre appropriées. La représentation obtenue est compatible avec celle des CP-nets, dont elle fournit une bonne approximation [83].

La théorie des possibilités fournit un cadre pour la décision qualitative sous incertitude où des critères de décision pessimistes et optimistes ont été axiomatisés [56]. La contrepartie de ces critères, lorsque la connaissance et les préférences sont sous la forme de *deux bases en LPB distinctes*, est donnée par les définitions [42] :

- l'utilité pessimiste $u_*(d)$ de la décision d est le maximum $\alpha \in \mathcal{S}$ s.t. $K_\alpha \wedge d \vdash_{PL} P_{\nu(\alpha)}$,

- l'utilité optimiste $u^*(d)$ de d est le maximum $\nu(\alpha) \in \mathcal{S}$ s.t. $K_\alpha \wedge d \wedge P_\alpha \not\equiv \perp$,
où \mathcal{S} est une échelle totalement ordonnée finie bornée, ν une bijection inversant l'ordre de cette échelle; K_α est un ensemble de formules logiques classiques rassemblant les connaissances certaines à un niveau au moins α , et où P_β est un ensemble de formules logiques classiques constitué d'un ensemble d'objectifs dont le niveau de priorité est *strictement* supérieur à β . Une décision pessimiste optimale assure la satisfaction de tous les objectifs de $P_{\nu(\alpha)}$ avec une priorité aussi basse que possible, en utilisant seulement une partie K_α de la connaissance qui a une grande certitude. Une décision optimiste optimale maximise la cohérence de tous les objectifs plus ou moins importants avec tous les éléments de connaissance plus ou moins certains.

3.8 Argumentation

Un langage de programmation logique possibiliste et réfutable qui combine des caractéristiques de la théorie de l'argumentation et de la programmation logique, intégrant également le traitement possibiliste de l'incertitude, a été proposé dans [1].

La logique possibiliste peut être utilisée pour représenter les états mentaux des agents (croyances éventuellement imprégnées d'incertitude et objectifs prioritaires), pour réviser les bases de croyances et pour décrire la procédure de décision pour sélectionner une nouvelle offre dans une négociation basée sur l'argumentation [3].

La logique des raisons [54] qui traite les formules (p, x) exprimant que " x est une raison pour p ", où la négation peut être appliquée à p , x et (p, x) , offre un cadre riche pour le raisonnement argumentatif.

3.9 Apprentissage automatique

L'étude de l'apprentissage des théories logiques possibilistes [73], a montré que de nombreux résultats d'apprentissage en temps polynomial pour la logique classique peuvent être transférés à l'extension possibiliste respective. La logique possibiliste bipolaire offre un cadre gradué pour étendre le cadre d'apprentissage de l'espace des versions [74].

La LPB peut également être appliquée à la programmation logique inductive (ILP). En effet, avoir un ensemble stratifié de règles logiques du premier ordre comme hypothèses en ILP est intéressant pour l'apprentissage à la fois de règles couvrant les cas normaux et de règles plus spécifiques pour les cas exceptionnels [79], [66].

Une cascade de produits min-max de matrices représentant des règles possibilistes de type si-alors présente une ressemblance structurelle avec un réseau neuronal min-max. Une telle cascade peut être montrée comme étant équivalente à un réseau neuronal min-max, chaque produit matriciel correspondant à une couche et la fonction d'activation utilisée étant l'identité; voir [5] pour plus de détails. [4] offre une approche neuro-symbolique possibiliste très complète.

3.10 Autres applications

D'autres applications peuvent être trouvées, comme la modélisation des désirs à l'aide de fonctions Δ [43], ou l'ex-

pression des buts des agents en logique possibiliste dans des jeux booléens lorsque les agents peuvent avoir une connaissance incomplète des préférences des autres [36].

Une autre application est le codage des politiques d'accès de contrôle [25]. Une description formelle des politiques de sécurité est nécessaire pour vérifier si les propriétés de sécurité sont satisfaites ou non. Les règles de contrôle d'accès, garantissant les propriétés de confidentialité et d'intégrité, sont codées en termes de bases de connaissances stratifiées. La stratification reflète la hiérarchie entre les rôles et est utile pour gérer les conflits.

4 Conclusion

Cet article a passé en revue un grand nombre de travaux sur le développement de la logique possibiliste et ses applications. La logique possibiliste est bien adaptée à la représentation d'informations incomplètes et de croyances acceptées plus ou moins ancrées. Elle reste proche de la logique classique et offre un cadre riche, simple et polyvalent pour la représentation de l'information et le raisonnement qualitatif sur l'incertitude. La théorie des possibilités, comme celle des probabilités, mérite qu'on s'y intéresse !

Références

- [1] T. Alsinet, C. I. Chesñevar, L. Godo, and G. R. Simari. A logic programming framework for possibilistic argumentation : Formalization and logical properties. *Fuzzy Sets Syst.*, 159 :1208–1228, 2008.
- [2] T. Alsinet, L. Godo, and S. Sandri. Two formalisms of extended possibilistic logic programming with context-dependent fuzzy unification : A comparative description. *Elec. Notes in Theo. Comput. Sci.*, 66 (5), 2002.
- [3] L. Amgoud and H. Prade. Reaching agreement through argumentation : A possibilistic approach. In *KR*, pages 175–182, 2004.
- [4] I. Baaj and P. Marquis. II-NeSy: A possibilistic neuro-symbolic approach. *cs.AI arXiv 2504.07055*, 2025.
- [5] I. Baaj, J. P. Poli, W. Ouerdane, and N. Maudet. Min-max inference for possibilistic rule-based system. In *FUZZ-IEEE*, pages 1–6. IEEE, 2021.
- [6] K. Bauters, S. Schockaert, M. De Cock, and D. Vermeir. Characterizing and extending answer set semantics using possibility theory. *Theory Pract. Log. Program.*, 15(1) :79–116, 2015.
- [7] S. Belabbès and S. Benferhat. Computing a possibility theory repair for partially preordered inconsistent ontologies. *IEEE Trans. Fuz. Syst.*, 30:3237–3246, 2022.
- [8] A. Belhadi, D. Dubois, F. Khellaf-Haned, and H. Prade. Reasoning with multiple-agent possibilistic logic. In *SUM*, LNCS 9858, 67–80. Springer, 2016.
- [9] N. Ben Amor, S. Benferhat, and K. Mellouli. Any-time propagation algorithm for min-based possibilistic graphs. *Soft Comput.*, 8 :150–161, 2003.
- [10] N. Ben Amor, D. Dubois, H. Gouider, and H. Prade. Possibilistic preference networks. *Inf. Sci.*, 460–461 :401–415, 2018.

- [11] S. Benferhat. Interventions and belief change in possibilistic graphical models. *Artif. Intell.*, 174(2) :177–189, 2010.
- [12] S. Benferhat, Z. Bouraoui, and Z. Loukil. Min-based fusion of possibilistic DL-Lite knowledge bases. *Proc. IEEE/WIC/ACM Int. Conf. on Web Intelligence (WI'13)*, Atlanta, 23-28, 2013.
- [13] S. Benferhat, G. Brewka, and D. Le Berre. On the relation between qualitative choice logic and possibilistic logic. In *IPMU vol.2*, pages 951–957, 2004.
- [14] S. Benferhat, D. Dubois, L. Garcia, and H. Prade. On the transformation between possibilistic logic bases and possibilistic causal networks. *Int. J. Approx. Reas.*, 29 :135–173, 2002.
- [15] S. Benferhat, D. Dubois, S. Kaci, and H. Prade. Possibilistic merging and distance-based fusion of propositional information. *Ann. Math. A I*, 34 :217–252, 2002.
- [16] S. Benferhat, D. Dubois, S. Kaci, and H. Prade. Bipolar possibility theory in preference modeling : Representation, fusion and optimal solutions. *Inf. Fusion*, 7 :135–150, 2006.
- [17] S. Benferhat, D. Dubois, S. Kaci, and H. Prade. Modeling positive and negative information in possibility theory. *Int. J. Intel. Syst.*, 23 :1094–1118, 2008.
- [18] S. Benferhat, D. Dubois, S. Kaci, and H. Prade. Bipolar possibilistic representations. *Proc. UAI*, 45-52, 2002.
- [19] S. Benferhat, D. Dubois, and H. Prade. Nonmonotonic reasoning, conditional objects and possibility theory. *Artif. Intell.*, 92 :259–276, 1997.
- [20] S. Benferhat, D. Dubois, and H. Prade. Practical handling of exception-tainted rules and independence information in possibilistic logic. *Appl. Intell.*, 9(2) :101–127, 1998.
- [21] S. Benferhat, D. Dubois, and H. Prade. An overview of inconsistency-tolerant inferences in prioritized knowledge bases. In *Fuzzy Sets, Logic and Reasoning about Knowledge*. 395-417, Kluwer, 1999.
- [22] S. Benferhat, D. Dubois, and H. Prade. Possibilistic and standard probabilistic semantics of conditional knowledge bases. *J. Log. Comput.*, 9 :873–895, 1999.
- [23] S. Benferhat, D. Dubois, and H. Prade. Towards a possibilistic logic handling of preferences. *Appl. Intell.*, 14(3) :303–317, 2001.
- [24] S. Benferhat, D. Dubois, H. Prade, and M.-A. Williams. A framework for iterated belief revision using possibilistic counterparts to Jeffrey’s rule. *Fundam. Inform.*, 99(2) :147–168, 2010.
- [25] S. Benferhat, R. El Baida, and F. Cuppens. A possibilistic logic encoding of access control. In *FLAIRS Conf.*, pages 481–485, 2003.
- [26] S. Benferhat, J. Hué, Sylvain Lagrue, and J. Rossit. Interval-based possibilistic logic. In *IJCAI*, pages 750–755, 2011.
- [27] S. Benferhat and S. Kaci. Logical representation and fusion of prioritized information based on guaranteed possibility measures : Application to the distance-based merging of classical bases. *Artif. Intell.*, 148(1-2) :291–333, 2003.
- [28] S. Benferhat, A. Levray, K. Tabia, and V. Kreinovich. Compatible-based conditioning in interval-based possibilistic logic. In *IJCAI*, 2777-2783, 2015.
- [29] S. Benferhat and H. Prade. Encoding formulas with partially constrained weights in a possibilistic-like many-sorted propositional logic. *Proc. 9th IJCAI*, Edinburgh, 1281-1286, 2005.
- [30] S. Benferhat and H. Prade. Compiling possibilistic knowledge bases. *ECAI*, 337-341, IOS Press, 2006.
- [31] S. Benferhat and S. Smaoui. Hybrid possibilistic networks. *Int. J. Approx. Reason.*, 44(3) :224–243, 2007.
- [32] S. Benferhat and C. Sossai. Reasoning with multiple-source information in a possibilistic logic framework. *Inf. Fusion*, 7(1) :80–96, 2006.
- [33] L. Boldrin and C. Sossai. Local possibilistic logic. *J. of Applied Non-Classical Logics*, 7(3) :309–333, 1997.
- [34] G. Brewka, S. Benferhat, and D. Le Berre. Qualitative choice logic. *Artif. Intel.*, 157 :203–237, 2004.
- [35] C. Cayrol, D. Dubois, and F. Touazi. Symbolic possibilistic logic : completeness and inference methods. *J. Log. Comput.*, 28(1) :219–244, 2018.
- [36] S. De Clercq, S. Schockaert, A. Nowé, and M. De Cock. Modelling incomplete information in Boolean games using possibilistic logic. *Int. J. Approx. Reason.*, 93 :1–23, 2018.
- [37] D. Dubois, L. Fariñas del Cerro, A. Herzig, and H. Prade. A roadmap of qualitative independence. In *Fuzzy Sets, Logics and Reasoning about Knowledge*, pages 325–350. Kluwer, 1999.
- [38] D. Dubois, P. Hajek, and H. Prade. Knowledge-driven versus data-driven logics. *J. Logic, Language, and Information*, 9 :65–89, 2000.
- [39] D. Dubois, J. Lang, and H. Prade. Theorem proving under uncertainty - A possibility theory-based approach. In *IJCAI*, pages 984–986, 1987.
- [40] D. Dubois, J. Lang, and H. Prade. Timed possibilistic logic. *Fund. Infor.*, 15 :211–234, 1991.
- [41] D. Dubois, J. Lang, and H. Prade. Possibilistic logic. In *Handbook of Logic in Artificial Intelligence and Logic Programming 3*. 439-513, OUP, 1994.
- [42] D. Dubois, D. Le Berre, H. Prade, and R. Sabbadin. Using possibilistic logic for modeling qualitative decision : ATMS-based algorithms. *Fundamenta Informaticae*, 37 :1–30, 1999.
- [43] D. Dubois, E. Lorini, and H. Prade. The strength of desires A logical approach. *Minds Mach.*, 27 :199–231, 2017.
- [44] D. Dubois and H. Prade. Possibilistic logic under matrix form. In *Fuzzy Logic in Knowledge Engineering*. Verlag TÜV Rheinland, 112-126, 1986.

- [45] D. Dubois and H. Prade. *Possibility Theory : An Approach to Computerized Processing of Uncertainty*. Plenum Press, 1988.
- [46] D. Dubois and H. Prade. Epistemic entrenchment and possibilistic logic. *Artif. Intell.*, 50 :223–239, 1991.
- [47] D. Dubois and H. Prade. Possibilistic logic : a retrospective and prospective view. *Fuzzy Sets Syst.*, 144(1) :3–23, 2004.
- [48] D. Dubois and H. Prade. Possibilistic logic - an overview. In *Computational Logic, Handbook of the History of Logic*, 9. 283–342, Elsevier, 2014.
- [49] D. Dubois and H. Prade. Inconsistency management from the standpoint of possibilistic logic. *Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems*, 23(Supp.-1) :15–30, 2015.
- [50] D. Dubois and H. Prade. Qualitative and semi-quantitative modeling of uncertain knowledge - A discussion. In *Computational Models of Rationality*, pages 280–296. College Publications, 2016.
- [51] D. Dubois and H. Prade. A crash course on generalized possibilistic logic. In *SUM*, volume 11142 of *LNCS*, pages 3–17. Springer, 2018.
- [52] D. Dubois and H. Prade. Possibilistic logic : From certainty-qualified statements to two-tiered logics - A prospective survey. In *JELIA*, volume 11468 of *LNCS*, pages 3–20. Springer, 2019.
- [53] D. Dubois and H. Prade. From possibilistic rule-based systems to machine learning - A discussion paper. *SUM*, LNCS 12322, 35–51, Springer. 2020.
- [54] D. Dubois and H. Prade. Boolean weighting in possibilistic logic. In *SUM*, volume 15350 of *LNCS*, pages 130–146. Springer, 2024.
- [55] D. Dubois and H. Prade. Possibilistic provenance. In *SUM*, LNCS 15350, 147–153. Springer, 2024.
- [56] D. Dubois, H. Prade, and R. Sabbadin. Decision-theoretic foundations of qualitative possibility theory. *Europ. J. of Operat. Research*, 128 :459–478, 2001.
- [57] D. Dubois, H. Prade, and S. Schockaert. Generalized possibilistic logic : Foundations and applications to qualitative reasoning about uncertainty. *Artif. Intell.*, 252 :139–174, 2017.
- [58] D. Dubois, H. Prade, and F. Touazi. A possibilistic logic approach to conditional preference queries. In *Proc. FQAS*. LNCS 8132, 376–388, Springer, 2013.
- [59] H. Farreny and H. Prade. Default and inexact reasoning with possibility degrees. *IEEE Trans. Syst. Man Cybern.*, 16(2) :270–276, 1986.
- [60] H. Farreny, H. Prade, and E. Wyss. Approximate reasoning in a rule-based expert system using possibility theory A case study. *IFIP Cong.*, 407–414. 1986.
- [61] P. Gärdenfors. *Knowledge in Flux*. MIT Press, 1988.
- [62] J. Hué, M. Westphal, and S. Wölfl. Towards a new semantics for possibilistic answer sets. In *KI*, LNCS, 8736, pages 159–170. Springer, 2014.
- [63] S. Kaci, S. Benferhat, D. Dubois, and H. Prade. A principled analysis of merging operations in possibilistic logic. In *UAI'00*, pages 24–31, 2000.
- [64] H. Koehler and S. Link. Possibilistic data cleaning. *IEEE Trans. Knowl. Data Eng.*, 34 :5939–5950, 2022.
- [65] O. Kuzelka, J. Davis, and S. Schockaert. Encoding Markov logic networks in possibilistic logic. In *UAI Conf.*, pages 454–463, 2015.
- [66] O. Kuzelka, J. Davis, and S. Schockaert. Induction of interpretable possibilistic logic theories from relational data. In *IJCAI*, pages 1153–1159, 2017.
- [67] C. Lafage, J. Lang, and R. Sabbadin. A logic of supporters. In *Information, Uncertainty and Fusion*, pages 381–392. Kluwer, 1999.
- [68] J. Lang. Possibilistic logic : complexity and algorithms. In *Algorithms for Uncertainty and Defeasible Reasoning*, pages 179–220. Kluwer, 2001.
- [69] A. Levray, S. Benferhat, and K. Tabia. Possibilistic networks : Computational analysis of MAP and MPE inference. *Int. J. Artif. Intel. Tools*, 29 :1–28, 2020.
- [70] S. Link and H. Prade. Relational database schema design for uncertain data. *Inf. Syst.*, 84 :88–110, 2019.
- [71] P. Nicolas, L. Garcia, I. Stéphan, and C. Lefèvre. Possibilistic uncertainty handling for answer set programming. *A. Math. Artif. Intell.*, 47 :139–181, 2006.
- [72] J. C. Nieves, M. Osorio, and U. Cortés. Semantics for possibilistic disjunctive programs. In *LPNMR*, LNCS 4483. 315–320, Springer, 2007.
- [73] C. Persia and A. Ozaki. On the learnability of possibilistic theories. In *IJCAI, 1870-1876*, 2020.
- [74] H. Prade and M. Serrurier. Bipolar version space learning. *Int. J. Intel. Syst.*, 23(10) :1135–1152, 2008.
- [75] G.I. Qi, J. Z. Pan, and Q. Ji. Extending description logics with uncertainty reasoning in possibilistic logic. In *ECSQARU*, LNCS 4724 828–839. Springer, 2007.
- [76] G.I. Qi and K.w. Wang. Conflict-based belief revision operators in possibilistic logic. *AAAI*, 800–806, 2012.
- [77] N. Rescher. *Plausible Reasoning*. Van Gorcum, 1976.
- [78] M. Richardson and P. M. Domingos. Markov logic networks. *Machine Learning*, 62 :107–136, 2006.
- [79] M. Serrurier and H. Prade. Introducing possibilistic logic in ILP for dealing with exceptions. *Artificial Intelligence*, 171 :939–950, 2007.
- [80] G. L. S. Shackle. *Expectation in Economics*. Cambridge University Press, 1949.
- [81] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.
- [82] W. Spohn. *The Laws of Belief : Ranking Theory and Its Philosophical Applications*. Oxford Univ. P., 2012.
- [83] N. Wilson, D. Dubois, and H. Prade. CP-nets, π -pref nets, and Pareto dominance. In *SUM*, volume LNCS 11940, pages 169–183. Springer, 2019.
- [84] L. A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Syst.*, 1(1) :3–28, 1978.

LeYOLO, nouvelle architecture embarquée pour la détection d'objets

Lilian Hollard¹, Lucas Mohimont¹, Nathalie Gaveau², Luiz Angelo Steffenel¹

¹ Université de Reims Champagne-Ardenne, CEA, LRC DIGIT, LICIS, Reims, France

² Université de Reims Champagne-Ardenne, INRAE, RIBP USC 1488, Reims, France

Résumé

La réduction du coût de calcul des réseaux neuronaux profonds est essentielle pour la détection d'objets en temps réel. Pourtant, les récents progrès reposent surtout sur le matériel plutôt que sur l'optimisation des modèles. Cela se remarque notamment dans les dernières architectures YOLO, où la vitesse prime sur la légèreté.

Pour répondre à ce défi, nous introduisons deux contributions majeures. D'abord, LeNeck, un cadre de détection rapide et précis, réduisant le nombre de paramètres. Ensuite, LeYOLO, un modèle optimisé pour YOLO, combinant compacité et haute précision. Ces solutions sont idéales pour les dispositifs à faible consommation, y compris les microcontrôleurs.

Mots-clés

Vision par ordinateur, Optimisation des réseaux de neurones, Architecture de réseaux de neurones, Microcontrôleurs.

Abstract

Reducing the computational cost of deep neural networks is essential for real-time object detection. However, recent progress has been based mainly on hardware rather than model optimization. This is particularly noticeable in the latest YOLO architectures, where speed precedes lightness. To address this challenge, we are introducing two significant contributions. Firstly, LeNeck is a fast and accurate detection framework that reduces the number of parameters. Secondly, LeYOLO is a model optimized for YOLO that combines compactness and high precision. These solutions are ideal for low-power devices, including microcontrollers.

Keywords

Computer vision, Neural Network Optimization, Neural Network Architecture, Microcontrollers.

1 Introduction

Un calcul efficace, un traitement en temps réel et une exécution à faible latence sont essentiels pour les dispositifs edge alimentés par l'IA, y compris les drones autonomes, les systèmes de surveillance, l'agriculture intelligente et les caméras intelligentes. Bien que le cloud computing offre

une alternative pour exécuter des modèles puissants, il présente des inconvénients tels que la latence, les contraintes de bande passante et les risques de sécurité [1, 43, 39]. Dans les applications pratiques de détection d'objets, les avancées de l'apprentissage profond se sont principalement concentrées sur l'optimisation de la vitesse pour les GPU à haute performance, souvent au détriment de l'efficacité sur le matériel à faible puissance.

Initialement introduit par Joseph Redmon et al. [25], les modèles YOLO sont connus pour leur vitesse d'inférence en détection d'objets. Ces modèles ont connu des améliorations architecturales significatives au fil des ans, tirant parti des puissances de calcul modernes.

Malgré leur vitesse inhérente, il y a eu un mouvement notable dans le développement des modèles YOLO ces dernières années. Avec les rapides avancées des capacités des GPU et les nouvelles innovations matérielles, l'accent s'est déplacé des modèles légers à ceux privilégiant la vitesse d'inférence [15, 17, 37, 14, 38]. En conséquence, les modèles YOLO sont devenus nettement plus rapides malgré l'augmentation des paramètres et des FLOP¹.

Notre travail met en évidence le fait que, malgré leur vitesse impressionnante sur les GPU, les modèles YOLO ont du mal avec le matériel sans accélération pour l'IA tel que les microcontrôleurs et les micro-ordinateurs embarqués. Par exemple, sur les microcontrôleurs STMicroelectronics - largement utilisés dans la robotique et les applications IoT - les modèles YOLO modernes prennent plus d'une seconde par inférence sur les puces les plus puissantes (Section 4, Tableau 4), les rendant inadaptés aux applications en temps réel. Sur les microcontrôleurs moins puissants, des améliorations supplémentaires sont nécessaires pour réduire le temps d'inférence, un défi que nous abordons dans cette étude. Ces contraintes posent un défi critique pour les industries dépendant de l'IA à faible puissance, où l'efficacité énergétique, la petite taille du modèle et l'utilisation optimisée des ressources sont essentielles.

Dans les tâches de classification, les recherches sur l'optimisation des comptes de paramètres et des coûts computationnels ont produit des modèles notables comme Mobile-

1. Nous décrivons les opérations en virgule flottante comme FLOP, définissant toutes les opérations arithmétiques que le réseau de neurones nécessite pour effectuer une inférence. Dans notre article, 1 FLOP est environ 2 MADD ou 2 MACC. Ainsi, la variation des benchmarks tels que MobileNet diffère de leur article original.

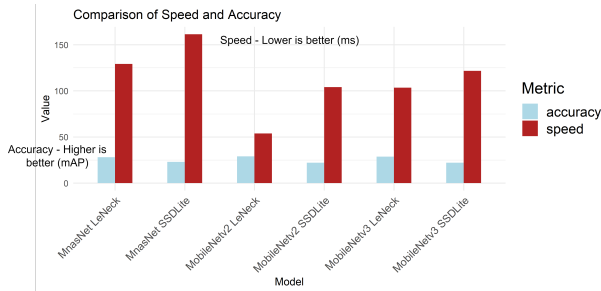


FIGURE 1 – Différence de vitesse (ms) et de précision (mAP) entre SSDLite et LeNeck sur STM32N6570-DK.

Nets [12, 27, 11] et EfficientNets [30, 31]. Bien que ces modèles soient remarquables, ils sont principalement reconnus pour leurs capacités de classification exceptionnelles plutôt que pour la détection d'objets. Les recherches se sont principalement concentrées sur les classificateurs légers, souvent associés à un ajout de détection d'objets comme SSD-Lite [20, 27]. Bien que les classificateurs à faible nombre de paramètres combinés avec SSDLite offrent une meilleure vitesse sur les microcontrôleurs, leur précision est inférieure à celle de YOLO.

Nos recherches ont identifié un écart crucial : il y a peu d'effort sur l'optimisation des architectures de détection d'objets qui équilibrent l'efficacité des paramètres et le coût computationnel tout en maintenant une précision au niveau des YOLO modernes. Cet écart oblige les développeurs à choisir entre des modèles YOLO haute performance mais coûteux en calcul et des alternatives à faible puissance comme SSDLite, qui sacrifient la précision pour la vitesse. Notre travail vise à combler cette partie manquante de recherche en introduisant des modèles de détection d'objets plus efficaces adaptés aux applications d'IA edge.

Cet article introduit deux contributions principales.

1. La première est une alternative à SSDLite appelée LeNeck qui comble l'écart entre les classificateurs à faible nombre de paramètres et les petits modèles YOLO. En utilisant LeNeck au lieu de SSDLite, nous maintenons une vitesse d'inférence similaire tout en obtenant une bien meilleure précision (Figure 1)).
2. La deuxième contribution est LeYOLO - une nouvelle famille de modèles YOLO légers et efficaces. LeYOLO correspond à la précision des échelles YOLO plus petites tout en améliorant considérablement la vitesse d'inférence sur les microcontrôleurs (Figure 2).

Nos résultats montrent que cette approche rivalise avec les modèles YOLO à des échelles comparables. Nous démontrons qu'il est possible d'optimiser l'architecture des réseaux de neurones pour la détection d'objets grâce à une nouvelle méthode d'échelle entre les classificateurs légers et les modèles YOLO.

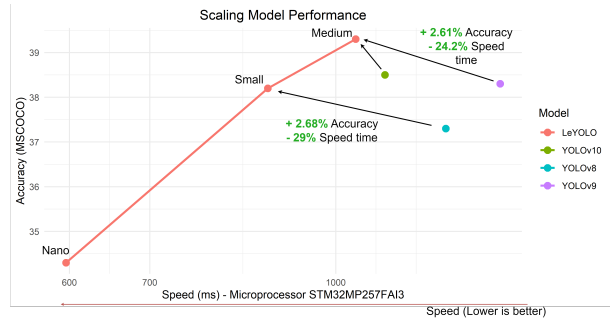


FIGURE 2 – Comparaison entre LeYOLO et les YOLO modernes, démontrant une meilleure précision pour moins de temps d'exécution sur STM32MP257FAI3.

2 Etat de l'art

Notre travail se concentre sur le développement d'une architecture optimale pour la détection d'objets en combinant deux approches clés : les détecteurs d'objets optimisés pour la vitesse et les classificateurs à faible coût conçus pour minimiser le nombre de paramètres en utilisant des techniques bien établies. LeYOLO et LeNeck intègrent des éléments connus pour leur efficacité dans la réduction de paramètres. Plus précisément, nous utilisons des Inverted Bottleneck, initialement introduits dans MobileNetV2 [27] et ensuite affinés par EfficientNet [30, 31] et GhostNet [9, 33]. Les convolutions pointwise [18] et depthwise sont des composants cruciaux dans l'optimisation de l'architecture, contribuant de manière significative à des modèles comme MNASNet [29].

L'essor des classificateurs à faible coût a conduit à SSD-Lite, une variante optimisée de SSD intégrant des convolutions groupées basées sur MobileNets. Initialement conçu pour réduire les coûts de détection en utilisant VGG [28], SSDLite partage des similitudes avec les premiers modèles YOLO [26]. Depuis lors, aucune méthode n'a significativement surpassé SSDLite, bien que SSDLiteX [16] ait tenté d'améliorer ses performances.

Du côté de YOLO, les recherches ont exploré la réduction des paramètres dans les architectures principales. Les efforts de tinier-yolo, efficient yolo, mobile densenet et autres [4, 41, 7, 40] ont intégré des éléments de classificateurs légers comme les convolutions depthwise et des techniques plus anciennes telles que les modules fire [13] pour minimiser l'utilisation des paramètres.

EfficientDet [40] partage la philosophie centrale de notre modèle : utiliser des couches à faible coût computationnel (concaténation et additions, convolutions depthwise et pointwise). Cependant, EfficientDet nécessite trop d'informations sémantiques et trop d'états de blocage (attente des couches précédentes, graphes complexes), ce qui le rend difficile à optimiser pour une vitesse d'exécution rapide.

Les auteurs de YOLOF [3] ont opté pour un modèle avec une seule entrée et une seule sortie dans le Neck². Bien

2. Partie du modèle qui partage plusieurs niveaux d'informations sémantiques

que cette conception soit théoriquement plus rapide et plus efficace en calcul, l'article YOLOF révèle une baisse significative de la précision lors de la comparaison d'un Neck à sortie unique (Single-in, Single-out - SiSO) avec un Neck à sorties multiples (Single-in, Multiple-out - SiMO).

Plus récemment, YOLOX [5] et YOLOv9 [38] ont introduit des alternatives légères avec des paramètres réduits. YOLOX remplace les convolutions standard par des convolutions depthwise de tailles de noyau plus grandes et traite des entrées d'image plus petites. YOLOv9 contribue de manière substantielle à l'optimisation des paramètres mais se concentre sur l'échelle YOLO standard plutôt que sur les architectures adaptées aux mobiles.

Enfin, Tinyssimo YOLO [24], basé sur les premiers modèles YOLO [25], se concentre sur la réduction des coûts computationnels pour permettre la détection d'objets sur les microcontrôleurs fonctionnant dans la gamme de puissance des milliwatts. Cependant, il ne parvient pas à atteindre la précision et l'efficacité même des plus petites variantes de YOLO ou des classificateurs basés sur SSDLite.

3 Optimisation des détecteurs d'objets en temps réel pour les microcontrôleurs

Les détecteurs d'objets modernes reposent sur des blocs d'architecture qui exploitent pleinement le matériel moderne. Les convolutions standard et les structures parallèles ou multi-branches sont couramment utilisées. Cependant, ces conceptions sont trop gourmandes en ressources pour les microcontrôleurs. Conçu pour être hautement efficace, le bloc de construction principal de LeYOLO optimise à la fois les paramètres et le mAP (mesure de précision pour la détection d'objet). Il s'appuie sur une structure bien connue appelée Inverted Bottleneck, couramment utilisée dans les réseaux de neurones efficaces comme MobileNets [12, 27, 11] et EfficientNets [40, 31].

Au lieu d'utiliser de grands filtres coûteux pour traiter les images, LeYOLO décompose le processus en étapes plus petites et plus efficaces en utilisant trois couches de convolution principales. Notre bloc applique une convolution 1×1 qui projette les cartes de caractéristiques des canaux C de $x \in R^{B,C,H,W}$ en un tenseur de dimension d (où $d \geq C$). Ensuite, une convolution depthwise $k \times k$ traite efficacement les caractéristiques spatiales. Enfin, une autre convolution pointwise 1×1 ramène les canaux à leur taille d'origine. Bien que de nombreux articles utilisant des Inverted Bottleneck modifient la convolution pointwise finale pour produire un nombre de canaux différent de l'entrée, LeNeck et LeYOLO ne suivent pas cette approche. Au lieu de cela, nous nous appuyons uniquement sur des convolutions standard séparées lors de la transition entre les tailles de cartes de caractéristiques après le sous-échantillonnage. Ces convolutions ajustent intrinsèquement à la fois le nombre de canaux et la taille de la carte de caractéristiques, éliminant ainsi le besoin de transformations supplémentaires au sein du Inverted Bottleneck.

Astuce d'optimisation. Normalement, la première convolution 1×1 étend les canaux avant le traitement. Cependant, si le nombre de canaux n'a pas besoin de changer (si $C == d$), nous supprimons la première convolution pointwise. Ce petit changement réduit considérablement le nombre de calculs, en particulier dans les premières couches où les images sont grandes.

Impact sur la vitesse et la précision. L'élimination des calculs inutiles rend le réseau plus rapide et plus efficace tout en maintenant une haute précision (Section 3.3). Cette optimisation est particulièrement bénéfique pour l'exécution de modèles de détection d'objets sur des dispositifs à faible puissance et à ressources limitées. Pour comparaison, SSDLite ne commence à partager des informations sémantiques qu'au niveau P4³ tandis que les détecteurs d'objets classiques et modernes commencent au niveau P3, qui fournit des détails spatiaux plus riches mais à un coût computationnel plus élevé. En réduisant stratégiquement les calculs redondants dans les premières couches, LeNeck atteint la même vitesse que SSDLite tout en exploitant le niveau P3 plus informatif, résultant en une meilleure performance de détection sans surcoût computationnel. Le modèle utilise la fonction d'activation SiLU σ , comme dans les versions modernes de YOLO (YOLOv7, YOLOv9) pour une meilleure performance.

Nous définissons les dimensions d'entrée et de sortie comme C et la dimension étendue comme d . Pour les filtres $W_1 \in R^{1,1,C,d}$, $W_2 \in R^{k,k,1,d}$, et $W_3 \in R^{1,1,d,C}$, notre approche peut être représentée comme suit :

$$y = \begin{cases} W_3 \otimes \sigma[W_2 \otimes \sigma(W_1 \otimes x)] & \text{si } d \neq C \\ W_3 \otimes \sigma[W_2 \otimes \sigma(W_1 \otimes x)] & \text{si } d = C \text{ et } W_1 = \text{Vrai} \\ W_3 \otimes \sigma[W_2 \otimes (x)] & \text{si } d = C \text{ et } W_1 = \text{Faux} \end{cases} \quad (1)$$

3.1 LeNeck - Détecteur d'objets polyvalent

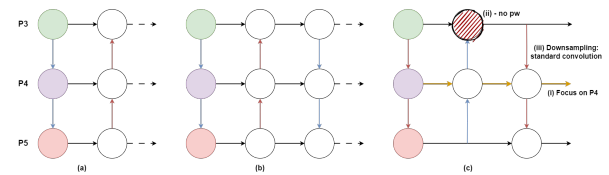


FIGURE 3 – Différence entre le Neck LeYOLO proposé et un agrégateur de caractéristiques sémantiques efficace. (a) Correspond à FPN [6]. (b) Représente PANnet [42]. Enfin, (c) est notre solution proposée.

Dans la détection d'objets, nous appelons le Neck la partie du modèle qui agrège plusieurs niveaux d'informations sémantiques, partageant les informations de couches distantes aux premières couches. Historiquement, les chercheurs ont utilisé un PANet [42] ou FPN [19] pour partager efficacement les cartes de caractéristiques, permettant plusieurs

3. P4 est le niveau sémantique de l'information correspondant à la taille de l'entrée divisée par 2^4 .

niveaux de détection en reliant plusieurs informations sémantiques P_i au PANet et leurs sorties respectives comme illustré dans la Figure 3(a). Pour créer LeNeck, nous avons identifié un aspect très important dans la composition des réseaux de neurones profonds. Nous avons remarqué qu'il y a constamment une répétition significative des couches au niveau sémantique équivalent à P4. Nous avons trouvé cela dans tous les MobileNets [12, 27, 11], dans l'optimisation des Inverted Bottleneck dans EfficientNets [30, 31] et EfficientDet [40], ainsi que dans les architectures plus récentes avec des mécanismes de self-attention comme MobileViTs [22, 23, 35], EdgeNext [21], et FastViT [34], qui sont conçus pour la vitesse. Plus intéressant encore, les modèles conçus par Neural Architecture Search (NAS) [29, 11, 30] utilisent également ce schéma. Par conséquent, nous introduisons LeNeck, un agrégateur de caractéristiques sémantiques efficace qui utilise le niveau sémantique P4 comme principal conducteur pour fusionner les informations de P3 et P5 (Figure 3.(i)). Le calcul à P3 et P5 n'est effectué qu'une seule fois, garantissant l'efficacité (P3 utilise trop de taille spatiale, et P5 utilise un nombre très étendu de canaux).

Nous réduisons le calcul - en particulier au niveau P3 en raison de la grande taille spatiale - en supprimant la première convolution pointwise (Figure 3.(ii)). Après une étude comparative (Section 3.3) réalisée sur le backbone de LeYOLO à l'échelle nano, nous avons saisi l'opportunité de supprimer les convolutions pointwise coûteuses en temps puisque les canaux d'entrée de P3 concaténés avec les caractéristiques suréchantillonnées de P4 résultent en la dimension d requise par la convolution depthwise intermédiaire de notre Inverted Bottleneck optimisé présenté dans la section 3. Chaque nombre de canaux d'entrée, ainsi que le nombre de canaux étendus du Inverted Bottleneck, ne dépasse jamais 6. L'entrée de P3 est $32C$ tandis que la dernière couche cachée du Neck de LeYOLO étend les canaux d égale 192.

Comme les convolutions standard ne sont pas très efficace en nombre de paramètres et de calcul, nous nous limitons à l'utiliser deux fois. De P3 à P4, et de P4 à P5 pour effectuer un sous-échantillonnage (Figure 3.(iii)).

3.2 Backbone de LeYOLO

Notre mise en œuvre implique la minimisation de l'échange d'informations inter-couches sous la forme de $I(X; h_1) \geq I(X; h_2) \geq \dots \geq I(X; h_n)$, avec n égal à la dernière couche cachée du backbone du réseau de neurones, en garantissant que le nombre de canaux d'entrée/sortie ne dépasse jamais une différence de ratio de 6 de la première couche cachée à la dernière. De plus, plutôt que d'augmenter la complexité computationnelle de notre modèle comme [37, 38, 10, 2], nous avons opté pour une mise à l'échelle plus efficace, intégrant la théorie du goulot d'étranglement inversé de Dangyoon Han et al. [8] qui stipulait que les convolutions pointwise ne devraient pas dépasser un ratio de 6 dans le Inverted Bottleneck.

TABLE 1 – Architecture du backbone de LeYOLO

Input	Operator	exp size	out size	NL	s
P0	conv2d, 3x3	-	16	SI	2
P1	conv2d, 1x1	16	16	SI	1
P1	bneck, 3x3, pw=False	16	16	SI	2
P2	bneck, 3x3	96	32	SI	2
P3	bneck, 3x3	96	32	SI	1
P3	bneck, 5x5	96	64	SI	2
P4	bneck, 5x5	192	64	SI	1
P4	bneck, 5x5	192	64	SI	1
P4	bneck, 5x5	192	64	SI	1
P4	bneck, 5x5	576	96	SI	2
P5	bneck, 5x5	576	96	SI	1
P5	bneck, 5x5	576	96	SI	1
P5	bneck, 5x5	576	96	SI	1

TABLE 2 – Amélioration de LeNeck et LeYOLO (best of).

Améliorations	mAP	GFLOP
base (LeYOLO nano)	34.3	2.64
+3x3	32.9	2.877
+5x5	34.9	3.946
+5x5 après P4	34.2	3.19
+Sous-échantillonnage 3x3	34.6	3.011
+aucun pw backbone et neck	34.1	2.823
+Ratio d'expansion de 2 au lieu de 3 dans LeNeck	34.3	2.64

3.3 Etude comparative

Une étude comparative en apprentissage automatique est une méthode de recherche qui teste l'impact de couches, de caractéristiques ou de techniques spécifiques en les désactivant ou en les remplaçant. L'objectif est d'identifier quels paramètres sont cruciaux pour la performance du modèle, guidant le développement d'un détecteur d'objets entièrement optimisé. Nous utilisons LeYOLO dans son intégralité (backbone + LeNeck) pour l'étude comparative afin d'affiner les deux contributions (Tableau 2).

Nous avons d'abord exploré diverses configurations de taille de noyau. Bien que des noyaux plus grands améliorent généralement les performances, ils nécessitent également plus de ressources de calcul. Le choix optimal était une convolution 5×5 après le sous-échantillonnage P4.

En suivant les idées de ConvNeXt, nous avons utilisé des convolutions séparées pour le sous-échantillonnage. Cependant, l'utilisation d'un noyau 3×3 au lieu de 5×5 dans cette configuration a conduit à de meilleurs résultats.

Enfin, nous avons fait deux optimisations critiques : la réduction du ratio d'expansion dans le Inverted Bottleneck de 3 à 2 et l'élimination de la première convolution pointwise coûteuse dans les premières couches du backbone et au niveau P3 dans le Neck. Ces modifications ont considérablement réduit le coût computationnel du modèle tout en entraînant une perte de précision de -0,3 mAP, que nous avons jugée négligeable compte tenu des gains d'efficacité.

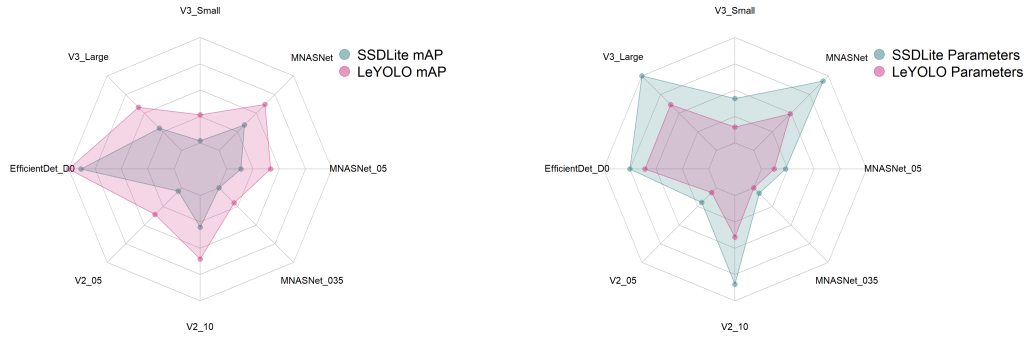


FIGURE 4 – LeYOLO comparé à SSDLite, avec un meilleur ratio paramètre-précision.

4 Résultats expérimentaux

Nous entraînons chaque réseau de neurones avec les mêmes hyperparamètres et techniques d’augmentation de données, tels que SGD, avec un taux d’apprentissage de 0,01 et un momentum de 0,9. Nous nous appuyons principalement sur l’augmentation de données mosaïque ainsi que sur hsv de {0,015,0,7,0,4} et une translation d’image de 0,1. En ce qui concerne les spécificités de l’entraînement, nous avons utilisé une taille de batch de 96 sur 4 GPU P100. La performance est évaluée sur l’ensemble de validation de MSCOCO en utilisant la précision moyenne (mAP). Pour LeYOLO, nous offrons une variété de modèles inspirés de la base architecturale présentée ci-dessus. Une approche classique implique la mise à l’échelle du nombre de canaux, de couches et de la taille d’entrée de l’image. Traditionnellement, la mise à l’échelle met l’accent sur les configurations de canaux et de couches, intégrant parfois divers schémas de mise à l’échelle (Tableau 3).

TABLE 3 – Différentes échelles pour l’entraînement de LeYOLO ainsi que leurs résultats.

Models	Nano	Small	Medium	Large
Input spatial size	640	640	640	768
Channels ratio	x1	x1.33	x1.33	x1.33
Layer ratio	x1	x1	x1.33	x1.33
mAP	34.3	38.2	39.3	41

LeYOLO évolue de la version Nano à la version Large avec une mise à l’échelle liée à ce qu’EfficientDet a apporté : répétition des canaux de 1.0 à 1.33, couches de 1.0 à 1.33, et taille spatiale pour les besoins d’entraînement de 640×640 à 768×768 . Plusieurs tailles spatiales sont utilisées à des fins d’évaluation, allant de 320×320 à 768×768 . Nous évaluons LeYOLO à des tailles spatiales réduites, tous les résultats étant présentés dans le Tableau 8.

Nous évaluons la vitesse du modèle sur deux microprocesseurs : STM32MP257FAI3 et STM32N6570-DK. Les deux utilisent des cœurs Arm Cortex, équilibrant faible consommation d’énergie et capacité de calcul efficace. Ces microcontrôleurs peuvent réaliser une inférence en temps réel à une résolution de 320×320 à 640×640 le calcul de-

vient plus exigeant, mais LeYOLO traite toujours chaque inférence en moins d’une seconde, surpassant les modèles YOLO modernes (Tableau 4).

TABLE 4 – Vitesse d’inférence de LeYOLO (640×640) et sa précision sur appareil embarqué (Onnx - STM32MP257FAI3).

Models	mAP	Speed(ms)
LeYOLO Nano	34.3	596
LeYOLO Small	38.2	877.9
LeYOLO Medium	39.3	1039
YOLOv10 Nano [36]	38.5	1099
YOLOv8 Nano [14]	37.3	1235
YOLOv9 Tiny [38]	38.3	1371

4.0.1 Détection d’objets mobile

LeYOLO surpasse les détecteurs d’objets de type YOLO sur les dispositifs embarqués ou ceux avec une puissance de calcul limitée. Nous fournissons un tableau détaillé (Tableau 8) montrant le nombre de FLOPs, et nous observons une corrélation entre cette métrique et la vitesse d’exécution sur les dispositifs à faible ressource de calcul (Tableau 4).

TABLE 5 – Vitesse d’inférence de LeNeck (320x320) et sa précision sur appareil embarqué (Onnx - STM32MP257FAI3).

Models	SSDLite	LeNeck	SSDLite	LeNeck
	Speed(ms)		mAP.95	
V3-Small	146.2	165.9	16.0	21.3
V3-Large	286.5	292.3	22	28.1
V2-1.0	259.2	256.3	22.1	28.6
MNASNet 0.5	167.7	155.3	18.5	24.6
MNASNet	306.2	262.3	23	28.9
LeYOLO Nano		165.4		25.2

Nous intégrons l’état de l’art des backbones à faible nombre de paramètres avec LeNeck. Quel que soit le backbone utilisé, tous les nombres de canaux, P3,P4 et P5 spécificités de répétition restent les mêmes. À P3, la première convolution pointwise n’est jamais utilisée, comme dans le LeYOLO de base, ce qui entraîne le premier filtre étant la convolution

TABLE 6 – Vitesse d’inférence de LeNeck (320x320) et sa précision sur appareil embarqué (Onnx - STM32N6570-DK).

Models	SSDLite	LeNeck	SSDLite	LeNeck
	Speed(ms)		mAP.95	
V3-Small	46.91	50.19	16.0	21.3
V3-Large	121.6	129.2	22	28.1
V2-1.0	104	103.5	22.1	28.6
MNASNet 0.5	86.31	36.03	18.5	24.6
MNASNet	161.3	53.78	23	28.9

TABLE 7 – Performance de LeNeck comparée à d’autres modèles de l’état de l’art combinés avec SSDLite sur MSCOCO.

Models	SSDLite	LeNeck	SSDLite	LeNeck
	Parameters(M)		mAP.95	
V3-Small	2.49	1.34	16.0	21.3
V3-Large	4.97	3.33	22	28.1
EfficientDetD0	3.9	3.29	34.6	37.1
V2-0.5	1.54	0.98	16.6	23.3
V2-1.0	4.3	2.39	22.1	28.6
MNASNet 0.35	1.02	0.7	15.6	20.0
MNASNet 0.5	1.68	1.22	18.5	24.6
MNASNet	4.68	2.8	23	28.9

depthwise de la taille exacte du nombre de canaux d’entrée équivalent du backbone.

LeNeck, en tant que détecteur d’objets général pour les classificateurs légers, conserve le même nombre de canaux⁴ et répétition des couches⁵ que la version Nano de LeYOLO. À partir d’une variété de classificateurs légers avec un faible nombre de paramètres et de FLOP, LeNeck a surpassé SSDLite dans tous les aspects de ce que nous attendons d’un modèle à faible coût - meilleure échelle de paramètres, meilleure précision, et enfin, bonne vitesse d’inférence - avec les résultats de vitesse d’inférence décrits dans les tableaux 5 - 6 et l’efficacité paramètres-précision décrite dans le Tableau 7 et la Figure 4.

4.0.2 Microcontrôleurs bas de gamme

Au-delà du traitement en temps réel, nous avons également benchmarké LeYOLO par rapport aux modèles YOLO modernes sur divers microcontrôleurs bas de gamme, comme le montre le Tableau 9, en utilisant la variante LeYOLO-Small (YOLOv10 n’est pas compatible avec ces types de microcontrôleurs). Selon le Tableau 8, LeYOLO-Small correspond à YOLOv8 à YOLOv10 en précision. De plus, il s’avère plus efficace en inférence sur ces microcontrôleurs, étendant la capacité de YOLO à fonctionner efficacement sur des dispositifs bas de gamme.

4.1 Analyse plus approfondie

Pour analyser plus en détail les résultats de vitesse, nous mettons en avant l’objectif de notre Inverted Bottleneck avec une couche pointwise optionnelle (voir la Section 3

pour plus d’informations). Dans LeYOLO Small, la convolution pointwise à haute résolution spatiale dans le backbone et au niveau P3 dans LeNeck entraîne une amélioration minimale de la précision, comme le montre la Section 3.3. Par rapport à l’Inverted Bottleneck classique, notre solution économise 8.5% de la vitesse d’inférence sur tous les STM32 benchmarkés dans l’article (Tableau 10-pw). Contrairement aux architectures YOLO standard, qui reposent sur une répétition de couches profondes avec moins de canaux, LeYOLO obtient une meilleure efficacité en utilisant un ratio d’extension de 2 au lieu de 3 tout en maintenant une profondeur de répétition minimale de 3. Ce choix de conception améliore la vitesse d’inférence de 17% tout en préservant la précision, comme le confirme notre étude expérimentale (Tableau 9-exp x3).

Notre modèle est presque aussi rapide que SSDLite tout en obtenant une précision significativement meilleure. La légère différence de vitesse provient de la taille de la carte de caractéristiques - SSDLite commence à P4, tandis que nous commençons à P3. Cependant, LeNeck reste suffisamment léger pour rivaliser avec SSDLite en vitesse. Étant donné qu’il conserve des informations spatiales plus riches, LeNeck fonctionne également mieux pour détecter diverses tailles d’objets (Tableau 8).

5 Discussions

LeNeck : Compte tenu de l’efficacité coût-efficacité de LeNeck, il existe une opportunité significative pour l’expérimentation sur différents backbones de modèles de classification de pointe. LeYOLO émerge comme une alternative prometteuse à SSD et SSDLite. Les résultats prometteurs obtenus sur MSCOCO avec notre solution suggèrent une applicabilité potentielle à d’autres modèles de classification.

Efficacité computationnelle : Nous avons mis en œuvre une nouvelle mise à l’échelle pour les modèles YOLO, prouvant qu’il est possible d’atteindre des niveaux de précision très élevés tout en utilisant très peu de ressources computationnelles (FLOP). LeYOLO fournit des résultats très rapides sur les dispositifs embarqués.

6 Conclusion

Tout au long de cet article, nous avons introduit plusieurs optimisations clés :

1. Amélioration des performances en aval du classificateur : Pour un budget de paramètres donné, LeNeck surpasse SSDLite en réduisant le nombre de paramètres tout en améliorant la précision sur MSCOCO. L’intégration de LeNeck avec les backbones existants à faible nombre de paramètres améliore la précision et l’efficacité à plusieurs échelles.
2. Une alternative viable aux modèles YOLO de petite taille : Le backbone optimisé de LeYOLO et LeNeck surpassent les variantes équivalentes des YOLO en détection d’objets. Les choix architecturaux derrière le backbone de LeYOLO entraînent une meilleure mise à l’échelle et un meilleur rapport précision-paramètres et FLOP.

4. de 32 couches à 96 avec une ratio d’expansion de 2

5. répétition $l = 3$

TABLE 8 – Etat de l’art des détecteur d’objets compatible avec les microcontrôleurs STM32.

Models	Input Size	mAP	mAP50	mAP75	S	M	L	FLOP(G)	Parameters (M)
MobileNetv3-S[12]	320	16.1	-	-	-	-	-	0.32	1.77
MobileNetv2-x0.5[27]	320	16.6	-	-	-	-	-	0.54	1.54
MnasNet-x0.5[29]	320	18.5	-	-	-	-	-	0.58	1.68
LeYOLO-Nano	320	25.2	37.7	26.4	5.5	23.7	48.0	0.66	1.1
MobileNetv3[11]	320	22	-	-	-	-	-	1.02	3.22
LeYOLO-Small	320	29	42.9	30.6	6.5	29.1	53.4	1.126	1.9
LeYOLO-Nano	480	31.3	46	33.2	10.5	33.1	52.7	1.47	1.1
MobileNetv2[27]	320	22.1	-	-	-	-	-	1.6	4.3
MnasNet[29]	320	23	-	-	-	-	-	1.68	4.8
LeYOLO-Small	480	35.2	50.5	37.5	13.3	38.1	55.7	2.53	1.9
MobileNetv1[12]	320	22.2	-	-	-	-	-	2.6	5.1
LeYOLO-Medium	480	36.4	52.0	38.9	14.3	40.1	58.1	3.27	2.4
LeYOLO-Small	640	38.2	54.1	41.3	17.6	42.2	55.1	4.5	1.9
YOLOv5-n[15]	640	28	45.7	-	-	-	-	4.5	1.9
EfficientDet-D0[32]	512	33.80	52.2	35.8	12	38.3	51.2	5	3.9
LeYOLO-Medium	640	39.3	55.7	42.5	18.8	44.1	56.1	5.8	2.4
YOLOv9-Tiny[38]	640	38.3	53.1	41.3	-	-	-	7.7	2
LeYOLO-Large	768	41	57.9	44.3	21.9	46.1	56.8	8.4	2.4

TABLE 9 – Vitesse d’inférence et précision de LeYOLO (640x640) sur des dispositifs embarqués.

Device	LeYOLO Small	YOLOv8	YOLOv9
	Speed (s)		
STM32H74I-DISCO	12.3	13.7	13.6
STM32F769I-DISCO	19	21.5	22.1
STM32F746G-DISCO	20	25	24.5
STM32F469I-DISCO	54.6	73.6	72.5

TABLE 10 – Amélioration de la vitesse d’inférence de LeYOLO Small (640x640)

Device	LeYOLO Small	pw	exp x3
	Speed (s)		
STM32H74I-DISCO	12.37	13.52	14.87
STM32F769I-DISCO	19.04	20.64	22.68
STM32F746G-DISCO	20	22.36	24.73
STM32F469I-DISCO	54.6	59.23	65.28

3. Vitesse d’inférence améliorée : LeYOLO et LeNeck obtiennent une meilleure vitesse d’inférence que les détecteurs d’objets à faible nombre de paramètres de pointe, grâce à leur architecture optimisée.

Nos contributions sont particulièrement efficaces sur les dispositifs mobiles, embarqués et à faible puissance, se rapprochant d’un équilibre idéal entre l’efficacité des paramètres et la performance de détection. La réduction de la taille du modèle tout en maintenant la précision permet la détection d’objets directement sur de petits dispositifs avec une surcharge computationnelle minimale. Ce raffinement étape par étape rapproche les modèles YOLO des applications pratiques de l’IA pour le edge.

Nous encourageons une expérimentation plus approfondie avec notre proposition, en explorant diverses variantes de jeux de données adaptées aux besoins spécifiques de l’in-

dustrie. Nous visons à fournir une gamme plus large de comparaisons pour LeYOLO dans des scénarios impliquant des dispositifs mobiles avec des ressources computationnelles très limitées.

Remerciements

Ce travail a été soutenu par Chips Joint Undertaking (Chips JU) dans le projet EdgeAI "Technologies Edge AI pour une performance embarquée optimisée" du projet, accord de subvention n° 101097300.

Références

- [1] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. Cloud computing and emerging IT platforms : Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems*, 25(6) :599–616, June 2009.
- [2] Yuxuan Cai, Yizhuang Zhou, Qi Han, Jianjian Sun, Xiangwen Kong, Jun Li, and Xiangyu Zhang. Reversible column networks. *arXiv preprint arXiv :2212.11696*, 2023.
- [3] Qiang Chen, Yingming Wang, Tong Yang, Xiangyu Zhang, Jian Cheng, and Jian Sun. You only look one-level feature. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13039–13048, 2021.
- [4] Wei Fang, Lin Wang, and Peiming Ren. Tinier-YOLO : A Real-Time Object Detection Method for Constrained Environments. *IEEE Access*, 8 :1935–1944, 2020.
- [5] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox : Exceeding yolo series in 2021. *arXiv preprint arXiv :2107.08430*, August 2021.

- [6] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Nas-fpn : Learning scalable feature pyramid architecture for object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7036–7045, 2019.
- [7] Mohammad Hajizadeh, Mohammad Sabokrou, and Adel Rahmani. MobileDenseNet : A new approach to object detection on mobile devices. *Expert Systems with Applications*, 215 :119348, April 2023.
- [8] Dongyoon Han, Sangdoo Yun, Byeongho Heo, and YoungJoon Yoo. Rethinking channel dimensions for efficient model design. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 732–741, 2021.
- [9] Kai Han, Yunhe Wang, Qi Tian, Jianyuan Guo, Chun-jing Xu, and Chang Xu. Ghostnet : More features from cheap operations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1580–1589, 2020.
- [10] Geoffrey Hinton. How to Represent Part-Whole Hierarchies in a Neural Network. *Neural Computation*, 35(3) :413–452, February 2023.
- [11] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019.
- [12] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets : Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv :1704.04861*, 2017.
- [13] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet : Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size. *arXiv preprint arXiv :1602.07360*, 2016.
- [14] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO, January 2023.
- [15] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, NanoCode012, Yonghye Kwon, Kalen Michael, TaoXie, Jiacong Fang, imyhxy, Lorna, (Zeng Yifu), Colin Wong, Abhiram V, Diego Montes, Zhi-qiang Wang, Cristi Fati, Jebastin Nadar, Laughing, UnglvKitDe, Victor Sonck, tkianai, yxNONG, Piotr Skalski, Adam Hogan, Dhruv Nair, Max Strobel, and Mrinal Jain. ultralytics/yolov5 : v7.0 - YOLOv5 SOTA Realtime Instance Segmentation, November 2022.
- [16] Hyeong-Ju Kang. Ssdlitex : Enhancing ssdlite for small object detection. *Applied Sciences*, 13(21), 2023.
- [17] Chuyi Li, Lulu Li, Hongliang Jiang, Kaiheng Weng, Yifei Geng, Liang Li, Zaidan Ke, Qingyuan Li, Meng Cheng, Weiqiang Nie, et al. Yolov6 : A single-stage object detection framework for industrial applications. *arXiv preprint arXiv :2209.02976*, September 2022.
- [18] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv :1312.4400*, 2013.
- [19] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.
- [20] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd : Single shot multibox detector. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 21–37, Cham, 2016.
- [21] Muhammad Maaz, Abdelrahman Shaker, Hisham Cholakkal, Salman Khan, Syed Waqas Zamir, Rao Muhammad Anwer, and Fahad Shahbaz Khan. EdgeNeXt : Efficiently Amalgamated CNN-Transformer Architecture for Mobile Vision Applications. In Leonid Karlinsky, Tomer Michaeli, and Ko Nishino, editors, *Computer Vision – ECCV 2022 Workshops*, Lecture Notes in Computer Science, pages 3–20, Cham, 2023.
- [22] Sachin Mehta and Mohammad Rastegari. Mobilevit : Light-weight, general-purpose, and mobile-friendly vision transformer. In *International Conference on Learning Representations*, 2022.
- [23] Sachin Mehta and Mohammad Rastegari. Separable Self-attention for Mobile Vision Transformers, June 2022.
- [24] Julian Moosmann, Marco Giordano, Christian Vogt, and Michele Magno. Tinyissimoyolo : A quantized, low-memory footprint, tinyml object detection network for low power microcontrollers. In *2023 IEEE 5th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, pages 1–5, 2023.
- [25] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once : Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, June 2016.
- [26] Joseph Redmon and Ali Farhadi. Yolov3 : An incremental improvement. *arXiv preprint arXiv :1804.02767*, 2018.
- [27] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobile-netv2 : Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.
- [28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv :1409.1556*, 2014.

- [29] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V. Le. MnasNet : Platform-Aware Neural Architecture Search for Mobile. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2815–2823, Long Beach, CA, USA, June 2019.
- [30] Mingxing Tan and Quoc Le. EfficientNet : Rethinking Model Scaling for Convolutional Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning*, pages 6105–6114. PMLR, May 2019.
- [31] Mingxing Tan and Quoc Le. EfficientNetV2 : Smaller Models and Faster Training. In *Proceedings of the 38th International Conference on Machine Learning*, pages 10096–10106. PMLR, July 2021.
- [32] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet : Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020.
- [33] Yehui Tang, Kai Han, Jianyuan Guo, Chang Xu, Chao Xu, and Yunhe Wang. GhostNetV2 : Enhance Cheap Operation with Long-Range Attention. *Advances in Neural Information Processing Systems*, 35 :9969–9982, December 2022.
- [34] Pavan Kumar Anasosalu Vasu, James Gabriel, Jeff Zhu, Oncel Tuzel, and Anurag Ranjan. Fastvit : A fast hybrid vision transformer using structural reparameterization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5785–5795, 2023.
- [35] Shakti N Wadekar and Abhishek Chaurasia. Mobilevitv3 : Mobile-friendly vision transformer with simple and effective fusion of local, global and input features. *arXiv preprint arXiv :2209.15159*, October 2022.
- [36] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Yolov10 : Real-time end-to-end object detection. *arXiv preprint arXiv :2405.14458*, 2024.
- [37] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7 : Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7464–7475, June 2023.
- [38] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao. Yolov9 : Learning what you want to learn using programmable gradient information. *arXiv preprint arXiv :2402.13616*, 2024.
- [39] Fangxin Wang, Miao Zhang, Xiangxiang Wang, Xiaoqiang Ma, and Jiangchuan Liu. Deep Learning for Edge Computing Applications : A State-of-the-Art Survey. *IEEE Access*, 8 :58322–58336, 2020.
- [40] Zixuan Wang, Jiacheng Zhang, Zhicheng Zhao, and Fei Su. Efficient Yolo : A Lightweight Model For Embedded Deep Learning Object Detection. In *2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6, July 2020.
- [41] Wang Yang, Ding BO, and Li Su Tong. TS-YOLO :An efficient YOLO Network for Multi-scale Object Detection. In *2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC)*, volume 6, pages 656–660, March 2022.
- [42] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaoqiang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.
- [43] Zhi Zhou, Xu Chen, En Li, Liekang Zeng, Ke Luo, and Junshan Zhang. Edge Intelligence : Paving the Last Mile of Artificial Intelligence With Edge Computing. *Proceedings of the IEEE*, 107(8) :1738–1762, August 2019.

Vers un *clustering* plus équilibré : augmentation de données par GMM et LLM

Noor Khalal¹, Abdallah Alaa-Eddine Djamai², Imed Keraghel^{1,2}, Mohamed Nadif¹

¹ Centre Borelli UMR 9010, Université Paris Cité

² Kernix Software

Résumé

En NLP, la gestion des thématiques sous-représentées est un défi, notamment en apprentissage non supervisé où le clustering peine à capturer les sujets minoritaires. Pour y remédier, nous proposons une méthode d'augmentation des données combinant les modèles de mélange gaussien (GMMs) et les grands modèles de langage (LLMs). Les GMMs identifient les clusters sous-représentés, tandis que les LLMs génèrent des documents synthétiques pour les enrichir. Nos expériences sur divers ensembles de données déséquilibrés montrent que cette approche préserve les performances du clustering et améliore l'interprétabilité des clusters, offrant une solution robuste et évolutive en NLP non supervisé.

Mots-clés

Augmentation des données, Grands Modèles de Langage, Modèles de Mélange Gaussien, Apprentissage non supervisé, Clustering.

Abstract

In NLP, handling underrepresented topics is challenging, particularly in unsupervised tasks where clustering may fail to capture minority topics effectively. To address this, we propose an unsupervised data augmentation method that combines Gaussian Mixture Models (GMMs) and Large Language Models (LLMs). GMMs identify underrepresented clusters, while LLMs generate synthetic documents to enrich them. Experiments on various imbalanced text datasets show that our approach maintains clustering performance and often improves interpretability, providing a robust and scalable solution for enhancing data representation in unsupervised NLP.

Keywords

Data Augmentation, Large Language Models, Gaussian Mixture Models, Unsupervised Learning, Clustering.

1 Introduction

Dans les tâches de NLP non supervisées, la qualité de la représentation des données est cruciale pour un clustering efficace. Cependant, les ensembles de données réels contiennent souvent des concepts sous-représentés, ce qui entraîne des groupes fortement déséquilibrés, difficiles à traiter pour la plupart des algorithmes de clustering. Ces

clusters sous-représentés peuvent contenir des informations significatives qui restent mal capturées par les algorithmes traditionnels, conduisant ainsi à des clusterings peu pertinents ou difficiles à interpréter.

Les techniques classiques d'augmentation de données, telles que le remplacement de synonymes et l'insertion aléatoire [23], augmentent la diversité des échantillons, mais elles s'appliquent de manière uniforme sans corriger les déséquilibres des données. De plus, de nombreuses approches existantes reposent sur des données annotées pour réaliser une augmentation ciblée [2, 10, 8], ce qui les rend inadaptées aux scénarios d'apprentissage non supervisé où les étiquettes de classe ne sont pas disponibles.

Les modèles génératifs, en particulier les grands modèles de langage (LLMs), ont récemment suscité un intérêt croissant en raison de leur capacité à produire du texte synthétique de haute qualité en capturant des relations sémantiques complexes au sein des données. Cependant, la plupart des travaux existants utilisent les LLMs pour une augmentation uniforme des ensembles de données [25], ce qui ne permet pas de résoudre le problème du déséquilibre.

Pour surmonter ces limitations, nous proposons une méthode combinant les modèles de mélange gaussien (GMMs) [1] et les LLMs pour une augmentation ciblée des données. Les GMMs analysent la distribution des embeddings et identifient les zones sous-représentées, où les LLMs génèrent ensuite des documents synthétiques pour améliorer leur représentation. La Figure 1 illustre ce processus sur le jeu de données Tweet Emotion : trois tweets sur l'optimisme sont fournis à un LLM, qui en génère un nouveau reflétant la même idée. Ce procédé enrichit la diversité du corpus et améliore sa représentation.

2 Contexte et travaux connexes

L'augmentation des données est une technique courante en NLP pour diversifier et accroître la taille des ensembles d'entraînement en générant de nouveaux échantillons [15, 2]. En classification de texte, les méthodes traditionnelles comme le remplacement de synonymes ou l'insertion aléatoire—regroupées sous Easy Data Augmentation (EDA) [23]—accroissent la variabilité des données mais n'adressent pas le déséquilibre des classes, étant appliquées uniformément. Pour corriger ce déséquilibre, des approches comme le rééchantillonnage et l'apprentissage sensible aux coûts ont été proposées [3, 10]. Cependant,

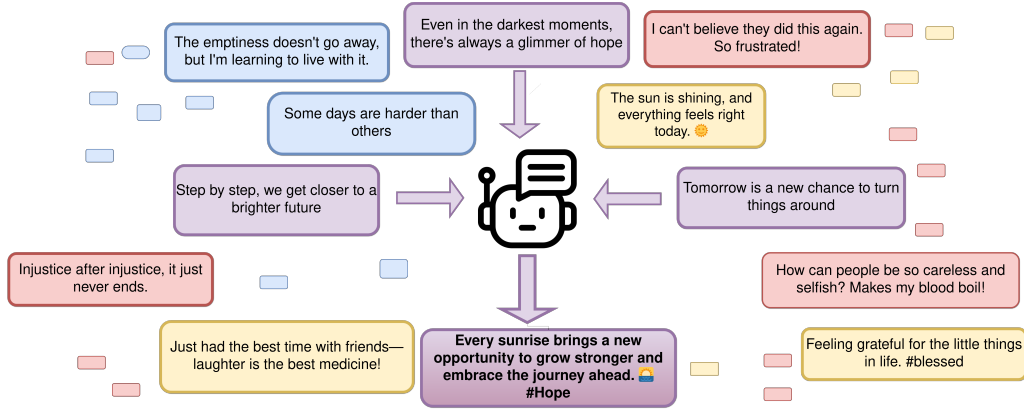


FIGURE 1 – Exemple issu du jeu de données Tweet Emotion : quatre clusters d’émotions (joie en jaune, tristesse en bleu, colère en rouge, optimisme en violet) où un LLM génère un nouveau tweet sur l’optimisme à partir de trois exemples représentatifs.

elles nécessitent des étiquettes de classe, limitant leur applicabilité en apprentissage non supervisé. Les modèles génératifs, tels que les GANs [6, 16] et les LLMs [12, 22], ont également été explorés pour l’augmentation des données, mais leur utilisation reste souvent uniforme et non ciblée [4]. Malgré ces avancées, peu de travaux combinent le clustering et les modèles génératifs pour traiter les classes sous-représentées en apprentissage non supervisé. Nos travaux comblent cette lacune en s’appuyant sur les GMMs pour détecter les clusters sous-représentés et sur les LLMs pour générer des documents synthétiques, sans dépendre de données annotées.

3 Contribution

Dans cette section, nous présentons notre approche d’augmentation des données textuelles en combinant des techniques d’encodage et des modèles gaussiens paramétriques [1]. Notre méthodologie repose sur plusieurs étapes clés : 1) représentation des documents, 2) clustering avec l’algorithme d’Expectation-Maximization (EM) [5], 3) génération de points de données synthétiques, et 4) utilisation d’un LLM pour l’augmentation des données. Cette approche est illustrée dans la Figure 2 et L’Algorithme 1.

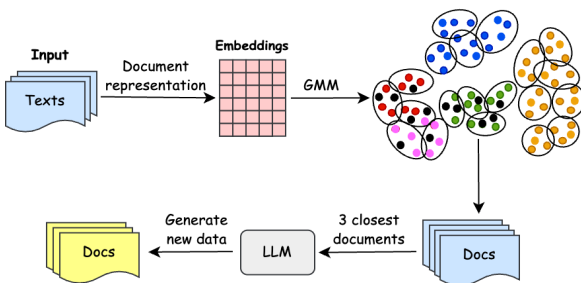


FIGURE 2 – Flux de travail pour générer des documents synthétiques. 1) Création d’embeddings, 2) clustering des embeddings via l’algorithme EM, 3) Génération de points synthétiques dans les clusters avec les meilleurs ratios volume/proportion, 4) Utilisation de ces points pour générer de nouvelles données via un LLM.

3.1 Représentation des documents

Pour le clustering, nous représentons les documents à l’aide d’un modèle d’encodage basé sur les Transformers, générant des embeddings qui capturent une information sémantique approfondie. Ces embeddings facilitent l’identification de connexions subtiles entre documents non étiquetés, améliorant ainsi la qualité du clustering [11]. Nous appliquons également UMAP [13] pour réduire la dimension des embeddings. Cette technique non linéaire préserve la structure des données tout en diminuant la complexité computationnelle, optimisant ainsi l’efficacité des algorithmes de clustering.

3.2 Clustering ciblé avec les GMMs

clustering avec les modèles de mélange gaussien (GMM). Dans un GMM fini, les données $\mathbf{x}_1, \dots, \mathbf{x}_n$ sont supposées être un échantillon de n instances indépendantes d’une variable aléatoire \mathbf{X} dans \mathbb{R}^d , où d est la dimension de l’espace. La densité des données est exprimée comme suit :

$$f(\mathbf{x}_i; \Theta) = \sum_{k=1}^g \pi_k \varphi_k(\mathbf{x}_i | \mu_k, \Sigma_k), \quad \forall i \in \{1, \dots, n\} \quad (1)$$

où $\Theta = (\pi_1, \dots, \pi_g, \mu_1, \dots, \mu_g, \Sigma_1, \dots, \Sigma_g)$, $\varphi_k(\mathbf{x}_i | \mu_k, \Sigma_k)$ est la densité de la k -ième composante pour l’observation \mathbf{x}_i avec les paramètres (μ_k, Σ_k) . Les π_k sont les poids de mélange (avec $\pi_k > 0, \sum_k \pi_k = 1$), et g est le nombre de composantes du mélange. Chaque cluster est ainsi représenté par une distribution gaussienne, dont les propriétés géométriques (volume, forme, orientation) sont définies par la matrice de covariance Σ_k [1]. L’estimation des paramètres Θ se fait en maximisant la log-vraisemblance :

$$L(\mathbf{X}; \Theta) = \sum_{i=1}^n \log \left(\sum_{k=1}^g \pi_k \varphi_k(\mathbf{x}_i | \mu_k, \Sigma_k) \right).$$

L’algorithme EM est utilisé pour maximiser cette fonction de manière itérative.

3.3 Augmentation ciblée des données

Génération de points de données synthétiques. Dans chaque cluster sous-représenté, de nouveaux points de données sont générés selon la distribution gaussienne associée. Le nombre de points générés suit une distribution multinomiale basée sur les poids π_k :

$$P(n_1, n_2, \dots, n_g) = \frac{n_{\text{samp}}!}{n_1! n_2! \dots n_g!} \pi_1^{n_1} \pi_2^{n_2} \dots \pi_g^{n_g}.$$

Ensuite, chaque composante k génère des échantillons selon une loi normale multivariée de paramètres μ_k et Σ_k . Cela garantit que les points synthétiques suivent la structure des clusters d'origine.

Génération de documents textuels. Pour chaque point généré, nous identifions ses trois plus proches voisins dans les données originales, puis un LLM génère un nouveau document en combinant leurs contenus, assurant ainsi cohérence et pertinence sémantique.

Algorithm 1 clustering et génération de documents

```

1: Entrée :  $D = \{d_1, \dots, d_n\}$ ,  $g$  (nombre de composantes),  $k^*$  (nombre de clusters à augmenter),  $\mathcal{M}$  (modèle d'encodage),  $\mathcal{M}^*$  (modèle instruction-tuné)
2:  $X_{\text{emb}} \leftarrow \mathcal{M}(D)$ 
3:  $X_R \leftarrow \text{UMAP}(X_{\text{emb}})$ 
4:  $\{C_1, \dots, C_g\} \leftarrow \text{EM}(X_R)$ 
5: for  $k \leftarrow 1, g$  do
6:    $S_k \leftarrow \frac{V_k}{\pi_k}$ 
7: end for
8: Trier  $C_k$  par  $S_k$  et retenir  $k^*$  clusters
9: Générer les nouveaux documents avec le LLM

```

4 Expériences

Nous évaluons l'efficacité de notre augmentation ciblée via le clustering, en comparant les performances avant et après augmentation avec des algorithmes comme KMeans.

4.1 Ensembles de Données

Les jeux de données utilisés dans nos expériences sont décrits dans la Table 1.

Données	Nb. Classes	Nb. Docs	Balance	#Tokens
Arxiv	12	7000	6.8×10^{-3}	10
Biorxiv	26	53787	4.1×10^{-4}	13
Medrxiv	51	17647	4.9×10^{-4}	16
Reddit	15	5114	1.2×10^{-3}	11
Tweet Emotion	4	3257	2.1×10^{-1}	16

TABLE 1 – Caractéristiques des jeux de données utilisés. **Balance** représente le ratio entre les classes minoritaires et majoritaires et **#Tokens** indique le nombre moyen de tokens.

4.2 Configuration Expérimentale

Nous utilisons le modèle *NoInstruct-Small-v0*¹, qui génère des embeddings de dimension 384, sélectionné pour ses bonnes performances sur MTEB². Pour réduire la com-

1. <https://huggingface.co/instructor/NoInstruct-Small-v0>
2. <https://huggingface.co/spaces/mteb/leaderboard>

plexité computationnelle, nous appliquons UMAP pour réduire la dimension des embeddings à 10.

Nous ajustons un GMM aux embeddings réduits X_R , avec un nombre de composantes g fixé à $\lceil nDocs^{1/3} \rceil$, suivant [24]. L'utilisation de matrices de covariance complètes permet à chaque cluster d'avoir sa propre variance.

Processus d'Augmentation Après obtention des clusters GMM, nous calculons le ratio volume/proportion $S_k = \frac{V_k}{\pi_k}$ pour chaque cluster, où V_k est le volume et π_k son poids. Les clusters sont triés par ordre décroissant de S_k afin d'identifier ceux couvrant un grand volume mais contenant peu de points de données. Nous sélectionnons les k^* clusters ayant les plus hauts S_k pour l'augmentation. Pour chaque cluster sélectionné, nous générons des embeddings synthétiques $(DA)_k$. Chaque embedding est associé aux trois documents les plus proches dans X_R , qui sont fournis au modèle *Mistral-7B-Instruct-v0.2* [9] afin de générer de nouveaux documents textuels. Ces documents $(DA)^*$ sont ensuite encodés et ajoutés à X_{emb} , formant ainsi l'ensemble de données augmenté.

5 Résultats et discussion

Nous évaluons l'impact de notre augmentation des données selon trois aspects. D'abord, nous analysons la distribution des clusters pour observer son influence sur l'équilibre et la répartition des documents dans l'espace latent. Ensuite, nous comparons les mots-clés afin d'évaluer l'amélioration de la couverture thématique et de l'interprétabilité des clusters. Enfin, nous mesurons la performance du clustering à l'aide des métriques NMI [21] et ARI [20].

5.0.1 Comparaison des mots-clés

Pour évaluer l'interprétabilité des classes, nous analysons la distribution des mots-clés dans les clusters KMeans avant et après augmentation des données. Les mots-clés sont extraits à l'aide de KeyBERT. Le tableau 2 présente les 12 mots-clés les plus fréquents pour chaque cluster dans les données augmentées et non augmentées sur plusieurs jeux de données : *Tweet_Emotions*, *Reddit* et les corpus scientifiques (*Arxiv*, *Biorxiv* et *Medrxiv*).

L'analyse révèle l'impact de l'augmentation sur les clusters sous-représentés et sur-représentés selon les domaines. Dans *Tweet_Emotions*, le cluster *Optimism* s'est enrichi de termes positifs comme *happy* et *smile*, tandis que des mots négatifs tels que *nervous* et *panic* ont disparu, affinant ainsi son orientation. Pour le cluster *Anger*, de nouveaux termes comme *game* et *racism* reflètent une expression plus large et intense de la colère, bien que la structure globale du cluster reste stable, comme attendu pour une catégorie sur-représentée.

Dans *Reddit*, le cluster sous-représenté *Skincare* est devenu plus précis avec des termes comme *cream* et *toner*, tandis que le cluster sur-représenté *Dogecoin* a peu évolué, conservant son focus sur *doge* avec une légère augmentation de termes comme *bought* et *currency*. L'impact est plus variable dans les corpus scientifiques. Dans *Arxiv*, le cluster *Economics* s'est élargi avec des termes comme *regression* et *sparsity*, reflétant un champ méthodologique plus large,

TABLE 2 – Comparaison des mots-clés et de leurs fréquences dans les clusters sur-représentés (+) et sous-représentés (-), avant et après l’augmentation des données. Le nombre en bas à droite indique la fréquence du mot. Les mots en gras sont nouveaux, tandis que les autres sont communs aux deux ensembles et soulignés s’ils ont la fréquence la plus élevée.

Cluster	Non Augmenté	Augmenté	Observations
Optimism (-)	depression ₅₈ , life ₃₈ , day ₂₆ , feel ₂₄ , nervous ₂₂ , lost ₂₁ , panic ₁₆ , optimism ₁₆ , despair ₁₅ , gloomy ₁₅ , love ₁₄ , shy ₁₃	depression ₅₈ , music ₄₀ , life ₄₀ , day ₃₆ , feel ₂₅ , lost ₂₂ , happy ₂₂ , love ₂₁ , optimism ₁₇ , sober ₁₇ , smile ₁₆ , birthday ₁₅	L’augmentation a renforcé les thèmes optimistes et positifs, apportant de la diversité tout en réduisant l’accent sur les termes négatifs ou neutres.
Anger (+)	angry ₄₁ , bully ₃₅ , outrage ₃₀ , terror ₂₈ , people ₂₆ , rage ₂₅ , dont ₂₄ , offended ₂₃ , revenge ₁₇ , irritate ₁₇ , insult ₁₇ , hate ₁₃	angry ₄₁ , bully ₃₅ , terror ₂₉ , people ₂₈ , outrage ₂₇ , rage ₂₅ , insult ₁₇ , hate ₁₃ , game ₁₀ , play ₁₀ , revenge ₉ , racism ₉	Les termes sont restés globalement stables, mais quelques nouveaux mots ont été ajoutés, exprimant une colère plus intense (ex. : <i>game</i> , <i>racism</i>).
Skincare (-)	sellus ₆ , pharmacy ₂ , moisturizer ₁ , brand ₁ , moisture ₁ , birthday ₁ , glossier ₁ , balm ₁ , rituals ₁ , babor ₁ , small ₁ , look ₁ ,	sellus ₈ skincare ₆ , sale ₅ , glossier ₄ , cream ₄ , toner ₄ , balm ₃ , treatments ₃ , dark ₃ , spot ₃ , sun ₃ , pharmacy ₂ , moisturizer ₂ ,	L’augmentation a introduit des termes spécifiques aux soins de la peau (ex. : <i>skincare</i> , <i>cream</i> , <i>toner</i>) et a légèrement orienté le focus vers les ventes (<i>sale</i>).
DogeCoin (+)	doge ₂₅₀ , bought ₁₇ , currency ₁₁ , market ₇ , coin ₇ , community ₇ , dollar ₆	doge ₂₅₄ , bought ₂₇ , currency ₁₇ , <u>dollar</u> ₁₁ , <u>coin</u> ₁₁ , <u>community</u> ₁₁ , price ₁₀ , market ₈	Les termes sont restés cohérents, avec de légères augmentations de fréquence. L’augmentation a légèrement mis l’accent sur les aspects financiers, sans modifier le focus principal sur <i>doge</i> .
Economics (-)	sustainable ₅ , inequality ₄ , frequency ₄ , likelihood ₃ , dimensional ₂ , regression ₂ , pandemic ₁ , governments ₁ , evolutionary ₁ , strategy ₁ , approach ₁ , macroecon ₁	model ₇ , regression ₅ , series ₄ , dimensional ₄ , learning ₃ , pandemic ₃ , sparsity ₂ , sustainable ₂ , approach ₂ , <u>likelihood</u> ₁	L’augmentation a ajouté des termes liés aux méthodes économiques (ex. : <i>series</i> , <i>sparsity</i>) et renforcé l’accent sur les applications statistiques.
CS (+)	learning ₄₇ , neural ₂₁ , networks ₂₀ , classification ₁₇ , adversarial ₁₅ , deep ₁₅ , models ₁₄ , detection ₁₂ , recognition ₁₁	learning ₇₄ , neural ₄₃ , <u>detection</u> ₃₅ , <u>networks</u> ₃₁ , <u>deep</u> ₃₁ , <u>classification</u> ₃₀ , recognition ₂₇ , adversarial ₂₂ , <u>models</u> ₂₁ ,	Les termes restent stables, avec un accent renforcé sur des notions clés en informatique comme <i>learning</i> , <i>neural</i> et <i>detection</i> .
A B&C (-)	drosophila ₁₄ , olfactory ₁₀ , neurons ₆ , circadian ₄ , cortex ₄ , pathway ₃ , learning ₃ , dopaminergic ₃ , model ₃ , light ₂ , endocrine ₂	drosophila ₂₅₇ , olfactory ₁₇₆ , <u>circadian</u> ₁₆₆ , sleep ₆₆ , clock ₆₂ , neurons ₅₄ , taste ₄₇ , <u>light</u> ₃₀ , sensory ₂₆ , rhythms ₂₄ , odorant ₂₄	Les nouveaux termes comme <i>sleep</i> , <i>clock</i> et <i>taste</i> élargissent le contexte comportemental.
Neuroscience(+)	neural ₆₉ , cortex ₅₈ , learning ₃₉ , visual ₃₈ , memory ₃₀ , auditory ₂₅ , temporal ₁₉ , cortical ₁₉ , speech ₁₇ , spatial ₁₆ , dynamics ₁₅ , prefrontal ₁₅	neural ₅₄₆ , cortex ₄₀₇ , <u>visual</u> ₃₁₈ , learning ₂₈₀ , brain ₂₅₁ , memory ₂₄₄ , <u>cortical</u> ₁₉₅ , auditory ₁₆₈ , attention ₁₃₂ , perception ₁₂₁ , dynamics ₁₁₉ , speech ₁₁₇	L’accent sur <i>neural</i> et <i>cortex</i> s’intensifie, tandis que <i>brain</i> et <i>perception</i> élargissent le focus aux processus cognitifs et sensoriels.
Neurology (-)	alzheimers ₃₅ , cognitive ₁₄ , disease ₁₂ , dementia ₁₀ , brain ₁₀ , genetic ₅ , impairment ₅ , biomarkers ₄ , diagnosis ₄ , amyotrophic ₄ , trials ₄ , risk ₄	alzheimers ₁₅₆ , parkinsons ₁₁₅ , cognitive ₁₁₅ , disease ₁₀₇ , brain ₉₇ , dementia ₈₅ , <u>impairment</u> ₄₅ , epilepsy ₄₄ , genetic ₃₀ , eeg ₂₉ , stroke ₂₈ , cognition ₂₅	Les termes <i>parkinsons</i> et <i>stroke</i> élargissent le champ aux maladies neurologiques, tandis que <i>alzheimers</i> et <i>dementia</i> gagnent en importance, renforçant le focus sur les maladies dégénératives et la fonction cérébrale.
Epidemiology(+)	sarscov2 ₈₉ , covid19 ₂₈ , antigen ₂₂ , testing ₁₉ , diagnostic ₁₆ , rtPCR ₁₅ , saliva ₁₄ , rna ₁₃ , detection ₁₀ , test ₉ , nasopharyngeal ₉ , viral ₈	sarscov2 ₃₂₉ , antigen ₁₂₅ , covid19 ₁₀₄ , <u>detection</u> ₇₀ , testing ₇₀ , saliva ₆₇ , <u>diagnostic</u> ₅₄ , rtPCR ₄₄ , <u>test</u> ₃₉ , <u>rna</u> ₃₂ , tests ₃₁ , screening ₃₀	L’accent sur le diagnostic et les tests se renforce avec l’augmentation de termes comme <i>sarscov2</i> , <i>antigen</i> et <i>covid19</i> . Les ajouts de <i>tests</i> et <i>screening</i> élargissent les approches de détection épidémiologique.

alors que le cluster *Computer Science* est resté stable avec quelques ajouts mineurs. Dans *Biorxiv*, le cluster *Animal Behavior and Cognition (A B&C)* a introduit des termes sensoriels comme *sleep* et *taste*, suggérant un intérêt accru pour les mécanismes sensoriels. Le cluster *Neuroscience* a

vu apparaître de nouveaux termes comme *brain*, tandis que des mots-clés centraux tels que *cortex* ont connu une augmentation significative (de 69 à 546 occurrences), reflétant l’augmentation proportionnelle des données générées. Dans *Medrxiv*, le cluster *Neurology* a mis en avant des termes

TABLE 3 – Résultats des performances de clustering (moyenne \pm écart-type). Les résultats sont reportés pour les ensembles de données non augmentés (N.A) et augmentés (A).

Algo.	Type	Métrique	Tweet_Emo.	Reddit	Arxiv	Medrxiv	Biorxiv
KMeans	N.A	NMI	20.32 \pm 2.28	54.74 \pm 1.73	44.22 \pm 0.91	30.07 \pm 0.21	34.03 \pm 0.17
		ARI	22.79 \pm 6.23	25.69 \pm 2.52	32.09 \pm 2.26	6.92 \pm 0.35	20.09 \pm 1.79
	A	NMI	22.10 \pm 3.49	55.08 \pm 0.58	44.28 \pm 0.57	34.62 \pm 0.21	30.39 \pm 0.21
		ARI	22.81 \pm 5.83	29.41 \pm 1.78	34.10 \pm 1.64	7.22 \pm 0.24	20.59 \pm 1.49
SKmeans	N.A	NMI	20.85 \pm 2.80	55.91 \pm 0.83	43.59 \pm 0.48	30.03 \pm 0.30	34.12 \pm 0.16
		ARI	19.39 \pm 6.72	29.88 \pm 1.83	32.75 \pm 1.49	7.17 \pm 0.44	22.24 \pm 2.25
	A	NMI	22.42 \pm 5.04	55.46 \pm 1.21	44.42 \pm 0.72	30.44 \pm 0.29	33.86 \pm 0.29
		ARI	23.67 \pm 9.74	31.53 \pm 2.78	33.58 \pm 1.60	7.65 \pm 0.34	23.36 \pm 3.01

comme *parkinsons* et *stroke*, renforçant la thématique des maladies neurologiques, tandis que le cluster *Epidemiology* a introduit *sarscov2* et *testing*, soulignant un accent mis sur le diagnostic et la détection épidémiologique.

Globalement, ces résultats montrent que l’augmentation a efficacement enrichi les clusters sous-représentés avec des termes spécifiques et variés, améliorant leur interprétabilité. Les clusters sur-représentés, comme attendu, ont montré peu de changements structurels.

5.0.2 Distribution des Clusters

La Figure 3 illustre la distribution des documents dans les clusters pour deux ensembles de données : Reddit et Arxiv. Ces ensembles, avec un nombre modéré de clusters (15 et 12), permettent une visualisation claire des schémas de répartition. Les graphiques en barres comparent trois scénarios : les étiquettes de classe originales (zigzags bleus), le regroupement KMeans sur les données non augmentées (briques rouges) et après augmentation (points roses).

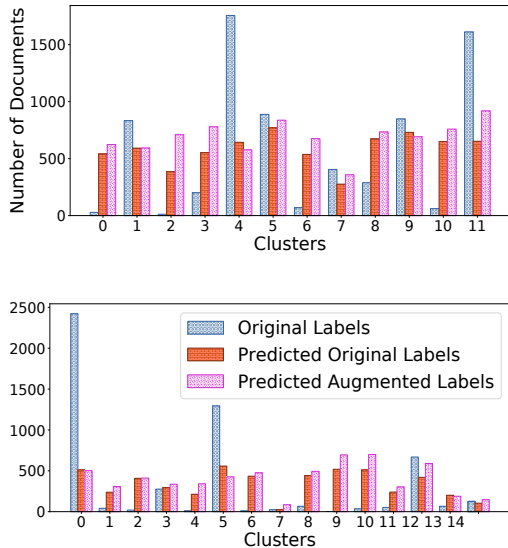


FIGURE 3 – Répartition des documents dans les clusters pour les ensembles de données Arxiv (en haut) et Reddit (en bas).

Dans les étiquettes initiales (zigzags bleus), le déséqui-

libre des ensembles de données est évident, avec certaines classes surreprésentées et d’autres sous-représentées. Le KMeans appliqué aux données d’origine (briques rouges) répartit uniformément les documents, réduisant la dominance des classes surreprésentées sans pour autant refléter fidèlement le déséquilibre initial.

Après l’augmentation des données (points roses), KMeans continue d’égaleriser la taille des clusters. Cependant, une tendance importante émerge : les pics des données augmentées (points roses) s’alignent souvent avec les clusters des classes minoritaires dans la distribution initiale (zigzags bleus).

En résumé, l’augmentation des données améliore la représentation des classes minoritaires, confirmant son efficacité pour corriger le déséquilibre des données.

5.0.3 Performance du clustering

Dans cette section, nous présentons l’analyse des performances du clustering, résumée dans le Tableau 3. Les résultats comparent les métriques de clustering (NMI et ARI) sur nos cinq ensembles de données, pour deux algorithmes : KMeans et Spherical KMeans, en utilisant les données avant et après augmentation. Les métriques, moyennées sur cinq exécutions, montrent que l’augmentation améliore souvent les performances, en particulier pour l’ARI, et maintient des résultats comparables même dans les pires scénarios, garantissant ainsi que la qualité du clustering n’est pas compromise.

6 Conclusion

Dans ce travail, nous avons proposé un nouveau cadre d’augmentation des données intégrant les GMMs et les LLMs afin de répondre aux défis liés au déséquilibre des classes dans les tâches de traitement automatique du langage naturel non supervisées. Notre approche cible spécifiquement les régions sous-représentées des ensembles de données, en utilisant les GMMs pour identifier avec précision ces clusters et les LLMs pour générer des documents synthétiques contextuellement pertinents. À travers des expériences approfondies sur plusieurs ensembles de données textuelles déséquilibrés, nous avons démontré que notre méthode maintient les performances du clustering, améliore la représentation des classes minoritaires et enrichit l’interprétabilité des clusters. Les résultats montrent que l’aug-

mentation ciblée des données est une stratégie efficace pour pallier les déséquilibres tout en préservant la qualité des algorithmes de clustering.

Dans cette contribution, nous avons utilisé des GMMs, mais il serait pertinent d'explorer d'autres modèles de mélanges, comme les modèles von-Mises Fisher [17, 18] ou les modèles de blocs latents [7, 14]. Une limitation de l'utilisation des données générées par des LLMs est le risque de renforcer les biais des ensembles d'entraînement [19]. Bien que notre approche aligne le texte généré avec les thèmes des clusters, les travaux futurs devraient évaluer et atténuer ces biais dans les données synthétiques.

Références

- [1] Jeffrey D. Banfield and Adrian E. Raftery. Model-based gaussian and non-gaussian clustering. *Biometrics*, 49(3) :803–821, 1993.
- [2] Markus Bayer, Marc-André Kaufhold, and Christian Reuter. A survey on data augmentation for text classification. *ACM Computing Surveys*, 55(7) :1–39, 2022.
- [3] Xunxin Cai, Meng Xiao, Zhiyuan Ning, and Yuan-chun Zhou. Resolving the imbalance issue in hierarchical disciplinary topic inference via llm-based data augmentation. In *ICDMW*, pages 1424–1429, 2023.
- [4] Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Wei Liu, Ninghao Liu, et al. Auggpt : Leveraging chatgpt for text data augmentation. 2023.
- [5] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society : Series B*, 39(1) :1–22.
- [6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- [7] Gérard Govaert and Mohamed Nadif. Clustering with block mixture models. *Pattern Recognition*, 36(2) :463–473, 2003.
- [8] Hongyu Guo, Yongyi Mao, and Richong Zhang. Augmenting data with mixup for sentence classification : An empirical study, 2019.
- [9] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. 2023.
- [10] Akbar Karimi, Leonardo Rossi, and Andrea Prati. Aeda : An easier data augmentation technique for text classification, 2021.
- [11] Imed Keraghel, Stanislas Morbieu, and Mohamed Nadif. Beyond words : a comparative analysis of LLM embeddings for effective clustering. In *IDA*, pages 205–216, 2024.
- [12] Varun Kumar, Ashutosh Choudhary, and Eunah Cho. Data augmentation using pre-trained transformer models. 2020.
- [13] Leland McInnes, John Healy, and James Melville. Umap : Uniform manifold approximation and projection for dimension reduction. 2018.
- [14] Mohamed Nadif and Gérard Govaert. Block clustering of contingency table and mixture model. In *International Symposium on Intelligent Data Analysis*, pages 249–259. Springer, 2005.
- [15] Siyuan Qiu, Binxia Xu, Jie Zhang, Yafang Wang, Xiaoyu Shen, Gerard De Melo, Chong Long, and Xiaolong Li. Easyaug : An automatic textual data augmentation platform for classification tasks. In *Companion proceedings of the web conference 2020*, pages 249–252, 2020.
- [16] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. 2015.
- [17] Aghiles Salah and Mohamed Nadif. Model-based von mises-fisher co-clustering with a conscience. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 246–254. SIAM, 2017.
- [18] Aghiles Salah and Mohamed Nadif. Directional co-clustering. *Advances in Data Analysis and Classification*, 13 :591–620, 2019.
- [19] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022) :755–759, Jul 2024.
- [20] Douglas Steinley. Properties of the hubert-arable adjusted rand index. *Psychological methods*, 9(3) :386, 2004.
- [21] Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec) :583–617, 2002.
- [22] Jens Van Nooten and Walter Daelemans. Improving dutch vaccine hesitancy monitoring via multi-label data augmentation with gpt-3.5. In *the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, volume 1, pages 251–270, 2023.
- [23] Jason Wei and Kai Zou. Eda : Easy data augmentation techniques for boosting performance on text classification tasks, 2019.
- [24] M Anthony Wong. A hybrid clustering method for identifying high-density clusters. *Journal of the American Statistical Association*, 77(380) :841–847, 1982.
- [25] Junjie Ye, Nuo Xu, Yikun Wang, Jie Zhou, Qi Zhang, Tao Gui, and Xuanjing Huang. Llm-da : Data augmentation via large language models for few-shot named entity recognition. *arXiv preprint arXiv :2402.14568*, 2024.

Search is not Dead!

Christophe lecoutre

University of Artois & CNRS, CRIL

lecoute@cril.fr

Résumé

La tendance actuelle de la communauté de programmation par contraintes (PPC) à utiliser largement la technologie SAT (c'est-à-dire la traduction des contraintes et/ou le raisonnement à partir des clauses) est intéressante (et s'est avérée efficace dans bon nombre de situations), mais elle éclipse quelque peu l'intérêt de la PPC. Si l'on considère les résultats des dernières compétitions XCSP3 (2022, 2023, 2024; par exemple, voir www.cril.fr/XCSP24), on peut observer qu'un solveur PPC "pur" comme ACE peut être comparativement très efficace pour trouver des bornes de bonne qualité. En fait, notre position est que "search is not dead", une référence à l'exposé invité à la conférence CP'13 par Peter Stuckey qui affirmait que "search is dead, long live proof". Il nous semble que se concentrer uniquement sur la preuve dès le début (du processus de résolution) n'est pas nécessairement la bonne approche, en particulier avec les nouveaux progrès réalisés pour conduire la recherche de solutions (notamment, le solution-based phase saving, et les trois heuristiques complémentaires actuelles basées sur les conflits).

Mots-clés

recherche arborescente, heuristiques, contraintes

Abstract

The current trend in the CP (Constraint Programming) community to widely use SAT technology (i.e., translating constraints and/or reasoning from clauses) is interesting (and has been shown to be effective in a number of situations) but somewhat detracts from the value of CP. Considering the results of the last (2022, 2023, 2024) XCSP3 competitions (e.g., see www.cril.fr/XCSP24), one can observe that a pure CP solver like ACE can be comparatively efficient for finding good quality bounds. Actually, our position is that "search is reborn", a reference to the invited talk at the conference CP'13 by Peter Stuckey who claimed that "search is dead, long live proof". It seems to us that focusing only on proof from the start (of the solving process) is not necessarily the right approach, especially with the new advances made in search (notably, solution-based phase saving, and the three current complementary state-of-the-art conflict-based heuristics).

Keywords

tree search, heuristics, constraints

1 Introduction

La programmation par contraintes (PPC) [18, 12, 11] est une technologie utile pour modéliser et résoudre des problèmes combinatoires sous contraintes. D'une part, on peut utiliser une bibliothèque comme PyCSP³ [16] pour modéliser les problèmes qui se posent dans divers domaines d'application (par exemple, l'ordonnancement, la planification, la cryptographie, la bio-informatique, la chimie organique, etc.) Les instances de problèmes peuvent alors être directement générées à partir de modèles et de données particulières. D'autre part, pour résoudre les instances (notamment représentées au format XCSP³ [6, 7]), on peut utiliser un solveur de contraintes comme ACE, dont il est question dans le présent document. ACE est un solveur de contraintes open-source, développé en Java, qui se focalise sur les variables entières (incluant les variables booléennes ou 0/1), différentes forme de contraintes table, les contraintes globales les plus importantes, les heuristiques de recherche état-de-l'art et l'optimisation (mono-critère).

2 Vers une recherche robuste en PPC

ACE est un solveur complet, qui effectue une exploration en profondeur de l'espace de recherche, avec retours-arrières. À chaque étape (nœud de l'arbre de recherche), une décision est prise (une affectation de variable ou une réfutation de valeur) et un processus de filtrage est exécuté (mécanisme de propagation de contraintes). Pour résoudre le problème de la forte variabilité des distributions de temps d'exécution [9], la recherche est relancée régulièrement, en suivant une progression géométrique. L'ordre dans lequel les variables sont choisies au cours de la construction de la branche courante de l'arbre de recherche est déterminé par une heuristique de choix de variables; une heuristique générique classique est dom/wdeg [5], combinée à un mécanisme simulant une certaine forme de sauts en arrière intelligents [15]. L'ordre dans lequel les valeurs sont choisies lors de l'affectation des variables est déterminé par une heuristique de choix de valeurs; pour l'optimisation, il est fortement recommandé d'utiliser d'abord, si disponible, la valeur de la dernière solution trouvée [20, 8].

Pour l'optimisation (instances du problème COP, Constraint Optimization Problem), on utilise une stratégie basée sur la mise à jour de la borne chaque fois qu'une solution est trouvée; il s'agit d'une sorte de stratégie de descente (apparentée au Branch and Bound), dont le

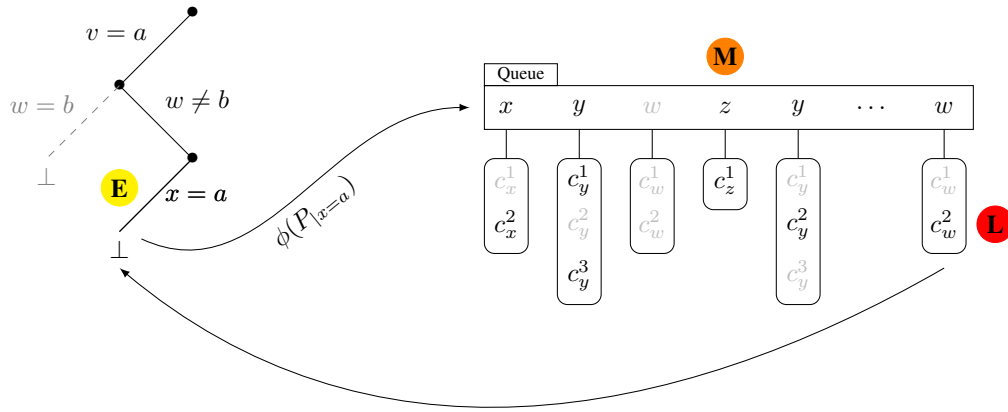


FIGURE 1 – Illustration de trois moments clés de la collecte d’informations concernant les conflits : ils correspondent au traitement précoce (Early), intermédiaire (Midway) et tardif (Late) des conflits.

principe est équivalent (dans l’hypothèse d’un problème de minimisation) à l’ajout d’une contrainte objectif spéciale $\text{obj} < \infty$ au réseau de contraintes (bien qu’elle soit initialement trivialement satisfaite), et à mettre à jour la limite de cette contrainte chaque fois qu’une nouvelle solution est trouvée. Cela signifie qu’à chaque fois qu’une solution S est trouvée avec un coût $B = \text{obj}(S)$, la contrainte objectif devient $\text{obj} < B$. Par conséquent, cette stratégie de descente progressive fournit une séquence de solutions de plus en plus intéressantes, jusqu’à prouver l’optimalité de la dernière solution trouvée.

Au moment de la rédaction de ce document, nous pensons que ACE est un solveur compétitif en raison des ingrédients suivants :

- redémarrage fréquent de la recherche
- enregistrement des nogoods de la branche courante au moment du redémarrage [14]
- pondération de contraintes pour la sélection des variables [5, 10, 21, 17, 4]
- raisonnement à partir du dernier conflit [15]
- solution(-based) phase saving [20, 8]

Fait important, dans [4], nous avons montré que trois manières d’exploiter les conflits pour guider la recherche présentent des comportements relativement complémentaires. Cela peut s’expliquer par le fait que les informations sont extraites à différents moments : au tout début du processus conduisant à un conflit (c’est-à-dire au moment de la décision), pendant la propagation de contraintes ou au moment où le dernier propagateur (algorithme de filtrage) est sollicité. On peut alors parler d’approches telles que le traitement opérationnel des conflits est *early* (E), *midway* (M) ou *late* (L). Ceci est illustré par la figure 1 où une nouvelle décision $x = a$ est prise, lors de la résolution d’un réseau de contraintes P , dans la continuité de deux décisions prises précédemment $v = a$ et $w \neq b$. Dans notre scénario, l’exécution de la propagation de contraintes ϕ sur (l’état actuel de) P après avoir assigné la valeur a à x , c’est-à-dire $\phi(P_{|x=a})$, révèle une nouvelle situation conflictuelle (dé-

signée par \perp). Le traitement précoce de ce nouveau conflit consiste à considérer la variable x impliquée dans la décision comme cause principale. C’est le principe de l’heuristique *frba/dom* [17]. Le traitement intermédiaire de ce conflit consiste à considérer que toutes les variables ayant joué un rôle (c’est-à-dire ayant été choisies) au cours de la propagation ont contribué à l’échec. C’est le principe qui sous-tend l’heuristique *pick/dom* [4]. Le traitement tardif de ce conflit consiste à considérer la dernière contrainte (ici, c_w^2) provoquant l’échec (c’est-à-dire supprimant la dernière valeur d’un domaine) comme l’objet d’intérêt. C’est le principe de la pondération des contraintes, comme dans *wdeg/dom* [5, 10, 21].

L’intérêt (la complémentarité) de ces trois heuristiques est démontré lors de la compétition XCSP³ 2024 (voir [3] et www.cril.fr/XCSP24), à laquelle deux versions de ACE [13] ont participé :

- ACE, avec son comportement par défaut (une seule heuristique de choix de variables est utilisée : *wdeg^{cacd}* [21] qui est un raffinement de *dom/wdeg*), et la première valeur (i.e., plus petite) est systématiquement choisie,
- ACE-mix, qui exploite les trois principales heuristiques de choix de variables mentionnées ci-dessus (ainsi que *wdeg^{cacd}*) et trois heuristiques de choix de valeurs (sélection de la valeur la plus petite, la plus grande, ou aléatoirement dans le domaine de la variable sélectionnée).

Il est clair que ACE-mix bénéficie de la complémentarité de ces heuristiques puisque cette version obtient de bien meilleurs scores que ACE dans les différents tracks de la compétition. Sans surprise, cela montre que la recherche devient (beaucoup) plus robuste lorsque l’on utilise des heuristiques complémentaires (ici, basées sur les conflits).

En fait, nous pensons que la tendance actuelle de la communauté PPC à utiliser largement la technologie SAT (c’est-à-dire traduire les contraintes en clauses et/ou raisonner à partir de clauses) est intéressante (et s’est avérée efficace dans

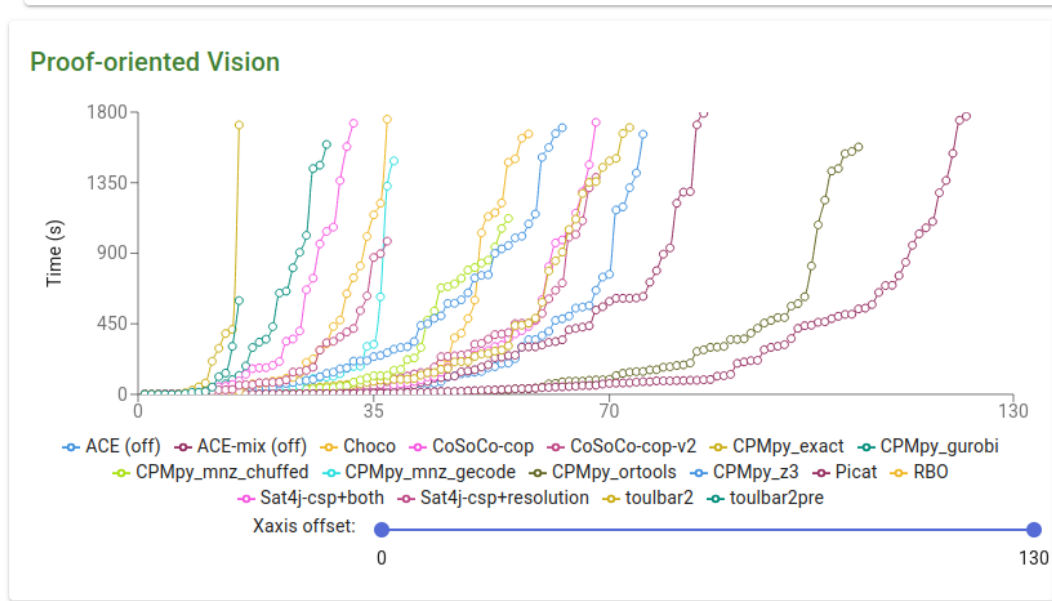


FIGURE 2 – Pour le track COP de la compétition XCSP³ 2024, avec une vision orientée preuve, Picat surpasse CPMpy_ortools, qui lui-même surpasse ACE-mix.

bon nombre de situations), mais qu’elle éclipse quelque peu la valeur de la PPC. En ce qui concerne les problèmes d’optimisation (main track COP), sur le site des résultats de la compétition XCSP³ 2024 (voir www.cril.fr/XCSP24), deux cactus plots sont visibles :

- un cactus plot, illustré à la figure 2, qui donne une vue “orientée preuve” des résultats (c’est à dire que seule les preuves d’optimalité comptent) : les approches basées sur SAT (et/ou sur la relaxation) sont très efficaces pour prouver l’optimalité ;
- un cactus plot, illustré à la figure 3, qui donne une vue “orientée recherche” (c’est à dire que les preuves d’optimalité sont ignorées) : les approches basée principalement sur la PPC sont très efficaces pour trouver des bornes de bonne qualité.

3 Search, Reborn!

Même des versions moins robustes de ACE, comme celles soumises en 2022 (voir [1] et www.cril.fr/XCSP22) et 2023 (voir [2] et www.cril.fr/XCSP23), étaient déjà très compétitives pour le main track COP. La vision orientée recherche montre déjà comparativement les très bonnes performances de ACE. En fait, notre position est que *search is not dead* (ou *search is reborn*), en référence à la conférence invitée [19] à CP’13 par Peter Stuckey qui affirmait que “search is dead, long live proof”. Comme la résolution d’un problème d’optimisation passe par trois étapes :

- la *phase d’accrochage* pendant laquelle une première solution doit être trouvée,
- la *phase de descente* pendant la quelle une solution optimale doit être trouvée, après un cheminement possiblement très long depuis la première solution,

- la *phase de preuve* pendant laquelle l’optimalité doit être prouvée,

il nous semble que se focaliser uniquement sur la preuve dès le départ n’est pas nécessairement la bonne approche, surtout avec les nouvelles avancées réalisées pour conduire la recherche (solution-based saving [20, 8] et les trois heuristiques complémentaires basées sur les conflits). Pour la phase de descente, la recherche telle qu’elle est effectuée par un solveur PPC classique peut, sous réserve d’utiliser des mécanismes de recherche complémentaires, devenir extrêmement compétitive (comme le montre ACE-mix).

Remerciements

Ce travail a bénéficié du soutien de l’ANR (France 2030), dans le cadre du projet MAIA (ANR-22-EXES-0009)

Références

- [1] G. Audemard, C. Lecoutre, and E. Lonca. Proceedings of the 2022 XCSP3 competition. Technical Report arXiv :2209.00917, CoRR, 2022. <https://arxiv.org/abs/2209.00917>.
- [2] G. Audemard, C. Lecoutre, and E. Lonca. Proceedings of the 2023 XCSP3 competition. Technical Report arXiv :2312.05877, CoRR, 2023. <https://arxiv.org/abs/2312.05877>.
- [3] G. Audemard, C. Lecoutre, and E. Lonca. Proceedings of the 2024 XCSP3 competition. Technical Report arXiv :2412.00117, CoRR, 2024. <https://arxiv.org/abs/2412.00117>.
- [4] G. Audemard, C. Lecoutre, and C. Prud’Homme. Guiding backtrack search by tracking variables during

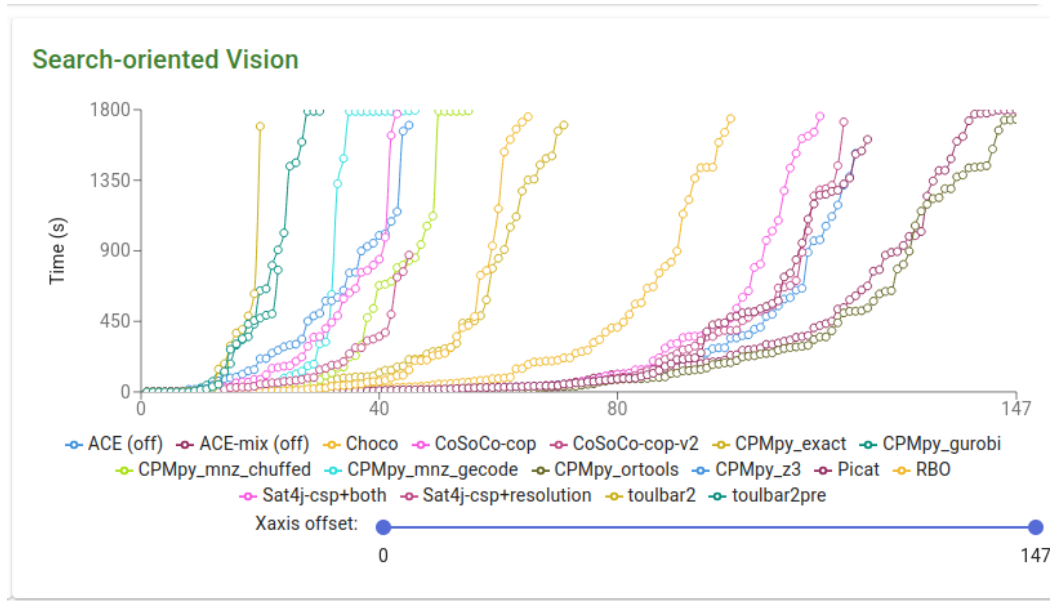


FIGURE 3 – Pour le track COP de la compétition XCSP³ 2024, avec une vision orientée recherche, ACE-mix et CPMpy_ortools sont assez proches et surpassent Picat.

- constraint propagation. In *Proceedings of CP'23*, pages 9 :1–9–7, 2023.
- [5] F. Boussemart, F. Hemery, C. Lecoutre, and L. Sais. Boosting systematic search by weighting constraints. In *Proceedings of ECAI'04*, pages 146–150, 2004.
- [6] F. Boussemart, C. Lecoutre, G. Audemard, and C. Piette. XCSP3 : an integrated format for benchmarking combinatorial constrained problems. Technical Report arXiv :1611.03398, CoRR, 2016. <https://arxiv.org/abs/1611.03398>.
- [7] F. Boussemart, C. Lecoutre, G. Audemard, and C. Piette. XCSP3-core : A format for representing constraint satisfaction/optimization problems. Technical Report arXiv :2009.00514, CoRR, 2020. <https://arxiv.org/abs/2009.00514>.
- [8] E. Demirovic, G. Chu, and P. Stuckey. Solution-based phase saving for CP : A value-selection heuristic to simulate local search behavior in complete solvers. In *Proceedings of CP'18*, pages 99–108, 2018.
- [9] C. Gomes, B. Selman, N. Crato, and H. Kautz. Heavy-tailed phenomena in satisfiability and constraint satisfaction problems. *Journal of Automated Reasoning*, 24 :67–100, 2000.
- [10] D. Habet and C. Terrioux. Conflict history based search for constraint satisfaction problem. In *Proceedings of SAC'19*, pages 1117–1122, 2019.
- [11] D. E. Knuth. *Art of Computer Programming, Volume 4, Fascicle 7 : Constraint Satisfaction*. Addison Wesley, March 2025.
- [12] C. Lecoutre. *Constraint Networks : techniques and algorithms*. International Scientific and Technical Encyclopedia (ISTE Ltd) - John Wiley Inc., June 2009. 592 pages. ISBN : 9781848211063.
- [13] C. Lecoutre. ACE, a generic constraint solver. Technical Report arXiv :2302.05405, CoRR, 2023. <https://arxiv.org/abs/2302.05405>.
- [14] C. Lecoutre, L. Sais, S. Tabary, and V. Vidal. Recording and minimizing nogoods from restarts. *Journal on Satisfiability, Boolean Modeling and Computation (JSAT)*, 1 :147–167, 2007.
- [15] C. Lecoutre, L. Sais, S. Tabary, and V. Vidal. Reasoning from last conflict(s) in constraint programming. *Artificial Intelligence*, 173(18) :1592–1614, 2009.
- [16] C. Lecoutre and N. Szczepanski. PyCSP³ : Modeling combinatorial constrained problems in Python. Technical Report arXiv :2009.00326, CoRR, 2020. <https://arxiv.org/abs/2009.00326>.
- [17] Hongbo Li, Minghao Yin, and Zhanshan Li. Failure based variable ordering heuristics for solving CSPs. In *Proceedings of CP'21*, pages 9 :1–9 :10, 2021.
- [18] F. Rossi, P. van Beek, and T. Walsh, editors. *Handbook of Constraint Programming*. Elsevier, 2006.
- [19] P. Stuckey. Those who cannot remember the past are condemned to repeat it. In *CP'13*, pages 5–6, 2013. www.youtube.com/watch?v=lxICHFRNngno.
- [20] J. Vion and S. Piechowiak. Une simple heuristique pour rapprocher DFS et LNS pour les COP. In *Proceedings of JFPC'17*, pages 39–45, 2017.
- [21] H. Watez, C. Lecoutre, A. Paparrizou, and S. Tabary. Refining constraint weighting. In *Proceedings of ICAI'19*, pages 71–77, 2019.

Système multi-agents et agents autonomes génératifs : état de l'art des connaissances et des ressources

Maxime Morge¹

¹ Univ Lyon, UCBL, CNRS, INSA Lyon, UMR 5205 LIRIS, F-59622 Villeurbanne, France

Maxime.Morge@univ-lyon1.fr

Résumé

Dotés de capacités exceptionnelles pour résoudre une variété de tâches complexes, les grands modèles de langage (LLMs) ouvrent de nouvelles perspectives pour la conception de Systèmes Multi-agents et d'Agents Autonomes (SMAA) génératifs, capables de s'adapter à des environnements ouverts et de simuler des dynamiques sociales complexes. Cet article propose une analyse approfondie des LLMs et de leur intégration dans les SMAA, en mettant en lumière leur potentiel pour surmonter les limites des approches traditionnelles. Nous examinons les défis liés à la conception de SMAA génératifs tout en identifiant les verrous technologiques et scientifiques actuels.

Mots-clés

Agent génératif, Architecture d'agents, Grand modèle de langue

Abstract

Endowed with exceptional capabilities to solve a variety of complex tasks, Large Language Models (LLMs) open new avenues for designing generative Autonomous Agents and Multi-Agent Systems (AAMAS) capable of adapting to open environments and simulating complex social dynamics. This paper provides an in-depth analysis of LLMs and their integration into AAMAS, highlighting their potential to overcome the limitations of traditional approaches. We examine the challenges associated with the design of generative AAMAS while identifying current technological and scientific barriers.

Keywords

Generative Agent, Agent Architecture, Large Language Model

1 Introduction

Grâce à leur capacité à générer du texte avec fluidité et précision au point qu'il devient difficile de les distinguer de ceux rédigés par des humains, les grands modèles de langage (*Large Language Models*, LLMs) comme GPT-4 [11], LLaMA3 [32], ou Mixtral 8x7B [33] s'imposent comme une voie de recherche particulièrement prometteuse vers l'intelligence artificielle générale. Ces modèles entraînés préalablement sur des très grands corpus, possèdent plu-

sieurs milliards de paramètres et, via des ajustements, ont des capacités surprenantes pour résoudre une grande variété de tâches dans de nouveaux contextes [37, 41] : dialogue en langage naturel, génération de code, résolution de problèmes mathématiques, etc.

Cet article propose une analyse approfondie des récents développements liés aux Systèmes Multi-agents et Agents Autonomes (SMAA) qui s'appuient sur les capacités des LLMs [36, 8]. Nous privilégions le terme « SMAA génératif » plutôt que les expressions « SMAA linguistique » ou « SMAA basé sur les LLMs », car, selon nous, c'est la capacité à prédire le jeton le plus probable en fonction d'un contexte donné qui permet d'envisager de remplacer les systèmes de production traditionnellement utilisés dans les architectures d'agent [17]. Nous défendons ici l'idée selon laquelle les SMAA génératifs ouvrent des perspectives prometteuses pour résoudre des problèmes dans des environnements ouverts et simuler des dynamiques sociales complexes. Les principales contributions de cet article sont :

1. un état de l'art détaillé¹ des LLMs et de leur intégration dans les systèmes multi-agents. Nous présentons une synthèse des principes fondamentaux des LLMs, des méthodes d'ajustement et des outils existants, offrant ainsi une vue d'ensemble des ressources disponibles² pour les chercheurs et praticiens de la communauté SMA ;
2. une discussion des défis spécifiques des SMAA génératifs. Nous examinons les limitations actuelles des architectures d'agent, tout en mettant en avant le potentiel des LLMs pour dépasser ces limites ;
3. des recommandations pour le développement futur des SMAA génératifs. Nous identifions les verrous scientifiques et technologiques pour surmonter les limites actuelles.

Ces contributions visent à enrichir la compréhension de cette voie de recherche pour favoriser l'émergence d'une intelligence artificielle plus générale et plus proche de celle des humains.

1. Même si nous avons autant que possible privilégié les articles publiés dans des conférences et revues avec comité de lecture, une part significative de la littérature dans ce domaine est diffusée par les auteurs via des archives ouvertes.

2. Une liste mise à jour de ces ressources est disponible à l'URL suivante : <https://gitlab.liris.cnrs.fr/mmorge/llm4aamas>

Plan. La section 2 présente les fondements et les principes des LLMs. La section 3 évoque les méthodes pour les adapter à une tâche spécifique. La section 4 dénombre quelques LLMs accessibles. La section 5 décrit les méthodes et stratégies de conception d'instructions (*prompt engineering*). La section 6 introduit les constituants des AA propulsés par des LLMs et la section 7 dénombre quelques outils logiciels pour leur conception. La section 8 présente un état des lieux des SMA génératifs existants et la section 9 des outils disponibles pour leur conception. La section 10 discute des orientations futures.

2 LLM

L'**Intelligence Artificielle** (IA) consiste en l'analyse, la conception, l'implémentation et l'optimisation de méthodes pour donner aux machines la capacité de reproduire/simuler l'intelligence humaine [26], i.e. la compréhension, l'apprentissage, la prise de décision, la résolution de problèmes, etc. Au-delà de l'IA faible, jusqu'ici prédominante et conçue pour accomplir des tâches spécifiques et limitées, l'IA générale vise à s'adapter à diverses situations.

Parmi les nombreux champs d'étude de l'IA, l'**apprentissage automatique** vise à donner aux machines la capacité d'améliorer leurs performances pour résoudre des tâches [4] (reconnaissance de la parole, traduction, perception visuelle, planification de tâches, prédiction) sans être explicitement programmée pour. En ce qui concerne l'apprentissage supervisé, la difficulté réside dans l'estimation à partir d'observations étiquetées d'un modèle de classification dont l'erreur statistique moyenne est la plus faible.

Parmi les nombreuses méthodes d'apprentissage adaptées à différents types de problèmes et de données, les **réseaux de neurones** (RN) se distinguent par leur structure composée de nœuds, appelés neurones formels. Ces neurones utilisent une fonction de transfert pour transformer leurs entrées en sorties. Ils sont reliés entre eux par des connexions qui reflètent leur capacité à s'influencer. Bien que leur architecture puisse varier, les réseaux de neurones sont généralement composés d'au moins trois couches distinctes : une couche d'entrée, une couche dite cachée et une couche de sortie. Dans cette configuration, chaque couche reçoit comme entrées les sorties de la couche précédente. L'entraînement des réseaux de neurones repose le plus souvent sur la méthode de rétropropagation du gradient, qui ajuste les poids des connexions en remontant des erreurs de la couche de sortie vers la couche d'entrée, dans le but de minimiser les erreurs et d'atteindre une classification aussi proche que possible de l'optimum [25].

Contrairement aux RNs classiques qui ne comportent généralement qu'une ou deux couches cachées, les **réseaux neuronaux profonds** comprennent des centaines de couches cachées. Ces couches multiples permettent de traiter des données avec une structure complexe (images, vidéos, texte, parole) pour extraire des représentations à différents niveaux d'abstraction à partir de vastes ensembles de données brutes. Cette approche a permis de diviser par 2 le taux

d'erreur de classification des images dans la base de données d'images ImageNet [16].

L'identification de relations complexes dans de vastes ensembles de données constitue un défi majeur en Traitement Automatique du Langage Naturel (TALN). Les **réseaux neuronaux récurrents** (RNN) [30], spécialement conçus pour traiter des données séquentielles, sont capables de capturer les relations contextuelles entre les éléments d'un texte, appelés jetons (*tokens*). Ces réseaux permettent de construire une représentation globale du sens des mots, de modéliser des phrases et d'en générer de nouvelles. Leur flexibilité permet d'aligner des représentations contextuelles, dépassant ainsi les limites de la traduction mot-à-mot [3]. Contrairement aux RNN, qui suivent un traitement strictement séquentiel, l'architecture *Transformers* [35] repose sur un mécanisme d'attention multi-tête. Ce mécanisme examine les interactions entre les jetons de manière combinatoire et indépendante de leur position dans la séquence, permettant ainsi un traitement parallèle de l'ensemble du texte. Cette approche a marqué une avancée significative dans le domaine.

Les modèles *Transformers*, sont qualifiés de **modèles génératifs**, car leur objectif est de prédire le jeton le plus probable en fonction d'un contexte donné. En particulier, les modèles de langue visent à modéliser la probabilité générative de séquences de mots, en attribuant une probabilité à chaque mot. Sélectionner systématiquement le jeton le plus probable à chaque étape peut conduire à ignorer des séquences avec une probabilité globale plus élevée. Pour remédier à cela, la recherche par faisceau (*beam search*) identifie la séquence ayant la plus grande probabilité globale. Par ailleurs, l'ajustement de la température T permet de moduler la répartition des probabilités : lorsque $T = 1$, la distribution reste inchangée, tandis que lorsque $T \rightarrow 0$, la probabilité du jeton le plus probable se rapproche de 1, écrasant ainsi les autres options. Malgré son coût calculatoire élevé, le processus génératif permet d'accomplir différents tâches de TALN telles que la recherche d'informations dans des documents, la génération de résumé, de traduction, de contenu ou la reformulation.

La mise à disposition de modèles de langue pré-entraînés sur de vastes corpus non étiquetés (par exemple, BERT [6]) marque un changement de paradigme [41]. Ces représentations contextuelles des mots, issues du pré-entraînement, démontrent de grandes capacités à résoudre une variété de tâches en TALN. Le principe fondamental des modèles GPT (*generative pre-trained transformer*) repose sur la compression des connaissances du monde dans un modèle *Transformer* à décodeur unique via la modélisation du langage, de manière à ce qu'il puisse restituer la sémantique de ces connaissances et servir d'outil polyvalent pour résoudre différentes tâches [24]. Lorsque l'échelle des paramètres dépasse un certain seuil, ces modèles ne se contentent pas d'améliorer significativement leurs performances, mais manifestent également des capacités surprenantes, dites émergentes, absentes des modèles de plus petite taille (e.g. BERT), pour résoudre des tâches générales complexes dans une perspective d'IA forte. Ainsi, la com-

munauté scientifique a introduit le terme de « **grand modèle de langue** » (*Large Language Model*, LLM) pour désigner les modèles de langue pré-entraînés comportant plusieurs milliards de paramètres.

Contrairement aux modèles de langage pré-entraînés antérieurs, les LLMs, qui possèdent un nombre de paramètres nettement plus élevé, nécessitent un volume de données d'entraînement plus important couvrant un large éventail de contenus. Le **pré-entraînement** établit les bases des capacités des LLMs. En s'entraînant sur des très grands corpus, les LLMs peuvent acquérir des compétences essentielles en compréhension et génération de langage [1]. À titre d'exemple, les données issues de wikipédia représente 3 % du jeu de données d'entraînement du modèle GPT3 qui contient 300 milliards de jetons soit 570 Go de texte. Dans ce paradigme, le modèle pré-entraîné doit être ajusté (*fine-tuning*) pour être adapté à une tâche spécifique.

3 Ajustement de LLM

Il y a deux principales méthodes pour ajuster les LLM pré-entraînés : l'ajustement par instructions (*instruction tuning*) et l'ajustement par alignement (*alignment tuning*). La première a pour objectif d'améliorer/débloquer les capacités des LLMs, tandis que la seconde cherche à aligner leur comportement sur des valeurs/préférences humaines.

Les LLMs ne répondent pas directement à une invite. Ils se contentent d'ajouter du texte à celle-ci. L'**ajustement par instructions** (*instruction tuning*) vise à combler l'écart entre l'objectif initial du modèle — générer du texte — et les attentes des utilisateurs, qui souhaitent que le modèle suive leurs instructions et accomplisse des tâches spécifiques. L'ajustement fin (*fine-tuning*) d'un modèle de langage pré-entraîné (LLM) requiert bien moins de données et de ressources de calcul, notamment lorsque des approches paramétriquement efficaces comme l'adaptation de faible rang (*Low-Rank Adaptation*, LoRA) [15] sont employées. Ce qui distingue l'ajustement par instructions des autres formes d'ajustement supervisé (*Supervised Fine-Tuning* SFT) réside dans la nature des données utilisées. Les ensembles de données conçus pour l'ajustement par instructions se composent exclusivement de tâches similaires aux requêtes que les utilisateurs pourraient formuler. Les exemples fournis incluent des réponses illustrant les résultats souhaités pour chaque tâche. L'objectif est d'améliorer la capacité des LLMs à répondre efficacement à des instructions en langage naturel [37].

L'ajustement par instructions repose sur un entraînement supervisé à partir de données formatées de démonstration associant des instructions à des résultats attendus. Par exemple, dans une tâche de question-réponse, une instruction telle que « répondez à cette question » est ajoutée pour contextualiser chaque exemple. L'ajustement par instructions permet de rendre les modèles plus adéquats et performants face à des requêtes spécifiques en langage naturel, tout en réduisant la complexité liée à l'ingénierie des invites (*prompts*) et à l'utilisation d'exemples (cf. Section 5).

L'**ajustement par alignement** (*alignment tuning*) vise à

améliorer les LLMs selon différents critères/valeurs humaines d'alignement (utilité, honnêteté, innocuité, etc.) et réduire les hallucinations (cf. Section 5). L'apprentissage par renforcement à partir de retours humains (*Reinforcement Learning From Human Feedback*, RLHF) utilise des algorithmes comme l'optimisation de politique proximale pour adapter les LLMs en fonction des retours humains en apprenant un modèle de récompense [21]. Cette approche intègre les humains dans la boucle d'entraînement pour développer des LLMs censurés. Cette technique a été largement utilisée pour ChatGPT [23].

En résumé, le développement de modèles GPT combine un entraînement préalable non supervisé et un ajustement supervisé. Tandis que le SFT adopte une méthode d'optimisation locale (au niveau du jeton) basée sur des données de démonstration, le RLHF adopte une méthode d'optimisation globale (au niveau du texte) en impliquant des préférences/valeurs humaines. Le développement des LLMs efface la distinction entre recherche et ingénierie, car la résolution de problèmes techniques complexes s'impose.

4 LLMs sur l'étagère

Les LLMs sont majoritairement développés par l'industrie³.

LLaMA3 [32] est un LLM entraîné préalablement sur un corpus multilingue de près de 15 trillions de jetons en utilisant 16 000 GPUs H100. Disponible en plusieurs configurations comprenant 405 milliards, 70 milliards ou 8 milliards de paramètres, il peut traiter une fenêtre contextuelle qui atteint 128 000 jetons. Pour adapter efficacement LLaMA aux autres langues que l'anglais, il est souvent nécessaire d'étendre le vocabulaire original à l'aide d'instructions ou de données dans la langue cible. Parmi ces modèles étendus, Alpaca 7B [31] est un modèle issu de LLaMA 7B ajusté à l'aide d'un jeu de 52 000 exemples d'instructions formatées.

DeepSeek-R1 [5] repose sur un ajustement par apprentissage par renforcement sur des tâches nécessitant du raisonnement et où la récompense est basée sur la cohérence et l'explicabilité. Ses capacités de raisonnement ont été distillées vers des modèles plus petits de 70 milliards à 1,5 milliards de paramètres grâce à la distillation de connaissances, permettant ainsi d'obtenir des modèles plus efficaces en termes de calcul tout en conservant des performances élevées.

Mixtral Small 3 possède la même architecture que DeepSeek-R1 [33]. Ce modèle économe est capable de fonctionner en local sur une machine standard.

Contrairement aux modèles précédents qualifiés de « modèle à poids ouverts », Lucie 7B [11] est un modèle *open-source* dont le code et les données d'entraînement sont publiés. Doté de 6,71 milliards de paramètres, ce modèle expérimental a été entraîné pendant 550 000 heures sur le superordinateur Jean Zay muni de 512 GPUs H100 avec

3. Par souci de concision, nous n'aborderons ici que quelques LLMs récents publiquement accessibles aux URLs suivantes :

<https://huggingface.co/models>,

<https://www.nomic.ai/gpt4all> et <https://ollama.com>.

seulement 3000 milliards de jetons. Sa capacité de contexte est de 4096 jetons.

GPT-4 [34] est un modèle propriétaire, bien qu'il soit accessible via une API publique payante permettant de l'utiliser sans avoir à l'exécuter localement. OpenAI n'a pas communiqué publiquement de détails spécifiques concernant la taille du corpus d'entraînement, les ressources informatiques utilisées et le nombre de paramètres du modèle. La fenêtre contextuelle peut atteindre 32 768 jetons.

5 Conception d'instructions

Après l'entraînement préalable et l'ajustement du LLM, la formulation adéquate des instructions en langage naturel (*prompt*) est essentielle pour résoudre efficacement des tâches.

La **conception d'instructions** (*prompt engineering*) désigne le processus de structuration des requêtes. Les éléments clés pour obtenir des résultats précis incluent : la description de la tâche, les données d'entrée, les informations contextuelles (documents, exemples), le formatage (e.g. markdown) et le style de l'instruction (e.g. attribuer un rôle au LLM).

Parmi les techniques de *prompting*, l'**apprentissage en contexte** (*in-context learning*) consiste à fournir au modèle des informations spécifiques sans nécessiter un nouvel entraînement [37, 7]. Selon cette approche, la création d'un prompt inclut une description précise de la tâche, la sélection d'exemples de démonstration associant instructions et résultats attendus, leur combinaison et formatage, et enfin l'ajout de l'instance de test comme entrée.

La **chaîne de pensée** (*chain-of-thought*) est une stratégie de *prompting* qui, au lieu de se limiter à des paires entrée-sortie, intègre des étapes de raisonnement intermédiaires servant de lien entre les entrées et les sorties [38]. Par exemple, l'expression « réfléchis étape par étape » peut améliorer la qualité du résultat. Cette approche n'est efficace que pour des modèles suffisamment grands, qui parviennent à donner l'illusion d'un raisonnement.

La **génération augmentée de récupération** (*Retrieval-Augmented Generation*, RAG) est une stratégie de *prompting* qui consiste à intégrer dans les instructions des informations pertinentes à partir de sources de données externes pour améliorer les réponses du modèle à l'aide de connaissances spécifiques et/ou récentes [10]. La RAG suit une procédure standard qui comporte quatre étapes :

1. indexation. Les sources de données (PDF, HTML, ODS, etc.) sont nettoyées, extraites sous forme de textes bruts segmentés et plongés sémantiquement dans un espace vectoriel (*word embedding*) pour être stockés dans une base de données vectorielle ;
2. récupération du contexte. Un score de similarité entre le plongement sémantique de la requête utilisateur et les vecteurs des segments du corpus indexé est calculé. Le RAG récupère les k segments les plus similaires pour étendre la fenêtre de contexte de la requête ;
3. construction de la requête. Le *prompt* doit orienter le modèle à utiliser les informations récupérées en incluant, par exemple, la directive suivante : « veuillez vous référer aux informations suivantes pour accomplir la tâche » ;
4. génération de la réponse. Le LLM peut également être guidé pour vérifier la qualité de la réponse générée. Il peut décider de relancer une nouvelle récupération d'informations si nécessaire, en fonction des nouvelles sorties.

Injecter des connaissances spécialisées/récentes dans les LLMs tout en préservant leurs capacités émergentes reste un défi non trivial.

Bien que les LLMs soient capables de générer des textes ou du code d'une qualité indistinguable à celle produite par des humains [39], ces perroquets stochastiques demeurent sujets à des limitations. Non seulement ils héritent des **biais** contenus dans leurs données d'entraînement, mais ils sont également enclins à produire des informations inexactes, appelées **hallucinations**, qui peuvent contredire une source existante ou ne pas être vérifiables à partir des sources disponibles. En dépit de leurs avancées, les LLMs rencontrent des difficultés dans la mémorisation, la génération de données structurées, ainsi que dans la résolution de tâches complexes de raisonnement. Ils échouent à améliorer le niveau de confiance en fournissant une réponse correcte tout en suivant un raisonnement invalide, ou en produisant une réponse erronée malgré un raisonnement correct. L'agentification des LLMs est considérée comme une voie prometteuse pour surmonter ces limitations [36].

6 Agent autonome génératif

Un agent autonome est une entité capable de percevoir son environnement, de prendre des décisions et d'effectuer des actions pour atteindre des objectifs spécifiques [26]. Souvent développés dans une approche d'intelligence artificielle faible, ces agents reposent sur des heuristiques interprétables ou des algorithmes d'apprentissage par renforcement. Conçus pour des environnements spécifiques, leur efficacité se restreint aux situations anticipées par le concepteur et elle est limitée dans des environnements ouverts. En revanche, les LLMs sont considérés comme des outils difficiles à interpréter mais polyvalents, i.e. capables de réaliser une grande variété de tâches dans de nouveaux contextes. Tirer parti des connaissances de sens commun intégrées aux LLMs représente une solution prometteuse pour doter les agents autonomes des capacités nécessaires à leur ajustement à de nouvelles tâches, tout en réduisant la dépendance à l'ingénierie des connaissances ou à l'apprentissage par essais-erreurs [36].

Pour réagir de manière autonome dans un environnement, un agent génératif doit interpréter ses perceptions (e.g. une demande utilisateur) à partir des connaissances stockées dans sa mémoire, raisonner et planifier des actions. Il doit exécuter le plan étape par étape avec l'aide d'outils et affiner le plan en fonction des retours de l'environnement.

Mémoire. Un agent génératif stocke dans :

- la mémoire de travail, son rôle (ses caractéristiques et ses objectifs) et ses perceptions en fonction de leur récence, de leur pertinence et de leur importance sous forme d’observations en langage naturel dans la fenêtre de contexte ;
- la mémoire épisodique, les expériences passées décrites en langage naturel dans la fenêtre de contexte ou encodées dans un espace vectoriel (RAG) ;
- la mémoire sémantique, les faits sur le monde encodés dans un RAG ou dans une base de données ;
- la mémoire procédurale ses paramètres qui peuvent être ajustés (*fine-tuned*) pendant la durée de vie de l’agent.

Ces connaissances peuvent être récupérées, mises à jour ou inférées. Un agent propulsé par un LLM a la capacité d’apprendre, i.e. écrire dans sa mémoire à long terme, et de raisonner, i.e. résumer ses expériences passées stockées en mémoire en des idées plus larges et abstraites, pour améliorer sa cohérence comportementale.

Planification. La planification permet de résoudre des tâches complexes, telles que le raisonnement mathématique ou la gestion de dialogues à plusieurs tours, en les décomposant en sous-tâches plus simples, puis en générant un plan d’action pour traiter ces sous-tâches. Un planificateur de tâches basé sur un LLM peut tirer parti de la mémoire pour stocker et récupérer les plans. Cette approche peut être mise en œuvre à l’aide de modèles comme les LLMs pour les tâches textuelles, ou à l’aide d’outils spécialisés tels qu’un interpréteur de code (par exemple, Python) ou un planificateur externe basé sur PDDL. Ensuite, l’agent exécute ses actions.

Actions. De multiples travaux visent à doter les LLMs de la capacité à utiliser des outils externes, tels qu’une calculatrice, un calendrier, un SGBD, un interpréteur de code, un moteur de recherche, un outil de traduction automatique, un système de questions-réponses, ou un outil d’IA (e.g. [28]). Ces travaux reposent le plus souvent soit sur un ajustement supervisé (cf. Section 3), soit sur la conception d’instructions (cf. Section 5). En revanche, la nature auto-supervisée de Toolformer permet aux LLMs d’apprendre par eux-mêmes, sans perdre leur caractère généraliste, comment et quand utiliser des outils externe via une simple API combinant ainsi le meilleur des deux mondes [27].

Perceptions. Un agent génératif utilise son environnement – qu’il soit physique, numérique ou dialogique – comme moyen d’exécution mais également comme source d’information. Il reçoit des entrées telles que des requêtes utilisateur ou des états environnementaux, et génère des sorties, comme des réponses ou des actions. Les signaux de retour, qu’ils proviennent du modèle lui-même ou de sources externes comme des utilisateurs humains, des outils ou des environnements virtuels [22], permettent à l’agent d’apprendre et s’adapter de manière itérative. Cet ancrage dans un environnement lui donne la possibilité d’affiner son plan initial pour obtenir de meilleurs résultats. Par exemple, des outils comme les interpréteurs de code sont largement employés dans le cadre des tâches de programmation. Un agent génératif, s’appuyant sur un VLM (*Vision Language Mo-*

del), qui perçoit de manière native sur des entrées à la fois textuelles et visuelles, reproduit sans doute de manière plus réaliste le comportement d’un humain qui ne lit pas le code HTML mais voit une page web.

Sumers *et al.* [29] propose une architecture cognitive inspirée de Soar [17] dont le moteur est un LLM. S’inspirer de l’histoire riche de l’IA symbolique et des sciences cognitives en reliant des idées établies depuis des décennies aux recherches de pointe sur les LLMs nous semble être une approche qui ouvre la voie au développement d’une IA plus générale et plus semblable à celle des humains.

7 Agent sur l’étagère

LangChain⁴ est un cadriciel *open-source* de conception d’instructions pour LLMs qui peut être utilisé pour définir des séquences de raisonnement de haut niveau, des agents conversationnels, des RAGs, des résumés de documents ou encore la génération de données synthétiques. LangGraph⁵ est une bibliothèque de plus bas niveau pour la conception d’architecture cognitive d’agent autonome dont le moteur de raisonnement est un LLM. Elle permet de contrôler finement le flux de données et l’état de la mémoire persistante. AutoGPT⁶ est une plateforme pour la création, le déploiement et la gestion d’agent génératif. Elle permet aux LLMs de gérer la mémoire à court et long terme ainsi que d’utiliser des outils externes tels que des moteurs de recherche. AutoGPT décompose la requête en un plan détaillé, exécute ce plan étape par étape avec l’aide d’outils comme des moteurs de recherche et ajuste le plan restant en fonction des retours de l’environnement. WorkGPT⁷ est un cadriciel offrant des fonctionnalités similaires.

Les ressources pour concevoir des agents propulsés par des LLM sont très nombreuses. Awesome LLM-Powered Agent⁸ en dresse une liste impressionnante.

8 SMA génératif

Un Système Multi-Agents (SMA) est composé de multiples entités de calcul autonomes, appelées agents, en interaction, situés dans un environnement qui prennent part à une organisation. Qu’il constitue une méthode distribuée pour la résolution de problème ou un modèle explicatif d’une dynamique sociale, un SMA se caractérise par l’émergence de processus complexes, impossibles à modéliser par des équations prédictives ou solvables.

La principale difficulté dans la conception d’un SMA réside dans le fait que les mécanismes internes régissant les comportements sociaux ne sont pas directement observables. Les architectures d’agent réactifs s’appuient sur des règles prédéfinies (e.g. machine à états finis), des équations symboliques (e.g. modèle de force sociale), des modèles stochastiques (e.g. SIR) ou des algorithmes d’apprentissage par renforcement (e.g. *Q-learning*). Ces approches

4. <https://www.langchain.com>

5. <https://langchain-ai.github.io/langgraph>

6. <https://github.com/Significant-Gravitas/AutoGPT>

7. <https://github.com/team-openpm/workgpt>

8. <https://github.com/hyp1231/awesome-llm-powered-agent>

montrent leurs limites face à des tâches complexes. À l'inverse, les architectures cognitives s'appuient sur des modèles internes explicites qui imitent la cognition humaine dans des processus comme la perception, la prise de décision, l'apprentissage ou la planification. Bien que plus difficile à concevoir, ces architectures délibératives permettent des comportements complexes pour capturer la résolution de problèmes par des humains. Les LLMs permettent d'envisager de repenser ces architectures pour surmonter les limites de complexité et d'expressivité.

Les LLMs ont la capacité de simuler une grande diversité de rôles sociaux afin de générer des représentations plus nuancées et fidèles des processus décisionnels, des dynamiques de communication et des mécanismes d'adaptation des agents [13]. Les LLMs possèdent la capacité d'émuler un véritable humain dans certaines expériences en économie expérimentale ou en psychologie sociale – un « *homo silicus* » – (e.g. [20]) et des capacités exceptionnelles dans des scénarios sans apprentissage préalable (*zero-shot*) [12]. Les agents propulsés par les LLMs peuvent simuler des perceptions, des raisonnements et des prises de décision réalistes, réagir de manière adaptative aux environnements sans instructions explicites prédéfinies en adaptant leurs réponses via des mécanismes d'apprentissage contextuel, générer de manière autonome des objectifs, interagir et communiquer en langage naturel [8]. Leur intégration enrichit la fidélité des processus cognitifs et des simulations.

Park *et al.* [22] étudient les comportements sociaux émergents d'une simulation multi-agents générative dans un environnement inspiré du jeu *The Sims*⁹. Le comportement crédible des agents et les dynamiques sociales qui émergent reposent sur une architecture d'agent permettant aux agents de :

1. récupérer des événements/interactions pertinentes. Le comportement de l'agent est guidé par ses expériences passées, i.e. une liste d'objets horodatés décrits en langage naturel ;
2. réfléchir sur ces souvenirs. Les souvenirs sont synthétisés en inférences de haut niveau, permettant à l'agent de tirer des conclusions afin d'orienter son comportement ;
3. planifier et réagir de manière cohérente. Les conclusions tirées des souvenirs, ainsi que le contexte actuel, sont traduites en plans d'action globaux. Ceux-ci sont ensuite détaillés en comportements spécifiques et en réactions.

AGENTVERSE [2] est un cadre général multi-agents qui simule les procédures de résolution de problèmes des groupes humains en quatre étapes : 1) le recrutement d'experts ajuste la composition du groupe en fonction de l'évolution de la résolution ; 2) la prise de décision collaborative consiste à faire participer les agents sélectionnés à des discussions communes pour élaborer des stratégies de résolution ; 3) l'exécution des actions par les agents dans leur environnement pour mettre en œuvre les actions planifiées ;

9. <https://github.com/al6z-infra/ai-town>

4) l'évaluation des écarts entre l'état actuel et les résultats souhaités. Si l'état actuel n'est pas satisfaisant, un retour d'information est fourni pour affiner les itérations suivantes. Les auteurs observent des comportements émergents positifs et négatifs : des comportements volontaristes assistant leurs pairs, des comportements de conformité où les agents déviants s'alignent sur l'objectif commun sous l'effet des critiques et des comportements destructeurs qui entraînent parfois des résultats indésirables.

En s'appuyant sur SANDBOX, une plateforme open-source pour simuler une société humaine, Liu *et al.* [19] proposent un cadre d'apprentissage de l'alignement (*Stable Alignment*) qui s'appuie sur des interactions sociales simulées en trois étapes : imitation, auto-critique et réaligement. L'émergence de normes sociales est encouragée par l'instauration de discussions sur des sujets sociétaux controversés ou des questions associées à des risques.

Gao *et al.* [9] propose un SMA génératif nommé S3 (*Social-network Simulation System*) pour simuler les processus de propagation dans un réseau social. Ce SMA reproduit les contenus générés par les utilisateurs en tenant compte de leurs caractéristiques démographiques, du contexte informationnel et des mécanismes complexes qui gouvernent la perception cognitive et la prise de décision des utilisateurs. Les auteurs combinent l'ajustement supervisé (cf. Section 3) et la conception d'instructions (cf. Section 5) pour déduire les caractéristiques démographiques des utilisateurs à partir de données textuelles, avec une attention particulière portée à la prédiction de l'âge, du genre et de la profession. Cette approche permet aux LLMs de prédire, en fonction des publications reçues, la propension d'un utilisateur à repartager un contenu ou à en créer un nouveau. Les LLMs sont ainsi mobilisés pour peupler la simulation en capturant les relations complexes entre les profils des utilisateurs et leurs comportements d'interaction.

9 SMA sur l'étagère

Comme évoqué dans la section 7, LangChain permet de développer des applications basées sur des agents mais ces derniers ne sont pas conçus pour communiquer et collaborer. La structuration des interactions LangGraph sous la forme de *workflows* rend difficile la gestion de scénarios où les agents doivent collaborer de manière complexe ou effectuer des tâches nécessitant une coordination approfondie, comme le partage dynamique de connaissances ou la prise de décision collective.

MetaGPT¹⁰ [14] est un *cadriciel* pour créer des SMA génératifs dédiés au développement logiciel. Il propose de modéliser le SMA comme une entreprise de services du numérique, en s'appuyant sur la spécialisation des rôles, l'organisation des flux de travail, ainsi que des mécanismes efficaces de partage d'information, tels que les *pools* de messages et les systèmes d'abonnement.

CAMEL¹¹ [18] propose un cadre multi-agents génératif pour l'accomplissement de tâches complexes. Il propose une

10. <https://github.com/geekan/MetaGPT>

11. <https://github.com/camel-ai/camel>

technique appelée *inception prompting* pour faire émerger des comportements de coopération. Cette bibliothèque *open-source* inclut des implémentations de divers agents, des *pipelines* de génération de données, des outils d'analyse de données et des jeux de données.

AutoGen¹²[40] est un cadriciel polyvalent *open-source* pour créer des systèmes multi-agents génératifs. Il permet des schémas de conversation plus flexibles que CAMEL, dynamiques intégrant (ou pas) une participation humaine.

10 Discussion

Sur le chemin vers le développement d'une intelligence artificielle plus générale et plus proche de celle des humains, les SMAA génératifs, avec leurs capacités de raisonnement et d'apprentissage continu, ouvrent des perspectives prometteuses pour résoudre des problèmes dans des environnements ouverts et simuler des dynamiques sociales complexes.

Les agents autonomes génératifs sont capables d'interpréter leur perception de l'environnement de manière contextuelle, d'apprendre en continu à partir de nouvelles expériences et d'effectuer des inférences logiques pour prendre des décisions et accomplir des tâches variées, y compris dans des situations non prévues par leur concepteur. Leur aptitude à décomposer des tâches complexes en sous-tâches plus simples et à générer des plans d'action détaillés leur permet de s'adapter à des scénarios divers et dynamiques.

Les SMA génératifs, quant à eux, se distinguent par leur capacité à partager des connaissances, à coordonner leurs actions et à ajuster leurs stratégies en réponse aux retours de l'environnement, ce qui les rend particulièrement efficaces pour la résolution de problèmes dans des environnements ouverts. En outre, la simulation à base d'agents génératifs hétérogènes, capables d'émuler des comportements sociaux humains et d'interagir avec d'autres agents ou humains dans des environnements virtuels, permet de reproduire des dynamiques sociales complexes.

Malgré des perspectives prometteuses, plusieurs défis concernant les SMAA génératifs doivent encore être relevés. Il est crucial d'assurer la reproductibilité des résultats à travers différents modèles et essais. Par ailleurs, les bancs d'essai et les critères d'évaluation actuels restent insuffisants. Les comportements générés peuvent répliquer des stéréotypes issus des données d'entraînement biaisés. De plus, le déploiement à grande échelle de SMAA génératifs est coûteux, en termes de consommation énergétique, de ressources de calcul, voire de crédits liés à l'utilisation des *tokens* pour des LLMs propriétaires. Ce déploiement nécessite la gestion de volumes d'informations pouvant dépasser la capacité de la fenêtre contextuelle de ces modèles. Cette limitation plaide en faveur d'architectures hybrides, disposant d'un état interne explicite et persistant, essentielle à la coordination et à la cohérence des comportements.

Remerciements. Le projet RORES-CL a obtenu le soutien financier du CNRS à travers les programmes interdisciplinaires de la MITI.

12. <https://github.com/microsoft/autogen>

Références

- [1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.
- [2] Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chi-Min Chan, Heyang Yu, Yaxi Lu, Yi-Hsin Hung, Chen Qian, et al. Agentverse : Facilitating multi-agent collaboration and exploring emergent behaviors. In *The Twelfth International Conference on Learning Representations*, 2023.
- [3] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation, 2014.
- [4] Antoine Cornuéjols, Laurent Miclet, and Vincent Barra. *Apprentissage artificiel - 3e édition : Deep learning, concepts et algorithmes*. Eyrolles, 2018.
- [5] DeepSeek-AI. Deepseek-r1 : Incentivizing reasoning capability in llms via reinforcement learning, 2025.
- [6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT : pre-training of deep bidirectional transformers for language understanding. In *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, NAACL-HLT, Volume 1*, pages 4171–4186. Association for Computational Linguistics, 2019.
- [7] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, Miami, Florida, USA, 2024. Association for Computational Linguistics.
- [8] Chen Gao, Xiaochong Lan, Nian Li, Yuan Yuan, Jingtao Ding, Zhilun Zhou, Fengli Xu, and Yong Li. Large language models empowered agent-based modeling and simulation : A survey and perspectives. *Humanities and Social Sciences Communications*, 11(1) :1–24, 2024.
- [9] Chen Gao, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li. S3 : Social-network simulation system with large language model-empowered agents, 2023.

- [10] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models : A survey, 2024.
- [11] Olivier Gouvert, Julie Hunter, Jérôme Louradour, Evan Dufraisse, Yaya Sy, Pierre-Carl Langlais, Anastasia Stasenko, Laura Rivière, Christophe Cerisara, and Jean-Pierre Lorré. The lucie-7b llm and the lucie training dataset : open resources for multilingual language generation, 2025.
- [12] Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. Large language model based multi-agents : A survey of progress and challenges, 2024.
- [13] Önder Gürcan. Llm-augmented agent-based modeling for social simulations : Challenges and opportunities. *HHAI 2024 : Hybrid Human AI Systems for the Social Good*, pages 134–144, 2024.
- [14] Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. MetaGPT : Meta programming for a multi-agent collaborative framework. In *The Twelfth International Conference on Learning Representations*, 2024.
- [15] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora : Low-rank adaptation of large language models, 2021.
- [16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.
- [17] John E Laird, Allen Newell, and Paul S Rosenbloom. Soar : An architecture for general intelligence. *Artificial intelligence*, 33(1) :1–64, 1987.
- [18] Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. Camel : Communicative agents for "mind" exploration of large language model society. In *Advances in Neural Information Processing Systems*, volume 36, pages 51991–52008. Curran Associates, Inc., 2023.
- [19] Ruibo Liu, Ruixin Yang, Chenyan Jia, Ge Zhang, Denny Zhou, Andrew M Dai, Diyi Yang, and Soroush Vosoughi. Training socially aligned language models on simulated social interactions, 2023.
- [20] Qiaozhu Mei, Yutong Xie, Walter Yuan, and Matthew O. Jackson. A turing test of whether ai chatbots are behaviorally similar to humans. *PNAS*, 121(9), 2024.
- [21] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc., 2022.
- [22] Joon Sung Park, Joseph O’Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents : Interactive simulacra of human behavior. In *Proc. of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST)*, New York, NY, USA, 2023. ACM.
- [23] Billy Perrigo. Exclusive : Openai used kenyan workers on less than \$2 per hour to make chatgpt less toxic. *Time Magazine*, 2023. January 18.
- [24] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. Technical report, OpenAI, 2018.
- [25] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088) :533–536, 1986.
- [26] Stuart Russell, Peter Norvig, Fabrice Popineau, Laurent Miclet, and Claire Cadet. *Intelligence artificielle : une approche moderne (4e édition)*. Pearson France, November 2021.
- [27] Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer : Language models can teach themselves to use tools. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 68539–68551. Curran Associates, Inc., 2023.
- [28] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt : Solving ai tasks with chatgpt and its friends in hugging face. In *Advances in Neural Information Processing Systems*, volume 36, pages 38154–38180. Curran Associates, Inc., 2023.
- [29] Theodore R. Sumers, Shunyu Yao, Karthik Narasimhan, and Thomas L. Griffiths. Cognitive architectures for language agents, 2024.
- [30] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks, 2014.
- [31] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford Alpaca : An instruction-following LLaMa model, 2023.
- [32] Meta team. The llama 3 herd of models, 2024.

- [33] Mistral AI team. Mixtral of experts, 2024.
- [34] OpenAI team. Gpt-4 technical report, 2024.
- [35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
- [36] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6) :186345, 2024.
- [37] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners, 2022.
- [38] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc., 2022.
- [39] Matt Welsh. The end of programming. *Communications of the ACM*, 66(1) :34–35, 2023.
- [40] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. Autogen : Enabling next-gen llm applications via multi-agent conversation, 2023.
- [41] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2024.

Detecting Anomalies Using Graph Neural Networks: A Review

Sena Ozgunay^{1,2}, Louise Travé-Massuyès¹, Jean-Michel Loubes^{2,3}, Raul Sena Ferreira⁴

¹ LAAS-CNRS, University of Toulouse, CNRS, Toulouse, France

² Institut de Mathématiques de Toulouse, France

³ INRIA, France

⁴ Continental Automotive France, Toulouse, France

sozgunay@laas.fr, louise@laas.fr, jean-michel.a.loubes@inria.fr, raul.ferreira@continental.com

Résumé

La détection d'anomalies vise à identifier des comportements atypiques au sein des systèmes complexes. Parmi les différentes approches développées dans ce domaine, les réseaux de neurones graphiques (GNN) se distinguent par leur efficacité. Dans cet article, nous proposons une revue des méthodes fondées sur les GNN pour la détection d'anomalies, et introduisons une nouvelle taxonomie, construite autour des mécanismes de prédiction d'anomalies utilisés.

Mots-clés

Détection d'anomalies, réseaux de neurones graphiques, séries temporelles.

Abstract

Anomaly detection is the process of identifying unusual behaviors in systems. In this wide-ranging field, graph neural networks (GNNs) are highly effective compared to the other proposed approaches in the literature. This article summarizes the representative GNN-based methods for anomaly detection and proposes a novel taxonomy based on how these methods predict anomalies.

Keywords

Anomaly detection, graph neural network, time series.

1 Introduction

Real-world systems are generally complex and include vast numbers of interconnected sensors. Anomaly detection is an important task but is difficult in such complex systems since capturing relationships between sensors and detecting sensors deviating from these is challenging [10]. Graph Neural Networks (GNNs) [13] have emerged as a robust AI method for anomaly detection in a wide range of application domains, including spam detection [4], fraud detection [11], cybersecurity [40], industrial system supervision [10], etc. In a financial transaction network, where nodes represent user accounts and edges represent transactions, an anomaly may correspond to a fraudulent account that interacts abnormally with others, such as making high-frequency or high-value transactions to unrelated accounts [38].

GNNs operate on graph-structured data, which can be constructed from various types of raw data such as tabular data, images, or time series, when a graph input is not available, and they aim to learn data representations (or embeddings) by capturing complex relationships from the graph. In many real-world scenarios, time series data constitute an essential setting for anomaly detection since anomalies may occur over time. Time series often involve multiple signals from system sensors that may exhibit spatial dependencies among sensors and temporal dependencies across different time steps [23]. In an industrial sensor network, time series data from different sensors can be modeled as a graph, where nodes represent sensor variables, and edges reflect their spatial or temporal correlations. An anomaly might occur as a sudden deviation in one sensor's behavior that breaks the normal relationship with others over time [10]. In all settings, GNNs offer a unified framework for capturing the underlying spatial or temporal patterns and identifying abnormal behaviors.

The survey [24] reviews GNN-based anomaly detection techniques. [23] analyzes GNN methods for different time series tasks, including anomaly detection. Moreover, [26] proposes a survey of anomaly detection approaches in graph data based on different types of anomalies, graphs, and graph-based methods, including GNNs. Unlike the recent survey [43] that summarizes different types of GNN-based anomaly detection methods based primarily on data distribution assumptions (such as whether normal or anomalous data boundaries are clearer), we propose an embedding-centric taxonomy focused on how the learned embeddings by GNNs are utilized for anomaly detection independently of the original anomaly boundary assumptions and the input raw data nature. Specifically, we classify representative GNN-based methods for anomaly detection based on whether embeddings are directly used for classification, for target value prediction, or for embedding similarity to identify anomalies, offering a complementary perspective to the existing literature. We also discuss the particular case of anomaly detection with GNN when the inputs are time series.

The rest of this paper is organized as follows. Section 2

gives a background on graphs and GNNs. Section 3 proposes the taxonomy based on different GNN methods for anomaly detection. Section 4 discusses the particular case of anomaly detection with GNNs where the inputs are time series, and Section 5 presents the conclusions and future directions.

2 Background

A graph is denoted as $G = (V, E)$, where V is the set of nodes, and E is the set of edges. Graphs can be presented by an adjacency matrix $A \in \mathbb{R}^{|V| \times |V|}$ where $A_{i,j} = 1$ if $(v_i, v_j) \in E$ and $A_{i,j} = 0$ otherwise. For a node $v_i \in V$, the set of neighbors $N(v_i)$ contains the nodes adjacent to v_i . A graph can have attributes or feature information on nodes/edges. Generally, a node feature matrix is denoted as $X \in \mathbb{R}^{|V| \times d}$, where d is the number of attributes [17]. A graph can be directed or undirected; that is, edges of directed graphs are directed from one node to another [7]. The graph can be homogeneous or heterogeneous. If only one type of node exists and one type of edge, the graph is homogeneous; otherwise, it is heterogeneous [29]. In a temporal aspect, a graph can be static or dynamic. In dynamic graphs, input features or the graph structure vary over time, reflecting the changing relationships and attributes within the network, while in static graphs, the graph structure remains fixed [17].

GNNs process graph-structured data to capture complex relationships and extract significant system information. The key principle behind GNNs is aggregating information from a node's local neighborhood to learn node representations or embeddings. That is, a node's state is affected by the states of its neighboring nodes.

For a given graph $G = (V, E, X)$, where $X \in \mathbb{R}^{|V| \times d}$ is the input feature matrix, x_i is the d -dimensional input node feature vector of v_i , GNN learns the node representation h_i for each $v_i \in V$. During a L -layer GNN training, the node embedding at the l -th GNN layer $h_i^{(l)}$ is updated based on aggregated information from v_i 's neighborhood $N(v_i)$. This update can be formalized as follows:

$$m_i^{(l)} = \text{AGG}^{(l)} \left(\left\{ h_j^{(l-1)}, \forall v_j \in N(v_i) \right\} \right), \quad (1)$$

$$h_i^{(l)} = \text{UPDATE}^{(l)} \left(h_i^{(l-1)}, m_i^{(l)} \right) \quad (2)$$

where UPDATE and AGG are differentiable functions with AGG as permutation invariant that can operate on multiple inputs (e.g., element-wise sum, mean, or max). This function takes node embeddings of v_i 's neighbors as input and computes a message $m_i^{(l)}$ by aggregating the information coming from neighbors. Then, the UPDATE function combines the message and previous v_i 's node embedding to obtain the current node embedding.

We note that the initial embeddings at $l = 0$ are input node features, i.e., $h_i^{(0)} = x_i$. After L iterations, the output of GNN is a set of node embeddings $\{h_i^{(L)}, \forall v_i \in V\}$ where each obtained node embedding includes information about its L -hop neighborhood.

Regarding the Equation 1, a simple GNN message-passing mechanism [13] can be constructed as follows:

$$h_i^{(l)} = \sigma \left(W_{self}^{(l)} h_i^{(l-1)} + W_{neigh}^{(l)} \sum_{v_j \in N(v_i)} h_j^{(l-1)} + b^{(l)} \right) \quad (3)$$

where $W_{self}^{(l)}, W_{neigh}^{(l)} \in \mathbb{R}^{d_{(l)} \times d_{(l-1)}}$ are trainable weight matrices with $d_{(l)}, d_{(l-1)}$ embedding dimensions of the corresponding layer and σ is a non-linearity function. $b^{(l)} \in \mathbb{R}^{d_{(l)}}$ is the bias term, which is often neglected. This simple GNN formulation can be decomposed as:

$$m_i^{(l)} = \sum_{v_j \in N(v_i)} h_j^{(l-1)}, \quad (4)$$

$$\text{UPDATE}^{(l)}(h_i^{(l-1)}, m_i^{(l)}) = \sigma(W_{self}^{(l)} h_i^{(l-1)} + W_{neigh}^{(l)} m_i^{(l)}) \quad (5)$$

Among the different GNN formulations, classical GNN layers are Graph Convolutional Networks (GCNs) [25], GraphSAGE (Graph Sample and Aggregate) [16], and Graph Attention Networks (GATs) [36], each with the functions for aggregation and message computation.

GCN and GraphSAGE suppose that all neighbors $v_j \in N(v_i)$ are equally important to the node v_i , that is, the messages coming from neighbors have the same weighting factor (importance), which is based on the node degree. However, some neighbors might be more relevant in some cases than others. Thus, Graph Attention Networks (GATs) introduce attention mechanisms to learn different weights for different neighbors [36]. The node embedding of v_i at the l -th GAT layer is defined as:

$$h_i^{(l)} = \sigma \left(\sum_{v_j \in N(v_i)} \alpha_{ij} W^{(l)} h_j^{(l-1)} \right) \quad (6)$$

where α_{ij} are attention weights across a pair of nodes v_i, v_j . To determine α_{ij} , we first compute attention coefficients e_{ij} , which indicate the importance of v_j to node v_i :

$$e_{ij} = a(W^{(l)} h_i^{(l-1)}, W^{(l)} h_j^{(l-1)}) \quad (7)$$

where a is an attention mechanism (e.g., a single-layer neural network). Then we normalize e_{ij} using the softmax function to obtain the final attention weight α_{ij} :

$$\alpha_{ij} = \text{softmax}_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{v_k \in N(v_i)} \exp(e_{ik})} \quad (8)$$

3 Anomaly Detection with GNNs

In graphs, anomalies may appear at the node-level, edge-level, or graph-level. At the node level, an anomaly can be seen on node attributes as a significantly different characteristic. Regarding neighborhood interaction, an anomaly can be present in structural patterns such as unusual degrees of connections or unexpected labels, etc. An edge can be anomalous at the edge-level if it has an abnormal edge weight compared to the distribution of the

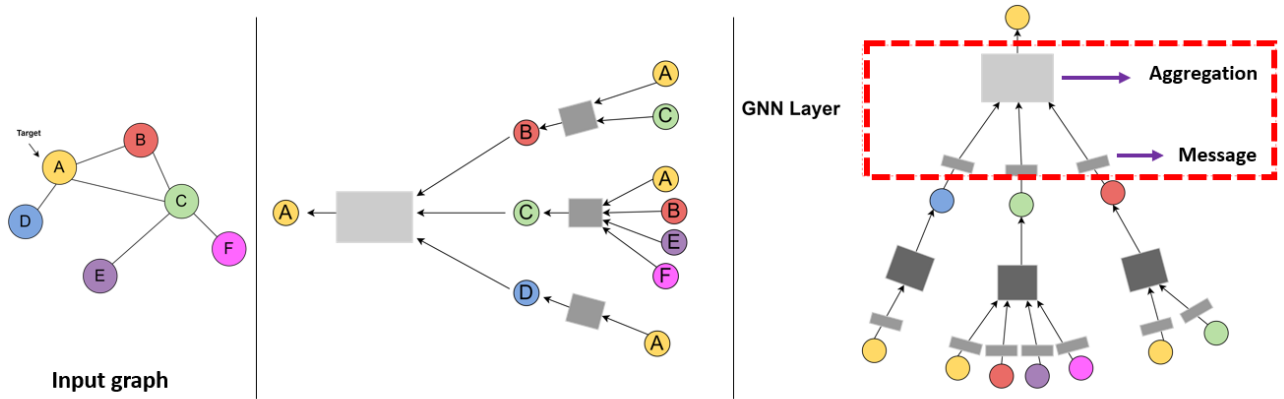


Figure 1: **Overview of GNN's two-layer version of a message-passing mechanism.** It shows the aggregation of messages from A's local neighbors on the graph, which are B, C, and D. Concept adapted from [17].

other edge weights. The discovery of an unexpected link or absence of any edge might also be pointed out as an anomaly. At the graph-level, a graph can be anomalous if it has unusual structural patterns or unexpected changes compared to other input graphs [26].

In anomaly detection, the training dataset may include normal and anomalous instances (fully or partially labeled) or only normal instances. In the former case, GNN can directly learn to identify instances as normal or anomalous based on known labels, while in the latter, GNN learns the expected behavior. The embeddings generated by GNN can be fed into the prediction phase using a classifier that performs a prediction function to decide whether an instance is anomalous. Further, the embeddings may be used to predict a value of a specific instance. Moreover, they can be compared with each other through similarities between them to infer anomaly information. In this survey we introduce a taxonomy that reflects these three ways of leveraging the embeddings learned by GNN for anomaly detection: via direct prediction, comparison with target values, or evaluation of embedding similarities.

3.1 Direct Prediction of Anomalies

The first category focuses on direct prediction of anomalies using a classifier function. Most methods in this category perform a training process under supervision, i.e., supervised (labeled normal and abnormal data) or semi-supervised (only labeled standard data). Depending on the task level, a given entity (node, edge, or graph)'s learned representation by GNN is used as input by a classifier that decides whether the entity is anomalous.

[4], **GEM** [29], and **HACUD** [20] aim to detect spam profiles/malicious accounts in social networks by considering a graph where accounts are nodes and communications between accounts are edges. Learned node embeddings by GNN-based models are passed to a classifier to predict whether an account is malicious.

SemiGNN [38] combines GNN with attention mechanisms to detect fraud in financial transaction networks that contain

different types of relationships. Final user embeddings are obtained by combining the learned relation-specific embeddings. Then the users are classified as fraudulent or not. Likewise, **CARE-GNN** [11] aims to detect camouflaged fraudsters by applying a similar approach on a multi-relation graph. On the other hand, **DeepHGNN** [39] detects anomalies in program executions by modeling the different relationships (events) between different node types (programs, files, etc.). It models the program's behavior by an attention-based GNN model and checks whether the given event corresponds to that program. **GAS** [28] identifies spam reviews using GCNs to model the interactions between reviews, users, and products. GAS learns embeddings for each review, which are then fed to a classifier to decide whether it is a spam comment. **iDetective** [46] introduces an attributed heterogeneous information network to identify key players in hack forums. Learned user embeddings are fed to Support Vector Machine (SVM) to classify the user as a key player (anomalous) or a non-key player (standard). **MVAN** [35] employs GNNs to detect anomalous players with fraudulent money trading activities in online games by leveraging multiple player relationships.

AddGraph [49] models a dynamic graph and identifies abnormal changes in the graph's connections for telecom and social networks. Learned node embeddings are then used to classify edges as anomalous or normal based on the computed edge anomaly score at the current timestamp.

3.2 Target Value Prediction

The GNN-based methods in this category predict future time series values by learning the expected behavior of systems. Due to that, they can be considered one-class methods since the training dataset contains observations from only one class (normal). An anomaly can be predicted via a forecasting-based model that predicts the target value at $t + 1$ based on historical normal data on a temporal window $[t - w, \dots, t]$. When the predicted value differs too much from the observed one, an anomaly is detected [34].

This step usually involves a single fully connected layer or Multilayer perceptron (MLP).

Anomalies can also be identified by a reconstruction-based model, which aims to reconstruct given inputs. The model can approximate well the normal inputs, but it fails to reconstruct abnormal ones [23].

MTAD-GAT [48], **GTAD** [14], and **HAD-MDGAT** [52] apply jointly a reconstruction module and a forecasting module after obtaining node embeddings from a GAT-based model. For each node, an anomaly score is defined based on the predicted value by the forecasting module and the reconstruction probability obtained from the reconstruction module. Similarly, [12] aims to detect leaks in water distribution networks (WDNs), where nodes are sensors and edges are pipes. With partially observed nodal pressures, the model reconstructs and predicts complete nodal pressure values, respectively. Then, edge-wise prediction errors are computed to detect leakages. On the other hand, **GReLeN** [45] uses only a reconstruction module after combining Variational Auto Encoder (VAE) and an attention-based GNN for learning graph structure.

GTA [7], **GDN** [10], **CST-GL** [50], and **STGDNN** [15] use a forecasting approach and aim to learn a graph structure along with the GNN-based model. For learning node embeddings, while GTA applies a transformer-based GNN, GDN and STGDNN use a GAT-based model.

DyGraphAD [21] models a multivariate time series as a set of dynamic correlation graphs. At the forecasting step, it predicts the dynamic correlation graph and the sensor values used to compute an anomaly score for the timestamp. On the other hand, **DGNN** [5] leverages dynamic neighborhood subgraphs based on correlations, and a GAT-based model to forecast sensor values. [27] also employs a forecasting approach, but for finding anomalies in interconnected traffic flows. The model uses GNN with attention mechanism to extract a context of the flows based on their past traffic activity and then forecast the next value of the flow. **GSC-MAD** [47] forecasts sensor values by analyzing changes in learning graph structure. An anomaly score at the timestamp is calculated by combining prediction deviations and differences between the new and stable graph structures. **GST-Pro** [51] deals with missing values by modeling dynamic graph neural controlled differential equations on multi-sensor data to learn embeddings for each sensor, which are then used for forecasting sensor values.

3.3 Embedding Similarity

In the third category, the focus is put on learned embeddings. Anomalies are detected by analyzing how similar or dissimilar embeddings are. This can be done using a clustering approach where nodes with similar embeddings are grouped into clusters, or using similarity metrics where the distance or similarity between node embeddings can indicate abnormality.

MatchGNet [40] applies a graph-matching approach based on the similarities between known benign and unknown programs. For a given unknown program, MatchGNet

computes its graph embedding, and then the cosine similarity between the ones of the existing programs is computed. Based on a threshold, the unknown anomalous programs are identified. From normal data, **OCGNN** [41] learns the node embeddings to find a minimal hypersphere that encloses them. The node is detected as anomalous if a given node's embedding is outside of this hypersphere. **GraphSAD** [6] aims to detect a sequence of abnormal observations in a time series split into subsequences. An abnormal subsequence is detected by comparing the learned embedding with reference subsequence embeddings based on its neighbors.

While our proposed taxonomy offers a general perspective on how GNN-learned embeddings can be used for anomaly detection, the specific nature of the input data can significantly influence how these methods are structured. In particular, time series data often exhibit temporal dependencies that require appropriate graph construction and adaptation of GNN architectures. In the following section, we analyze a subset of the surveyed GNN-based anomaly detection methods that consider time series as input, and discuss how they represent spatial and temporal dependencies within graph-based frameworks.

4 Anomaly Detection with GNNs in Time Series

Time series is a key data type in most systems that record masses of data over time [18, 23]. Formally we denote a time series X as:

$$X = \{X_1, X_2, \dots, X_T\} \in \mathbb{R}^{N \times T} \quad (9)$$

where $X_t = (x_{t,1}, x_{t,2}, \dots, x_{t,N})$ is the vector of values across N variables at time t , and $x_{t,i}$ is the value of the i -th variable at time t [44]. Among the methods mentioned in Section 3, [29, 35, 49, 42] and all the methods presented in Section 3.2 focus on anomaly detection in multivariate time series, while [6] deals with univariate time series.

Anomalies in time series may occur as a single point (e.g., at a given time step), referred to as point anomalies, or as a subsequence that deviates significantly from the expected behavior [44], referred to as collective anomalies [3]. For a long time series, fixed-length samples generated by a sliding window of length T are used as inputs. For point anomalies, the objective of the anomaly detection in multivariate time series is to decide if a point at a given timestamp i is an anomaly or not, i.e., compute an output vector $y \in \mathbb{R}^T$, where $y_i \in \{0, 1\}$ indicates whether the timestamp i is an anomaly [48].

In complex systems, where time series reveal the evolution of measurements in multiple sensors, anomalies may arise not only from the characteristics of variables but also from the relationships between variables [48]. Due to the graph's capacity to exhibit complex data structures, GNNs have emerged as a powerful tool for revealing complex relationships on non-Euclidean data [18]. GNNs can capture the links between variables within a multivariate time series (spatial dependency) and dependencies between

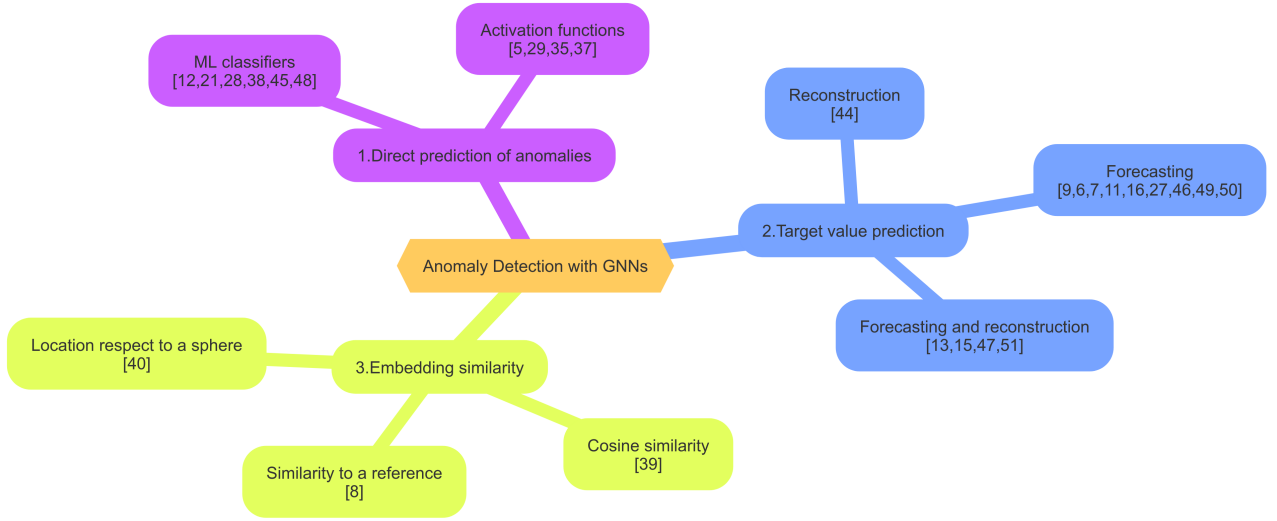


Figure 2: **Taxonomy of the representative GNN-based methods for anomaly detection.** Direct prediction of anomalies can be done by interpreting the output layer of the GNN where an activation function (softmax, ReLU, etc.) is applied, or the learned embeddings can be fed into a separate classifier (MLP, logistic regression, SVM, etc.). The methods in the second category can use a forecasting or reconstruction-based approach. Further, some methods combine these two techniques. The third category contains methods that compare the similarity of the learned embeddings of GNN to detect anomalies. They can compare by using the cosine similarity, the similarity to a reference, or the similarity based on a defined sphere location.

different time points (temporal dependency). A recent survey by [23] offers a comprehensive overview of GNN methods in time series for several major tasks, including anomaly detection. Other recent surveys [18, 44] review the time series anomaly detection techniques using graphs and deep learning methods (including GNNs).

A graph structure can be learned directly from input time series data or constructed by computing several metrics or functions, such as spatial proximity, attribute similarity, etc. Spatial proximity aims to compute proximity between nodes' locations based on, for example, the shortest path between nodes [23]. Meanwhile, attribute similarity determines the similarity between node attributes and then connects similar nodes. This can be done, among other ways, by computing Euclidean distance, cosine similarity, Pearson Correlation Coefficients (PCC), or dynamic time warping (DTW) [31] between time series [18, 23].

Multivariate time series data can be introduced as a *spatial-temporal graph*. It is a set of graphs $\mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_T\}$ with $\mathcal{G}_t = (A_t, X_t)$ a graph at time t where $A_t \in \mathbb{R}^{N \times N}$ is the adjacency matrix and $X_t \in \mathbb{R}^{N \times d}$ is the node-feature matrix with d being the feature dimension [23]. The adjacency matrix A_t may remain fixed (static case) or evolve in time (dynamic case). Thus, spatial-temporal graphs allow GNN to capture in an explicit way both:

- Spatial (inter-variable) dependencies: Relationships between nodes (e.g., sensors, devices).
- Temporal (intra-variable) dependencies: Evolving patterns over time (e.g., periodic behaviors or sequential dependencies) [44].

Several techniques exist in the spatial-temporal graph structures to integrate temporal dependency into GNNs. One approach is to include time-specific features for each node, for example, the average or variance of past values [22]. Another approach uses a sliding window, where the data can be divided into overlapping fixed-length windows, creating a separate graph for each window [48]. This allows the GNN to learn spatial relationships in each window while implicitly considering the temporal context. Another approach is to insert some hybrid architectures that combine GNNs with temporal modeling methods such as RNN [1], LSTM [19], or GRU [8], allowing them to model time dependencies explicitly. Moreover, Temporal GAT (TGAT) [9], which employs a self-attention mechanism with a functional time encoding, or Temporal Convolutional Networks (TCNs) [2], which use a 1D fully-convolutional network (FCN) [30] architecture and causal convolutions, might be applied to extract temporal dependencies. In such hybrid architectures, the spatial and temporal layers can be combined directly (e.g., using GCNs for spatial learning and GRUs for temporal patterns) or by stacking the temporal layer on the GNN spatial layer (e.g., processing a GRU after a GNN) [23, 44].

[12, 27], GEM [29], GDN [10], and GSC-MAD [47] can extract spatial dependencies between nodes, but they ignore temporal dependencies between different timestamps. Meanwhile, AddGraph [49] extends GCN to a temporal GCN with an attentional GRU to capture the nodes' long- and short-term temporal patterns. Similarly, GReLeN [45] applies an architecture combined with GRU and GNN. However, MTAD-GAT [48] employs a GAT

model as a spatial-temporal encoder with spatial and temporal GAT layers. The spatial GAT learns spatial dependencies between sensors, while the temporal GAT extracts temporal relationships from a given input in a sliding window; it considers a fully connected graph where timestamps in the sliding window are nodes. Concatenated learned embeddings across layers are fed into a GRU as a forecasting module to capture long-term temporal dependencies as in HAD-MDGAT [52]. However, in HAD-MDGAT, GRU output is used in forecasting and reconstruction modules for model inference. Besides, Event2Graph [42] extends the TGAT architecture to model the temporal dependency on the node level. At the same time, GTA [7] employs a transformer-based architecture within a hierarchical causal convolution to extract temporal dependencies. GTAD [14] uses a TCN to extract temporal information and a variant of GAT for spatial dependencies. The obtained embeddings are fed into the forecasting and reconstruction models, which both contain a TCN module to avoid losing temporal information.

DyGraphAD [21] applies DTW for constructing dynamic correlation graphs for each input subsequence from a multivariate time series that computes similarities between pairwise time series. Then, embeddings of these graphs and representations of subsequences are learned via a temporal-based GNN architecture. CST-GL [50] combines GCN and TCN for spatial and temporal convolutions, respectively. Similarly, DGNN [5] uses GRU, a forecasting module to predict values at the next time step and extract temporal dependency. Further, STGDNN [15] extends the GDN [10] architecture as a spatial-temporal GNN framework by inserting spatial-temporal relationships in the graph structure. GST-Pro [51] exploits spatial and temporal dependencies by modeling Dynamic Graph Neural Controlled Differential Equations.

5 Conclusion

In this paper, we reviewed the representative GNN-based methods for anomaly detection and proposed a taxonomy based on how these use learned embeddings for predicting anomalies. We discussed the specific case of the GNNs for anomaly detection, where the inputs are time series.

This review highlights that GNNs are well-suited solutions for the anomaly detection task due to their capacity for modeling complex systems. The message-passing technique allows GNNs to generate embeddings that contain rich neighborhood information. These learned embeddings are then used in many ways to identify anomalies. Our taxonomy helps researchers better understand anomaly detection using GNNs by clarifying the differences in the ways anomalies are identified using the generated embeddings from GNNs.

In the anomaly detection, it's crucial to understand why a particular event is flagged as anomalous and how to validate the relevance of these detected anomalies [33]. Therefore, as future work, we intend to study the anomaly detection in industrial production lines using GNNs, and explain the

detected anomalies using counterfactual explanations by focusing on alternative scenarios [37]. That is, we aim to generate counterfactual explanations that would indicate what changes to an anomalous instance would make it be considered normal [33]. Another research direction is to explore how to determine which system components are malfunctioning through model-based diagnosis [32].

Acknowledgments

This work is supported by the AI Interdisciplinary Institute ANITI funded by the France 2030 program under the Grant agreements n°ANR-19-PI3A-0004 and n°ANR-23-IACL-0002.

References

- [1] 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), 2018.
- [2] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *CoRR*, abs/1803.01271, 2018.
- [3] Mohammad Braei and Sebastian Wagner. Anomaly detection in univariate time-series: A survey on the state-of-the-art. *CoRR*, abs/2004.00433, 2020.
- [4] Anshika Chaudhary, Himangi Mittal, and Anuja Arora. Anomaly detection using graph neural networks. In *2019 international conference on machine learning, big data, cloud and parallel computing (COMITCon)*, pages 346–350. IEEE, 2019.
- [5] Bowen Chen, Hancheng Lu, Yuang Chen, Haoyue Yuan, and Minghui Wang. Dgmn: Dynamic graph neural networks for anomaly detection in multivariate time series. In *SEKE*, pages 415–420, 2023.
- [6] Weiqi Chen, Zhiqiang Zhou, Qingsong Wen, and Liang Sun. Time series subsequence anomaly detection via graph neural networks, 2023.
- [7] Zekai Chen, Dingshuo Chen, Xiao Zhang, Zixuan Yuan, and Xiuzhen Cheng. Learning graph structures with transformer for multivariate time-series anomaly detection in iot. *IEEE Internet of Things Journal*, 9(12):9179–9189, 2021.
- [8] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *SSST@EMNLP*, 2014.
- [9] da Xu, chuanwei ruan, evren korpeoglu, sushant kumar, and kannan achan. Inductive representation learning on temporal graphs. In *International Conference on Learning Representations*, 2020.

- [10] Ailin Deng and Bryan Hooi. Graph neural network-based anomaly detection in multivariate time series. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 4027–4035, 2021.
- [11] Yingdong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, and Philip S Yu. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 315–324, 2020.
- [12] Garðar Örn Garðarsson, Francesca Boem, and Laura Toni. Graph-based learning for leak detection and localisation in water distribution networks. *IFAC-PapersOnLine*, 55(6):661–666, 2022.
- [13] Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE international joint conference on neural networks, 2005.*, volume 2, pages 729–734. IEEE, 2005.
- [14] Siwei Guan, Binjie Zhao, Zhekang Dong, Mingyu Gao, and Zhiwei He. Gtad: Graph and temporal neural network for multivariate time series anomaly detection. *Entropy*, 24(6):759, 2022.
- [15] Ziyu Guo, Weiyang Kong, and Yubao Liu. Spatial-temporal graph deviation neural networks for anomaly detection in time series. In *Journal of Physics: Conference Series*, volume 2735, page 012016. IOP Publishing, 2024.
- [16] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.
- [17] William L Hamilton. *Graph representation learning*. Morgan & Claypool Publishers, 2020.
- [18] Thi Kieu Khanh Ho, Ali Karami, and Narges Armanfard. Graph-based time-series anomaly detection: A survey. *arXiv preprint arXiv:2302.00058*, 2023.
- [19] S Hochreiter. Long short-term memory. *Neural Computation MIT-Press*, 1997.
- [20] Binbin Hu, Zhiqiang Zhang, Chuan Shi, Jun Zhou, Xiaolong Li, and Yuan Qi. Cash-out user detection based on attributed heterogeneous information network with a hierarchical attention mechanism. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 946–953, 2019.
- [21] Xiangheng Huang, Ningjiang Chen, Ziyue Deng, and Suqun Huang. Multivariate time series anomaly detection via dynamic graph attention network and informer. *Applied Intelligence*, 54(17–18):7636–7658, June 2024.
- [22] Guangyin Jin, Yuxuan Liang, Yuchen Fang, Zezhi Shao, Jincui Huang, Junbo Zhang, and Yu Zheng. Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 2023.
- [23] Ming Jin, Huan Yee Koh, Qingsong Wen, Daniele Zambon, Cesare Alippi, Geoffrey I Webb, Irwin King, and Shirui Pan. A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [24] Hwan Kim, Byung Suk Lee, Won-Yong Shin, and Sungsu Lim. Graph anomaly detection with graph neural networks: Current status and challenges. *IEEE Access*, 10:111820–111829, 2022.
- [25] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.
- [26] Prabin B Lamichhane and William Eberle. Anomaly detection in graph structured data: A survey. *arXiv preprint arXiv:2405.06172*, 2024.
- [27] Hamid Latif, José Suárez-Varela, Albert Cabellos-Aparicio, and Pere Barlet-Ros. Detecting contextual network anomalies with graph neural networks. In *Proceedings of the 2nd on Graph Neural Networking Workshop 2023*, pages 25–30, 2023.
- [28] Ao Li, Zhou Qin, Runshi Liu, Yiqun Yang, and Dong Li. Spam review detection with graph convolutional networks. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 2703–2711, 2019.
- [29] Ziqi Liu, Chaochao Chen, Xinxing Yang, Jun Zhou, Xiaolong Li, and Le Song. Heterogeneous graph neural networks for malicious account detection. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 2077–2085, 2018.
- [30] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [31] Chotirat Ann Ratanamahatana and Eamonn Keogh. Everything you know about dynamic time warping is wrong. In *Third workshop on mining temporal and sequential data*, volume 32. Citeseer, 2004.
- [32] Raymond Reiter. A theory of diagnosis from first principles. *Artificial intelligence*, 32(1):57–95, 1987.

- [33] Max Schemmer, Joshua Holstein, Niklas Bauer, Niklas Kühn, and Gerhard Satzger. Towards meaningful anomaly detection: The effect of counterfactual explanations on the investigation of anomalies in multivariate time series. *CoRR*, abs/2302.03302, 2023.
- [34] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. Anomaly detection in time series: a comprehensive evaluation. *Proceedings of the VLDB Endowment*, 15(9):1779–1797, 2022.
- [35] Jianrong Tao, Jianshi Lin, Shize Zhang, Sha Zhao, Runze Wu, Changjie Fan, and Peng Cui. Mvan: Multi-view attention networks for real money trading detection in online games. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2536–2546, 2019.
- [36] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.
- [37] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.
- [38] Daixin Wang, Jianbin Lin, Peng Cui, Quanhui Jia, Zhen Wang, Yanming Fang, Quan Yu, Jun Zhou, Shuang Yang, and Yuan Qi. A semi-supervised graph attentive network for financial fraud detection. In *2019 IEEE international conference on data mining (ICDM)*, pages 598–607. IEEE, 2019.
- [39] Shen Wang, Zhengzhang Chen, Ding Li, Zhichun Li, Lu-An Tang, Jingchao Ni, Junghwan Rhee, Haifeng Chen, and Philip S Yu. Attentional heterogeneous graph neural network: Application to program reidentification. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 693–701. SIAM, 2019.
- [40] Shen Wang and S Yu Philip. Heterogeneous graph matching networks: Application to unknown malware detection. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 5401–5408. IEEE, 2019.
- [41] Xuhong Wang, Baihong Jin, Ying Du, Ping Cui, Yingshui Tan, and Yupu Yang. One-class graph neural networks for anomaly detection in attributed networks. *Neural computing and applications*, 33:12073–12085, 2021.
- [42] Yuhang Wu, Mengting Gu, Lan Wang, Yusan Lin, Fei Wang, and Hao Yang. Event2graph: Event-driven bipartite graph for multivariate time series forecasting and anomaly detection. In *AMLS*, 2022.
- [43] Ziqi Yuan, Qingyun Sun, Haoyi Zhou, Minglai Shao, and Xingcheng Fu. A comprehensive survey on gnn-based anomaly detection: taxonomy, methods, and the role of large language models. *International Journal of Machine Learning and Cybernetics*, pages 1–26, 2025.
- [44] Zahra Zamanzadeh Darban, Geoffrey I Webb, Shirui Pan, Charu Aggarwal, and Mahsa Salehi. Deep learning for time series anomaly detection: A survey. *ACM Computing Surveys*, 57(1):1–42, 2024.
- [45] Weiqi Zhang, Chen Zhang, and Fugee Tsung. Grelen: Multivariate time series anomaly detection from the perspective of graph relational learning. In *IJCAI*, pages 2390–2397, 2022.
- [46] Yiming Zhang, Yujie Fan, Yanfang Ye, Liang Zhao, and Chuan Shi. Key player identification in underground forums over attributed heterogeneous information network embedding framework. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 549–558, 2019.
- [47] Zhen Zhang, Zhiqiang Geng, and Yongming Han. Graph structure change-based anomaly detection in multivariate time series of industrial processes. *IEEE Transactions on Industrial Informatics*, 2024.
- [48] Hang Zhao, Yujing Wang, Juanyong Duan, Congrui Huang, Defu Cao, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, and Qi Zhang. Multivariate time-series anomaly detection via graph attention network. In *2020 IEEE international conference on data mining (ICDM)*, pages 841–850. IEEE, 2020.
- [49] Li Zheng, Zhenpeng Li, Jian Li, Zhao Li, and Jun Gao. Addgraph: Anomaly detection in dynamic graph using attention-based temporal gcn. In *IJCAI*, volume 3, page 7, 2019.
- [50] Yu Zheng, Huan Yee Koh, Ming Jin, Lianhua Chi, Khoa T Phan, Shirui Pan, Yi-Ping Phoebe Chen, and Wei Xiang. Correlation-aware spatial-temporal graph learning for multivariate time-series anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.
- [51] Yu Zheng, Huan Yee Koh, Ming Jin, Lianhua Chi, Haishuai Wang, Khoa T Phan, Yi-Ping Phoebe Chen, Shirui Pan, and Wei Xiang. Graph spatiotemporal process for multivariate time series anomaly detection with missing values. *Information Fusion*, 106:102255, 2024.
- [52] Liwen Zhou, Qingkui Zeng, and Bo Li. Hybrid anomaly detection via multihead dynamic graph attention networks for multivariate time series. *IEEE Access*, 10:40967–40978, 2022.

Une approche organisationnelle pour améliorer l'explicabilité et le contrôle dans l'apprentissage par renforcement multi-agent*

Julien Soulé^{1,2}, Jean-Paul Jamont¹, Michel Occello¹, Louis-Marie Traonouez², Paul Théron³

¹ Univ. Grenoble Alpes, Grenoble INP, LCIS, 26000, Valence, France

² Thales Land and Air Systems, BU IAS, Rennes, France

³ AICA IWG, La Guillerme, France

julien.soule@lcis.grenoble-inp.fr, jean-paul.jamont@lcis.grenoble-inp.fr,
michel.occello@lcis.grenoble-inp.fr,
louis-marie.traonouez@thalesgroup.com, paul.theron@orange.fr

Résumé

Des agents entraînés peuvent exhiber des comportements collectifs, d'où l'on peut extrapoler des rôles et objectifs implicites par analogie d'une organisation structurée et fonctionnelle. Nous proposons MOISE+MARL, un cadre qui contraint les agents à évoluer selon une telle organisation pour améliorer le contrôle et l'explicabilité en apprentissage par renforcement multi-agent. MOISE+MARL guide les agents vers des rôles et missions en ajustant dynamiquement leurs actions et récompenses. Il inclut une analyse post-entraînement pour inférer des spécifications organisationnelles implicites. Expérimenté sur divers environnements, il démontre un alignement entre les comportements des agents, les spécifications définies et celles inférées.

Mots-clés

Apprentissage par renforcement multi-agent, Explicabilité, Contrôle, Organisation

Abstract

Trained agents can exhibit collective behaviors, from which roles and implicit objectives can be extrapolated by analogy with a structured and functional organization. We propose MOISE+MARL, a framework that constrains agents to evolve within such an organization to enhance control and explainability Multi-Agent Reinforcement Learning. MOISE+MARL guides agents toward roles and missions by dynamically adjusting their actions and rewards. It also includes a post-training analysis to infer implicit organizational specifications. Tested across various environments, it demonstrates alignment between agent behaviors, predefined specifications, and inferred ones.

Keywords

Multi-Agent Reinforcement Learning, Explainability, Control, Organization

1 Introduction

L'apprentissage par renforcement multi-agent [19] (*Multi-Agent Reinforcement Learning - MARL*) permet de trouver une politique conjointe qui régit les actions individuelles des agents et leurs interactions pour atteindre un objectif sans gérer explicitement leur coordination. Dans des environnements nécessitant des interactions sociales, les agents peuvent aboutir à des comportements semblables à des rôles et objectifs implicites, qui les rapprochent en partie d'une **organisation** (structurelle et fonctionnelle) telle que décrite dans *MOISE+* [14].

Toutefois, l'identification de ces rôles et objectifs émergents reste complexe en raison de comportements souvent bruités ou irréguliers. Afin d'interpréter des comportements comme des rôles et objectifs d'une organisation implicite, nous introduisons l'**adéquation organisationnelle**. Ce concept propose une vision organisationnelle pour envisager deux problèmes encore peu explorés que sont le contrôle et l'explicabilité dans le MARL au travers de : i) **L'évaluation de l'adéquation organisationnelle** consiste à mesurer l'alignement d'une politique conjointe avec une organisation explicite où les comportements sont réguliers. La littérature, souvent centrée sur les rôles [1, 33, 35], manque d'approches systématiques dotées de moyens quantitatifs ; ii) **Le contrôle de l'adéquation organisationnelle** consiste à orienter les agents vers des politiques conformes à une organisation via des contraintes ou incitations définies. Ce contrôle induit la réduction de l'espace de recherche, l'amélioration de la convergence et le respect des contraintes de sécurité, sans recourir au *Hierarchical Reinforcement Learning* (HRL).

Cet article présente le cadre **MOISE+MARL** que nous avons introduit dans [30]. MOISE+MARL apporte une contribution synthétique en simplifiant la complexité de nos premiers travaux sur l'intégration d'un modèle organisationnel en MARL [29, 28] en se recentrant sur les notions de rôles et objectifs, permettant une meilleure utilisabilité et scalabilité. Il combine un formalisme Markovien et le modèle organisationnel *MOISE+* [14] pour permettre de

*Cet article est traduit d'un article long accepté à AAMAS [30].

spécifier la logique des rôles et objectifs. Une fois configuré, ce cadre permet d'attribuer des rôles et objectifs aux agents en ajustant dynamiquement les actions et la récompense. Dans MOISE+MARL, nous proposons également la méthode *Trajectory-based Evaluation in MOISE+MARL* (TEMM) pour inférer des rôles et objectifs implicites à partir de trajectoires collectées via des techniques d'apprentissage non supervisé, évaluant ainsi quantitativement l'adéquation organisationnelle. Contrairement au HRL, qui décompose les tâches en interne [23, 18, 26], MOISE+MARL guide les agents vers des rôles et objectifs de façon externe. Nous avons évalué MOISE+MARL avec : i) Quatre environnements différents, chacun entraînant des politiques nécessitant des organisations implicites variées, afin d'évaluer la généralisabilité du cadre ; ii) Quatre algorithmes MARL de familles distinctes, pour mesurer leur adéquation avec MOISE+MARL lors de l'entraînement et de l'analyse post-entraînement ; iii) Un ensemble de spécifications organisationnelles par environnement, permettant une évaluation manuelle et quantitative de leurs impacts. Une observation manuelle montre qu'un agent adoptant un rôle et engagé dans une mission, s'aligne effectivement sur le comportement attendu, confirmant la mesure de l'adéquation organisationnelle obtenue via TEMM. Les rôles et missions inférés s'alignent sur les spécifications prédéfinies, démontrant la cohérence interne du cadre. Par ailleurs, les algorithmes basés sur la politique et les acteur-critique produisent des politiques stables, tandis que ceux basés sur la valeur présentent une variabilité plus importante.

La section 2 présente les travaux sur l'adéquation organisationnelle, la section 3 introduit MOISE+MARL, la section 4 décrit TEMM, la section 5 expose le protocole expérimental et la section 6 les résultats, puis la section 7 conclut.

2 Travaux connexes

Cette section explore des travaux qui lient des aspects de l'organisation dans le MARL.

2.1 Évaluation de l'adéquation organisationnelle

Certains travaux se sont intéressés à l'inférence de rôles ou d'objectifs pour mesurer l'adéquation organisationnelle ou des concepts similaires. Wilson et al. [5] proposent un transfert de rôles dans les environnements multi-agent pour faciliter l'adaptation entre environnements, mais leur modèle se limite à des rôles spécifiques liés aux tâches. Berenji et Vengerov [7] étudient la coordination et l'inférence de rôles dans des missions *Unmanned Aerial Vehicles* pour renforcer la coopération, sans permettre d'inférer les rôles implicites requis. Yusuf et Baber [34] utilisent des méthodes bayésiennes pour coordonner des agents diversifiés, mais leur approche manque d'abstraction des rôles et n'évalue pas l'alignement avec une structure organisationnelle globale. Serrino et al. [27] explorent l'inférence dynamique des rôles dans des environnements sociaux, se concentrant sur des rôles opérationnels immédiats plutôt que sur des rôles implicites.

Les travaux identifiés intègrent des mécanismes internes impliquant une inférence « faite-main » implicite des rôles pour servir une coordination globale. Néanmoins, aucun ne se focalise sur l'inférence générale des rôles et objectifs ni sur la mesure de l'adéquation organisationnelle.

2.2 Contrôle de l'adéquation organisationnelle

Le contrôle de l'adéquation organisationnelle consiste à aligner les politiques des agents sur une organisation prédéfinie via des contraintes ou incitations. Par exemple, Achiam et al. [16] présentent le *Constrained Policy Optimization* (CPO), qui ajuste les politiques avec des contraintes de sécurité tandis que nous souhaitons aller plus loin en appliquant des contraintes externes qui modifient dynamiquement l'espace d'action pour orienter les agents vers des comportements organisationnels.

Ray et al. [4] intègrent des contraintes dans la récompense via des multiplicateurs de Lagrange mais ne les exploite pas pour structurer les comportements de plusieurs agents. De même, alors que Garcia et al. [12] et Alshiekh et al. [21] se concentrent sur l'exploration sécurisée (*shielding*), nous cherchons à guider les agents vers des comportements alignés avec des rôles.

Plutôt que de décomposer les tâches en sous-tâches comme le HRL [13], nous voulons contraindre le MARL de manière externe, assurant une granularité modulaire et des comportements affinés. Enfin, la coordination décentralisée par partage des connaissances [15] illustre l'importance d'une communication maîtrisée pour garantir l'adéquation organisationnelle dans les systèmes complexes.

Poursuivant les approches comme le *shielding* ou le CPO, nous souhaitons intégrer des contraintes organisationnelles externes en MARL standard, modifiant actions et récompenses pour s'aligner sur des rôles et objectifs.

3 Le cadre MOISE+MARL

Cette section présente le formalisme utilisé pour décrire le cadre MOISE+MARL.

3.1 Cadre de Markov pour le MARL

Pour appliquer les techniques de MARL, nous nous appuyons sur le *Decentralized Partially Observable Markov Decision Process* (Dec-POMDP) [22]. Les Dec-POMDP modélisent naturellement la coordination décentralisée multi-agent en situation d'observabilité partielle, ce qui les rend particulièrement adaptés à l'intégration de contraintes organisationnelles. Contrairement aux *Partially Observable Stochastic Games* (POSG), le Dec-POMDP comprend une fonction de récompense commune, favorisant ainsi la collaboration [8].

Un Dec-POMDP $d \in D$ (où D est l'ensemble des Dec-POMDP) est défini comme un 7-uplet $d = \langle S, \{A_i\}, T, R, \{\Omega_i\}, O, \gamma \rangle$, où $S = \{s_1, \dots, s_{|S|}\}$ est l'ensemble des états possibles ; $A_i = \{a_1^i, \dots, a_{|A_i|}^i\}$ est l'ensemble des actions possibles pour l'agent i ; T représente l'ensemble des probabilités de transition, avec $T(s, a, s') = \mathbb{P}(s'|s, a)$ qui correspond à la probabilité

de passer de l'état s à l'état s' suite à l'action a ; $R : S \times A \times S \rightarrow \mathbb{R}$ est la fonction de récompense, attribuant une récompense en fonction de l'état initial, de l'action effectuée et de l'état résultant; $\Omega_i = \{o_1^i, \dots, o_{|\Omega_i|}^i\}$ est l'ensemble des observations possibles pour l'agent i ; O représente l'ensemble des probabilités d'observation, où $O(s', a, o) = \mathbb{P}(o|s', a)$ est la probabilité d'obtenir l'observation o après avoir effectué l'action a et atteint l'état s' ; et enfin, $\gamma \in [0, 1]$ est le facteur d'actualisation.

Le formalisme suivant est utilisé avec MOISE+MARL pour résoudre le Dec-POMDP [8, 3] : i) \mathcal{A} représente l'ensemble des n agents; ii) Π désigne l'ensemble des **politiques**, où une politique $\pi \in \Pi$, $\pi : \Omega \rightarrow A$, associe de manière déterministe une observation à une action, représentant ainsi la stratégie interne de l'agent; iii) Π_{joint} représente l'ensemble des **politiques conjointes**, avec une politique conjointe $\pi_{joint} \in \Pi_{joint}$, $\pi_{joint} : \Omega^n \rightarrow A^n = \Pi^n$, qui sélectionne une action pour chaque agent en fonction de leurs observations respectives, constituant ainsi une collection de politiques utilisée par les agents d'une même équipe; iv) H est l'ensemble des **historiques**, où un historique (ou trajectoire) sur $z \in \mathbb{N}$ étapes (typiquement le nombre maximal d'étapes dans un épisode) est représenté par le z -uplet $h = \langle \langle \omega_k, a_k \rangle \mid k \leq z, \omega \in \Omega, a \in A \rangle$, capturant les observations et actions successives; v) H_{joint} désigne l'ensemble des **historiques conjointes**, avec un historique conjoint $h_{joint} \in H_{joint}$ sur z étapes défini comme l'ensemble des historiques individuels : $h_{joint} = \{h_1, h_2, \dots, h_n\}$; vi) $V_{joint}(\pi_{joint}) : \Pi_{joint} \rightarrow \mathbb{R}$ désigne la **récompense cumulative attendue** sur un horizon fini (en supposant $\gamma < 1$ ou si le nombre d'étapes dans un épisode est fini), où π_{joint} représente la politique conjointe pour l'équipe i , les politiques conjointes des autres équipes, $\pi_{joint, -i}$, étant considérées comme fixes.

Nous définissons la **résolution du Dec-POMDP** comme la recherche d'une politique conjointe $\pi_{joint} \in \Pi_{joint}$ qui atteint au moins une récompense cumulative attendue de s , où $s \in \mathbb{R}$.

3.2 Le modèle organisationnel MOISE⁺

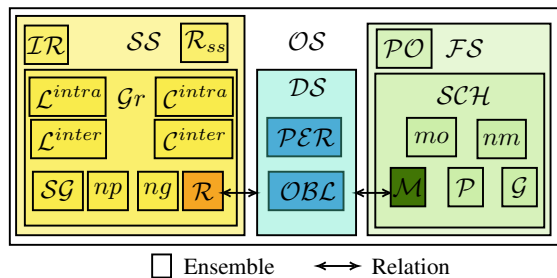


FIGURE 1 – Une vue synthétique de MOISE⁺

Spécifications structurelles (SS) : Elles définissent la structure des agents, notées $SS = \langle \mathcal{R}, \mathcal{IR}, \mathcal{G} \rangle$, où \mathcal{R} est l'ensemble des rôles, avec une relation d'héritage \mathcal{IR} (c'est-à-dire, $\rho_1 \sqsubset \rho_2$ si ρ_1 hérite de ρ_2). De plus, \mathcal{G} spécifie des groupes sous la forme

$\langle \mathcal{R}, \mathcal{SG}, \mathcal{L}^{intra}, \mathcal{L}^{inter}, \mathcal{C}^{intra}, \mathcal{C}^{inter}, np, ng \rangle$, où \mathcal{L} désigne les liens (connaissance, communication, autorité) et \mathcal{C} les compatibilités entre rôles, avec np et ng indiquant respectivement le nombre de rôles et de sous-groupes.

Spécifications fonctionnelles (FS) : Elles décrivent les objectifs des agents et sont notées $FS = \langle \mathcal{SCH}, \mathcal{PO} \rangle$. Le schéma social \mathcal{SCH} comprend les objectifs globaux \mathcal{G} , les missions \mathcal{M} et les plans \mathcal{P} organisant ces objectifs (via l'opérateur op pour séquence, choix ou parallèle). Les missions regroupent des ensembles d'objectifs (mo) et le nombre d'agents par mission est donné par nm , tandis que \mathcal{PO} représente les préférences (ex. $m_1 \prec m_2$).

Spécifications déontiques (DS) : Elles précisent la relation entre rôles et objectifs, notées $DS = \langle \mathcal{OBL}, \mathcal{PER} \rangle$. Les contraintes temporelles \mathcal{TC} fixent les périodes pour les permissions/obligations (ex. Any pour tout moment). Les obligations \mathcal{OBL} imposent aux agents en rôle ρ_a d'exécuter la mission m aux moments tc , tandis que les permissions \mathcal{PER} les autorisent. La fonction rd associe à chaque rôle une spécification sous la forme $\langle tc, y, m \rangle$, avec $y = 0$ pour permission et $y = 1$ pour obligation.

Les autres spécifications structurelles (compatibilités, liens) sont inhérentes aux rôles. De même, les objectifs (incluant missions et mo) sont inhérents aux autres spécifications fonctionnelles (plans, cardinalités, ordres de préférence). Considérer les rôles, les missions et les permissions/obligations est suffisant pour lier MOISE⁺ au Dec-POMDP.

3.3 Liaison de MOISE⁺ avec le MARL

Alors qu'AGR [10] (Agent Group Role) est un cadre informel qui introduit des rôles par groupes, MOISE⁺ offre une description plus détaillée et flexible des structures et fonctions d'un système multi-agent (MAS), facilitant ainsi la formalisation des politiques en MARL.

Guides de Contraintes : Trois relations décrivent la logique des rôles et objectifs dans le cadre Dec-POMDP : i) **Guide d'action de rôle** $rag : H \times \Omega \rightarrow \mathcal{P}(A \times \mathbb{R})$: Pour chaque couple (h, ω) ($h \in H, \omega \in \Omega$), il associe un ensemble d'actions $A_\omega \subseteq A$ avec une rigidité $ch \in [0, 1]$ (par défaut $ch = 1$), restreignant ainsi le choix de la prochaine action; ii) **Guide de récompense de rôle** $rrg : H \times \Omega \times A \rightarrow \mathbb{R}$: Défini par $rrg(h, \omega, a) = r_m$ si $a \notin A_\omega$ (avec $rag(h, \omega) = A_\omega \times \mathbb{R}$), et 0 sinon, afin d'encourager le respect du rôle; iii) **Guide de récompense d'objectif** $grg : H \rightarrow \mathbb{R}$: Attribue un bonus r_b à la récompense globale si h contient une sous-séquence caractéristique $h_g \in H_g$ correspondant à un objectif.

Lieux : Relient les spécifications de MOISE⁺ aux Guides de Contraintes et aux agents : i) **Agent vers Rôle** $ar : \mathcal{A} \rightarrow \mathcal{R}$ (relation bijective); ii) **Rôle vers Guide de Contrainte** $rcg : \mathcal{R} \rightarrow rag \cup rrg$: Associe à chaque rôle une relation rag ou rrg ; iii) **Objectif vers Guide de Contrainte** $gcg : \mathcal{G} \rightarrow grg$: Relie chaque objectif à son guide grg .

La résolution du problème MOISE+MARL consiste à

Définition 1 Fonction de valeur d'état adaptée aux **Guides de Contrainte** en mode AEC :

$$V^{\pi^j}(s_t) = \sum_{\substack{a_t \in A \text{ si } rn() < ch_t, \\ a_t \in A_t \text{ sinon}}} \pi_i(a_t | \omega_t) \sum_{s_{t+1} \in S} T(s_{t+1} | s_t, a_t) \left[R(s_t, a_t, s_{t+1}) + \sum_{m \in \mathcal{M}_i} v_m(t) \frac{grgm(h_{t+1})}{1-p+\epsilon} + (1-ch_t) \times rrg(\omega_t, a_{t+1}) + V^{\pi_{i+1 \bmod n}^j}(s_{t+1}) \right]$$

Avec $rag(h_t, \omega_t) = A_t \times \mathbb{R}$, $\langle a_t, ch_t \rangle \in A_t \times \mathbb{R}$; et $rn : \emptyset \rightarrow [0, 1]$, une fonction aléatoire uniforme

Avec $\omega_t = O(\omega_t | s_t, a_t)$; $h_t = \{h_0 = \langle \rangle, h_{t+1} = \langle h_t, \langle \omega_{t+1}, a_{t+1} \rangle \rangle\}$; $grgm(h) = \sum w_i \times grgi(h)$; $\epsilon \in \mathbb{R}_{>0}$;

$v_m(t) = \{1 \text{ si } t \in t_c; \text{ sinon } 0\}$; et $\mathcal{M}_i = \{m_j \mid \langle ar(i), m_j, t_c, p \rangle \in \mathcal{M}\}$ ($grgi, w_i \in mo(m)$)

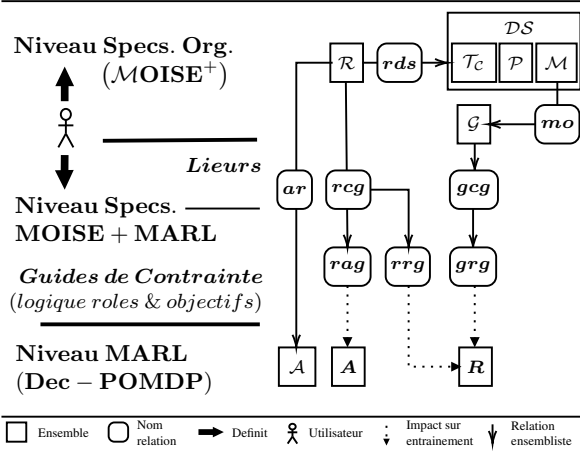


FIGURE 2 – Une vue minimale du cadre MOISE+MARL : Les utilisateurs définissent d'abord les spécifications de $MOISE^+$, qui incluent des rôles (\mathcal{R}) et des missions (\mathcal{M}) associées via rds . Ensuite, ils créent des spécifications MOISE+MARL en définissant d'abord des **Guides de Contraintes** tels que rag et rrg pour spécifier la logique des rôles, et grg pour la logique des objectifs. Ensuite, des **Lieurs** sont utilisés pour connecter les agents aux rôles via ar et pour relier la logique des **Guides de Contrainte** aux spécifications définies de $MOISE^+$. Une fois cette configuration réalisée, des rôles peuvent être attribués aux agents, et le cadre MARL s'adapte en conséquence pendant l'entraînement.

trouver une politique conjointe $\pi^j = \{\pi_0^j, \pi_1^j, \dots, \pi_n^j\}$ maximisant la fonction de valeur d'état V^{π^j} (ou atteignant un seuil minimal), qui représente la récompense cumulative attendue à partir d'un état initial $s \in S$ en suivant les actions conjointes $a^j \in A^n$ sous l'effet de **Guides de Contraintes**. Cette fonction est définie pour des agents agissant de manière séquentielle et cyclique (mode AEC) (voir la Définition 1). La Figure 2 illustre les liens entre $MOISE^+$ et le Dec-POMDP via MOISE+MARL.

À chaque instant $t \in \mathbb{N}$ (initialement $t = 0$), l'agent $i = t \bmod n$ doit assumer le rôle $\rho_i = ar(i)$. Pour chaque spécification déontique valide $d_i = rds(\rho_i) = \langle tc_i, y_i, m_i \rangle$, l'agent est autorisé (si $y_i = 0$) ou obligé (si $y_i = 1$) d'exécuter la mission $m_i \in \mathcal{M}$ (avec $\mathcal{G}_{m_i} = mo(m_i)$). L'agent choisit d'abord une action parmi celles attendues A_t si une valeur aléatoire est inférieure à la rigidité ch_t , sinon parmi l'ensemble A ; ainsi, un $ch_t = 1$ impose une contrainte forte.

L'action appliquée à s_t conduit à l'état suivant s_{t+1} , génère la prochaine observation ω_{t+1} et une récompense. Celle-ci est la somme de la récompense globale et des ajustements organisationnels : i) un bonus (via les Guides de Récompense d'Objectif), pondéré par $\frac{1}{1-p+\epsilon}$ pour ajuster son impact, et ii) une pénalité (via les Guides de Récompense de Rôle), pondérée par la rigidité ch_t pour ajuster son impact. Le calcul de la récompense cumulative se poursuit dans s_{t+1} avec l'agent suivant $(i+1) \bmod n$.

3.4 Faciliter l'implémentation des Guides de Contrainte

Puisque les rôles, objectifs et missions sont de simples étiquettes, leur définition est implicite. Cependant, implémenter une relation rag , rrg ou grg nécessite de définir de nombreux historiques, souvent redondants, rendant une définition extensionnelle fastidieuse. De plus, la logique de chaque **Guide de Contrainte** analyse la trajectoire de l'agent pour vérifier son appartenance à un ensemble prédéfini. Par exemple, rag détermine les actions attendues selon l'appartenance de la trajectoire à un ensemble donné et la nouvelle observation.

Une approche consiste à laisser l'utilisateur définir ses **Guides de Contrainte** via une logique personnalisée (par script, par exemple). Dans ce cas, la relation $b_g : H \rightarrow \{0, 1\}$ formalise la décision d'appartenance d'un historique à un ensemble H_g . Pour simplifier l'implémentation, nous proposons un *Trajectory-based Pattern* (TP), inspiré du Traitement Automatique du Langage, noté $p \in P$, permettant de définir intentionnellement un ensemble d'historiques.

Un TP implique que toute observation ou action réelle considérée est connue et associée à une étiquette $l \in L$ (via $l : \Omega \cup A \rightarrow L$) afin d'être gérée de manière pratique. Un TP $p \in P$ est défini comme suit : p est soit une « séquence feuille » notée comme un couple historique-cardinalité $s_l = \langle h, \{c_{min}, c_{max}\} \rangle$ (où $h \in H$, $c_{min} \in \mathbb{N}$, $c_{max} \in \mathbb{N} \cup \{^*\}$); soit une « séquence nœud » notée comme un couple composé d'un tuple de séquences et d'une cardinalité $s_n = \langle \langle s_{l_1}, s_{l_2}, \dots \rangle, \{c_{min}, c_{max}\} \rangle$. Par exemple, le pattern $p = \langle [o_1, a_1, [o_2, a_2]](0, 2) \rangle(1, ^*)$ peut être formalisé comme la séquence nœud $\langle \langle \langle o_1, a_1 \rangle, \langle 1, 1 \rangle \rangle, \langle \langle o_2, a_2 \rangle, \langle 0, 2 \rangle \rangle \rangle(1, ^*)$, indiquant l'ensemble des historiques H_p contenant au moins une fois la sous-séquence constituée d'une première paire $\langle o_1, a_1 \rangle$ suivie d'au maximum deux répétitions de la paire $\langle o_2, a_2 \rangle$.

4 La méthode TEMM

Comme indiqué en [section 2](#), aucun travaux ne répond pleinement à nos exigences pour déterminer les rôles et objectifs implicites ainsi que l'adéquation organisationnelle. Nous proposons donc la méthode *Trajectory-based Evaluation in MOISE+MARL* (TEMM) pour inférer des spécifications comme des rôles ou des missions.

TEMM repose sur l'apprentissage non supervisé pour généraliser ces spécifications à partir des trajectoires collectées lors de multiples épisodes. En quantifiant l'écart entre les spécifications implicites inférées et les comportements observés, TEMM mesure l'adéquation organisationnelle, c'est-à-dire la conformité d'une politique aux spécifications inférées. TEMM repose sur des définitions propres à chaque spécification organisationnelle de $MOISE^+$ (historiques conjoints, etc.), inférées progressivement via des techniques non supervisées. Une description informelle est disponible ².

1) Inférence des rôles et héritage

Un rôle ρ est défini comme une politique dont l'historique contient une Séquence Commune la Plus Longue (SCL). Un rôle ρ_2 hérite de ρ_1 si sa SCL est incluse dans celle de ρ_1 . TEMM emploie le clustering hiérarchique pour extraire ces SCL, représentées sous forme de dendrogramme, et mesure l'écart entre les séquences actuelles et inférées, définissant ainsi l'adéquation organisationnelle structurelle.

2) Inférence des objectifs, plans et missions

Un objectif correspond à un ensemble d'observations conjointes communes, atteint via les historiques d'agents performants. TEMM construit pour chaque historique conjoint un graphe de transition, fusionne ces graphes et, à l'aide de K-means, identifie des groupes de trajectoires. Les ensembles d'observations restreints extraits pour chaque groupe sont considérés comme des objectifs implicites, permettant d'inférer les plans sur la base des choix et séquences. Une mission est définie comme l'ensemble des objectifs réalisés par un ou plusieurs agents, et la distance entre les objectifs inférés et l'observation conjointe actuelle permet de calculer l'adéquation organisationnelle fonctionnelle.

3) Inférence des obligations et permissions

Une obligation se produit lorsqu'un agent, jouant le rôle ρ , réalise exclusivement les objectifs d'une mission dans un intervalle donné, tandis qu'une permission permet d'atteindre d'autres objectifs sous conditions. TEMM identifie l'association agent-mission et détermine si l'agent est contraint (obligation) ou bénéficie de flexibilité (permission). L'adéquation organisationnelle globale est la somme de l'adéquation structurelle et fonctionnelle.

Globalement, bien que le K-means et le clustering hiérarchique requièrent une configuration manuelle pour éviter les erreurs, TEMM recommande de vérifier et d'ajuster manuellement les rôles et objectifs obtenus pour éliminer les perturbations éventuelles.

². L'implémentation « MOISE+MARL API » (MMA), les hyperparamètres et spécifications utilisés sont disponibles à <https://github.com/julien6/MOISE-MARL>.

5 Cadre expérimental

Cette section détaille le cadre expérimental utilisé pour évaluer le cadre MOISE+MARL.

5.1 Implémentation de MOISE+MARL

Nous avons développé une API Python ², pour implémenter MOISE+MARL. Cette API structure le modèle $MOISE^+$ en classes de données imbriquées afin de définir les spécifications organisationnelles (rôles, objectifs, permissions...). Nous utilisons la bibliothèque *PettingZoo* [17] (similaire à *Gymnasium* [6]) pour la gestion des environnements en y intégrant un dictionnaire personnalisable pour le mappage des étiquettes d'observation/action (l), ainsi que le support des TP pour définir et faire correspondre les motifs.

Chaque **Guide de Contrainte** (rag , rrg et grg) est implémenté comme une classe distincte. Les utilisateurs peuvent les définir via des fonctions personnalisées ou des règles JSON (par exemple, rag associe un couple (TP, dernière observation) à des actions attendues, et grg applique des bonus selon des TP spécifiques). La classe globale MMA intègre ces guides et relie les agents aux rôles via des relations telles que ar , intégrant ainsi les spécifications de $MOISE^+$.

Une fois configuré, MMA encapsule l'environnement avec un wrapper *PettingZoo* qui applique des masques d'actions et ajuste les récompenses pour garantir le respect des spécifications durant l'entraînement. Il intègre également *MARL-lib* [24] pour accéder aux algorithmes MARL de pointe sur un cluster haute performance.

Enfin, la méthode TEMM, avec des hyperparamètres optimisés manuellement, est utilisée après l'entraînement pour inférer les rôles et objectifs implicites via clustering hiérarchique et K-means. Cette analyse génère des sorties visuelles (dendrogrammes, graphes de transition) et permet d'exporter les trajectoires JSON des comportements organisationnels inférés.

5.2 Environnements utilisés

Nous testons MOISE+MARL dans quatre environnements MARL, modélisés comme des scénarios Dec-POMDP et chacune présentant des défis distincts en termes d'organisations requises pour atteindre au mieux l'objectif global :

- i) **Predator-Prey** : Plusieurs prédateurs coopèrent pour capturer une proie, testant la coordination pour atteindre un objectif collectif [25];
- ii) **Overcooked-AI** : Jeu de cuisine en équipe où les agents préparent et servent des plats dans des cuisines de complexité croissante [20]. Cet environnement évalue la coordination et l'allocation des tâches avec des rôles clairs (chef, assistant, serveur);
- iii) **Warehouse Management** : Les agents gèrent un entrepôt en coordonnant les livraisons vers des points de demande, influençant leur spécialisation (transport, gestion des stocks);
- iv) **Cyber-Defense Simulation** : Simulation de défense d'un réseau contre des cyberattaques. Les agents identifient et contrent les menaces tout en respectant des règles de sécurité strictes, testant ainsi leur sûreté [31].

Ces environnements, encapsulables via l'API *PettingZoo*,

s'intègrent avec notre implémentation de MOISE+MARL et facilitent l'application des spécifications organisationnelles.

5.3 Algorithmes MARL utilisés

Nous avons évalué notre cadre avec plusieurs algorithmes MARL : i) **MADDPG (Multi-Agent Deep Deterministic Policy Gradient)** [25] : Un algorithme d'apprentissage centralisé avec exécution décentralisée, permettant à chaque agent d'avoir une politique déterministe tout en utilisant l'information globale lors de l'entraînement; ii) **MAPPO (Multi-Agent Proximal Policy Optimization)** [9] : Une version adaptée de PPO pour les systèmes multi-agent, optimisée pour une convergence stable de la politique conjointe dans des scénarios complexes; iii) **Q-Mix** [32] : Un algorithme basé sur les valeurs Q qui apprend à combiner les Q-valeurs individuelles des agents en une valeur conjointe afin d'optimiser la coopération; iv) **COMA (Counterfactual Multi-Agent)** [11] : Un algorithme acteur-critique capable d'estimer l'impact des actions d'un agent individuel sur la récompense globale de l'équipe.

5.4 Spécifications organisationnelles

Pour chaque environnement, nous avons défini un ensemble de spécifications organisationnelles. Ces spécifications comprennent les rôles, les missions, ainsi que les permissions et obligations. Voici une description informelle de ces spécifications ² : i) **Predator-Prey** : Des rôles de prédateur et de proie sont définis, chaque prédateur ayant des objectifs spécifiques tels que « capturer la proie » ou « bloquer les voies d'évasion »; ii) **Overcooked-AI** : Les agents adoptent trois rôles principaux : chef, assistant et serveur. Le chef est responsable de la cuisson et de l'assemblage des plats, l'assistant s'occupe de la découpe et de l'approvisionnement en ingrédients, et le serveur se charge de la livraison des plats aux clients. Les missions consistent principalement à préparer et à servir un nombre déterminé de plats dans un délai imparti; iii) **Warehouse Management** : Les agents adoptent des rôles tels que « transporteur » et « gestionnaire d'inventaire », avec des missions liées à la gestion des flux logistiques et à l'optimisation des livraisons; iv) **Cyber-Defense Simulation** : Les agents occupent des rôles de défenseurs de réseau, avec des obligations telles que la détection d'intrusions, la levée d'alertes aux autres agents pour protéger l'essaim de drones.

5.5 Configuration matérielle

Toutes les expériences ont été menées sur un cluster académique haute performance, avec des nœuds GPU (NVIDIA A100, V100 et AMD MI210). Chaque configuration algorithme-environnement a été exécutée 5 fois en parallèle pour assurer des résultats fiables. Les hyperparamètres ² (taux d'apprentissage, facteurs d'actualisation, taux d'exploration) proviennent soit des banques de MARLlib, soit d'une recherche sur grille réalisée via *Optuna* [2].

5.6 Métriques d'évaluation et protocole

L'évaluation prend en compte l'efficacité des politiques et l'impact des spécifications organisationnelles en reposant sur les métriques suivantes : i) **Récompense Cumulative** : Mesure l'efficacité de la politique dans l'atteinte des objectifs de l'environnement; ii) **Écart-type de la Récompense** : Reflète la stabilité des politiques apprises au cours des épisodes; iii) **Taux de Convergence** : Indique la rapidité avec laquelle les politiques atteignent une performance stable; iv) **Taux de Violation des Contraintes** : Évalue le respect des contraintes organisationnelles par la politique, ce qui est crucial pour la sécurité; v) **Score de Cohérence** : Mesure l'alignement entre les comportements appris et les spécifications organisationnelles; vi) **Score de Robustesse** : Évalue la capacité des agents à maintenir leur performance face à une série de scénarios difficiles; vii) **Niveau d'Adéquation Organisationnelle** : Quantifie l'adéquation organisationnelle avec TEMM.

Notre protocole compare le *Baseline de Référence* (RB) sans contraintes organisationnelles au *Baseline Organisé* (OB) utilisant MOISE+MARL. Pour le RB, nous utilisons MMA pour entraîner les agents dans chaque environnement (jusqu'à convergence ou limite d'épisodes) sans appliquer de spécifications organisationnelles, puis nous sélectionnons l'algorithme obtenant la Récompense Cumulative maximale. Pour l'OB, nous réinitialisons environnements et agents, appliquons via MMA des spécifications prédéfinies (chaque agent se voit attribuer un rôle) et ré-entraînons ces agents avec l'algorithme le plus performant du RB. Les métriques permettent alors des comparaisons. Nous évaluons l'impact de MOISE+MARL en vérifiant si les comportements des agents s'alignent avec les rôles définis (à l'aide de l'Écart-type de Récompense, du Taux de Convergence et du Score de Robustesse). Une différence significative du Niveau d'Adéquation Organisationnelle entre RB et OB, et une corrélation entre les rôles et ce niveau, confirmera l'efficacité du cadre. Enfin, nous comparons MOISE+MARL à AGR+MARL (qui ne considère que les rôles) pour évaluer l'importance des missions.

6 Résultats

Cette section discute des résultats obtenus sur les quatre environnements.

6.1 Adéquation organisationnelle quantitative et cohérence

Comme l'illustre le [Tableau 1](#), l'adéquation organisationnelle est systématiquement plus élevée dans l'OB, confirmant que MOISE+MARL aligne efficacement le comportement des agents sur les spécifications organisationnelles. Par exemple, dans **Predator-Prey** avec **MADDPG**, l'OB atteint un niveau d'adéquation de 0.87 (soit +49% par rapport aux 0.43 du RB), tandis que dans **Overcooked-AI** avec **MAPPO**, on observe 0.91 (+89%). Même constat pour **Warehouse Management** avec **Q-Mix**, où l'adéquation passe de 0.50 (RB) à 0.90 (OB).

De façon générale, contraindre les agents par des spécifi-

TABLE 1 – Résultats détaillés pour chaque environnement et algorithme favorisé, pour le RB et l’OB.

Env.	Alg.	Spec. Org.	Récom. Cum.	Écart-Type	Taux Conv.	Taux Viol.	Score Cohé.	Score Rob.	Niv. Adéq.
Predator-Prey	MADDPG		200.1	21.5	0.65	12.3%	-	0.65	0.43
Predator-Prey	MADDPG	Oui	245.8	15.2	0.85	0.0%	0.81	0.83	0.87
Overcooked-AI	MAPPO		348.2	15.6	0.75	7.1%	-	0.71	0.48
Overcooked-AI	MAPPO	Oui	391.2	10.4	0.92	0.0%	0.89	0.89	0.91
Warehouse M.	Q-Mix		257.4	18.9	0.74	7.8%	-	0.68	0.50
Warehouse M.	Q-Mix	Oui	307.1	13.8	0.88	0.0%	0.88	0.86	0.90
Cyber-Defense	COMA		162.4	17.3	0.70	12.2%	-	0.67	0.45
Cyber-Defense	COMA	Oui	188.9	11.2	0.86	0.0%	0.76	0.80	0.83

cations organisationnelles diminue la déviation de récompense et accélère la convergence, indiquant un impact notable sur leur comportement. Nous avons observé manuellement, notamment dans **Predator-Prey**, que les politiques entraînées correspondent bien à une organisation structurée et fonctionnelle implicite.

Le **score de cohérence** demeure également élevé (jusqu’à 0.76 dans le contexte bruité de **Cyber-Defense**), montrant que, malgré les perturbations, les spécifications organisationnelles inférées sont proches de celles appliquées.

6.2 Performance et stabilité selon les algorithmes

Les résultats indiquent que les algorithmes basés sur la politique et les algorithmes acteur-critique, tels que **MADDPG** et **MAPPO**, bénéficient considérablement du cadre MOISE+MARL, notamment en termes de cohérence et de stabilité. Par exemple, dans l’environnement **Overcooked-AI**, **MAPPO** a vu son écart-type de récompense passer de 15.6 (RB) à 10.4 (OB), reflétant une politique plus stable avec moins de fluctuations comportementales. De même, **MADDPG** dans **Predator-Prey** a montré une diminution similaire, passant de 21.5 en RB à 15.2 en OB, indiquant une fiabilité accrue.

En revanche, les algorithmes basés sur la valeur, comme **Q-Mix**, ont maintenu une haute performance en récompense cumulative, mais ont affiché une variabilité plus importante en termes de cohérence. Par exemple, dans l’environnement **Warehouse Management**, **Q-Mix** a atteint un écart-type de récompense de 13.8 en OB, soit une amélioration notable par rapport aux 18.9 en RB, mais toujours supérieur à la stabilité observée dans les algorithmes basés sur la politique. Cela suggère que, bien que **Q-Mix** soit efficace pour atteindre les objectifs de la tâche, il pourrait nécessiter un ajustement supplémentaire pour les rôles avec MOISE+MARL afin d’améliorer la cohérence.

6.3 Impact des contraintes organisationnelles sur la convergence, la robustesse et le taux de violation des politiques

L’application des contraintes organisationnelles a permis d’accélérer les taux de convergence dans tous les environnements. Dans l’environnement **Cyber-Defense**, **COMA** avec MOISE+MARL a convergé à un taux de 0.86, contre 0.70 en RB. Des tendances similaires ont été observées dans

l’environnement **Warehouse Management** avec **Q-Mix**, qui est passé de 0.74 en RB à 0.88 en OB. Cette convergence accélérée peut être attribuée aux rôles et aux mises, qui réduisent l’espace de recherche des politiques.

En outre, nous avons observé que les taux de violation des contraintes étaient systématiquement plus élevés lorsque les contraintes organisationnelles étaient définies avec une rigidité de contrainte plus faible. Dans l’environnement **Overcooked-AI**, **MAPPO** a enregistré un taux de violation nul avec une rigidité de contrainte de 1, contre 7.1% avec une rigidité de 0. De même, dans **Warehouse Management**, **Q-Mix** a vu le taux de violation passer de 7.8% à zéro lorsque la rigidité augmentait. Cela vient renforcer l’efficacité du cadre dans l’amélioration du respect des comportements souhaités.

De plus, nous avons observé une amélioration constante de la robustesse lorsque les spécifications organisationnelles étaient appliquées aux agents. Par exemple, **MADDPG** dans **Predator-Prey** et **MAPPO** dans **Overcooked-AI** ont obtenu des scores de cohérence élevés, respectivement 0.81 et 0.89, indiquant que les agents suivaient de près les rôles inférés. La robustesse s’est également améliorée, avec **MAPPO** dans **Overcooked-AI** atteignant un score de robustesse de 0.89, contre 0.71 en RB, soulignant une meilleure résilience face aux perturbations.

Cependant, un biais potentiel peut être souligné : les spécifications organisationnelles ont été conçues pour englober toutes les observations, évitant ainsi les situations nouvelles non gérées.

6.4 Comparaison entre MOISE+MARL et AGR+MARL

TABLE 2 – Comparaison de la performance entre MOISE+MARL et AGR+MARL.

Framework	Env.	Taux Conv.	Score Rob.	Niv. Adéq.	Récom. Cum.
MOISE+MARL	PP	0.85	0.83	0.87	245.8
AGR+MARL	PP	0.75	0.69	0.56	208.4
MOISE+MARL	OA	0.92	0.89	0.91	391.2
AGR+MARL	OA	0.82	0.75	0.58	348.9
MOISE+MARL	WM	0.88	0.86	0.90	307.1
AGR+MARL	WM	0.76	0.72	0.61	278.6

Impact des objectifs intermédiaires Le [Tableau 2](#) met en lumière l’effet de ces objectifs dans MOISE+MARL. Dans

Overcooked-AI, **MAPPO** obtient une récompense cumulative de 391.2 et une adéquation organisationnelle de 0.91, soit 33% de plus qu’AGR+MARL (0.58). Dans **Warehouse Management**, **Q-Mix** sous MOISE+MARL atteint 307.1 de récompense (contre 278.6 pour AGR+MARL) et un score de robustesse supérieur (0.86 vs 0.72).

Ces résultats soulignent l’importance des objectifs intermédiaires pour des comportements plus stables et mieux orientés vers l’objectif. MOISE+MARL surpasse ainsi AGR+MARL en récompense, robustesse et adéquation organisationnelle dans **Predator-Prey**, **Warehouse Management** et **Overcooked-AI**. Enfin, l’augmentation du nombre de contraintes organisationnelles accroît quasi linéairement la durée d’entraînement, d’après nos premières résultats².

7 Conclusion et travaux futurs

Nous proposons le cadre MOISE+MARL pour améliorer le contrôle et l’explicabilité des agents en MARL par l’intégration d’un modèle organisationnel explicite. Nos résultats montrent une meilleure convergence et stabilité des politiques, ainsi qu’un alignement des comportements observés avec les spécifications attendues et inférées de façon agnostique.

Cependant, reposant sur des spécifications prédéfinies, MOISE+MARL peut peiner à prendre en compte le surcoût computationnel. Trois axes de recherche émergent donc : i) Développer des mécanismes adaptatifs pour faire évoluer dynamiquement rôles et missions ; ii) Explorer des méthodes automatisées (*Large Language Models*) pour générer des spécifications organisationnelles ; iii) Améliorer la scalabilité de TEMM et proposer d’autres approches. Ces perspectives ouvriront la voie à une meilleure intégration de l’organisation dans le MARL, renforçant notamment la robustesse, sûreté et l’explicabilité des agents pour des systèmes réels.

Remerciements

Ce travail est financé par *Thales Land Air Systems* et s’inscrit dans les travaux de la chaire *Cyb’Air* ainsi que de l’*AICA IWG*.

Références

- [1] A. Isakov et al. Cooperative-competitive decision-making in resource management : A reinforcement learning perspective. In *Advances in Machine Learning and Automated Learning*. 2024.
- [2] Akiba Takuya et al. Optuna : A next-generation hyperparameter optimization framework. In *Proc. of the 25th ACM SIGKDD Int. Conf. on Knowledge Discovery & Data Mining*, 2019.
- [3] Stefano V. Albrecht and Jacob Y. Foerster. Survey on recent advances in cooperative multi-agent reinforcement learning. *Journal of Artificial Intelligence Research*, 2024.
- [4] Alex Ray et al. Benchmarking safe exploration in deep reinforcement learning. In *arXiv :1910.01708*, 2019.
- [5] Andrew Wilson et al. Learning and transferring roles in multi-agent mdps. In *Proc. of AAAI*, 2008.
- [6] Ariel Kwiatkowski et al. Gymnasium : A standard interface for reinforcement learning environments, 2024.
- [7] Hamid R Berenji and David Vengerov. Learning, cooperation, and coordination in multi-agent systems. *Inference Systems Corporation, Technical report*, 2000.
- [8] Aurélie Beynier and Alain Mouaddib. A decentralized approach for reinforcement learning in cooperative multi-agent systems. In *Proc. of the 23rd Int. Joint Conf. on Artificial Intelligence*, 2013.
- [9] Chengjie Yu et al. The surprising effectiveness of ppo in cooperative multi-agent games. *NeurIPS*, 2021.
- [10] Jacques Ferber, Olivier Gutknecht, and Fabien Michel. Agent/Group/Roles : Simulating with Organizations. In *4th Int. Workshop on Agent-Based Simulation*, 2003.
- [11] Jakob Foerster et al. Counterfactual multi-agent policy gradients. *Int. Conf. on Machine Learning*, 2018.
- [12] Javier Garcia and Fernando Fernandez. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 2015.
- [13] Mohammad Ghavamzadeh and Sridhar Mahadevan. Hierarchical reinforcement learning with cooperative agents. In *Proc. of the 23rd Int. Conf. on Machine Learning*, 2006.
- [14] Hubner, Jomi F et. al. Developing organised multiagent systems using the moise+ model : programming issues at the system and agent levels. *Int. Journal of Agent-Oriented Software Engineering*, 2007.
- [15] Jakob Foerster et al. Learning to communicate with deep multi-agent reinforcement learning. *NeurIPS*, 2018.
- [16] Joshua Achiam et al. Constrained policy optimization. In *Proc. of the 34th Int. Conf. on Machine Learning*, 2017.
- [17] Justin K Terry et al. Pettingzoo : Gym for multi-agent reinforcement learning. *Proc. of the NeurIPS 2020 Track on Datasets and Benchmarks*, 2020.
- [18] K. Matsuyama et al. Cord : Generalizable cooperation via role diversity. *arXiv preprint*, 2025.
- [19] Tiziano Maisonhaute, Fabien Michel, and Jean-Christophe Soulié. État de l’art sur les approches en apprentissage par renforcement multi-agent. In *32ème journées francophones sur les systèmes multi-agents*, 2024.
- [20] Micah Carroll et al. Overcooked-ai : A benchmark for multi-agent learning under partial observability. *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2020.

- [21] Mohammed Alshiekh et al. Safe reinforcement learning via shielding. *Proc. of the 32nd AAAI Conf. on Artificial Intelligence*, 2018.
- [22] Frans A. Oliehoek and Christopher Amato. *A Concise Introduction to Decentralized POMDPs*. 2016.
- [23] Y. Qi, J. Cao, and B. Wu. Bidirectional q-learning for recycling path planning of used appliances under strong and weak constraints. *Communications in Transportation Research*, 2024.
- [24] Qi Hu et al. Marllib : A comprehensive library for multi-agent reinforcement learning. *arXiv preprint arXiv :2106.05912*, 2021.
- [25] Ryan Lowe et al. Multi-agent actor-critic for mixed cooperative-competitive environments. *NeurIPS*, 30, 2017.
- [26] N. Sao Mai. *The intrinsic motivation of reinforcement and imitation learning for sequential tasks*. PhD thesis, HAL Archive, 2024.
- [27] Jack Serrino, Max Kleiman-Weiner, et al. Finding friend and foe in multi-agent games. In *NeurIPS*, 2019.
- [28] Julien Soule, Jean-Paul Jamont, Michel Occello, Louis-Marie Traonouez, and Paul Théron. A marl-based approach for easing SMA organization engineering. In *Proc. of the 20th Int. Conf. Artificial Intelligence Applications and Innovations*, 2024.
- [29] Julien Soule, Jean-Paul Jamont, Michel Occello, Louis-Marie Traonouez, and Paul Théron. Une approche basée sur l'apprentissage par renforcement pour l'ingénierie organisationnelle d'un SMA. In *32èmes journées francophones sur les systèmes multi-agents*, 2024.
- [30] Julien Soule, Jean-Paul Jamont, Michel Occello, Louis-Marie Traonouez, and Paul Théron. An organizationally-oriented approach to enhancing explainability and control in multi-agent reinforcement learning. *Proc. of the 24th Int. Conf. on Autonomous Agents and Multiagent Systems*, 2025. à paraître.
- [31] Standen Maxwell et al. Cyborg : A gym for the development of autonomous cyber agents, 2021.
- [32] Tabish Rashid et al. Qmix : Monotonic value function factorisation for deep multi-agent reinforcement learning. *Proc. of the 35th Int. Conf. on Machine Learning*, 2018.
- [33] W. Wen et al. Role play : Learning adaptive role-specific strategies in multi-agent interactions. *arXiv preprint*, 2024.
- [34] Sagir M Yusuf and Christopher Baber. Inferential reasoning for heterogeneous multi-agent missions. *Int. Journal of Electrical and Computer Engineering*, 2020.
- [35] Z. Xie et al. Roco : Role-oriented communication for efficient multi-agent reinforcement learning. *SSRN Electronic Journal*, 2024.

Gestion des ressources de communication pour satellites dans une fédération de constellations d'observation de la Terre*

Hénoïk Willot¹, Jean-Loup Farges¹, Gauthier Picard¹, Philippe Pavero²

¹ DTIS, ONERA, Université de Toulouse

² Airbus Defence and Space

Résumé

Cet article aborde le problème nouveau, découlant du concept novateur de fédération de missions dans le domaine de l'observation de la Terre grâce au développement de réseaux de stations terrestres, rencontré par le système de communication et de gestion des ressources des satellites (SCRMS). Ce problème décisionnel est celui de la sélection des contacts, sous contraintes de satisfaction des besoins de communication des satellites, en minimisant le coût et les brouillages. Nous avons développé plusieurs méthodes de résolution : un programme linéaire, des schémas de recherche incomplète et des schémas d'allocation gloutonne.

Mots-clés

Observation de la Terre, ressources de communication, GSaaS, programmation linéaire en nombres entiers, recherche incomplète.

1 Introduction

Traditionnellement, les missions spatiales consacrées à l'observation de la Terre disposent de leurs propres ressources. Chaque mission possède sa constellation de satellites, ses stations au sol et son centre de contrôle. Les stations au sol sont dédiées à la communication avec les satellites et permettent de charger des plans d'observation et de télécharger des observations. Plus récemment, des stations de communication tierces utilisent le paradigme de la station au sol en tant que service (GSaaS) ont été développées [1]. Dans ce paradigme, les clients réservent des ressources de communication auprès d'un fournisseur de segments terrestres au lieu de construire leurs propres stations.

Une couche de fédération multi-agents pour coordonner plusieurs missions d'observation de la Terre indépendantes a été proposée [2]. L'objectif de cette fédération est de permettre aux clients qui demandent l'acquisition de vastes zones d'accéder facilement à plusieurs constellations de satellites et de sites de communication afin de composer et de télécharger leurs acquisitions en un temps réduit comparé aux demandes conventionnelles non coordonnées. Le système de communication et de gestion des ressources des satellites (SCRMS) doit gérer les communications en utilisant non seulement les stations terrestres appartenant aux

missions de la fédération, mais aussi le GSaaS.

Le présent travail est consacré au développement et à l'évaluation expérimentale d'algorithmes permettant de résoudre ce problème d'allocation. Sa principale contribution est la fourniture de lignes directrices solides pour la sélection d'algorithmes pour la fonction de réservation de sites de communication.

2 Problème de sélection

Le SCRMS est chargé de fournir des possibilités de communication avec le sol à tous les satellites de toutes les missions fédérées. Il gère la communication entre d'une part N satellites et d'autre part des sites comprenant une ou plusieurs stations au sol, chaque station contrôlant une antenne spécifique. Pour cela il recherche des fenêtres de communication pour le transfert de données en liaison montante et descendante. Les missions fournissent les orbites de leurs satellites et les réseaux de stations au sol publient les coordonnées et les masques de visibilité de leurs stations.

La propagation des orbites et le calcul des événements conduisent pour chaque satellite i à L_i contacts potentiels avec le sol. Chaque contact $l \in \{1, \dots, L_i\}$ est caractérisé par son site au sol, et sa fenêtre temporelle $[u_{i,l}, \bar{u}_{i,l}]$, permettant de déduire sa durée $d_{i,l}$. La variable de décision associée à un contact est $x_{i,l} \in \{0, 1\}$, valant 1 si le contact l du satellite i est choisi dans la réservation.

La sélection de contacts permet de remplir en partie les besoins de communications d'un satellite. Chaque satellite i a K_i besoins à remplir, définis par une durée de communication $D_{i,k}$, à remplir sur une fenêtre temporelle $[\underline{t}_{i,k}, \bar{t}_{i,k}]$, avec une liste de sites réservables $\mathcal{S}_{i,k}$ et une bande de radiocommunication (bande S pour téléverser les futurs plan d'observation et les opérations de maintenance, et bande X pour le téléchargement des observations). Un contact peut donc être choisi pour participer à la validation d'un besoin k si sa fenêtre temporelle est incluse dans celle du besoin, que le site fait partie des sites réservables et que la bande de radiocommunication est la bonne. Notons $\mathcal{Z}_{i,l}$ l'ensemble des besoins qu'un contact l est en mesure de satisfaire.

Notre problème de décision est donc celui de sélectionner des contacts tel que le plan obtenu puisse vérifier pour chaque satellite i et chacun de ses besoins k

$$\sum_{l \in \{1, \dots, L_i\} / k \in \mathcal{Z}_{i,l}} d_{i,l} x_{i,l} \geq D_{i,k}.$$

Nous pouvons ainsi optimiser deux critères : le coût total

*Ce document résume en français un article accepté à la conférence CPAIOR'25 [3].

de la sélection, en additionnant le coût des contacts retenus (dépendant du modèle de coût du GSaaS, basé sur la durée du contact ou sur un coût fixe par contact), et le conflit et brouillage total induit par la sélection. Deux contacts l et m des satellites i et j , respectivement, peuvent générer une quantité $b_{i,l,j,m}$ de conflit et brouillage si leurs fenêtres temporelles se chevauchent et qu'ils partagent le même site (pas nécessairement la même antenne). Le total est obtenu en faisant la somme du conflit et brouillage généré par toutes les paires de contacts. Nous considérons ici un ordre lexicographique sur les critères, nous minimisons en premier le coût (resp. le conflit et brouillage) et à valeur égale nous minimisons le conflit et brouillage (resp. le coût).

3 Méthodes de résolution

Round-robin. La procédure round-robin permet d'envisager une distribution équitable d'articles entre des agents. Nous associons ici les besoins aux agents et les contacts aux articles. Chaque besoin est considéré à tour de rôle et sélectionne un contact supplémentaire. L'ordre d'examen des besoins dépend de l'évaluation de ces derniers ou de leur difficulté, déclinant deux méthodes. L'évaluation du besoin k du satellite i s'obtient en sélectionnant la meilleure valeur sur les critères calculée en ajoutant un des contacts non sélectionnés qui peuvent contribuer au besoin k . La dureté du besoin est proportionnelle au ratio de sa durée de communication et de la durée totale possible parmi les contacts non sélectionnés.

Procédures gloutonnes. Un schéma d'allocation glouton sélectionne les contacts un par un en faisant le choix qui semble le meilleur sur le moment, sans réviser cette décision par la suite. Nous considérons deux manières de choisir : soit l'on se concentre sur les besoins, soit les contacts. Dans le premier cas, les besoins sont sélectionnés de manière itérative pour être pleinement satisfaits, guidé par leur évaluation ou par leur dureté (similaire au Round-robin). La deuxième méthode conduit à un glouton guidé par la sélection du contact avec la meilleure évaluation.

Recherches locales. Les procédures round-robin et gloutonnes sont des heuristiques constructives : elles produisent des solutions partielles à chaque étape jusqu'à ce qu'elles atteignent l'admissibilité. À l'inverse, les recherches locales travaillent sur des solutions complètes en essayant d'améliorer les critères tout en maintenant l'admissibilité. Ces méthodes nécessitent une solution initiale et la définition du voisinage d'une solution. Pour choisir la solution initiale nous nous servons de deux heuristiques : si le coût est optimisé en premier nous sélectionnons pour chaque satellite i les contacts par ordre croissant de coût jusqu'à ce que ses besoins soient remplis, et sinon nous utilisons la procédure du glouton guidé par l'évaluation du besoin. Avec un voisinage obtenu par la dé-sélection d'un unique $x_{i,l}$ (en interdisant celles qui violent la contrainte sur les besoins), nous définissons deux schémas de recherche : une recherche du meilleur d'abord, qui sélectionne la solution avec la meilleure évaluation parmi toutes les solutions déjà évaluées mais non développées, et une recherche

en profondeur d'abord, qui sélectionne la meilleure solution voisine de la solution courante. Avec un plus grand voisinage obtenu en dé-sélectionnant un nombre donné de contacts nous pouvons également reconstruire l'admissibilité à l'aide d'une recherche à large voisinage (*Large Neighborhood Search*).

Programmation linéaire en nombres entiers. Les contraintes sur les besoins ainsi que l'expression du coût sont déjà linéaires et s'intègrent directement. L'expression du conflit et du brouillage est cependant quadratique, étant donné qu'elle contient le produit $x_{i,l}x_{j,m}$ afin d'ajouter la quantité $b_{i,l,j,m}$ ssi les deux contacts l et m sont sélectionnés. Cette expression est cependant facilement linéarisable sans approximation en introduisant une variable binaire intermédiaire $y_{i,l,j,m}$, remplaçant ce produit, et trois contraintes linéaires pour chaque paire de contacts.

4 Résultats et perspectives

Les performances sont comparées sur un scénario réaliste de 60 jours, avec une fédération de 3 constellations de 2,4 et 4 satellites respectivement, et des besoins qui évoluent tout au long de cette période. Les méthodes doivent planifier chaque jour les contacts des 10 prochains. Les résultats obtenus sont satisfaisants, surtout pour la modélisation linéaire qui respecte la contrainte de temps de calcul imposée (moins de 5 minutes pour obtenir le plan). Certains aspects du problème réel ne sont cependant pas encore pris en compte, comme la possibilité que le GSaaS refuse un contact. Il est aussi possible d'avoir des forfaits de communication avec un nombre de contacts prépayé et un coût supplémentaire pour chaque contact excédent. De plus les besoins considérés sont routiniers, dont seul la charge évolue, mais il existe aussi des besoins urgents, liés à l'observation d'une zone spécifique. Il est alors nécessaire de réserver des contacts avant et après le survol de cette zone pour mettre à jour les plans d'observation et récupérer ces dernières le plus rapidement possible. Ces travaux sont réalisés dans le projet européen DOMINO-E (<https://domino-e.eu/>).

Références

- [1] Elisa Carcaillon and Berylia Bancquart. Market perspectives of ground segment as a service. *71st International Astronautical Congress (IAC)*, 2020.
- [2] Jean-Loup Farges, Filippo Studzinski Perotto, Cédric Pralet, Gauthier Picard, Cyril de Lussy, Jonathan Guerra, Philippe Pavero, and Fabrice Planchou. Going beyond mono-mission earth observation : Using the multi-agent paradigm to federate multiple missions. In *23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS-24)*, 2024.
- [3] Hénôik Willot, Jean-Loup Farges, Gauthier Picard, and Philippe Pavero. Satellite communication resources management in a earth observation federation of constellations. In *International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research (CPAIOR-25)*, 2025.

Approches exactes pour le problème de satisfiabilité diversifiée

Zhifei Zheng, Sami Cherif, Rui Sá Shibasaki, Chu-Min Li
Laboratoire MIS UR 4290, Université de Picardie Jules Verne, Amiens, France

{zhifei.zheng;sami.cherif;rui.sa.shibasaki;chu-min.li}@u-picardie.fr

Résumé

La satisfiabilité diversifiée est un problème d'optimisation combinatoire qui cherche à identifier k modèles satisfaisant une formule CNF donnée tout en maximisant la somme des distances entre chaque paire de modèles. Cet article présente deux nouvelles méthodes exactes de résolution. La première approche s'appuie sur la programmation quadratique en nombres entiers, tandis que la seconde est basée sur une formulation pseudo-booléenne linéaire, pouvant également être adaptée à la satisfiabilité maximum (MaxSAT). Notre évaluation expérimentale approfondie démontre l'efficacité et la performance des deux méthodes proposées sur diverses instances. De plus, pour certaines familles d'instances, les approches proposées parviennent à obtenir des solutions optimales.

Mots-clés

Satisfiabilité diversifiée, Programmation en Nombres Entiers, Satisfiabilité Maximum, Modélisation

Abstract

Diverse Satisfiability is a combinatorial optimization problem that seeks to identify k models satisfying a given CNF formula while maximizing the sum of pairwise Hamming distances between them. This paper introduces two novel exact solution methods. The first approach relies on quadratic integer programming, while the second is based on a linear Pseudo-Boolean formulation, which can also be adapted to Maximum Satisfiability (MaxSAT). Our comprehensive experimental evaluation demonstrates the efficiency and effectiveness of both proposed methods across various benchmark instances. Furthermore, for some instance families, the proposed approaches manage to obtain optimal solutions.

Keywords

Diverse Satisfiability, Integer Programming, Maximum Satisfiability, Modeling

1 Introduction

La satisfiabilité propositionnelle (SAT) est un problème de décision consistant à déterminer si une formule en forme normale conjonctive (CNF) donnée peut être satisfaite par une affectation de variables [6], également appelée modèle de la formule. SAT est le premier à avoir été démontré NP-complet [8], et constitue un paradigme largement utilisé pour la modélisation de nombreux problèmes du

monde réel, notamment en vérification matérielle et logicielle [1, 7], en planification et ordonnancement [24, 25], ou encore en bio-informatique [16, 17]. La version classique du problème SAT vise à trouver un unique modèle pour une formule CNF donnée, mais de nombreuses applications réelles exigent l'identification de plusieurs modèles présentant des propriétés structurelles diverses. Dans ce cas, le fait de s'arrêter au premier modèle ne permet pas de mener à bien l'analyse souhaitée ou risque l'omission d'informations cruciales. Un exemple représentatif qu'on peut citer est celui de la vérification bornée de modèles (Bounded Model Checking, BMC) [1], où chaque modèle de la formule correspond à un chemin de vérification possible. Toutefois, se limiter à une unique solution peut accroître le risque d'omettre des erreurs spécifiques, dans la mesure où celle-ci ne permet pas de capturer l'ensemble des comportements potentiels du système vérifié. Cette limitation met en évidence la nécessité de générer plusieurs solutions présentant une forte diversité structurelle, en vue d'accroître la robustesse des processus de débogage et de vérification.

Dans cette optique, Nadel a introduit le problème de la satisfiabilité diversifiée (Diverse Satisfiability) dans [20]. Étant donné une formule CNF, ce problème consiste à identifier k modèles de la formule maximisant la diversité structurelle. Celle-ci est quantifiée à l'aide de la somme des distances de Hamming entre toutes les paires de modèles sélectionnés, assurant ainsi une dispersion maximale dans l'espace des solutions. Dans ce même travail, l'auteur propose une méthode incomplète s'appuyant sur une heuristique de décision dédiée. Un problème connexe introduit dans la littérature est celui de la distance de Hamming maximale pour les problèmes de satisfaction de contraintes (CSP), initialement formulé par Crescenzi et Rossi dans [9]. Néanmoins, ce problème, bien qu'applicable à des instances à domaines finis, se limite à la recherche de deux modèles distincts. Dès lors, dans le cas binaire, il devient essentiellement équivalent à une instance de la satisfiabilité diversifiée avec $k = 2$.

Étant donné que le problème SAT est déjà NP-complet, le problème de la satisfiabilité diversifiée, qui requiert l'identification de k modèles satisfaisants distincts, représente un défi computationnel encore plus complexe. En particulier, pour un paramètre donné p , Misra et al. [19] démontrent que, même dans le cas restreint où $k = 2$, déterminer deux modèles qui diffèrent exactement (ou au moins) sur p variables demeure un problème NP-difficile. Dans [2], An-

gelsmark et Thapper établissent une borne supérieure de complexité en $O(1.7338^n)$ pour la résolution du problème de distance de hamming maximale sur les formules 2-SAT (c'est-à-dire la satisfiabilité diversifiée avec $k = 2$, où l'instance ne contient que des clauses de longueur 2). Plus généralement, ils montrent que, pour une formule l -SAT où l désigne la longueur maximale des clauses, une borne supérieure en $O((2a)^n)$ peut être obtenue, sous l'hypothèse que l'instance sous-jacente de l -SAT peut être résolue en $O(a^n)$ pour une certaine constante a .

Dans cet article, nous nous focalisons sur l'introduction d'approches exactes pour le problème de satisfiabilité diversifiée, lesquelles, à notre connaissance, n'ont pas encore été explorées dans la littérature. Nous proposons deux approches génériques, l'une reposant sur la programmation entière quadratique (QIP) et l'autre sur la programmation pseudo-boulienne linéaire, cette dernière pouvant être aisément transformée en une instance de satisfiabilité maximum (MaxSAT), l'extension naturelle de SAT en problème d'optimisation. Notre approche QIP permet de formuler la fonction objectif de manière naturelle sous forme quadratique, tandis que les approches linéaires reformulent l'objectif comme une combinaison linéaire d'une série de variables booléennes, s'appuyant sur deux fonctions de pondération : la pondération directe (DW) et la pondération incrémentale (IW). L'efficacité des approches proposées est ensuite évaluée au travers d'expérimentations approfondies.

La suite de cet article est structurée comme suit. La section 2 présente les définitions formelles, les notations ainsi que les fondements théoriques du problème de satisfiabilité diversifiée, en plus d'introduire les paradigmes standards de la programmation entière quadratique, la programmation pseudo-boulienne linéaire, et la satisfiabilité maximum. La section 3 expose nos approches exactes, incluant la formulation quadratique ainsi que les formulations pseudo-bouliennes linéaires et leur transformation en modèles MaxSAT. La section 4 fournit une analyse détaillée des résultats expérimentaux obtenus. Enfin, la section 5 récapitule nos contributions et discute les perspectives de nos travaux.

2 Préliminaires

2.1 Satisfiabilité diversifiée

Soit X un ensemble de variables booléennes prenant des valeurs dans $\{Vrai, Faux\}$ (ou $\{0, 1\}$). Un littéral l est soit une variable $x \in X$, soit sa négation \bar{x} . Le littéral positif $l = x$ est dit satisfait lorsque x est affectée à *Vrai*, tandis que le littéral négatif $l = \bar{x}$ est satisfait lorsque x est affectée à *Faux*. Une clause C est une disjonction (\vee) de littéraux, et peut également être représentée comme un ensemble de littéraux. Plus précisément, on peut écrire une clause sous la forme $C = C^+ \cup C^-$, où C^+ est l'ensemble des littéraux positifs et C^- l'ensemble des littéraux négatifs. De plus, une clause est satisfaite dès lors que l'un de ses littéraux est satisfait. Une formule en forme normale conjonctive (CNF) est une conjonction (\wedge) de clauses, et elle est satisfaite si toutes ses clauses sont sa-

tisfaites. Une formule CNF ϕ composée de m clauses peut être représentée comme un ensemble de clauses, c'est-à-dire $\phi = \{C_1, \dots, C_m\}$. Étant donnée une formule CNF ϕ définie sur les variables de X , le problème de satisfiabilité propositionnelle (SAT) consiste à déterminer s'il existe une affectation $\alpha : X \rightarrow \{Vrai, Faux\}$ qui satisfait ϕ . Lorsqu'une telle affectation existe, on l'appelle modèle de ϕ et on dit que la formule est satisfiable.

Nous introduisons ci-après de manière formelle la notion de distance de Hamming entre deux affectations, qui correspond au nombre de valeurs différentes attribuées aux variables booléennes dans une paire d'affectations. Nous définissons ensuite une notion essentielle à la formulation du problème de satisfiabilité diversifiée, à savoir la mesure de *diversité* pour des ensembles d'affectations.

Définition 1 (Distance de Hamming entre affectations). *Étant donné un ensemble de variables booléennes X et deux affectations α_1, α_2 des variables de X , la distance de Hamming entre α_1 et α_2 , notée $Dis(\alpha_1, \alpha_2)$, est définie comme suit :*

$$Dis(\alpha_1, \alpha_2) = |\{x \in X \mid \alpha_1(x) \neq \alpha_2(x)\}| \\ = \sum_{x \in X} |\alpha_1(x) - \alpha_2(x)|$$

Définition 2 (Diversité). *Étant donné un ensemble de variables booléennes X et un ensemble $A = \{\alpha_1, \dots, \alpha_k\}$ de k affectations des variables dans X , la diversité de A , notée $Div(A)$, est définie comme suit :*

$$Div(A) = \sum_{i=1}^{k-1} \sum_{j=i+1}^k Dis(\alpha_i, \alpha_j) \quad (1)$$

Étant donnée une formule CNF ϕ et un entier $k \geq 2$, le problème de la satisfiabilité diversifiée (Diverse satisfiability, ou Diverse SAT) consiste à trouver un ensemble de k modèles de ϕ avec une diversité maximale. Plus formellement, nous cherchons :

$$A = \operatorname{argmax}_{A \in \{0,1\}^{k \times |X|}} Div(A)$$

Dans [20], Nadel évoque la notion de diversité pour une variable, c'est-à-dire le nombre de ses assignations à *Vrai* multiplié par le nombre de ses assignations à *Faux*, telle que définie formellement ci-dessous. Cette notion lui a servi de critère pour évaluer la contribution d'une seule variable à la diversité, dans l'heuristique proposée par l'auteur. Nous formalisons la correspondance entre la diversité classique et celles des variables dans Proposition 1.

Définition 3 (Diversité des Variables). *Étant donné un ensemble de variables booléennes X et un ensemble $A = \{\alpha_1, \dots, \alpha_k\}$ de k affectations des variables de X , la diversité de la variable $x \in X$ dans A , notée $Div(x, A)$, est définie comme suit :*

$$Div(x, A) = T(x, A) \cdot F(x, A)$$

où :

- $T(x, A) = |\{\alpha \in A \mid \alpha(x) = 1\}|$,
- $F(x, A) = |\{\alpha \in A \mid \alpha(x) = 0\}|$.

Proposition 1. *Étant donné un ensemble de variables booléennes X et un ensemble $A = \{\alpha_1, \dots, \alpha_k\}$ de k affectations des variables dans X , on a :*

$$Div(A) = \sum_{x \in X} Div(x, A) \quad (2)$$

Preuve.

$$\begin{aligned} Div(A) &= \sum_{i=1}^{k-1} \sum_{j=i+1}^k Dis(\alpha_i, \alpha_j) \\ &= \sum_{i=1}^{k-1} \sum_{j=i+1}^k |\{x \in X \mid \alpha_i(x) \neq \alpha_j(x)\}| \\ &= \sum_{i=1}^{k-1} \sum_{j=i+1}^k \sum_{\substack{x \in X \\ \alpha_i(x) \neq \alpha_j(x)}} 1 \\ &= \sum_{x \in X} \sum_{i=1}^{k-1} \sum_{\substack{j=i+1 \\ \alpha_i(x) \neq \alpha_j(x)}}^k 1 \\ &= \sum_{x \in X} T(x, A) \cdot F(x, A) = \sum_{x \in X} Div(x, A) \end{aligned}$$

□

Dans le reste de cet article, nous calculons la diversité des modèles à l'aide de l'équation 2 puisqu'elle permet de représenter plus naturellement le problème dans nos formulations.

2.2 Programmation quadratique en nombres entiers

La programmation quadratique en nombres entiers (Quadratic Integer Programming, QIP) est un paradigme de programmation mathématique avec des variables entières, une fonction objectif quadratique et des contraintes linéaires. Comme défini dans [22], la forme classique d'un modèle QIP est la suivante :

$$\begin{aligned} (QIP) \quad & \max_x x^\top Hx + c^\top x \\ & \text{s.t. } x \in \mathcal{P} \\ & x \in \mathbb{Z}^n \end{aligned}$$

où $H \in \mathbb{Q}^{n \times n}$ est une matrice symétrique, $c \in \mathbb{Q}^n$ et \mathcal{P} est un polyèdre, représenté par $\mathcal{P} = \{x : Ax \leq b\}$ où $A \in \mathbb{Q}^{m \times n}$ et $b \in \mathbb{Q}^m$.

La programmation pseudo-booléenne linéaire (Linear Pseudo-Boolean Programming, LPB) est définie dans le domaine des variables booléennes. Selon [10], elle peut être formulée sous le format suivant :

$$\begin{aligned} (LPB) \quad & \max f(x_1, \dots, x_n) \\ & \text{s.t. } f_j(x_1, \dots, x_n) \geq 0, \quad j \in \{1, \dots, m\} \\ & x \in \{0, 1\}^n \end{aligned}$$

où $f(x_1, \dots, x_n) = \sum_{i=1}^n w_i \cdot x_i$, $w_i \in \mathbb{Z}$.

2.3 Satisfiabilité maximum

La satisfiabilité maximum (MaxSAT) est l'extension naturelle de SAT en problème d'optimisation. Dans cet article, nous utilisons sa version la plus générique, à savoir MaxSAT partiel pondéré [4, 15]. Ce problème prend en entrée une formule bipartite pondérée $\phi = H \cup S$ contenant des variables booléennes dans X . H est l'ensemble des clauses dures qui doivent être satisfaites, comme dans SAT. S est l'ensemble des clauses souples, composé de clauses pondérées (C, W_C) , où W_C est un poids entier positif associé à la clause C . L'objectif de MaxSAT partiel pondéré est d'obtenir une affectation α qui maximise (resp. minimise) la somme des poids des clauses souples satisfaites (resp. falsifiées) tout en satisfaisant toutes les clauses dures. Formellement, soit $cost_\alpha(\phi)$ la somme des poids des clauses souples de ϕ falsifiées par l'affectation α , le problème MaxSAT partiel pondéré consiste ainsi à déterminer $optimum(\phi) = \min_\alpha cost_\alpha(\phi)$.

Dans le reste de l'article, nous utilisons MaxSAT pour faire référence à sa variante partielle pondérée. De plus, la littérature introduit des contraintes pseudo-boléennes (PB) qui peuvent être efficacement encodées sous forme clause [23]. Ces contraintes prennent la forme suivante :

$$\sum_j a_j \cdot l_j \triangleright b$$

où $a_j \in \mathbb{N}$, $b \in \mathbb{N}$, l_j est un littéral et $\triangleright \in \{=, \geq, \leq\}$. Un cas particulier de contrainte PB survient lorsque tous les coefficients des littéraux sont égaux à un et peuvent donc être omis [23]. Ces contraintes sont couramment appelées contraintes de cardinalité et prennent donc la forme la suivante :

$$\sum_j l_j \bowtie b$$

où l_j est un littéral, $b \in \mathbb{N}$ et $\bowtie \in \{=, \leq, \geq\}$.

3 Formulations pour la satisfiabilité diversifiée

Dans cette section, nous présentons trois formulations distinctes du problème de satisfiabilité diversifiée, à savoir une formulation basée sur un modèle de programmation quadratique en nombres entiers (QP) et deux formulations linéaires pseudo-boléennes (DW et IW). Nous commençons par la première formulation QP, présentée en 3.1, qui transforme naturellement le problème d'optimisation initial en une fonction objectif quadratique. En 3.2, nous exposons l'idée générale permettant de faire le lien entre une formulation avec variables entières et objectif quadratique, et une formulation avec variables booléennes et objectif linéaire, conduisant ainsi aux deux modèles linéaires proposés en 3.3. Enfin, la section 3.4 présente la transformation des formulations linéaires pseudo-boléennes vers des formulations MaxSAT.

3.1 Formulation quadratique

Dans cette sous-section, nous présentons une formulation en programmation quadratique en nombres entiers (QIP)

$$\max \sum_{j \in N} O_j \cdot (k - O_j) \quad (\text{QP-0})$$

s.c

$$\sum_{l_j \in C_h^-} (1 - V_{i,j}) + \sum_{l_j \in C_h^+} V_{i,j} \geq 1 \quad \forall (i, h) \in K \times M \quad (\text{QP-1})$$

$$O_j = \sum_{i=\{1, \dots, K\}} V_{i,j} \quad \forall j \in N \quad (\text{QP-2})$$

$$O_j \in \mathbb{N}^+, V_{i,j} \in \{0, 1\} \quad \forall (i, j) \in K \times N \quad (\text{QP-3})$$

FIGURE 1 – Modèle quadratique pour la satisfiabilité diversifiée

pour le problème de satisfiabilité diversifiée. Cette formulation s'appuie naturellement sur la Proposition 1, qui exprime la diversité comme la somme des diversités individuelles des variables. Étant donnée une formule CNF ϕ contenant n variables et m clauses, notre objectif est d'obtenir k modèles maximisant la diversité. Pour cela, nous définissons un modèle QIP à l'aide des deux séries de variables suivantes, où $K = \{1, \dots, k\}$, $N = \{1, \dots, n\}$ et $M = \{1, \dots, m\}$:

- **Variables entières** $O_j \quad \forall j \in N$, qui représentent le nombre d'affectations à *Vrai* de la $j^{\text{ème}}$ variable dans l'ensemble A des k affectations, c'est à dire $T(x_j, A)$.
- **Variables binaire** $V_{i,j} \quad \forall (i, j) \in K \times N$, qui représentent la valeur booléenne de la $j^{\text{ème}}$ variable dans la $i^{\text{ème}}$ affectation $\alpha_i \in A$, c'est à dire $\alpha_i(x_j)$.

Nous obtenons la formulation quadratique (QP) illustrée dans la Figure 1. L'équation QP-0 est une reformulation de Proposition 1, indiquant que l'objectif de maximisation, à savoir la diversité, est égal à la somme des diversités individuelles des variables. L'équation QP-1 garantit que l'ensemble A des k affectations construites constitue bien des modèles de la formule CNF ϕ donnée en entrée. L'équation QP-2 établit le lien sémantique entre les variables booléennes $V_{i,j}$ et les variables entières O_j . Cette formulation requiert $O(k \cdot n)$ variables et $O(k \cdot m + n)$ contraintes.

3.2 De la formulation quadratique aux formulations linéaires

Dans la sous-section précédente, la formulation proposée utilise des variables entières ainsi qu'une fonction objectif quadratique, conformément à la relation établie dans Proposition 1. Toutefois, la présence d'un terme quadratique dans la fonction objectif peut constituer un obstacle important à la résolution efficace du problème. Par conséquent, nous proposons une approche alternative consistant à transformer la formulation QIP en une formulation linéaire pseudo-booléenne. Autrement dit, il s'agit de transformer la fonction objectif quadratique initiale en une combinaison linéaire de variables booléennes.

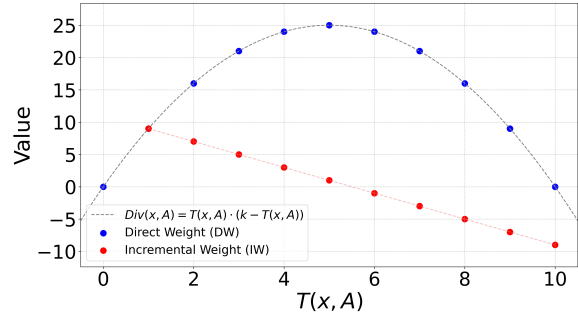


FIGURE 2 – Les fonctions de poids DW et IW pour un scénario avec $k = 10$

Pour chaque variable booléenne d'origine $x_j \in X$ dans la formule CNF ϕ donnée, nous introduisons d'abord une série de variables booléennes auxiliaires destinées à représenter le nombre d'affectations à *Vrai* de x_j à travers les k modèles, soit $T(x_j, A)$. Il convient de souligner que la relation de correspondance entre ces variables auxiliaires et l'entier $T(x_j, A)$ doit être cohérente avec les fonctions de pondération. Nous développons ensuite deux fonctions de pondération distinctes, appelées respectivement *pondération directe* (DW) et *pondération incrémentale* (IW), permettant de reformuler le terme quadratique original $Div(x_j, A) = T(x_j, A) \cdot (k - T(x_j, A))$.

Une illustration de ces deux fonctions de pondération dans un scénario comportant $k = 10$ modèles est présentée dans la Figure 2. La première fonction, DW , associe directement à chaque valeur possible de $T(x_j, A)$ sa contribution correspondante à la diversité. Les points bleus répartis sur la courbe parabolique indiquent que le maximum de diversité est atteint lorsque x_j est assignée à *Vrai* dans exactement la moitié des modèles. La seconde fonction, IW , adopte une approche incrémentale en exprimant $Div(x_j, A)$ comme la somme de contributions marginales. Les points rouges de la Figure 2 illustrent la variation de la valeur de diversité lorsqu'on incrémente $T(x_j, A)$ d'une unité. Il est à noter que ces valeurs incrémentales deviennent négatives au-delà du point médian, traduisant les rendements décroissants associés à une répartition déséquilibrée des affectations à *Vrai*. Formellement, soit $r = T(x_j, A)$, nous définissons la fonction de *pondération directe* (DW) comme suit :

$$\mathcal{D}_r = r \cdot (k - r) \quad \forall r \in \{0, \dots, k\}$$

et la fonction de *pondération incrémentale* (IW) par :

$$\mathcal{I}_r = \mathcal{D}_r - \mathcal{D}_{r-1} = -2r + k + 1 \quad \forall r \in \{1, \dots, k\}$$

On peut remarquer que pour tout $r \in \{1, \dots, k\}$, on a $\mathcal{D}_r = \sum_{r'=1}^r \mathcal{I}_{r'}$. Ces deux fonctions de pondération distinctes offrent donc des approches différentes mais mathématiquement équivalentes pour reformuler la fonction objectif d'origine, comme cela sera détaillé dans la suite.

$$\max \sum_{j=1}^n \sum_{r=0}^k \mathcal{D}_r \cdot U_{j,r} \quad (\text{DW-0})$$

s.c

$$\sum_{l_j \in C_h^-} (1 - V_{i,j}) + \sum_{l_j \in C_h^+} V_{i,j} \geq 1 \quad \forall (i, h) \in K \times M \quad (\text{DW-1})$$

$$\sum_{r \in (\{0\} \cup K)} U_{j,r} = 1 \quad \forall j \in N \quad (\text{DW-2})$$

$$\sum_{r \in (\{0\} \cup K)} r U_{j,r} = \sum_{i \in K} V_{i,j} \quad \forall j \in N \quad (\text{DW-3})$$

$$V_{i,j} \in \{0, 1\}, U_{j,r} \in \{0, 1\} \quad \forall (i, j) \in K \times N \\ \forall (r, j) \in (\{0\} \cup K) \times N \quad (\text{DW-4})$$

FIGURE 3 – Modèle linéaire pour la satisfiabilité diversifiée avec le pondération directe DW .

3.3 Formulations linéaires

3.3.1 Formulation par pondération directe

Nous présentons d'abord la formulation pour le problème de satisfiabilité diversifiée avec la fonction de pondération DW , qui repose sur les variables suivantes, où $K = \{1, \dots, k\}$ et $N = \{1, \dots, n\}$:

- **Variables binaires** $U_{j,r} \quad \forall (r, j) \in (\{0\} \cup K) \times N$, qui représentent le fait que le nombre d'affectations à *Vrai* de la $j^{\text{ème}}$ variable dans l'ensemble des k modèles soit égal à r , c'est à dire $T(x_j, A) = r$.
- **Variables binaire** $V_{i,j} \quad \forall (i, j) \in K \times N$, qui représentent la valeur booléenne de la $j^{\text{ème}}$ variable dans la $i^{\text{ème}}$ affectation $\alpha_i \in A$, c'est à dire $\alpha_i(x_j)$.

Ainsi, la formulation avec la fonction de pondération DW , $\mathcal{D}_r = r \cdot (k - r)$, peut être obtenue comme indiqué dans la Figure 3, où $M = \{1, \dots, m\}$. L'équation [DW-0](#) formule la fonction objectif, où chaque terme $\mathcal{D}_r \cdot U_{j,r}$ indique qu'il y a une contribution à la diversité $\mathcal{D}_r = r \cdot (k - r)$ lorsque la variable x_j est assignée à *Vrai* exactement dans r des k modèles. L'équation [DW-1](#) garantit que chaque modèle α_i satisfait la formule CNF originale ϕ , c'est-à-dire que pour toute clause C_h de ϕ et chaque modèle α_i , au moins un littéral de la clause doit être satisfait selon les assignations de variables dans α_i . L'équation [DW-2](#) impose qu'il y ait un seul nombre d'assignations à *Vrai* de la variable j parmi les k modèles. L'équation [DW-3](#) établit la relation de correspondance entre les variables $U_{j,r}$ et les assignations réelles des modèles $V_{i,j}$. Il y a $O(k \cdot n)$ variables et $O(k \cdot m + n)$ contraintes dans la formulation DW .

3.3.2 Formulation par pondération incrémentale

Dans cette sous-section, nous développons une autre formulation pour le problème de satisfiabilité diversifiée en utilisant la fonction IW . Les variables utilisées dans cette formulation sont les suivantes :

- **Variables binaires** $U_{j,r} \quad \forall (r, j) \in (\{0\} \cup K) \times N$, qui représentent le fait que le nombre d'affectations à *Vrai* de la $j^{\text{ème}}$ variable dans l'ensemble des k af-

$$\max \sum_{j=1}^n \sum_{r=0}^k \mathcal{I}_r \cdot U_{j,r} \quad (\text{IW-0})$$

s.c

$$\sum_{l_j \in C_h^-} (1 - V_{i,j}) + \sum_{l_j \in C_h^+} V_{i,j} \geq 1 \quad \forall (i, h) \in K \times M \quad (\text{IW-1})$$

$$U_{j,r} \leq U_{j,r-1} \quad \forall (r, j) \in (K \setminus \{1\}) \times N \quad (\text{IW-2})$$

$$\sum_{r \in K} U_{j,r} = \sum_{i \in K} V_{i,j} \quad \forall j \in N \quad (\text{IW-3})$$

$$V_{i,j} \in \{0, 1\}, U_{j,r} \in \{0, 1\} \quad \forall (i, j) \in K \times N \\ \forall (r, j) \in K \times N \quad (\text{IW-4})$$

FIGURE 4 – Modèle linéaire pour la satisfiabilité diversifiée avec le pondération incrémentale IW .

fectations A soit supérieure ou égale à r , c'est à dire $T(x_j, A) \geq r$.

- **Variables binaire** $V_{i,j} \quad \forall (i, j) \in K \times N$, qui représentent la valeur booléenne de la $j^{\text{ème}}$ variable dans la $i^{\text{ème}}$ affectation $\alpha_i \in A$, c'est à dire $\alpha_i(x_j)$.

La formulation linéaire avec IW , définie par $\mathcal{I}_r = -2r + k + 1$, est obtenue comme illustré dans la Figure 4. La structure de IW reste similaire à celle de DW . L'équation [IW-0](#) exprime la contribution incrémentale de l'affectation *Vrai* de la variable x_j à la diversité globale. L'équation [IW-1](#) impose la satisfiabilité de l'ensemble des k affectations obtenues. L'équation [IW-2](#) assure l'ordre dans l'utilisation des variables $U_{j,r}$ ce qui, en conjonction avec [IW-3](#), permet de capturer l'information sémantique de $U_{j,r}$. Les complexités en nombre de variables et de contraintes dans la formulation IW sont respectivement de $O(n \cdot k)$ et $O(k \cdot (m + n))$.

3.4 Formulations MaxSAT

Lorsqu'on se focalise sur l'exigence intrinsèque du problème de satisfiabilité diversifiée, on ne peut pas ignorer la tâche centrale consistant à obtenir des modèles en logique propositionnelle, une tâche pour laquelle MaxSAT semble intuitivement être mieux adapté. Dans cette sous-section, nous présentons la transformation des formulations linéaires précédentes vers MaxSAT. Nous reformulons les modèles sous forme clausale, accompagnée de contraintes pseudo-booléennes (PB) ou de cardinalité, comme illustré dans la Figure 5. De plus, nous signalons que l'équation [IW-0*](#) peut contenir des poids négatifs, ce qui n'est pas conforme au standard original de MaxSAT. Elle est donc transformée à l'aide de l'équation suivante :

$$\begin{cases} (U_{j,i}, \mathcal{I}_r) & \mathcal{I}_r \geq 0, \\ (\bar{U}_{j,i}, -\mathcal{I}_r) & \mathcal{I}_r < 0, \end{cases} \quad \forall (r, j) \in K \times N \quad (\text{IW-0})$$

Un exemple illustratif est présenté dans l'exemple suivant. Cette technique de normalisation permet de représenter correctement la fonction de poids incrémentale dans le cadre standard de MaxSAT, qui exige des poids non négatifs pour toutes les clauses souples.

Exemple 1. Si on prend $k = 4$ and $n = 3$, selon la définition $\mathcal{I}_r = -2r + k + 1$, on a :

$$\mathcal{I}_1 = 3, \mathcal{I}_2 = 1, \mathcal{I}_3 = -1, \mathcal{I}_4 = -3$$

Pour chaque variable x_j où $j \in \{1, 2, 3\}$, avant normalisation, les clause souples présentent des poids négatifs selon *IW-0** :

$$(U_{j,1}, 3), (U_{j,2}, 1), (U_{j,3}, -1), (U_{j,4}, -3)$$

La contribution des x_j peut être formulée comme suit :

$$obj_{original} = 3U_{j,1} + U_{j,2} - U_{j,3} - 3U_{j,4}$$

Après normalisation de *IW-0**, on obtient :

$$(U_{j,1}, 3), (U_{j,2}, 1), (\bar{U}_{j,3}, 1), (\bar{U}_{j,4}, 3)$$

avec la contribution suivante :

$$obj_{normalized} = 3U_{j,1} + U_{j,2} + \bar{U}_{j,3} + 3\bar{U}_{j,4}$$

Puisque $\bar{U}_{j,r} = 1 - U_{j,r}$, on peut réécrire les objectifs comme suit :

$$\begin{aligned} obj_{normalized} &= 3U_{j,1} + U_{j,2} + (1 - U_{j,3}) + 3(1 - U_{j,4}) \\ &= 3U_{j,1} + U_{j,2} - U_{j,3} - 3U_{j,4} + 4 \\ &= obj_{original} + 4 \end{aligned}$$

L'ajout d'une constante à toutes les valeurs de la fonction objectif ne modifie pas la solution optimale d'une instance MaxSAT donc la formulation normalisée permet bien d'identifier les mêmes ensembles optimaux de modèles diversifiés.

4 Résultats expérimentaux

Dans cette section, nous présentons les résultats expérimentaux des approches proposées. Le protocole expérimental, incluant l'environnement de test, les instances traitées, ainsi que les solveurs utilisés, est détaillé en 4.1. La section 4.2 présente les performances globales, tandis que la section 4.3 compare les formulations basées sur les deux fonctions de pondération distinctes.

4.1 Protocole expérimental

Tous les tests ont été menés sur la plateforme MatriCS¹, équipée d'un système CentOS 8.6 et d'un processeur Intel Xeon E5-2680 v4 fonctionnant à une fréquence de 2,40 GHz, avec une capacité Turbo Boost allant jusqu'à 3,30 GHz. Le temps limite fixé pour tous les solveurs est de 7200 secondes. Les instances utilisées dans nos expériences proviennent de deux sources : 20 instances issues de la vérification de matériel, telles que proposées et utilisées dans [20], ainsi que 88 instances provenant de SATLIB², pour un total de 108 instances réparties en 8 familles distinctes. La taille de ces instances varie de 61 variables et 300 clauses

Soft Clauses

$$(U_{j,r}, r \cdot (k - r)) \quad \forall (r, j) \in (\{0\} \cup K) \times N \quad (\text{DW-0})$$

Hard Clauses

$$\left(\bigvee_{l_j \in C_h^-} \bar{V}_{i,j} \right) \vee \left(\bigvee_{l_j \in C_h^+} V_{i,j} \right) \quad \forall (i, h) \in K \times M \quad (\text{DW-1})$$

$$\sum_{r \in (\{0\} \cup K)} U_{j,r} = 1 \quad \forall j \in N \quad (\text{DW-2})$$

$$\sum_{r \in (\{0\} \cup K)} r U_{j,r} = \sum_{i \in K} V_{i,j} \quad \forall j \in N \quad (\text{DW-3})$$

(a) MaxSAT-DW

Soft Clauses

$$(U_{j,r}, \mathcal{I}_r) \quad \forall (r, j) \in K \times N \quad (\text{IW-0*})$$

Hard Clauses

$$\left(\bigvee_{l_j \in C_h^-} \bar{V}_{i,j} \right) \vee \left(\bigvee_{l_j \in C_h^+} V_{i,j} \right) \quad \forall (i, h) \in K \times M \quad (\text{IW-1})$$

$$U_{j,r} \rightarrow U_{j,r-1} \quad \forall (r, j) \in (K \setminus \{1\}) \times N \quad (\text{IW-2})$$

$$\sum_{r \in K} U_{j,r} = \sum_{i \in K} V_{i,j} \quad \forall j \in N \quad (\text{IW-3})$$

(b) MaxSAT-IW

FIGURE 5 – Formulations MaxSAT avec DW (a) et IW (b).

à 83908 variables et 276106 clauses. Toutes les instances garantissent l'existence d'au moins 10 modèles distincts.

Les solveurs de l'état de l'art suivants ont été utilisés dans nos expériences afin d'évaluer les méthodologies proposées : CPLEX [11] pour la formulation QIP proposée en 3.1 ainsi que pour les deux formulations pseudo-bouliennes linéaires introduites en 3.3 ; MaxHS [21], WMaxCDCL [14], et OpenWBO [18] pour les deux formulations MaxSAT décrites en 3.4. Nous notons que ces trois solveurs MaxSAT représentent respectivement des approches basées sur la programmation linéaire en nombres entiers (ILP), sur la séparation et évaluation (Branch-and-Bound), et sur les appels aux oracles SAT. Pour réaliser les encodages CNF des contraintes de cardinalité et pseudo-bouliennes, nous avons utilisé la bibliothèque PySAT³ [12, 13], avec les contraintes de cardinalité encodées à l'aide de l'encodage *Cardinality Network* [3], qui permet d'encoder une contrainte de cardinalité de borne \mathcal{K} sur \mathcal{N} littéraux avec une complexité $O(\mathcal{N} \log^2 \mathcal{K})$ en termes de clauses et de variables auxiliaires. Concernant les contraintes pseudo-bouliennes, nous avons utilisé le mode "Best" proposé par PySAT, qui sélectionne automatiquement la méthode d'encodage la plus adaptée.

Pour justifier la non-trivialité des solutions obtenues, nous avons également mis en place une approche naïve par énumération, basée sur le solveur SAT CaDiCaL [5]. Le solveur est exécuté de manière itérative k fois. Après chaque

1. <https://www.matrics.u-picardie.fr>

2. <https://www.cs.ubc.ca/~hoos/SATLIB/benchm.html>

3. <https://pysathq.github.io/>

itération, une nouvelle clause est ajoutée à la formule pour bloquer l'assignation satisfaisante nouvellement trouvée. Une fois les k itérations terminées, le programme est suspendu et la diversité est calculée à partir des k modèles obtenus.

4.2 Performances globales

Les analyses de performance des différentes approches testées à travers l'ensemble des familles, pour les valeurs de k égales à 2, 5 et 10, sont présentées dans les Tableaux 1, 2 et 3. Ces tableaux présentent les performances sous deux aspects : la capacité des approches proposées à résoudre le problème à l'optimalité et la diversité obtenue par celles-ci, comparée à l'énumérateur naïf CaDiCaL. À partir de ces tableaux, il ressort que toutes les formulations proposées parviennent à obtenir des solutions optimales dans certaines familles, notamment pour des valeurs plus petites de k , comme celle de 2. De manière générale, les formulations linéaires avec les fonctions de poids DW et IW , obtiennent le plus grand nombre de solutions optimales pour toutes les valeurs de k . L'écart se renforce lorsque la valeur de k augmente jusqu'à 10, illustrant ainsi une bonne scalabilité de l'approche linéaire pour la résolution du problème.

Pour les solveurs MaxSAT, OpenWBO coopérant avec DW obtient le plus grand nombre d'instances résolues à l'optimalité, soit 89 sur 108, lorsque $k = 2$, tandis que MaxHS coopérant avec IW bascule en première position lorsque $k = 5$ et $k = 10$, résolvant respectivement 67 et 33 instances. WMaxCDCL présente des performances comparables à celles de MaxHS et OpenWBO lorsque $k = 2$ et $k = 10$, bien que ses performances diffèrent des deux autres lorsque $k = 10$.

De plus, on note que, dans plusieurs cas, bien que la formulation QIP avec CPLEX parvienne à atteindre la valeur optimale prouvée par certains autres solveurs, elle échoue à prouver l'optimalité, avec respectivement 7, 21, et 15 instances non prouvées lorsqu'on prend $k = 2$, $k = 5$ et $k = 10$. Dans l'ensemble, ces résultats démontrent que, malgré le fait que toutes les approches proposées sont capables de résoudre de manière optimale le problème de satisfiabilité diversifiée, le choix du solveur et de la fonction de poids a un impact significatif sur les performances, certaines combinaisons montrant des résultats supérieurs en fonction des cas spécifiques.

En termes de diversité, pour $k = 2, 5, 10$, les meilleurs résultats obtenus par un solveur exact est respectivement de 3.58, 2.50, et 3.34 fois meilleure que celle de la solution obtenue par l'énumération naïve avec CaDiCaL. Lorsqu'on restreint l'analyse aux cas avec optimalité prouvée, ces facteurs deviennent respectivement de 113, 63, et 23. L'écart de performance substantiel entre l'énumérateur naïf et les approches exactes proposées démontre clairement la difficulté inhérente du problème, et confirme la nécessité de développer des approches exactes performantes.

On peut également observer qu'avec l'augmentation de la valeur de k , les approches exactes proposées échouent parfois à obtenir des solutions faisables dans le temps imparti. Cela est particulièrement le cas pour WMaxCDCL

et CPLEX et pourrait être dû au fait que l'objectif interne des solveurs exacts est d'obtenir des solutions optimales, de sorte que les résultats intermédiaires incomplets ne reflètent pas pleinement le processus de résolution. Par exemple, l'approche linéaire avec IW résolue avec le solveur CPLEX ne parvient à obtenir que 4 résultats faisables pour la famille difficile "hardware" lorsque $k = 5$, mais tous ces résultats sont optimaux. Malgré cette difficulté, les approches proposées offrent toujours une qualité satisfaisante pour les solutions sous-optimales, ce qui démontre une grande robustesse.

En se focalisant sur la comparaison de la capacité de résolution optimale de toutes les approches exactes proposées, les Figures 6, 7, 8 offrent une vision plus complète du nombre cumulé d'instances résolues par chaque approche dans le temps imparti.

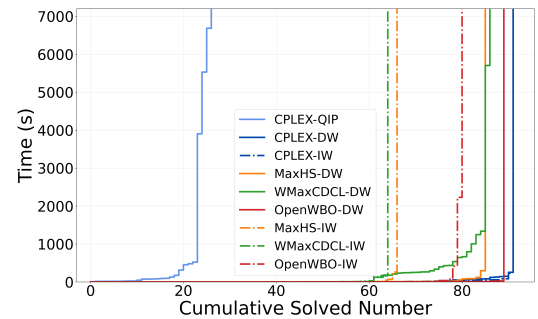


FIGURE 6 – Nombre cumulé d'instances résolues pour $k = 2$ en fonction du temps de résolution en secondes.

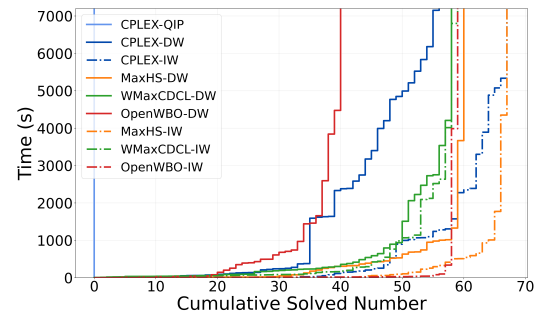


FIGURE 7 – Nombre cumulé d'instances résolues pour $k = 5$ en fonction du temps de résolution en secondes.

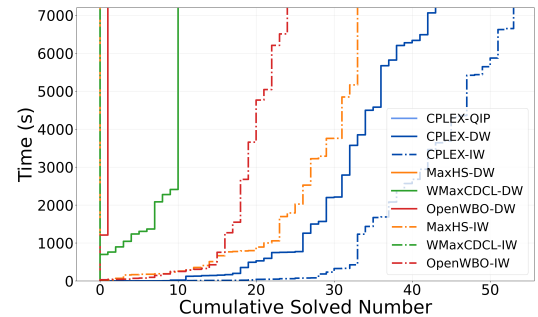


FIGURE 8 – Nombre cumulé d'instances résolues pour $k = 10$ en fonction du temps de résolution en secondes.

Famille	#I	#Var	#CL	Énum.	CPLEX			MaxSAT (DW)			MaxSAT (IW)		
				CaDi.	QIP	DW	IW	MaxHS	WMaxCDCL	OpenWBO	MaxHS	WMaxCDCL	OpenWBO
flat100-	30	155	562	[2]	23(6686)[73]	30(11)[103]	30(8)[103]	30(2)[103]	30(15)[103]	30(2)[103]	30(5)[103]	30(16)[103]	30(6)[103]
flat100+	30	425	1584	[2]	1(6685)[63]	30(1047)[283]	30(730)[283]	30(486)[283]	30(357)[283]	30(50)[283]	30(81)[283]	30(435)[283]	25(51)[235]
ais	4	155	2730	[18]	1(4)[6]	4(196)[34]	4(60)[34]	3(56)[34]	4(583)[34]	4(35)[34]	3(69)[34]	3(171)[22]	3(425)[22]
morphed	20	500	3100	[2]	0(-)[0]	20(65)[200]	20(25)[200]	19(166)[200]	20(9836)[200]	19(21)[199]	1(8)[200]	0(-)[0]	18(46)[197]
logistics	4	1881	11682	[3]	1(5531)[216]	3(58)[365]	3(108)[368]	3(286)[364]	2(443)[218]	2(13)[293]	2(304)[366]	1(28)[218]	2(2302)[285]
hardware	20	49391	161548	[5483]	0(-)[596]	4(249)[1177]	4(195)[1180]	0(-)[18835]	0(-)[0]	4(66)[10052]	0(-)[18381]	0(-)[0]	2(74)[9149]
All	108	9476	31620	[1018]	26(18905)[156]	91(1626)[377]	91(1126)[378]	85(996)[3647]	86(16482)[154]	89(187)[2018]	66(466)[3563]	64(651)[116]	80(2904)[1836]

TABLE 1 – Résultats pour $k = 2$. Pour chaque famille d’instances, le nombre d’instances ($\#I$), le nombre moyen de variables ($\#Var$) et le nombre moyen de clauses ($\#CL$) sont indiqués. Ensuite, la diversité des résultats obtenue par l’approche d’énumération naïve utilisant CaDiCaL (CaDi.) est rapportée. Les performances de l’approche QIP et de l’approche linéaire avec CPLEX ainsi que les trois solveurs MaxSAT, à savoir MaxHS, WMaxCDCL et OpenWBO, sur les formulations avec deux fonctions de poids distinctes (DW, IW) sont présentées sous la forme de $\#OPT(tot_time_{opt})[Div_{all}]$, où $\#OPT$ représente le nombre de solutions optimales obtenues dans le temps limite, tot_time_{opt} est la somme des temps de résolution pour ces solutions optimales et Div_{all} est la diversité moyenne calculée en fonction des meilleures solutions faisables (ou optimales) obtenues pour toutes les instances dans la famille concernée. Dans le cas où aucune solution faisable n’est trouvée dans le temps limite, la diversité est considérée comme étant nulle (égale à 0 comme pénalité) sinon la diversité de la meilleure solution faisable est considérée. Parmi toutes nos approches, celle ayant résolu de manière optimale le plus grand nombre d’instances dans le temps le plus court est indiquée en gras. De plus, la meilleure diversité globale Div_{all} parmi toutes les approches, y compris l’énumération naïve avec CaDiCaL, est soulignée.

Famille	#I	#Var	#CL	Énum.	CPLEX			MaxSAT (DW)			MaxSAT (IW)		
				CaDi.	QIP	DW	IW	MaxHS	WMaxCDCL	OpenWBO	MaxHS	WMaxCDCL	OpenWBO
flat100-	30	155	562	[50]	0(-)[595]	30(2169)[827]	30(81)[827]	30(3558)[827]	30(5311)[827]	20(7597)[817]	30(609)[827]	30(4472)[827]	28(403)[823]
flat100+	30	425	1584	[32]	0(-)[59]	22(56712)[1493]	29(26379)[2160]	29(13603)[2249]	27(25914)[1973]	20(15181)[2148]	30(3023)[2267]	28(25118)[2067]	30(796)[2267]
ais	4	155	2730	[280]	0(-)[50]	1(5834)[119]	1(2348)[124]	1(429)[314]	1(645)[124]	0(-)[304]	1(506)[327]	1(765)[124]	1(3987)[306]
morphed	20	500	3100	[30]	0(-)[87]	0(-)[1977]	3(10458)[1985]	0(-)[1855]	0(-)[0]	0(-)[1197]	6(9311)[2000]	0(-)[0]	0(-)[2000]
logistics	4	1881	11682	[38]	0(-)[0]	0(-)[995]	0(-)[1608]	0(-)[2604]	0(-)[0]	0(-)[1682]	0(-)[2652]	0(-)[0]	0(-)[1782]
hardware	20	49391	161548	[42853]	0(-)[1919]	3(16991)[3850]	4(4427)[3850]	0(-)[82752]	0(-)[0]	0(-)[87145]	0(-)[100503]	0(-)[0]	0(-)[8613]
All	108	9476	31620	[7976]	0(-)[555]	56(81706)[1765]	67(43693)[1974]	60(17589)[16630]	58(31870)[782]	40(22779)[17257]	67(13449)[19952]	59(30354)[808]	59(5185)[2901]

TABLE 2 – Résultats pour $k = 5$. Le format des données est aligné avec Tableau 1.

Famille	#I	#Var	#CL	Énum.	CPLEX			MaxSAT (DW)			MaxSAT (IW)		
				CaDi.	QIP	DW	IW	MaxHS	WMaxCDCL	OpenWBO	MaxHS	WMaxCDCL	OpenWBO
flat100-	30	155	562	[337]	0(-)[1501]	30(14651)[3410]	30(1253)[3410]	0(-)[3224]	10(14085)[1815]	1(1209)[2897]	27(21543)[3410]	0(-)[2144]	17(31749)[3410]
flat100+	30	425	1584	[179]	0(-)[20]	5(18333)[1487]	16(43420)[3905]	0(-)[8323]	0(-)[0]	0(-)[8286]	6(20519)[9350]	0(-)[1531]	7(3270)[9350]
ais	4	155	2730	[1226]	0(-)[0]	0(-)[211]	0(-)[211]	0(-)[1272]	0(-)[0]	0(-)[1230]	0(-)[1366]	0(-)[0]	0(-)[1278]
morphed	20	500	3100	[168]	0(-)[0]	8(45029)[7988]	7(31263)[7993]	0(-)[6548]	0(-)[0]	0(-)[5106]	0(-)[6371]	0(-)[0]	0(-)[8000]
logistics	4	1881	11682	[208]	0(-)[0]	0(-)[0]	0(-)[1942]	0(-)[9230]	0(-)[0]	0(-)[7772]	0(-)[9975]	0(-)[0]	0(-)[7760]
hardware	20	49391	161548	[105280]	0(-)[0]	0(-)[14790]	0(-)[15886]	0(-)[330172]	0(-)[0]	0(-)[34762]	0(-)[42646]	0(-)[0]	0(-)[21896]
All	108	9476	31620	[19724]	0(-)[423]	43(78013)[5586]	53(75936)[6534]	0(-)[65952]	10(14085)[504]	1(1209)[10823]	33(42062)[13042]	0(-)[1021]	24(35019)[9416]

TABLE 3 – Résultats pour $k = 10$. Le format des données est aligné avec Tableau 1.

Lorsque $k = 2$, DW surpasse généralement IW , à l'exception du seul cas où CPLEX- DW et CPLEX- IW parviennent à résoudre rapidement toutes les instances. À mesure que k augmente à 5 et 10, IW montre généralement une meilleure performance que le poids direct. Par ailleurs, on observe que la formulation quadratique (QIP) montre des performances plus faibles par rapport aux approches linéaires. De plus, l'approche linéaire avec CPLEX se comporte de manière comparable aux solveurs MaxSAT uniquement lorsque $k = 5$, mais elle présente un avantage dans tous les autres cas. Ces résultats confirment que nos approches exactes peuvent résoudre efficacement les problèmes de satisfiabilité diversifiée de taille raisonnable, les approches linéaires montrant des résultats particulièrement prometteurs.

4.3 Poids direct vs poids incrémental

Pour analyser les différences entre les deux fonctions de poids proposées, DW et IW , nous menons une comparaison détaillée de leurs performances à travers différentes familles d'instances et configurations de solveurs. Les Figures 9 et 10 présentent les comparaisons des temps d'exécution des deux fonctions de poids pour $k = 2$ et $k = 5$. Chaque point dans les diagrammes de dispersion correspond à une instance, avec ses coordonnées déterminées par le temps d'exécution pour les deux fonctions de poids. Lorsqu'une instance n'a pas été résolue dans le temps imparti, la limite du temps d'exécution est reportée. Les points situés en dessous de la diagonale indiquent les instances où IW réalise de meilleures performances, tandis que les points au-dessus de la diagonale sont en faveur de DW .

Nous observons que les distributions des points suivent des tendances distinctes lorsque la valeur de k varie. Lorsque $k = 2$, DW montre généralement des performances supérieures, avec 33.33%, 62.96%, 95.37%, 92.59% des points situés au-dessus ou sur la ligne diagonale respectivement pour les solveurs CPLEX, MaxHS, WMaxCDCL, OpenWBO. À mesure que k augmente à 5, nous observons une tendance où IW surpasse DW , avec des pourcentages de points de 84.26%, 89.81%, 81.48%, 72.22% situés sous ou sur la ligne diagonale respectivement pour les solveurs CPLEX, MaxHS, WMaxCDCL et OpenWBO. Dans le cas de $k = 10$, nous omettons les figures correspondantes car toutes les approches, à l'exception des modèles linéaires avec CPLEX, n'ont pas réussi à résoudre de nombreuses instances, mais nous observons clairement dans le Tableau 3 que IW obtient les meilleurs résultats. Enfin, sur la base des résultats obtenus pour toutes les valeurs de k , nous constatons l'avantage de IW par rapport à DW à mesure que k augmente.

5 Conclusion

Dans cette étude, nous avons proposé des formulations exactes pour le problème de la satisfiabilité diversifiée, en utilisant la programmation entière quadratique (QIP) ainsi que des modèles de programmation linéaire pseudo-bouléenne, pouvant être adaptées en formulations MaxSAT. Ces approches permettent de formuler l'objectif de la satis-

fiabilité diversifiée sous l'angle de la diversité des variables. L'approche QIP présente une formulation naturelle de l'objectif sous forme quadratique, tandis que les formulations linéaires et MaxSAT reposent sur deux fonctions de poids distinctes, exploitant pleinement les capacités des solveurs modernes à résoudre des problèmes d'optimisation linéaire complexes. Les résultats expérimentaux, menés sur un large éventail d'instances avec différentes valeurs du paramètre k , représentant le nombre de modèles, démontrent l'efficacité et la robustesse des méthodologies proposées.

Dans le cadre de nos travaux futurs, nous envisageons de concevoir des algorithmes dédiés pour la satisfiabilité diversifiée qui dépassent les limitations des schémas de résolution globaux tout en conservant la structure et la représentation de l'objectif proposées dans nos approches actuelles. Par ailleurs, à mesure que nous élargissons nos expériences à un éventail d'instances plus large, nous espérons établir de nouvelles solutions optimales, afin de soutenir les chercheurs intéressés par ce domaine. Ces repères élargis faciliteront également la conception et l'évaluation d'algorithmes heuristiques et d'approximation pour le problème de la satisfiabilité diversifiée.

Remerciements

Ce travail est partiellement soutenu par le projet ANR-24-CE23-6126 (BforSAT) et par la chaire IA ANR-19-CHIA-0013-01 (MASSAL'IA) co-financée par l'agence nationale de recherche et le gestionnaire du réseau de distribution d'électricité Enedis. Il a bénéficié d'un accès aux ressources HPC de la « Plateforme MatriCS » de l'Université de Picardie Jules Verne qui est cofinancée par l'Union Européenne avec le Fonds Européen de Développement Régional (FEDER) et le Conseil Régional des Hauts-De-France entre autres.

Références

- [1] S. Agbaria, D. Carmi, O. Cohen, D. Korchemny, M. Lifshits, and A. Nadel. Sat-based semiformal verification of hardware. In R. Bloem and N. Sharygina, editors, *FMCAD 2010*, pages 25–32. IEEE, 2010.
- [2] O. Angelsmark and J. Thapper. Algorithms for the maximum hamming distance problem. In B. Faltings, A. Petcu, F. Fages, and F. Rossi, editors, *CSCLP 2004*, volume 3419 of *LNCS*, pages 128–141. Springer, 2004.
- [3] R. Asín, R. Nieuwenhuis, A. Oliveras, and E. Rodríguez-Carbonell. Cardinality networks and their applications. In O. Kullmann, editor, *SAT 2009*, volume 5584 of *LNCS*, pages 167–180. Springer, 2009.
- [4] F. Bacchus, M. Jarvisalo, and R. Martins. Maximum satisfiability. In *Handbook of satisfiability*, pages 929–991. IOS Press, 2021.
- [5] A. Biere, K. Fazekas, M. Fleury, and M. Heisinger. CaDiCaL, Kissat, Paracooba, Plingeling and Treengeling entering the SAT Competition 2020. In T. Balyo, N. Froleyks, M. Heule, M. Iser, M. Jarvisalo, and M. Suda, editors, *Proc. of SAT Competition 2020 – Solver and Benchmark Descriptions*, volume B-2020-1 of *Department of Computer Science Report Series B*, pages 51–53. University of Helsinki, 2020.

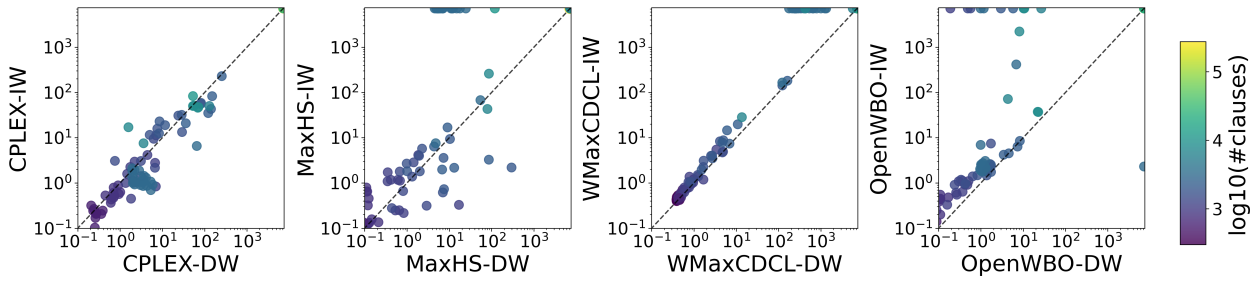


FIGURE 9 – Comparaison des temps d’exécution en échelle logarithmique des deux fonctions de poids pour $k = 2$.

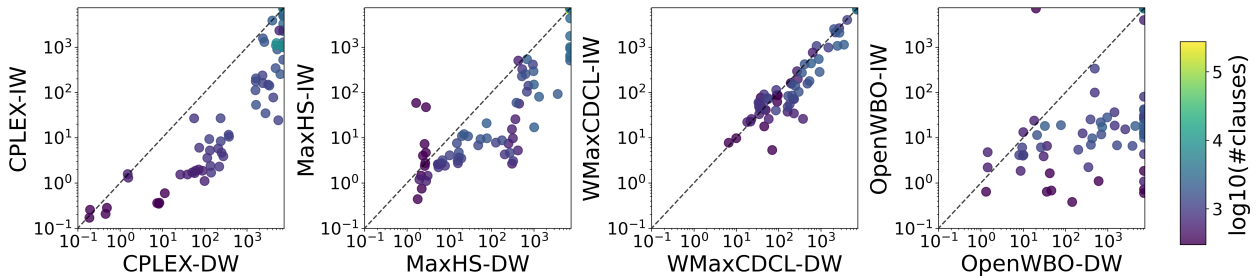


FIGURE 10 – Comparaison des temps d’exécution en échelle logarithmique des deux fonctions de poids pour $k = 5$.

- [6] A. Biere, M. Heule, H. van Maaren, and T. Walsh, editors. *Handbook of Satisfiability - Second Edition*, volume 336 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2021.
- [7] E. M. Clarke, D. Kroening, and F. Lerda. A tool for checking ANSI-C programs. In K. Jensen and A. Podolski, editors, *TACAS 2004*, volume 2988 of *LNCS*, pages 168–176. Springer, 2004.
- [8] S. A. Cook. The complexity of theorem-proving procedures. In M. A. Harrison, R. B. Banerji, and J. D. Ullman, editors, *STOC 1971*, pages 151–158. ACM, 1971.
- [9] P. Crescenzi and G. Rossi. On the hamming distance of constraint satisfaction problems. *Theor. Comput. Sci.*, 288(1):85–100, 2002.
- [10] P. Hammer Ivănescu and S. Rudeanu. Boolean methods in operations research and related areas. *Oekonometrie und Unternehmensforschung* (, 7, 1968.
- [11] IBM Corporation. *IBM ILOG CPLEX Optimization Studio*, 2022.
- [12] A. Ignatiev, A. Morgado, and J. Marques-Silva. PySAT : A Python toolkit for prototyping with SAT oracles. In *SAT*, pages 428–437, 2018.
- [13] A. Ignatiev, Z. L. Tan, and C. Karamanos. Towards universally accessible SAT technology. In *SAT*, pages 4 :1–4 :11, 2024.
- [14] C. Li, Z. Xu, J. Coll, F. Manyà, D. Habet, and K. He. Boosting branch-and-bound maxsat solvers with clause learning. *AI Commun.*, 35(2):131–151, 2022.
- [15] C. M. Li and F. Manyà. MaxSAT, Hard and Soft Constraints. In A. Biere, M. Heule, H. van Maaren, and T. Walsh, editors, *Handbook of Satisfiability - Second Edition*, volume 336 of *Frontiers in Artificial Intelligence and Applications*, pages 903–927. IOS Press, 2021.
- [16] I. Lynce and J. Marques-Silva. SAT in bioinformatics : Making the case with haplotype inference. In A. Biere and C. P. Gomes, editors, *SAT 2006*, volume 4121 of *LNCS*, pages 136–141. Springer, 2006.
- [17] P. Manolios, M. G. Oms, and S. O. Valls. Checking pedigree consistency with PCS. In O. Grumberg and M. Huth, editors, *TACAS 2007*, volume 4424 of *LNCS*, pages 339–342. Springer, 2007.
- [18] R. Martins, V. Manquinho, and I. Lynce. Open-wbo : A modular maxsat solver,. In C. Sinz and U. Egly, editors, *SAT 2014*, volume 8561 of *LNCS*, pages 438–445. Springer, 2014.
- [19] N. Misra, H. Mittal, and A. Rai. On the parameterized complexity of diverse sat. *arXiv preprint arXiv :2412.09717*, 2024.
- [20] A. Nadel. Generating diverse solutions in SAT. In K. A. Sakallah and L. Simon, editors, *SAT 2011*, volume 6695 of *LNCS*, pages 287–301. Springer, 2011.
- [21] A. Niskanen, J. Berg, and M. Järvisalo. Maxhs in maxsat evaluation 2022. *MaxSAT Evaluation 2022*, page 35, 2022.
- [22] A. D. Pia, S. S. Dey, and M. Molinaro. Mixed-integer quadratic programming is in np. *Mathematical Programming*, 162 :225–240, 2017.
- [23] O. Roussel and V. Manquinho. Pseudo-boolean and cardinality constraints. In A. Biere, M. Heule, H. van Maaren, and T. Walsh, editors, *Handbook of Satisfiability - Second Edition*, volume 336 of *Frontiers in Artificial Intelligence and Applications*, pages 1087–1129. IOS Press, 2021.
- [24] P. Surynek. Lazy compilation of variants of multi-robot path planning with satisfiability modulo theory (SMT) approach. In *IROS 2019*, pages 3282–3287. IEEE, 2019.
- [25] H. Xu, R. A. Rutenbar, and K. A. Sakallah. sub-sat : a formulation for relaxed boolean satisfiability with applications in routing. In S. S. Sapatnekar and M. Pedram, editors, *ISPD 2002*, pages 182–187. ACM, 2002.

Activité AFIA

1^{er} août 2023 – 31 juillet 2025

Éditeurs : Conseil d'Administration de l'AFIA – Année 2025

Tables des matières

À propos de l'AFIA	III
Conseil d'Administration 2024	V
Conseil d'Administration 2025	V
Collège Création d'Événements Collaboratifs, Inclusifs et Ludiques en IA	VII
Collège Industriel	IX
Collège Humanités, Société et Intelligence Artificielle	XI
Collège Représentation et Raisonnement	XIII
Collège Science de l'Ingénierie des Connaissances	XV
Collège Systèmes Multi-Agents et Agents Autonomes	XVII
Collège Technologies du Langage Humain	XIX
Prix de Thèse IA 2024	XXI
Prix de Thèse IA 2025	XXI
Bulletins	XXIII
Journée thématique FIIA 2023	XXV
Journée thématique PDIA 2024	XXVI
Journée thématique FIIA 2025	XXVII
Journée thématique PDIA 2025	XXVIII
Journée thématique EFIA 2025	XXIX
Journée commune Modèles hybrides & IA 2023	XXX
Journée commune EGC & IA 2024	XXXI
Journée commune Humanités Numériques & IA 2024	XXXII
Journée commune Santé & IA 2024	XXXIII
Journée commune Société & IA 2024	XXXV
Journée commune Agent & IA 2024	XXXVII
Journée commune Société & IA 2025	XXXVIII
Journée commune MAFTEC-12 2025	XXXIX
Journée commune GAS 2025	XLI
Journée commune Connaissances, Décisions & IA 2025	XLII
Plate-forme Intelligence Artificielle 2024	XLIV
Plate-forme Intelligence Artificielle 2025	XLV

À propos de l'AFIA

L'objet de l'**AFIA**, Association Loi 1901 sans but lucratif, est de promouvoir et de favoriser le développement de l'Intelligence Artificielle (IA) sous ses différentes formes, de regrouper et de faire croître la communauté française en IA et, à la hauteur des forces de ses membres, d'en assurer la visibilité.

L'**AFIA** anime la communauté par l'organisation de grands rendez-vous. Se tient ainsi chaque été une semaine de l'IA, la Plate-forme IA (**PFIA 2023** à Strasbourg, **PFIA 2024** à La Rochelle, **PFIA 2025** à Dijon) au sein de laquelle se tiennent la Conférence Nationale d'Intelligence Artificielle (**CNIA**), les Rencontres des Jeunes Chercheurs en IA (**RJCIA**) et la Conférence sur les Applications Pratiques de l'IA (**APIA**) ainsi que des conférences/journées thématiques hébergées qui évoluent d'une année à l'autre, sans récurrence obligée.

Ainsi, **PFIA 2025** a hébergé du 30 juin au 4 juillet 2025 à Dijon, outre la 28^e **CNIA**, les 23^{es} **RJCIA** et la 11^e **APIA** : les 6 conférences **CAP**, **IC**, **JFPC**, **JFSMA**, **JIAF** et **SFC**, 4 journées thématiques (**Défense & IA**, **Humanité & IA**, **Société & IA**, **IA en BFC**), 6 ateliers thématiques (**ACAI**, **GDR RADIA**, **IN-OVIVE**, **Jeux & IA**, **MAFTEC**, **Santé & IA**), et plusieurs **tutoriels hébergés**.

Forte du soutien de ses 497 adhérents à jour de leur cotisation en juillet 2025, l'**AFIA** assure :

- le maintien d'un site Web dédié à l'IA reproduisant également les **Brèves** de l'IA ;
- une *journée industrielle* « Forum Industriel en IA » (**FIIA 2025**) ;
- une *journée recherche* « Perspectives et Défis en IA » (**PDIA 2024**) ;
- une *journée enseignement* « Enseignement et Formation en IA » (**EFIA 2023**) ;
- une « École Saisonnière en IA » (**ESIA2025**, précédente édition en **2023**) ;
- la remise annuelle d'un **prix de thèse** en IA ;
- le soutien à 7 collèges ayant leur propre activité :
 - collège **Création d'Événements Collaboratifs, Inclusifs et Ludiques en IA** (octobre 2021),
 - collège **Humanités, Société et Intelligence Artificielle** (janvier 2025),
 - collège **Industriel** (janvier 2016),
 - collège **Représentation et Raisonnement** (avril 2017),
 - collège **Science de l'Ingénierie des Connaissances** (avril 2016),
 - collège **Systèmes Multi-Agents et Agents Autonomes** (janvier 2017),
 - collège **Technologies du Langage Humain** (juillet 2019) ;
- la parution trimestrielle des **Bulletins** de l'**AFIA** ;
- un lien entre ses membres et sympathisants sur les réseaux sociaux **LinkedIn**, **Facebook** et **Twitter** ;
- le **parrainage** scientifique, mais aussi éventuellement financier, d'événements en IA ;
- la diffusion mensuelle de **Brèves** sur les actualités de l'IA en France (**abonnement** ou **envoi** à la liste) ;
- la réponse aux consultations officielles ou officieuses (Ministères, Missions, Organismes) ;
- la réponse aux questions de la presse, écrite ou orale, également sur internet ;
- la divulgation d'offres de **collaborations**, de **formations**, d'**emploi**, de **thèses** et de **stages**.

L'**AFIA** organise aussi des **journées communes** avec d'autres associations. Pour 2025 : **Société & IA** avec **AFIHM**; **MAFTEC 12** avec **GDR RADIA**.

Enfin, l'**AFIA** encourage la participation de ses membres aux grands événements de l'IA, dont **PFIA**. Ainsi, les membres de l'**AFIA**, pour leur inscription à **PfIA**, bénéficient d'une réduction équivalente à deux fois le coût de leur adhésion, leur permettant d'assister à **PFIA 2025** sur 5 jours au tarif de 138€ TTC !

Rejoignez-nous vous aussi et **adhérez** à l'**AFIA** pour contribuer au développement de l'IA en France. L'adhésion peut être individuelle ou au titre de personne morale. Merci également de susciter de telles adhésions en diffusant ce document autour de vous !

**Afia**Association française
pour l'Intelligence Artificielle

Conseil d'Administration 2024

Président	Benoit LE BLANC	GT Bureau, Prix Thèse, International, Représentation
V.-Président	Thomas GUYET	GT Bureau, GT Plateforme, Représentation
Trésorière	Isabelle SESÉ	GT Bureau, Adhésions, Partenariats
Secrétaire	Grégory BONNET	GT Bureau, GT Rédaction, Adhésions, Prix Thèse
Porte-parole	Emmanuel ADAM	GT Communication, Brèves, Réseaux
Rédacteur	Dominique LONGIN	GT Rédaction, Collections HAL
Webmestre	Catherine ROUSSEY	GT Web & Mail, Réseaux
	Azzedine BENABBOU	Invité, GT Web & Mail
	Zied BOURAOU	GT Collèges
	Gayo DIALLO	Collège Industriel
	Bernard GEORGES	Collège Industriel
	Domitile LOURDEAUX	GT Collèges
	Frédéric MARIS	GT Enseignement
	Davy MONTICOLO	GT Plateforme, GT Journées
	Jose MORENO	Invité, GT Plateforme
	Gauthier PICARD	GT Enseignement
	Valérie REINER	Collège Industriel
	Céline ROUVEIROL	Collège Industriel
	Fatiha SAIS	GT Journées
	Ahmed SAMET	GT Plateforme, GT Journées, GT Enseignement

Conseil d'Administration 2025

Président	Benoit LE BLANC	GT Bureau, Prix Thèse, International, Représentation
V.-Président	Thomas GUYET	GT Bureau, GT Plateforme, Représentation
Trésorière	Isabelle SESÉ	GT Bureau, Adhésions, Partenariats
Secrétaire	Grégory BONNET	GT Bureau, GT Rédaction, Adhésions, Prix Thèse
Porte-parole	Pierre MONNIN	GT Communication, Brèves, Réseaux
Rédacteur	Dominique LONGIN	GT Rédaction, Collections HAL
Webmestre	Jean-Guy MAILLY	GT Web & Mail, Réseaux
	Azzedine BENABBOU	Invité, GT Web & Mail
	Zied BOURAOU	GT Collèges
	Gayo DIALLO	Collège Industriel
	Domitile LOURDEAUX	GT Collèges
	Davy MONTICOLO	GT Plateforme, GT Journées
	Jose MORENO	GT Enseignement
	Gauthier PICARD	GT Enseignement
	Valérie REINER	Collège Industriel
	Céline ROUVEIROL	Collège Industriel
	Fatiha SAIS	GT Journées
	Ahmed SAMET	GT Plateforme, GT Journées, GT Enseignement
	Thierry VIDAL	Parrainage

Collège Création d'Événements Collaboratifs, Inclusifs et Ludiques en IA

Objectif du collège

Le Collège [Création d'Événements Collaboratifs, Inclusifs et Ludiques en IA](#) (CECILIA) de l'AFIA défend l'apprentissage de l'IA grâce à la pratique coopérative et l'expérimentation. Il a deux objectifs :

- mettre à disposition des ressources pour l'Intelligence Artificielle par la pratique, en particulier pour les étudiant(e)s/lycéen(ne)s (par exemple lors de la Nuit de l'info) ;
- favoriser les rencontres dans la communauté IA au travers d'événements instructifs, ludiques et conviviaux (par exemple lors de PFIA).

Ce collège a repris les activités du Collège Compétition 2018-2020, pour les étendre à d'autres activités comme des ateliers inspirés des *game jams* pour être plus inclusives en termes de public et d'approches scientifiques. Ainsi le collège CECILIA a organisé un atelier « Jam Création de textes poétiques ou drôles » lors de PFIA'2022 et un atelier « Jam DriveToGæther » lors de PFIA'2023.

Pour mémoire, le Collège Compétition a organisé pour l'AFIA les compétitions et animations : « IA sur Robot » lors de RFIA'16 à Clermont-Ferrand, et « BotContest » lors de PFIA'17 à Caen, « DriveToGæther » lors de PFIA'19.

Programme de travail

À ce jour, les événements identifiées par le collège sont les suivants :

- **Nuit de l'Info.** Les participantes et les participants à la Nuit de l'Info répondent en une nuit sur un défi national de programmation, tout en relevant divers défis connexes. Leur programme et un document doivent être remis le lendemain matin avant 8h. Ces résultats sont évalués par un jury national et les jurys des défis connexes. Cette compétition a lieu tous les ans en décembre et l'AFIA y participe en y proposant un défi : intégrer de l'IA dans l'application développée. Le jury de ce défi se compose de membres du collège et d'autres personnes de l'AFIA.
- **IA sur Robots.** Le but de ce défi est de mettre en évidence l'IA au sein d'une ou plusieurs plateformes robotiques, dans un scénario figé avec des règles, et une mesure des performances. Ce défi est ouvert à toutes et tous : personnels travaillant dans la recherche, l'enseignement, l'ingénierie, étudiantes et étudiants ainsi que le grand public. Ce défi s'intitule « DriveToGæther » et dispose d'un scénario et d'un règlement. Les projets sélectionnés seront testés par le public (par exemple lors de PFIA). Un de nos souhaits est de formaliser le règlement et les supports de cet événement pour le rendre facilement organisable par des tierces organisations locales (lycées, clubs, etc.).
- **IA et Créativité.** Le but de cet événement est de faire expérimenter au public le potentiel créatif, l'originalité, la performance stratégique et/ou l'adaptabilité des méthodes d'IA. Il se déroule sur une ou plusieurs soirées (par exemple lors de PFIA), dans un cadre coopératif et festif.
- Un rassemblement annuel des membres du collège pour revenir sur les activités organisées (par exemple lors de PFIA).

Comité de pilotage

Le comité de pilotage se réunit au moins mensuellement et plus avant les événements.

- Anne-Gwenn BOSSER, École Nationale d’Ingénieurs de Brest, Lab-STICC ;
- Florence DUPIN DE SAINT-CYR, Université Toulouse 3 Paul Sabatier, IRIT ;
- Liana ERMAKOVA, Université de Bretagne Occidentale, Brest, HCTI ;
- Thomas GUYET, Inria, Lyon ;
- Philippe MORIGNOT, chercheur indépendant, Paris
- Nicolas PÉPIN-HERMANN, BA Healthcare, Rennes.

Un comité d’organisation est déterminé en fonction des événements, et intègre des membres du comité de pilotage et des membres de comité d’organisation de l’événement. Nous projetons de lancer un appel à participation pour constituer un comité consultatif afin de fédérer une communauté d’actrices et d’acteurs intéressés par l’IA pour tous (responsables d’enseignement ou de FabLab, etc.).

Contacts

Coordinateur du collège : Florence DUPIN DE SAINT-CYR

Adresse de contact : collegececilia-ca@googlegroups.com

Un membre du comité de pilotage est membre du conseil d’administration de l’AFIA :

- Thomas GUYET, thomas.guyet@afia.asso.fr.

Références

- Florence Dupin de Saint-Cyr, Nicolas Yannick Pepin, Nassim Mokhtari, Philippe Morignot, Julien Vianey, Anne-Gwenn Bosser, Liana Ermakova (2024) *DriveToGaether: a Turnkey Collaborative Robotic Event Platform*, 16th Int. Conf. on Agents and Artificial Intelligence (ICAART 2024), SciTePress, pages 404-411, doi : 10.5220/0012463800003636.
- Anne-Gwenn Bosser, Liana Ermakova, Florence Dupin de Saint-Cyr, Pierre De Loor, Victor Charpenay, Nicolas Yannick Pépin, Benoît Alcaraz, Jean-Victor Autran, Alexandre Devillers, Juliette Grosset, Aymeric Hénard, Florian Marchal (2022) *Poetic or Humorous Text Generation: Jam Event at PFIA2022* 13th Conf. and Labs of the Evaluation Forum (CLEF 2022), CEUR Workshop Proceedings, CEUR-WS.org, vol 3180, pages 1719-1726.

(Version de juillet 2024)

Collège Industriel

Objectif du collège

L'objet du Collège Industriel (CI) de l'AFIA est de favoriser les échanges en France dans le domaine de l'IA entre sa composante industrielle et sa composante académique ainsi que diverses actions de promotion de l'IA. Le rejoindre c'est, pour une société, en plus des bénéfices accordés à toutes les personnes morales de l'AFIA (pointeur vers site Web adhésions) :

- accroître la visibilité du CI de l'AFIA ;
- pouvoir faire état de ses relations académiques et leurs recherches de partenariat académique sur des problématiques ciblées ;
- proposer aux collèges thématiques des actions intéressant les membres du CI ;
- pouvoir discuter avec les autres sociétés et adhérents au CI de problématiques dans le domaine de l'IA, et partager des solutions en garantissant la confidentialité des échanges ;
- promouvoir l'IA auprès des décideurs et dirigeants industriels ;
- contribuer à équilibrer tous les domaines de l'IA et leurs hybridations ;
- témoigner auprès des collèges thématiques de cas d'usage qui intéressent le CI ;
- témoigner de l'apport de l'IA dans l'industrie lors d'événements AFIA (FIIA, tutoriels). ;
- faire du lobbying au niveau français auprès des Ministères, des pôles de compétitivité, de l'ANR et tout autre organisme, également au niveau européen ;
- faire connaître aux académiques ses besoins en recrutement ;
- offrir des opportunités à la communauté académique de valoriser leurs formations.

En outre, les sociétés membres du CI à jour de leur cotisation mensuelle au printemps apparaissent comme partenaires de la Plateforme Intelligence Artificielle de la même année.

Programme de travail

En délégation du CA de l'AFIA, le programme de travail du CI consiste à :

- contribuer au pilotage d'événements annuels à forte visibilité, le Forum Industriel de l'IA (FIIA) et la Conférence sur les Applications Pratiques de l'IA (APIA) ;
- cartographier les relations académiques et industrielles (services du 1^{er} Ministre, MA, MC, MEAE, MESRI, MI, MINEF, MJ, MS, MTES, ALLISTENE, CNRS, IMT, INRAE, INRIA, ONERA + IRT) ;
- solliciter les collèges thématiques de l'AFIA pour des contacts ou des interventions ;
- solliciter les collèges thématiques de l'AFIA pour des partenariats de projet ;
- organiser des réunions régulières au sein du collège ;
- organiser des réunions avec invités externes ;
- coprogrammer le prochain *AI Summit France* ;
- diffuser des bulletins ou des dossiers du collège en français avec résumés en anglais ;
- motiver les facilités d'accès à toutes les approches d'IA et leurs hybridations dans les formations, open-sourcer.

Ce programme est complété en début d'année civile par les membres du collège. Le CI se réunit mensuellement pour coordonner les avancées sur les actions engagées par le collège et en décider d'autres, échanger sur un sujet particulier et/ou sur l'actualité en IA sur le mois écoulé. Les réunions du CI font l'objet de comptes rendus qui sont distribués à ses membres et au CA de l'AFIA.

Comité de pilotage

Le CI est composé de l'ensemble des sociétés s'étant acquittées des droits d'adhésion pour l'année en cours et d'au moins deux académiques membres du CA de l'AFIA. Son comité de pilotage est constitué de dix personnes physiques, dont au moins : le coordinateur du CI, le responsable de la feuille de route et le responsable des séminaires. Ces responsables sont désignés pour une durée d'une année par les membres du CI lors de la première réunion annuelle. En 2022, le Collège Industriel est composé de :

- Bruno CARRON et Frédéric PERLANT, AIRBUS, Elancourt ;
- Alain BERGER, ARDANS, Montigny-le-Bretonneux ;
- Mustapha DERRAS, Youssef MILOUDI et Valérie REINER, BERGER-LEVRAULT, Boulogne Billancourt ;
- Stéphane DURAND et Bruno PATIN, DASSAULT Aviation, Saint-Cloud ;
- Pierre FEILLET et Christian DE SAINTE-MARIE, IBM FRANCE, Gentilly ;
- Ghislain ATEMEZING et Christophe PRIGENT, MONDECA, Paris ;
- Jean-Pierre DESMOULINS, Jean-Baptiste FANTUN et Véronique VENTOS, NUKKAI, Paris ;
- Julien BOHNE, Bernard GEORGES et Christelle LAUNOIS, SOCIETE GENERALE, Val de Fontenay ;
- Patricia BESSON, Juliette MATTIOLI et David SADEK, THALES, Palaiseau ;
- Yves DEMAZEAU et Céline ROUVEIROL, CA AFIA, Grenoble et Paris.

Contacts

Coordinateur et Responsable de la feuille de route : yves.demazeau@afia.asso.fr.

Responsable des séminaires : valerie.reiner@berger-levrault.com.

Quatre membres du comité de pilotage sont membres du conseil d'administration de l'AFIA :

- Yves DEMAZEAU, yves.demazeau@afia.asso.fr ;
- Bernard GEORGES, bernard.georges.777@gmail.com ;
- Valérie REINER, valerie.reiner@berger-levrault.com ;
- Céline ROUVEIROL, celine.rouverirol@afia.asso.fr.

(Version de juillet 2024)

Collège Humanités, Société et Intelligence Artificielle

Objectif du collège

Le collège [Humanités, Société et Intelligence Artificielle](#) (HumaniSIA) de l'[AFIA](#) s'est donné pour mission de développer une communauté de chercheuses et chercheurs venant de tous horizons et s'intéressant à la convergence des différents domaines de recherche autour des aspects éthiques, sociaux et environnementaux de l'intelligence artificielle.

Ce collège organise et aide à l'organisation de conférences autour des interactions entre l'intelligence artificielle et la société. Il participe également aux travaux des autres collèges de l'AFIA et contribue à la diffusion des connaissances au profit du grand public. Il contribue à la mise en avant de jeunes chercheuses et chercheurs et à leur formation, en les encourageant à participer aux activités de ce collège et plus largement à celles de l'AFIA.

Ce collège s'intéresse plus particulièrement aux risques qui peuvent se poser en matière d'environnement et de gestion des ressources fondamentales, d'inégalités, d'emploi, de discriminations et d'exclusion de tout ordre qu'un usage maladroît ou volontairement néfaste de l'intelligence artificielle pourrait avoir sur nos sociétés et nos démocraties.

Programme de travail

Pour la période 2024 – 2026, le programme du collège HumaniSIA est prioritairement dédié à la constitution de la communauté HumaniSIA à l'intérieur comme à l'extérieur de l'AFIA.

À cet effet, nous participerons activement à l'intérieur de l'AFIA :

- à la Plate-Forme Intelligence Artificielle (PFIA) regroupant une fois par an les chercheuses et chercheurs en IA et que nous nous attacherons à faire connaître auprès des autres disciplines.
- aux journées thématiques organisées par les collèges de l'AFIA intéressés par notre démarche.

À l'extérieur de l'AFIA, nous avons l'intention de développer des coopérations avec des sociétés savantes proches de l'AFIA pour contribuer à faire connaître l'AFIA auprès des chercheur.se.s d'autres disciplines et à en développer l'influence. Les premières coopérations envisagées le sont avec :

- l'association [Évolution Artificielle](#) ;
- l'Association Francophone d'Interaction Humain-Machine ([AFIHM](#)) ;
- le groupement de recherche Raisonnement, Apprentissage, et Décision en Intelligence Artificielle ([RADIA](#)) du CNRS.

Pour encourager les jeunes chercheuses et chercheurs à intégrer l'AFIA, nous avons également l'intention de soutenir l'organisation d'un colloque de doctorant.e.s autour de l'approche transversale de l'IA.

Activités prévues pour 2025 :

- mars 2025 : journée commune AFIA-AFIHM à Lille, sur le campus de l'Inria : les thèmes porteront sur l'impact de l'IA dans la société ainsi que l'emploi de l'IA dans la création artistique, avec notamment la question des droits d'auteur ainsi que la consommation d'énergie ;

- juillet 2025 : participation à la plateforme IA à Dijon, organisée par l’AFIA ; le thème de la journée Société et IA de 2025 s’intitule : « Expertise artificielle ? L’apparition des systèmes d’IA dans l’expertise ».
- courant 2ème semestre 2025 : journées doctorales à destination des jeunes chercheurs (à Strasbourg).

Comité de pilotage

Le comité de pilotage est composé des personnes suivantes :

- Benoît LEBLANC, École Nationale Supérieure de Cognitique ;
- Davy MONTICOLO, Université de Lorraine ;
- Céline CASTETS-RENARD, Université d’Ottawa et ANITI (Institut d’IA de Toulouse) ;
- Samia CHEHBI GAMOURA, École de Management de Strasbourg, Université de Strasbourg ;
- Alain GOUDEY, Neoma Business School ;
- Évelyne LUTTON, INRAE ;
- Thierry MÉNISSIER, Université Grenoble Alpes et MIAI (Institut d’IA de Grenoble) ;
- Fabrice JAOUËN, Junia ;

Le comité de pilotage peut être amené à inviter des membres de la communauté à participer aux discussions et réunions du collège.

Contacts

Coordinateur du collège : fab.jaouen@gmail.com.

Deux membres du comité de pilotage sont membres du conseil d’administration de l’AFIA :

- Benoît LEBLANC, benoit.leblanc@ensc.fr ;
- Davy MONTICOLO, davy.monticolo@univ-lorraine.fr.

(Version de mars 2025)

Collège Représentation et Raisonnement

Objectif du collège

L'objectif du Collège **Représentation et Raisonnement** (R&R) de l'**AFIA** est d'animer les communautés de recherche françaises dans ce domaine. Les thématiques de recherche sont relatives aux méthodes et outils fondamentaux de l'Intelligence Artificielle. Elles portent sur :

- la définition de modèles de représentation des informations (croyances, connaissances, préférences, obligations et permissions, actions, incertitude, confiance, réputation) comme les langages des logiques classiques ou non classiques, les modèles possibilistes, les ontologies, les langages à base de contraintes, les représentations graphiques, etc. ;
- la définition et l'automatisation de raisonnements sur ces informations : raisonnement spatio-temporel, dynamique des informations, révision de croyances, fusion d'informations symboliques, raisonnement par argumentation, raisonnement causal, raisonnement abductif, raisonnement à partir de cas, etc. ;
- la perspective algorithmique et de représentation pour des concepts utilisés dans des théories connexes comme la théorie des jeux ou la théorie du choix social (équilibre, stratégie gagnante, manipulation, etc.) : théorie des jeux algorithmique et choix social computationnel ;
- la mise au point de méthodes de codage des informations et d'algorithmes de traitement efficaces : compilation de connaissances, SAT, ASP, etc. ;
- la modélisation formelle de l'interaction : entre utilisateurs et systèmes informatiques, entre entités informatiques autonomes (agents) ;
- et généralement le lien avec différentes techniques liées à la décision, la planification, l'ordonancement, le diagnostic, l'apprentissage, les sciences des données, etc.

Ces thématiques couvrent de très nombreux contextes d'application, comme par exemple le Web sémantique, le Web des données, les systèmes de recommandation ou d'aide à la décision, les agents conversationnels et assistants personnels, la programmation des jeux, la robotique, etc.

Programme de travail

Le collège R&R est impliqué dans les activités suivantes :

- IAF : Journées d'Intelligence Artificielle Fondamentales. Ces journées ont lieu tous les ans, à l'initiative du comité IAF. Elles sont articulées autour de 3 ou 4 exposés de synthèse invités, ainsi que d'un programme constitué après appel à communication ;
- JFPDA : Journées Francophones sur la Planification, la Décision et l'Apprentissage pour la conduite de systèmes. Ces journées ont lieu tous les ans, à l'initiative des anciens présidents de comités de programme JFPDA et des participants à la liste de diffusion jfpda@loria.fr. Elles sont articulées autour de 2 ou 3 exposés de synthèse invités, ainsi que d'un programme constitué après appel à communication ;
- JFPC : Journées Francophones de Programmation par Contraintes. Ces journées ont lieu tous les ans à l'initiative de l'AFPC (Association Française pour la Programmation par Contraintes). Elles sont articulées autour de 2 ou 3 exposés de synthèse invités, ainsi que d'un programme constitué après appel à communication ;

- des ateliers thématiques, qui ont lieu lors d’autres événements et la contribution à l’organisation de journées communes.

Le collège consacre une partie de son budget de fonctionnement à l’attribution de bourses permettant à des étudiants d’assister à l’un des événements organisés par le collège, ou d’autres manifestations pertinentes au regard des thématiques scientifiques.

Comité de pilotage

Le comité de pilotage du collège est constitué des personnes suivantes :

- Elise BONZON, Université Paris-Descartes, LIPADE, Paris ;
- Zied BOURAOUI, co-président du comité de programme IAF, Université d’Artois, CRIL, Lens ;
- Sylvie DOUTRE, co-présidente du comité de programme IAF, Université Toulouse 1 Capitole, IRIT, Toulouse ;
- Sébastien KONIECZNY, ancien directeur du GDR IA, CNRS, CRIL, Lens ;
- Frédéric MARIS, membre du CA et représentant pour l’AFIA, Université Toulouse 3 Paul Sabatier, IRIT, Toulouse ;
- Nicolas MAUDET, Université Pierre et Marie Curie, LIP6, Paris ;
- François SCHWARZENTRUBER, ancien président du comité de programme JFPDA, ENS Rennes, IRISA, Rennes ;
- Laurent SIMON, membre du CA de l’AFIA, président de l’AFPC, Université de Bordeaux, LaBRI, Bordeaux ;
- Elise VAREILLES, membre du CA de l’AFPC ISAE Supaero, Toulouse ;
- Bruno ZANUTTINI, ancien président du comité de programme JFPDA, Université Caen Normandie, GREYC, Caen.

La composition du comité est amenée à être modifiée bi-annuellement.

Contacts

Coordinateur du collège : frederic.maris@irit.fr.

Listes de diffusion : bull-i3@irit.fr, bull-ia@gdria.fr, jfpda@loria.fr.

Deux membres du comité de pilotage du collège sont membres du Conseil d’Administration de l’AFIA :

- Frédéric MARIS, frederic.maris@irit.fr ;
- Laurent SIMON, lsimon@labri.fr.

(Version de juillet 2024)

Collège Science de l'Ingénierie des Connaissances

Objectif du collège

L'Ingénierie des Connaissances (IC) est une thématique de l'Intelligence Artificielle (IA). Sa finalité est la production de systèmes « intelligents et explicables », capables d'aider l'humain dans ses activités et ses prises de décisions. La construction des modèles de connaissances (principalement à base d'ontologies) et leur évaluation reposent sur une prise en compte des contextes applicatifs des différents domaines d'étude et des cas d'usage qui leur correspondent. La représentation formelle permet des raisonnements automatiques sur ces connaissances et sur les données qui leur sont associées, pouvant être complexes, hétérogènes, évolutives et distribuées. Elle permet aussi des tests de validité et de cohérence des modèles développés. L'opérationnalisation des modèles pose des questions de passage à l'échelle et d'explicabilité des résultats destinés aux utilisateurs des systèmes construits.

Les thématiques de recherche de l'IC sont relatives aux méthodes et outils de l'IA. Elles traitent un ensemble de thèmes portant, entre autres, sur les graphes et les modèles de connaissances dont nous listons les principaux :

- construction, réutilisation et mise à jour de ressources sémantiques, liage de données, alignement complexe, gestion des données et connaissances évolutives ;
- découverte de connaissances : fouille de ressources structurées et non structurées, découverte de classes et de propriétés, découverte de règles ;
- validation et évaluation : métriques d'évaluation, explicabilité des résultats, détection d'erreurs, interaction avec les utilisateurs ;
- gestion de données complexes, spatiales et temporelles,
- gestion des flux de données hétérogènes : imprécision, incertitude, interopérabilité sémantique ; mise en œuvre dans le contexte de l'internet des objets (IoT) ;
- éthique et Ingénierie des Connaissances.

Programme de travail

Le collège [Science de l'Ingénierie des Connaissances](#) (SIC) de l'[AFIA](#) a premièrement un rôle de diffusion de l'information. Il gère une liste de diffusion (comptant plus de 800 inscrits) sur l'ingénierie des connaissances et maintient une page web sur le site Web de l'AFIA. Il contribue également à l'animation et au dynamisme de la communauté de recherche en ingénierie des connaissances. Il est moteur dans l'organisation de différentes manifestations, comme :

- la conférence nationale Ingénierie des Connaissances (IC) qui a lieu chaque année lors de PFIA et les ateliers qui lui sont associés ;
- les sessions spéciales Interaction Management in Digitized Ecosystems organisées dans le cadre du chapitre français IEEE Systems Man and Cybernetics (SMC) ;
- la conférence internationale « *Terminology & Ontology : Theories and applications* » (TOTh) ;
- les journées dédiées au web sémantique dans le monde professionnel (SemWebPro) ;
- l'édition de numéros spéciaux en lien avec les thématiques du collège.

Les membres du collège contribuent à la mise en place de plusieurs « Journées Communes » de l'AFIA avec d'autres sociétés savantes, d'ateliers sur la Plate-Forme Intelligence Artificielle (PFIA) et de journées thématiques internes au collège. Ils participent activement à des événements internationaux tels que EKAW, K-CAP, ESWC, ECAI, TheWebConf ou encore ISWC.

Comité de pilotage

Le comité de pilotage du collège est constitué de chercheurs ou enseignants chercheurs spécialisés dans le domaine de l'ingénierie des connaissances, tous membres des comités de programme de la conférence nationale « journées francophones d'Ingénierie des Connaissances » (IC). Ils font également le lien entre des sociétés savantes (ARIA (Association Francophone de Recherche d'Information (RI) et Applications), ATALA (Association pour le Traitement Automatique des Langues), AIM (Association d'Informatique Médicale)) et avec les GDRs RADIA et MADICS, humanité numérique :

- Marie-Hélène ABEL, Université de Technologie de Compiègne, HEUDIASYC ;
- Jean CHARLET, Assistance Publique-Hôpitaux de Paris & INSERM, LIMICS ;
- Sylvie DESPRÉS, Université Sorbonne Paris Nord,, LIMICS ;
- Catherine FARON, Université Nice Sophia Antipolis, I3S et Inria ;
- Nathalie HERNANDEZ, Université Toulouse 2 Jean Jaurès , IRIT ;
- Nathalie PERNELLE, Université Sorbonne Paris Nord, LIPN ;
- Maxime LEFRANÇOIS, Mines Saint Etienne, IMT, LIMOS ;
- Nathalie PERNELLE, EURECOM, Sophia Antipolis ;
- Catherine ROUSSEY, INRAE Centre Occitanie Montpellier, MISTEA ;
- Fatiha SAÏS, Université Paris Saclay, LISN ;
- Cassia TROJAHN, Université Jean Jaurès Toulouse 2, IRIT ;
- Haifa ZARGAYOUNA, Université Sorbonne Paris Nord, LIPN.

Contacts

Coordinateur du collège : sylvie.despres@univ-paris13.fr.

Listes de diffusion : info-ic@inria.fr.

Mail pour contacter le comité de pilotage du collège : collegeSIC@afia.asso.fr.

Deux membres du comité de pilotage sont membres du conseil d'administration de l'AFIA :

- Fatiha SAÏS, fatiha.sais@lisn.fr ;
- Catherine ROUSSEY, catherine.roussey@inrae.fr.

(Version de juillet 2024)

Collège Systèmes Multi-Agents et Agents Autonomes

Objectif du collège

Le collège **Systèmes Multi-Agents et Agents Autonomes** (SMAA) de l'AFIA a pour mission d'organiser les activités du collège au sein de l'AFIA et d'assurer l'interaction entre l'AFIA et la communauté francophone SMA et Agents Autonomes, concernant leur volet IA. Il participe à l'animation scientifique au sein de l'AFIA, en coordination avec les communautés concernées, pour susciter l'implication des membres du collège dans les événements organisés par l'AFIA (p. ex. PFIA, CNIA) tant en participant aux comités scientifiques, qu'en organisant des manifestations d'intérêt pour la communauté IA en France.

Le collège SMAA évolue dans deux directions :

- Accentuation des interactions avec les communautés robotique, automatique d'une part et simulation, systèmes complexes d'autre part. Concrètement cela se traduira par l'organisation d'événements joints sur des travaux associant SMA et/ou agents conversationnels animés à ces disciplines. Lorsque d'autres champs d'IA seront ciblés, ces événements seront préparés en association avec les autres collèges de l'AFIA concernés.
- Mise en place de webinaires réguliers, issues des équipes impliquées dans le collège SMAA. Les séminaires d'équipes, soutenances de thèses, les soutenances de HDR pourront, sur base du volontariat être diffusées à l'ensemble du collège SMAA.

Programme de travail

Les missions du collège SMAA concernent l'organisation de manifestations scientifiques (conférences, journées thématiques), l'édition de dossiers techniques ou de numéros spéciaux de journaux sur des thématiques d'intérêt pour la communauté.

Le collège SMAA accompagne notamment l'organisation régulière des JFSMA et de WACAI :

- JFSMA 2021 à PFIA Bordeaux ;
- JFSMA 2022 à Saint-Etienne ;
- WACAI 2022 (lieu à définir).

Le collège SMAA accompagne également l'organisation des journées communes suivantes :

- Journée commune MACS & IA ;
- Journée commune Robotique & IA ;
- Journée commune Simulation & IA.

Il propose également un soutien similaire à d'autres événements, ponctuels ou récurrents, relevant de son périmètre scientifique.

Plus largement le collège envisage des actions d'animation à destination des doctorants, et des actions de médiation scientifique et de communication à destination du public.

Comité de pilotage

Le comité de pilotage du collège SMAA est constitué de chercheurs spécialisés dans le domaine des systèmes multi-agents et des agents autonomes, tous membres des comités de programme ou du comité consultatif des Journées Francophones en Systèmes Multi-Agents (JFSMA) et/ou du Workshop Affect, Compagnon Artificiel, Interaction (WACAI).

- Emmanuel ADAM, Université Polytechnique Hauts-de-France, LAMIH, Valenciennes ;
- Fabien MICHEL, Université de Montpellier, LIRMM, Montpellier ;
- Frédéric MIGEON, Université Toulouse 3 Paul Sabatier, IRIT, Toulouse ;
- Maxime MORGE, Université de Lille 1, CRISTAL, Lille ;
- Magalie OCHS, Université Aix-Marseille, LSIS, Marseille ;
- Gauthier PICARD, ONERA, Toulouse ;
- Nicolas SABOURET, Université Paris-Sud, LIMSI, Saclay ;
- Olivier SIMONIN, INSA Lyon, CITI, Lyon ;
- Mahdi ZARGAYOUNA, IFSTTAR, Paris.

Contacts

Coordinateur du collège : emmanuel.adam@uphf.fr.

Listes de diffusion : sma@loria.fr, acai@poleia.lip6.fr.

Deux membres du comité de pilotage sont membres du conseil d'administration de l'AFIA :

- Emmanuel ADAM, emmanuel.adam@uphf.fr ;
- Gauthier PICARD, gauthier.picard@onera.fr.

(Version de juillet 2024)

Collège Technologies du Langage Humain

Objectif du collège

Le collège **Technologies du Langage Humain** (TLH) de l'**AFIA** s'intéresse aux méthodes de communication humain-machine naturelle, pouvant s'étendre à une interaction humain-humain médiée. Elles permettent d'analyser, d'interpréter et de produire des actes du langage écrit, parlé ou signé, ainsi que d'interagir avec des données langagières. Les TLH englobent le Traitement Automatique des Langues (TAL), la Communication Parlée (CP) et leurs applications emblématiques comme la Recherche d'Information (RI) et la Traduction Automatique.

L'étude du langage humain est multidisciplinaire et requiert des compétences en Linguistique, Psychologie, Sciences Cognitives, Numérique et en Intelligence Artificielle (IA). Les TLH occupent une place centrale en IA sous le prisme du Test de Turing, couvrant la représentation (*e. g.*, plongements lexicaux, analyse syntaxique), le raisonnement (*e. g.*, systèmes de question-réponse, analyse sémantique), la planification (*e. g.*, argumentation), l'apprentissage (*e. g.*, analyse de sentiments), ou même l'intelligence collective (*e. g.*, détection de communautés). Créer des modèles capables d'interagir naturellement dans un langage donné impose une compréhension fine de l'acoustique, de la phonétique et de la prosodie pour l'oral, mais aussi de la morphologie, de la syntaxe, de la sémantique et de la pragmatique pour l'écrit ou le signé. Ainsi seulement, les TLH permettent un accès élargi à l'information à travers moteurs de recherche, traduction, résumé de textes, veille ou systèmes de question-réponse. La compréhension globale du langage permet également de caractériser les textes et la parole suivant leurs objectifs communicationnels. Ainsi, l'analyse des sentiments et des émotions, l'identification de discours haineux, la détection de plagats ou de contenus synthétiques, l'identification et la vérification du locuteur proposent autant de cadres applicatifs importants pour les sociétés numériques.

Puisqu'à l'ère du numérique les données langagières prolifèrent dans des quantités toujours plus importantes et accessibles, dans un nombre grandissant de pays, les champs d'applications des TLH sont nombreux. Ainsi, les archives numériques, les réseaux sociaux, les plateformes collaboratives, les entretiens clients, les textos, les courriels, les commentaires sur des plateformes de vente en ligne, les messages oraux, les vidéos en ligne sont autant de matières premières pour le développement d'applications des TLH. En particulier, quelques domaines privilégiés sont la santé, l'éducation, le droit, le journalisme et le handicap, mais d'autres émergent comme la finance, l'agriculture, la sécurité, le marketing et les humanités numériques.

Parallèlement au développement de modèles des TLH, et notamment les LLM, de nombreux défis connexes doivent être pris en compte dans le cadre d'une démarche globale, comme la confiance (*e. g.*, reproductibilité, explicabilité, confidentialité), l'éthique (*e. g.*, biais d'apprentissage, représentativité, anonymisation) et l'évaluation (*e. g.*, métriques dédiées) des systèmes proposés.

Le collège TLH de l'**AFIA** a donc pour mission de promouvoir l'animation et l'interaction scientifiques entre les communautés TAL, CP et RI, et l'ensemble des communautés en IA ayant des intérêts communs dans le but de consolider les collaborations transversales.

Programme de travail

Afin de mener à bien sa mission, le collège TLH s'engage à soutenir l'organisation de manifestations scientifiques (conférences, ateliers, tutoriels), animer des groupes de travail, éditer des dossiers

techniques, organiser des journées thématiques et diffuser et communiquer autour des recherches des communautés françaises du TAL, de la CP, de la RI et de l'IA.

Le collège TLH s'engage à servir de canal de communication entre l'AFIA et ses collègues ainsi qu'avec l'ATALA (Association pour le Traitement Automatique des Langues), l'ARIA (Association Francophone de Recherche d'Information et Applications) et l'AFCP (Association Francophone de la Communication Parlée).

Comité de pilotage

Le comité de pilotage du collège est constitué de 10 chercheuses et chercheurs spécialisés dans le domaine du TAL, de la CP et de la RI.

- Georgeta BORDEA, La Rochelle Université, L3i, La Rochelle
- Peggy CELLIER, INSA Rennes, IRISA, Rennes
- Gaël DIAS, Université de Caen Normandie, GREYC, Caen
- Corinne FREDOUILLE, Avignon Université, LIA, Avignon
- Sahar GHANNAY, Université Paris-Saclay, LISN, Saclay
- Martin LENTSCHAT, Université Toulouse Jean-Jaures, LERASS, Toulouse
- Vincent MARTIN, LORIA, Nancy
- José MORENO, Université de Toulouse, IRIT, Toulouse
- Marie TAHON Le Mans Université, LIUM, Le Mans
- Manel ZARROUK, Université Sorbonne Paris Nord, LIPN, Villetaneuse

Contacts

Coordinatrices du collège :

- Marie TAHON marie.tahon@univ-lemans.fr.
- Peggy CELLIER (peggy.cellier@irisa.fr) ;

Listes de diffusion :

- ln@cines.fr (communauté TAL) ;
- parole@listes.afcp-parole.org (communauté CP) ;
- info-aria@lsis.org (communauté RI).

Un membre du comité de pilotage est membre du CA de l'AFIA : José MORENO (jose.moreno@irit.fr).

(Version de septembre 2025)

Prix de Thèse IA 2024

Le jury

- Président : Andreas HERZIG ;
- Membres : Thierry ARTIÈRES, Isabelle BLOCH, Tristan CAZENAVE, Élisabeth FROMONT, Thomas GUYET, Pierre MARQUIS, Nicolas MAUDET, Fatiha SAÏS, Thomas SCHIEX, Nicolas THOME.

Les lauréats

- 1^{er} prix : Virginie DO, « *Fairness in recommender systems : insights from social choice* », dir. : Nicolas USUNIER, Jérôme LANG et Jamal ATIF, 11/07/2023, Meta AI / Université Paris Dauphine-PSL ;
- accessit (ex-æquo) : Pierre MARION, « *Mathematics of deep learning : generalization, optimization, continuous-time models* », dir. : Gérard BIAU et Jean-Philippe VERT, 20/11/2023, LPSM / Sorbonne Université ;
- accessit (ex-æquo) : Yuan YIN, « *Physics-aware deep learning and dynamical systems : hybrid modeling and generalization* », dir. : Patrick GALLINARI et Nicolas BASKIOTIS, 28/06/2023, ISIR / Sorbonne Université.

Prix de Thèse IA 2025

Le jury

- Président : Andreas HERZIG ;
- Membres : Thierry ARTIÈRES, Farah BENAMARA, Tristan CAZENAVE, Cédric DEMONCEAUX, Élisabeth FROMONT, Thomas GUYET, Jérôme LANG, Pierre MARQUIS, Nicolas MAUDET, Fatiha SAÏS, Thomas SCHIEX, Michèle SEBAG, Matthieu SERRURIER.

Les lauréats

- 1^{er} prix : Thomas FEL, « *Sparks of Explainability : Recent Advancements in Explaining Large Vision Models* », dir. : Thomas SERRE, 25/07/2024, Université de Toulouse ;
- accessit (ex-æquo) : Edwige CYFFERS, « *Differential Privacy for Decentralized Learning* », dir. : Aurélien BELLET, 05/12/2024, Université de Lille ;
- accessit (ex-æquo) : Marc JOURDAN, « *Solving Pure Exploration Problems with the Top Two Approach* », dir. : Emilie KAUFMANN et Rémy DEGENNE, 14/06/2024, Université de Lille.

Bulletins

Le [Bulletin](#) de l'[AFIA](#) fournit un cadre de discussions et d'échanges au sein de la communauté. Toutes les contributions, pour peu qu'elles aient un intérêt général, sont les bienvenues. Le bulletin contient des rubriques régulières de comptes rendus des conférences, journées et autres événements que l'AFIA organise ou parraine, les résumés d'HDR et de Thèses de Doctorat, et un Dossier qui dresse un état de l'art sur un domaine particulier de l'IA, présente des équipes françaises de recherche en IA (académiques ou industrielles), ou PFIA. Les bulletins de l'AFIA sont accessibles librement depuis le site de l'AFIA ou [son portail HAL](#).

Le comité de rédaction

Le comité de rédaction 2024 avait Dominique LONGIN comme rédacteur en chef, aidé de Emmanuel ADAM, Grégory BONNET (rédacteur adjoint) et Gaël LEJEUNE. En 2025, Emmanuel ADAM a laissé sa place à Caio CORRO.

Bulletin 122

Dossier « PFIA 2023 »

octobre 2023

- Ce bulletin portait sur la Plate-Forme Intelligence Artificielle ([PFIA 2023](#)) qui s'est déroulée à Strasbourg du 3 au 7 juillet 2023. Il regroupe les comptes rendus : des 9 conférences hébergées (APIA, CAp, CNIA, IC, JFPC, JFSMA, JIAF, RJCIA et SFC) ; de ses 4 journées communes (« Jeux et IA », « Santé et IA », « Affects, Compagnons Artificiels et Interactions » et « Résilience et IA ») ; d'un résumé des travaux menés par les jeunes docteurs primés pour leur thèse.
- Ce bulletin contient également les comptes rendus des journées AFIA/GRD RADIA (sur le thème « Modèles hybrides & IA ») et « Jam DriveToGaether ».

Bulletin 123

Dossier « Équipes industrielles en IA »

janvier 2024

- Ce bulletin, comme tous les deux ans, contient un dossier sur les équipes de recherche industrielles en IA. Ce dossier, monté par Christophe BORTOLASO (Berger-Levrault) du Collège [Industriel](#) de l'AFIA, a l'originalité de débiter par un texte collectif du collège sur les LLM et l'IA générative. Il est suivi par les contributions de 4 entreprises différentes (Jeolis Solutions, Thalès, Cosling et son Choco Solver et enfin CS Group).
- En fin de bulletin sont présentées les thèses en IA soutenues durant le trimestre écoulé.

Bulletin 124

Dossier « Regards croisés : AFIA & GDR RADIA »

avril 2024

- Ce bulletin est un peu particulier par rapport aux précédents car son dossier, que nous devons à Meghyn BIENVENUE et Nicolas MAUDET, présente et met en miroir les 10 GT du GDR RADIA avec les 8 [collèges](#) de l'AFIA. Un travail très intéressant qui met en lumière toute la complémentarité de ces deux structures.
- Ce bulletin présente également les comptes rendus de FIIA 2023 et PDIA 2024, ainsi que les thèses et HDR soutenue sur le trimestre écoulé.

Bulletin 125 **Dossier « IA & Création artistique »** **juillet 2024**

- Ce bulletin est aussi un peu particulier puisque son dossier, sous la responsabilité de Samuel BIANCHINI, est résolument interdisciplinaire. Il regroupe 7 contributions, pour plonger dans l'univers passionnant de la création artistique sous toutes ses formes.
- En fin de bulletin, la liste des thèses soutenues durant le 2^e trimestre 2024.

Bulletin 126 **Dossier « PFIA 2024 »** **octobre 2024**

- Le dossier de ce Bulletin est dirigé par Fatiha SAÏS. Il est consacré à PFIA 2024 (La Rochelle, du 1^{er} au 5 juillet 2024) qui a hébergé 5 conférences (APIA, CNIA, IC, JIAF et RJCIA), 3 journées (Agent & IA, Santé & IA, Société & IA), 7 ateliers, des tutoriels, des conférences invitées et la remise des 3 prix de thèse AFIA.
- Ce Bulletin recense également les thèses et HDR du 3^e trimestre de 2024 dont nous avons eu connaissance.

Bulletin 127 **Dossier « Équipes académiques en IA »** **janvier 2025**

- Ce Bulletin est dirigé par Gaël LEJEUNE, et son dossier « IA & Francophonie » a été confié à Pascale KUNTZ et Gayo DIALO.
- Ce bulletin présente également la liste des thèses et HDR soutenues durant le 4^e trimestre 2024 et dont nous avons eu connaissance.

Bulletin 128 **Dossier « IA & Neurosciences »** **avril 2025**

- Ce Bulletin dirigé par Dominique LONGIN contient un dossier « IA et neurosciences » dirigé par Frédéric ALEXANDRE et regroupant pas moins de 25 contributions.
- Ce Bulletin contient également un compte rendu de FIJA 2025 et des journées IA & Société en partenariat avec l'AFIHM et MAFTEC 12 en partenariat avec le GDR RADIA, ainsi que la liste des thèses et HDR soutenues pendant le 1^{er} trimestre 2025 et dont nous avons eu connaissance.

Bulletin 129 **Dossier « IA & Économie »** **juillet 2025**

- Ce second dossier thématique de 2025 a été dirigé par Jean-Daniel KANT et rassemble près d'une dizaine de contribution différentes. Bulletin dirigé par Grégory BONNET.
- Ce Bulletin contient également les comptes rendus de PDIA 2025 et GAS 2025, ainsi que les thèses et HDR dont nous avons eu connaissance et soutenue durant la période du 2^e trimestre 2023.

FIIA 2023

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)) organise son huitième « Forum Industriel de l'IA » sur le thème « Large Language Models (LLM) & IA ». La date a été en effet différée en 2024 pour des raisons d'organisation.

La journée est constituée de présentations pour permettre des échanges accrus entre académiques et industriels. Une dernière session permettra des échanges entre les industriels et universitaires invités, sur les verrous technologiques identifiés, et les perspectives de résolutions.

Date et lieu

- Date : 9 février 2024
- Lieu : [TOTEM](#) / Institut des Systèmes Complexes Paris IdF, 11 Place Nationale, 75013 Paris

Programme

- 09h00 « Mot d'accueil de l'[AFIA](#) » par Valérie REINER (Coordinatrice du collège industriel de l'[AFIA](#)).
- 09h10 « Introduction de la journée » par Davy MONTICOLO (Représentant du collège industriel de l'[AFIA](#)) et Bruno CARRON (Airbus Defence and Space).
- 09h25 « *Connaissances et IA* » par Guilherme ALVES (INRIA).
- 10h10 « *Approches IA hybrides et applications au domaine de la défense* » par Claude FENDZI et Geraud FAYE (Airbus Defence and Space).
- 10h55 Pause.
- 11h10 « *ChatDOC : une IA Générative interne pour interroger des documents clients* » par Ay-men SHABOU et Mohamed DHOUB (DataLab Groupe Crédit Agricole).
- 11h55 « *Mathématiques et IA* » par Marianne CLAUSEL (Institut Elie Cartan Nancy).
- 12h30 Buffet.
- 13h55 « *Ingénierie des connaissances, ontologie* » par Cécilia ZANNI (INSA Rouen).
- 14h40 « *Applications de l'IA* » par Christophe BORTOLASO (Berger Levrault).
- 15h25 Table ronde : « *Apports de la journée, Perspectives (verrous et ouvertures)* » par l'ensemble des orateurs et des oratrices.
- 16h10 Conclusion et prochains travaux (talks, bulletin, PFIA, APIA) par Davy MONTICOLO (Représentant du collège industriel de l'[AFIA](#)) et Bruno CARRON (Airbus Defence and Space).
- 16h30 Clôture.

Organisation

Cet événement est organisé par Davy MONTICOLO (Représentant du collège industriel de l'[AFIA](#)) et Bruno CARRON (Airbus Defence and Space).

Inscriptions

Les inscriptions à la journée (gratuites pour les membres AFIA, 30€ sinon) sont obligatoires et à effectuer sur le site : <https://afia.asso.fr/inscription-fia/>.

PDIA 2024

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)) organise sa neuvième journée « Perspectives et Défis en Intelligence Artificielle » sur le thème « NEUROSCIENCE ET IA ».

La journée est hybride, pouvant être donc suivie à distance.

Date et lieu

- Date : 09 Avril 2024
- Lieu : [ENSC](#), 109 avenue Roul, 33400 Talence

Programme

- 09h30 Accueil.
- 10h00 Présentation de l'AFIA par Benoit LEBLANC (Président de l'AFIA) et introduction de la journée PDIA – Perspectives et Défis en IA – Neuroscience et IA par Thomas BORAUD (Campus Neuroscience).
- 10h15 « *IVirtual Brain Twins at the interface of AI and the brain* » par Viktor JIRSA (Institut de Neurosciences des Systèmes, Marseille).
- 11h15 « *IDeep Learning in Medical Imaging : What's Needed for Training Data ?* » par Francesca GALASSI (Empenn lab, Inria Rennes).
- 12h15 Pause.
- 13h45 « *IHippocampal cells inside a random but embodied computational model* » par Naomi CHAI-EICHEL (Institut des Maladies Neurodégénératives, CNRS, Inria, U. Bordeaux).
- 14h45 « *IA neuro-inspirée pour le codage prédictif et neurosymbolique dans l'acquisition du langage et la planification* » par Alex PITTI (ETIS Lab, CY Alliance, Cergy Paris).
- 15h45 Pause.
- 16h15 « Algorithmes d'apprentissage et de prise de décision à la croisée des chemins entre l'Intelligence Artificielle et les Neurosciences » par Mehdi KHAMASSI (ISIR, Sorbonne Université).
- 17h15 « Comment exploiter et faire évoluer les relations IA-Neurosciences ? » Table ronde.
- 18h00 Clôture.

Organisation

Cet événement est organisé par Frédéric ALEXANDRE (LABRI, INRIA), Arthur LEBLOIS (Université de Bordeaux, Neurocampus).

Inscriptions

Les inscriptions à la journée sont obligatoires uniquement pour les personnes souhaitant être présentes dans l'amphithéâtre. Pour bénéficier des pauses et du déjeuner, il est nécessaire de s'inscrire avant le 5 avril 2024.

- L'inscription est gratuite pour les personnes adhérentes à l'AFIA.
- Elle est de 30€ pour les personnes non adhérentes.

FIIA 2025

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)) organise son neuvième FORUM INDUSTRIEL de l'IA sur le thème «IA Responsable et maîtrise des risques». Cette journée est labellisée «En route vers le Sommet» dans le cadre du Sommet pour l'action sur l'IA.

La journée est constituée de présentations pour permettre des échanges accrus entre académiques et industriels. Une dernière session permettra des échanges entre les industriels et universitaires invités, sur les verrous technologiques identifiés, et les perspectives de résolutions.

Date et lieu

- Date : 28 janvier 2025
- Lieu : [TOTEM](#) / Institut des Systèmes Complexes Paris IdF, 11 Place Nationale, 75013 Paris

Programme

- 09h00 « Mot d'accueil de l'[AFIA](#) » par Valérie REINER (Collège Industriel l'[AFIA](#)).
- 09h10 « Introduction de la journée » par Davy MONTICOLO (Représentant du collège industriel de l'AFIA) et Walid ERRAY (Groupe Crédit Agricole).
- 09h20 « *Dépasser ce qui brille et embrasser la complexité des sociétés modernes pour s'engager intelligemment dans l'intelligence artificielle* » par Fabrice JAOUEN (Ministère des Armées / Collège HumaniSIA).
- 10h05 « *IA Générative en Production : maîtriser ses risques et renforcer la sécurité grâce au Red-Teaming* » par Aymen SHABOU et Arnaud PARAN (Groupe Crédit Agricole).
- 10h50 Pause.
- 11h10 « *Chaire IA de confiance et responsable – Focus sur l'analyse et mesure des risques de confidentialité des grands modèles de langage* » par Jérémie DENTAN et Sonia VANIER (Ecole Polytechnique).
- 11h45 « *IA responsable : vision de bout-en-bout en vue de standardisation* » par Celine REVERDY, Fateh KAKAI et Michel BARRETEAU (Thales Group).
- 12h30 Buffet.
- 13h40 « *Très grands modèles de langue pour le domaine médical : avancées, enjeux et défis* » par Richard DUFOUR (Université de Nantes).
- 14h25 « *Moins d'Erreurs, Plus de Confiance : Vers une Réduction des Hallucinations dans les Assistants Intelligents et le Résumé Juridique* » par Mokhtar Boumedyen BILLAMI et Nihed BENDAHDMAN (Berger Levrault).
- 15h10 « *Les LLM au défi : passer à l'échelle tout en réduisant leur taille* » par Christophe CERISARA (LORIA).
- 15h55 Table ronde : « *Questions / Réponses* » par tous les conférenciers.
- 15h55 Conclusion de la journée par Valérie REINER (Coordinatrice du collège industriel de l'AFIA), Davy MONTICOLO (Représentant du collège industriel de l'AFIA) et Walid ERRAY (Groupe Crédit Agricole).

Organisation

Cet événement est organisé par Davy MONTICOLO (Représentant du collège industriel de l'[AFIA](#)) et Walid ERRAY (Groupe Crédit Agricole).

Inscriptions

Les inscriptions à la journée (gratuites pour les membres AFIA, 30€ sinon) sont obligatoires et à effectuer à partir du site de l'AFIA : <https://afia.asso.fr/les-journees/fia-2025/>.

PDIA 2025

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)) organise sa onzième journée « Perspectives et Défis de l'IA » (PDIA) sur le thème « Désinformation, Démocratie et IA ».

La journée s'articulera autour de 6 exposés invités en informatique, mathématiques et droit. Elle comprendra également 4 interventions courtes de doctorant(e)s.

Date et lieu

- Date : 10 juin 2025
- Lieu : CNAM Paris, Amphithéâtre Gaston Planté, accès 35, 1er étage, 2 rue Conté, Paris

Programme

- 09h00 Accueil et café.
- 09h50 Introduction.
- 10h00 « Knowledge is Power : Fighting Disinformation with Data and AI » par Ioana MANOLESCU (Inria, CEDAR).
- 10h50 « If generative AI becomes the mirror, we will end as Narcisse » par David CHAVALARIAS (CNRS, CAMS et EHESS).
- 11h40 « AI Models with Systemic Risks—Manipulation and Disinformation : A Concept at the Crossroads of the AI Act, GDPR, and Digital Services Act » par Juliette SÉNÉCHAL (Université de Lille).
- 12h30 Buffet.
- 14h00 « Influence Campaign – The Romanian Example » par Gabriel BEN ZENOU (Inria, CEDAR, AMIAD) – présentation jeune chercheur.
- 14h15 « On Assessing the Political Biases of Multilingual Large Language Models » par Paul LERNER (Sorbonne Université) – présentation jeune chercheur.
- 14h30 « Argumentation Perspectives for Better Debates » par Bruno YUN (Université Claude Bernard Lyon 1, LIRIS).
- 15h20 Pause.
- 15h45 « Computational Social Choice and Democracy » par Jérôme LANG (CNRS, LAMSADE).
- 16h35 « Between surveillance and sousveillance : what balance in a democracy ? » par Elia VERDON (Université de Bordeaux, CERCCLE).
- 17h25 Clôture.

Organisation

Cet événement est organisé par Fatiha SAÏS (LISN, Université Paris Saclay) pour le Conseil d'Administration de l'AFIA, et Srdjan VESIC (CRIL, Université d'Artois), Cédric DU MOUZA (CEDRIC, CNAM Paris), et Nada MIMOUNI (CEDRIC, CNAM Paris) pour PDIA.

Inscriptions

Les inscriptions à la journée sont **obligatoires**. Pour des raisons logistiques (restauration), il est fortement conseillé de s'inscrire avant le 03 Juin 2025. **Participation en présentiel**, au CNAM Paris, pour bénéficier des échanges directs, des pauses cafés et du déjeuner.

L'inscription est **gratuite pour les personnes adhérentes à l'AFIA** : [inscription adhérents](#). Elle est de **30€ pour les personnes non adhérentes** : [inscription non adhérents](#).

EFIA 2025

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)) organise sa cinquième journée « Enseignement et Formation » sur le thème « LLM & Enseignement ».

La journée a pour objectif d'échanger autour des nouvelles pratiques dans l'enseignement dans le cadre de l'utilisation de l'IA Générative. Ce sera l'occasion d'identifier les innovations pédagogiques, sa mise en pratique et les défis encore existants pour la communauté enseignante, pour les établissements d'enseignement et pour les étudiants.

Date et lieu

— Date :

14 octobre 2025

— Lieu :

Université de Toulouse

Programme

- 09h00 Ouverture de Benoit LE BLANC (Président de l'AFIA). Introduction par André AOUN et José MORENO (membre du CA de l'AFIA).
- 09h30 « Parlez IA ! Secrets d'une ingénierie de prompt efficace » par André AOUN (Université de Toulouse).
- 10h00 « Réflexions pour un usage pertinent de l'IA dans l'enseignement et l'apprentissage » par Franck AMADIEU (Université de Toulouse).
- 10h30 « TBA » par Prenom NOM (AFFILIATION).
- 11h00 Pause.
- 11h30 « TIPS : Technologie Interactive pour Pratiques de Scénarios » par Stéphanie LOUP (AFFILIATION), Olivier CATTEAU (AFFILIATION) et Cédric TEYSSIE (AFFILIATION).
- 12h30 Buffet.
- 14h00 « Augmenter l'apprentissage avec l'IA : cas d'usage, défis et impact » par Mar PEREZ-SANAGUSTIN (AFFILIATION).
- 14h30 « TBA » par Prenom NOM (AFFILIATION).
- 15h00 Pause.
- 15h30 « Table ronde sur l'Intelligence Artificielle Générative au service de l'enseignement » par André AOUN (AFFILIATION), Olivier CATTEAU (AFFILIATION), Cédric TEYSSIE (AFFILIATION) et Mar PEREZ-SANAGUSTIN (AFFILIATION).
- 17h00 Clôture.

Organisation

Cet événement est organisé par André AOUN (Université de Toulouse) et José MORENO (Université de Toulouse) pour le Conseil d'Administration de l'AFIA.

Inscriptions

Lien d'inscription à venir

Modèles hybrides & IA 2023

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)) et le groupe de travail Modèles Hybrides d'IA ([MHyIA](#)) du groupement de recherche Raisonnement, Apprentissage, et Décision en Intelligence Artificielle ([GDR RADIA](#)) organisent conjointement une 1^{re} journée commune « Modèles hybrides & IA » sur le thème « IA neuro-symbolique » au sein de la conférence « *European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty* » ([ECSQARU 2023](#)).

Date et lieu

- Date : 19 septembre 2023
- Lieu : la salle des colloques, Bâtiment "La Maison de la Recherche", Université d'Artois, Rue Maurice Schuman, 62000 Arras

Programme

- 09h00 Ouverture de Fatiha SAÏS (représentant l'AFIA) et Zied BOURAOUI (représentant le GDR RADIA).
- 09h15 « *Aligning embeddings with symbolic knowledge : towards a tight integration of learning and reasoning* » par Steven SCHOCKAERT (Cardiff University).
- 10h15 Pause.
- 10h30 « *Interpretable Neural-Symbolic Concept Reasoning* » par Giuseppe MARRA (KU Leuven).
- 11h30 « *Integrating Combinatorial Solvers and Neural Models* » par Pasquale MINERVINI (University of Edinburgh).
- 12h30 Clôture.

Organisation

Cet événement est organisé par Zied BOURAOUI (CRIL, Université d'Artois), Pierre MONNIN (I3S, Université Côte d'Azur) et Fatiha SAÏS (LISN, Université Paris Saclay).

Inscriptions

La participation est gratuite mais l'inscription est obligatoire, à effectuer à partir de ce lien, pour recevoir le lien Zoom de la demi-journée.

EGC & IA 2024

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)) et l'association internationale francophone d'Extraction et de Gestion des Connaissances ([Association EGC](#)) organisent, avec l'aide du groupe de travail « Gestion et Analyse des données Spatiales et Temporelles » ([GAST](#)), une journée commune « Gestion et Analyse des données Aériennes et Satellitaires » ([G2AS' 24](#)) sur le thème de la représentation, de la Gestion, de l'Analyse et du stockage des données Aériennes et Satellitaires. Cette journée réunira les acteurs de la recherche académique ou industrielle autour de cette thématique.

Date et lieu

— Date :

17 avril 2024

— Lieu :

[EPITA](#), 14-16 rue Voltaire, 94270 Le Kremlin-Bicêtre

Programme

- 09h00 Mot d'accueil par Aurélie LEBORGNE (GT GAST), par Thierry GERAUD (LRE EPITA) et par Thomas GUYET (AFIA).
- 09h30 « Présentation du projet SESAME Surveillance de l'État de SANTé des Mangroves » par Franck NIVOLE (IRD Nouvelle-Calédonie, UMR Espace-DEV).
- 10h00 « Machine and deep learning for earth observation : advanced approaches and practical use cases » par Roberto INTERDONATO (CIRAD, UMR TETIS, INRIA, Montpellier).
- 10h30 Pause.
- 10h45 « Données satellitaires : de l'application locale à des modèles de fondation » par Sylvain LOBRY (LIPADE, Université Paris Cité).
- 11h15 « Interplay between data assimilation and Machine Learning for time series of satellite images » par Lucas DRUMET (Lab-STICC, IMT Atlantique).
- 11h45 Posters.
- 12h00 Pause.
- 13h30 Discussion Posters.
- 14h00 « Analyse d'images aériennes historiques : application à une étude épidémiologique » par Laure TOUGNE (LIRIS, Université Lumière Lyon 2).
- 14h30 « Utilisation de l'imagerie satellitaire – L'Espace au service de la Terre » par Henri GIRAUD (SERTIT, Université de Strasbourg).
- 15h00 « La Terre vue par la géodésie spatiale : des mesures massives pour une meilleure compréhension des processus géophysiques sous-jacents » par Samuel NAHMANI (UMR IPGP / Université Paris Cité, CNRS, IGN ; Université Gustave Eiffel, ENSG, IGN).
- 15h30 Pause.
- 15h45 « Apprentissage profond pour le traitement de séries temporelles d'images satellites » par Loïc LANDRIEU (IMAGINE – LIGM / École Nationale des Ponts et Chaussées).
- 16h15 « Traitement d'images de télédétection SAR et optiques pour la cartographie de l'occupation des sols » par Flora WEISSGERBER (ONERA/DTIS/SAPIA).
- 16h45 Clôture.

Organisation

Cet événement est organisé par Clément IPHAR (LETG, UBO, Brest), Guillaume TOCHON (LRE, EPITA, Kremlin-Bicêtre), Aurélie LEBORGNE (ICube, UNISTRA, Strasbourg), Loïc SALMON (ISEA, UNC, Nouméa, Nouvelle-Calédonie), et Nida MEDDOURI (LRE, EPITA, Kremlin-Bicêtre).

Humanités Numériques & IA 2024

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)) et les groupes de travail Masses de Données, Informations et Connaissances en Sciences ([MADICS](#)) et Méthodes et Applications pour la Géomatique et l'Information Spatiale ([MAGIS](#)) organisent conjointement une première journée commune sur le thème « Humanité Numérique et IA ».

Le vocable humanités numériques s'est imposé pour désigner les travaux de recherche relevant tantôt : de la création, la gestion et la formalisation de processus sociaux à l'aide d'outils mathématiques et informatiques ; ou encore de la formaliser de processus humains. L'analyse automatique de documents anciens, de traitement automatique du langage naturel, de recherche d'informations ou encore de simulation, posent de véritables défis scientifiques aux approches développées dans le domaine de l'intelligence artificielle.

Date et lieu

- Date : 3 mai 2024
- Lieu : Datalab, BnF, Quai François Mauriac, 75706 Paris
- Distanciel : Egalement en distanciel via Zoom

Programme

- 09h15 Ouverture par Fatiha SAÏS (AFIA), Nathalie HERNANDEZ(GdR MADICS), Nathalie ABADIE (GdR MAGIS), Tiphaine VACQUE (BnF) et Marie CARLIN (Datalab, BnF).
- 09h55 « La politique dans la machine : identifier, mesurer et limiter l'information politique apprise par les algorithmes » par Tim FAVERJON(médialab Siences Po) et Pedro RAMACIOTTI(médialab Siences Po, ISC-PIF, INSHS-CNRS).
- 10h15 «ISIDORE 2030 : refactorisation d'un moteur de recherche à l'ère des IA de traitement et des IA génératives» par Stéphane POUYLLAU (Huma-Num)..
- 12h30 Clôture.

Organisation

Cet événement est organisé par Zied BOURAOUI (CRIL, Université d'Artois), Pierre MONNIN (I3S, Université Côte d'Azur) et Fatiha SAÏS (LISN, Université Paris Saclay).

Inscriptions

La participation est gratuite mais l'inscription est obligatoire, à effectuer à partir de ce lien, pour recevoir le lien Zoom de la demi-journée.

Santé & IA 2024

L'Association française d'Informatique Médicale ([AIM](#)) et le collège Science de l'Ingénierie des Connaissances de l'(AFIA) organisent une septième journée commune Santé et IA, inscrite au sein de la Plate-Forme Intelligence Artificielle 2024 ([PFIA 2024](#)).

Cette journée a pour objectif de faire un point sur les travaux menés actuellement en ingénierie des connaissances dans le domaine de la santé.

En effet, l'ingénierie des connaissances peut permettre de répondre aux enjeux majeurs tels que la progression du savoir médical, l'aide à la décision (qu'elle soit diagnostique, thérapeutique ou pronostique), et plus largement d'apporter des solutions permettant de favoriser l'accès aux informations et connaissances médicales. Ces méthodes peuvent être appliquées à de nombreux cas d'usage au service des patients (que ce soit à l'échelle individuelle ou d'une population), pour les professionnels de santé, étudiants en santé, chercheurs, décideurs et le grand public. Les données de santé ont de multiples caractéristiques qui soulèvent des problématiques liées à l'extraction d'information, à la sécurité des données à caractère personnel, à l'intégration de données réparties dans des systèmes hétérogènes, à la recherche d'information, au traitement de données massives et à la compréhension des données.

Articles et présentations sont disponibles sur cette page : afia.asso.fr/les-journees-communes/sante-et-ia-2024

Date et lieu

— Date :

1er Juillet 2024

— Lieu :

PFIA 24, La Rochelle

Programme (session matinale)

- 9h30 « Chirurgie métabolique de précision » par François PATTOU (*conférence invitée*)(Université de Lille, Inserm, CHU Lille, Institut Pasteur de Lille, U1190-EGID).
- 10h30 Pause.
- 10h50 « Deep Reinforcement Learning for Controlled Piecewise Deterministic Markov Process in Cancer Treatment Follow-up » par Alice CLEYNEN, Benoîte DE SAPORTA, Orlane ROSSINI, Régis SABBADIN, Meritxell VINYALS.
- 11h15 « Contributions à l'Ordonnancement des Interventions en Chirurgie Ambulatoire : Q-learning et Flow-Shop Hybride » par Lydia BOUCHLAGHEM, Adnène GUESSOUM, Fatima GHEDJATI.
- 11h30 « Amélioration de la sécurité chirurgicale avec un jumeau numérique prédictif : le rôle des systèmes multi-agents et de l'apprentissage par renforcement » par Bruno PEREZ.
- 11h45 « Interpretable AI for Dermoscopy Images of Pigmented Skin Lesions » par Marianne DEFRESNE, Élise COUTIER, Paul FRICKER, Folkert BLOK, Hang NGUYEN.
- 12h10 « INM-Explain – Expliquer les controverses médicales : Application au cas des interventions non médicamenteuses » par Houria SAYAH, Alya ZOUZOU, Jonathan DUCKES, Audric GIRONDIN, Maéva MAÏO, Maximilien SERVAJEAN, Sandra BRINGAY.
- 12h30 Pause.

Programme (session post-méridienne)

- 14h50 « Une Ontologie du Parcours de Soins » par François-Élie CALVIER, Thomas GUYET, Nolwenn LEMEURE-ROUILLARD.
- 15h15 « Entrepôts de Données de Santé et Protection de la Vie Privée : Synthèse de discussions Inter-CHU » par Antoine RICHARD, Manal AHIKKI, Marc BERARD, Camille BOIN, Antoine BOUTET, Stéphane BREANT, Alice CALLIGER, Ariel COHEN, Jean-François COUCHOT, Denis DELAMARRE, Caroline DUNOYER, Thibaut FABACHER, Lucas GAUTHIER, David GIMBERT, Camille GIRARD-CHANUDET, Faustine GLAIS, Romain GRIFFIER, Martin HILKA, Yannick JACOB, Vianney JOUHET, David LAIYMANI, Leonardo MOROS, Joris MULLER, David PELLECUER, Thomas PETIT-JEAN, Maxime SALAUN, François TALBOT, Perceval WAJSBURT, Kevin YAUY.
- 15h30 « Chain Classifier pour le transcodage LOINC » par Théodore MICHEL-PICQUE, Sandra BRINGAY, Pascal PONCELET, Namrata PATEL, Guilhem MAYORAL.
- 15h45 « Is DTW resilient to noise and effective for EEG functional connectivity assessment ? » par Maxime BEDOIN, Nesma HOUMANI, Rita YABOURI, Jérôme BOUDY, Kiyoka KINUGAWA.
- 16h00 Pause.
- 16h20 « Récentes avancées de l'inférence en langue naturelle pour les essais cliniques » par Mathilde AGUIAR, Pierre ZWEIGENBAUM, Nona NADERI.
- 16h45 « Des pipelines faciles à réutiliser pour comparer les performances d'outils de reconnaissance d'entités nommées sur les textes cliniques en français » par Thibault HUBERT, Ghislain VAILLANT, Olivier BIROT, Camila ARIAS, Antoine NEURAZ, Bastien RANCE, Adrien COULET.
- 17h10 « Équilibrer qualité et quantité : comparaison de stratégies d'annotation pour la reconnaissance d'entités nommées en cardiologie » par Virgile BARTHET, Laura MONCEAUX-CACHARD, Christine JACQUIN, Cyril GROUIN, Joconde WELLER, Pascal DEGroote, Emmanuel MORIN, Pierre ZWEIGENBAUM.

Organisation

Cet événement est organisé par Adrien COULET (HeKA, Inria Paris & Inserm, Université Paris Cité), Fleur MOUGIN (AHeaD, Université de Bordeaux & Inserm) et Lina SOUALMIA (LITIS & LIMICS, Normandie Universités & Inserm).

Société & IA 2024

L'**AFIA**, le groupe de travail Aspects Computationnels de l'Éthique (**ACE**) du **GDR RADIA** et **Inria Bordeaux** ont organisé les journées « Société et IA » les 1er et 2 juillet 2024 dans le cadre de la Plate-Forme Intelligence Artificielle de l'AFIA (PFIA 2024).

Depuis plusieurs années, des comités réunis à l'initiative d'université, d'États, de puissances supranationales comme la Commission Européenne, de sociétés savantes ou d'organisation non gouvernementales réfléchissent aux questions d'éthique de l'Intelligence Artificielle et à sa régulation. Ces réflexions ont abouti entre autres sur la notion de systèmes informatiques dignes de confiance qui sont à mettre en perspective avec les problématiques en éthique artificielle.

Ces journées avaient pour objectif de réunir les communautés travaillant sur l'Intelligence Artificielle de confiance, l'éthique artificielle et plus généralement sur tout ce qui est en lien avec l'impact social de l'Intelligence Artificielle. Dans une volonté d'ouverture tant aux communautés de recherche travaillant déjà sur ces problématiques, qu'aux non-spécialistes intéressés, nous avons encouragé toutes les contributions relatives à ces sujets, qu'elles portent sur les aspects techniques, juridiques, philosophiques ou sociologiques de l'Intelligence Artificielle ou sur les impacts industriels de son déploiement.

Date et lieu

— Date :

1-2 Juillet 2024

— Lieu :

PFIA 24, La Rochelle

Programme (Lundi 1^{er} Juillet 2024)

- 14h50 « Ouverture des journées » par Frédéric ALEXANDRE, Grégory BONNET, Ikram CHRAIBIKaadoud, Jean-Gabriel GANASCIA.
- 15h00 « Un besoin de Confiance Artificielle pour l'Intelligence Artificielle » (*conférence invitée*) par Laurent SIMON.
- 16h00 Pause.
- 16h20 « Détection de biais et intégration de connaissances expertes pour l'explicabilité en IA » par Matthieu DELAHAYE, Lina FAHED, Florent CASTAGNINO et Philippe LENCA.
- 16h50 « Modéliser la confiance d'un agent décisionnel » par Baptiste PESQUET et Frédéric ALEXANDRE.
- 17h20 « L'explicabilité appliquée aux modèles de diffusion » par Raphael TEITGEN, Jeanine HARB et Jeanne LE PEILLET.

Programme (Mardi 2 Juillet 2024)

- 09h00 « La réglementation de l'intelligence artificielle dans l'Union européenne » (*conférence invitée*) par Nathalie NEVEJANS.
- 10h00 Pause.
- 10h20 « La normalisation de l'IA : un déluge de réinterprétations de l'AI Act » par Hélène HERMAN et Mélanie GORNET.
- 10h50 « IA générative et désinformation : quel impact sur les rapports de force en géopolitique ? » par Alice MARANNE, Clara FONTAINE-SAY et Ikram CHRAIBI KAADOUD.

- 11h20 « Quel sens donner à l'IA de confiance ? » (*conférence invitée*) par Cédric BRUN.
- 12h30 Pause.
- 14h50 « L'intelligence artificielle à la lumière de la mythologie grecque : rendre compréhensible les impacts de l'IA pour le grand public » par Fabrice MUHLENBACH.
- 15h20 « Cadre conceptuel pour agents autonomes éthiques : application aux agents conversationnels » par Robert VOYER, Thierno TOUNKARA..
- 16h00 Pause.
- 16h20 « Définition de la compatibilité pour des préférences morales : une condition basée sur la cohérence de Suzumura » par Guillaume GERVOIS, Gauvain BOURGNE et Marie-Jeanne LESOT.
- 16h50 « Modèle d'éthique pour les MDP multi-agents » par Mihail STOJANOVSKI, Nadjat BOURDACHE, Grégory BONNET et Abdel-Ilah MOUADDIB.
- 17h20 « Équité subjective par les explications » par Sarra TAJOURI et Alexis TSOUKIÀS..

Organisation

Cet événement est organisé par Frédéric ALEXANDRE (Inria Bordeaux), Grégory BONNET (GREYC, Université de Caen), Ikram CHRAIBI KAADOUD (Inria Bordeaux), Jean-Gabriel GANASCIA (LIP6, Sorbonne Université)

Agent & IA 2024

L'AFIA, le groupe de travail Affects, Compagnons Artificiels et Interactions ([ACAI](#)) et la communauté des Journées Francophones des Systèmes Multi-Agents ([JFSMA](#)) ont organisé la journée « Agent et IA » le 4 juillet 2024 dans le cadre de la Plate-Forme Intelligence Artificielle de l'AFIA (PFIA 2024).

Cette journée a porté sur les agents et avait pour thème l'humain dans la boucle. Elle était composée de deux parties :

- la matinée a été dédiée à un tutoriel animé par le groupe de travail ACAI : Affects, Compagnons Artificiels et Interactions, avec un focus sur l'interaction humain-agent ;
- l'après-midi a été dédiée à la présentation de systèmes multi-agents dans le domaine des *Smart-Cities*, avec un focus sur l'aspect mobilité.

Cette journée s'est voulue accessible et a visé également un objectif pédagogique, vous donnant quelques pistes pour développer vous-mêmes vos agents logiciels.

Date et lieu

- Date : 4 Juillet 2024
- Lieu : PFIA 24, La Rochelle

Programme (session ACAI)

- 10h20 « Interaction entre une personne et un ou plusieurs agent(s) conversationnel(s) animé(s) » par Brian RAVENET et Nicolas SABOURET.
- 12h30 Pause.

Programme (session JFSMA)

- 14h50 « Travaux et plateformes de la communauté Mobilité Intelligente de l'IMT » par Arnaud DONIEC et Flavien BALBO.
- 15h20 « Architecture et comportements pour la simulation multiagent de véhicules et de l'infrastructure connectée en condition atmosphérique difficiles » par Stéphane GALLAND.
- 15h40 « Vers des véhicules autonomes socialement désirables » par Joris DINNEWETH.
- 16h00 Pause.
- 16h20 « Résolution de conflits entre trajectoires de vol planifiées (contrats 4D) de drones dans le trafic urbain » par Gauthier PICARD.
- 16h40 « Projet VILAGIL et action VILAGIL-MaaS » par Valérie CAMPS et Elsy KADDOUM.
- 17h20 « Projet autoCampus » par Marie-Pierre GLEIZES.
- 17h40 Table ronde.

Organisation

Cet événement est organisé par Valérie CAMPS (Université Paul Sabatier), Elsy KADDOUM (Université Toulouse 2 Jean-Jaurès), Brian RAVENET (IUT Orsay) et Nicolas SABOURET (Université Paris-Saclay).

Société & IA 2025

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)) et l'Association Association Francophone de l'Interaction Humain-Machine ([AFIHM](#)) organisent conjointement une journée commune « IA & Société » de débats sur l'influence de l'IA dans la société.

Les interventions seront réparties en 2 demi-journées. Chaque demi-journée s'achève par un débat entre les intervenants et un échange avec les participants.

Date et lieu

- Date : 10 mars 2025
- Lieu : Centre Inria de l'université de Lille, Parc scientifique de la Haute-Borne 40, avenue Halley - Bât A - Park Plaza 59650 Villeneuve-d'Ascq

Programme

- 09h30 Café d'accueil.
- 09h50 Présentation de la journée.
- 10h00 « IA Générative : quand les sciences humaines s'en mêlent ! » par Alexandra DELMAS et Juliette MASSART (onepoint).
- 10h30 « L'IA générative, la création et le droit d'auteur » par Céline CASTETS-RENARD (Université d'Ottawa).
- 11h00 « Quels rôles peut jouer l'IA en matière de répression pénale ? » par Jean-Claude PLANQUE (Université de Lille).
- 11h30 « Créer et juger avec l'IA : quels coûts et quelles conséquences ? (débat) ».
- 12h30 Pause déjeuner et posters.
- 14h00 « Pratique et politique de l'intelligence artificielle dans l'Art » par Sarah FDILI ALAOUI (University of Arts, Londres).
- 14h30 « IA Explicable (eXplainable AI – XAI) : Fondements et Applications » par Samia CHEHBI GAMOURA (Ecole de Management de Strasbourg).
- 15h00 « Comment appréhender les résultats complexes des modèles intégrés de transformation du climat ? » par Thierry CHEVALIER (Capgemini).
- 15h30 « Comment s'approprier l'IA pour créer et expliquer des phénomènes complexes ? (débat) ».
- 16h30 Cocktail de clôture.

Organisation

Cet événement est organisé par Fabrice JAOUËN (ISEN) pour l'AFIA et le collège HumaniSIA et Bruno FRUCHARD (Inria Lille) pour l'AFIHM.

Inscriptions

Les inscriptions à la journée sont gratuites et obligatoires et à effectuer sur le [lien suivant](#). Un déjeuner sous forme de buffet sera organisé sur place et inclus dans l'inscription.

MAFTEC-12 2025

Le collège **SMAA** de l'Association Française pour l'Intelligence Artificielle (**AFIA**) et les groupes de travail **MAFTEC** et **ACE** du GDR **RADIA** organisent conjointement une 12^e journée commune « MAFTEC » sur le thème « ACE/MAFTEC et MAFTEC/SMAA ».

L'objectif principal du GT MAFTEC est de fédérer les recherches sur les différentes thématiques de la planification. Le but des travaux concernés est de modéliser et résoudre des problèmes complexes du monde réel, dans lesquels de nombreux agents (humains ou automatiques) interagissent de manière coopérative et robuste via des actions physiques, de communication et d'observation, ceci afin d'atteindre des buts communs dans un environnement partiellement imprédictible.

Dans un contexte où les états et les puissances supra-étatiques multiplient les recommandations et le règlement de l'intelligence artificielle, le groupe de travail Aspects computationnels de l'éthique (ACE) a comme objectifs : fédérer une communauté de recherche autour des problématiques en éthique artificielle ; développer les travaux en éthique artificielle ; tisser des liens interdisciplinaires (aspects juridiques et sociologiques de l'intelligence artificielle).

Date et lieu

- Date : 26-28 mars 2025
- Lieu : Maison des Sciences Humaines Alpes, 1221 rue des universités, St-Martin d'Hères

Programme (Jeudi 27 mars)

Introduction

- 09h00 Présentation MAFTEC : Tiago DE LIMA, Bruno ZANUTTINI
- 09h05 Présentation SMAA : Emmanuel ADAM, Gauthier PICARD
- 09h10 Présentation ACE : Grégory BONNET, Jean-Gabriel GANASCIA

MAFTEC + SMA

- 09h15 « Négociation pour la consommation adaptative d'allocation continue » par Ellie BEAU-PRÉZ, Anne-Cécile CARON, Maxime MORGE, Jean-Christophe ROUTIER.
- 09h40 « Un modèle agent pour la résolution de problèmes d'optimisation complexes. Application à la planification de missions de constellations de satellites hétérogènes » par Elsy KADDOUM, Jean-Pierre GEORGÉ.
- 10h05 « Coordination Temporelle de Plans d'Agents Interdépendants : Apports et Pistes d'Amélioration d'une Approche Distribuée » par Ajdin SUMIC, Thierry VIDAL.
- 10h30 Pause.

MAFTEC, aspects multi-agents

- 11h00 « Méthode de planification basée sur des contraintes de flux pour une mission de progression multi-agents avec des agents déployables et des contraintes de communication » par Émile SIBOULET, Simon LACROIX, Arthur BIT-MONNOT.
- 11h25 « Planification et exécution pour systèmes multi-robots trans-médias : structuration du problème et coordination/gestion des contraintes temporelles et séquentielles » par Virgile DE LA ROCHEFOUCAULD, Simon LACROIX.
- 11h50 « Apprentissage par renforcement multi-objectifs distributionnel » par Farès CHOUBAKI, Aurélie BEYNIER, Nicolas MAUDET, Paolo VIAPPANI.

— 12h15 Repas.

MAFTEC, planification et connaissances

- 14h00 « A Simple Integration of Epistemic Logic and Reinforcement Learning » par Thorsten ENGESSER, Thibaut LE MARRE, Emiliano LORINI, François SCHWARZENTRUBER, Bruno ZANUTTINI.
- 14h25 « Interprétation post-hoc de politiques pour les POMDP » par Geoffrey LAFOREST, Olivier BUFFET, Alexandre NIVEAU, Bruno ZANUTTINI.
- 14h50 « Planification pour l'adaptation d'agent autonome : du plan d'action à un plan d'expérience » par Victor LAVAIRYE, François TERRIER, Florian NOYRIT.
- 15h15 Pause.

MAFTEC, calcul de stratégies

- 15h45 « Améliorer l'efficacité de la planification QBF avec des nouveaux codages d'arbres compacts » par Frédéric MARIS.
- 16h10 « Computationally Feasible Strategies » par Catalin DIMA, Wojtek JAMROGA.
- 16h35 « The Complexity of Pure Maxmin Strategies in Two-Player Extensive-Form Games » par Junkang LI, Bruno ZANUTTINI, Véronique VENTOS.
- 17h00 « Une Recherche Optimale Gloutonne Basée sur MaxSAT pour la Planification TOHTN » par Gaspard QUENARD, Damien PELLIER.
- 17h25 **rump session.**

Programme (vendredi 28 mars 2025)

MAFTEC + ACE

- 09h25 « Un modèle éthique pour les MDP multi-agents » par Mihail STOJANOSKI, Nadjat BOURDACHE, Grégory BONNET, Abdel-Ilhah MOUADDIB.
- 09h50 « Moral Evaluation of Speech Acts - Truthfulness, Lies and Ethical Dilemmas - Modeled and Implemented with ASP » par Benjamin ICARD, Jean-Gabriel GANASCIA, Gauvain BOURGNE.
- 10h15 Pause.
- 10h45 « A Model of Compliant and Epistemic Human-Aware Task Planner which Anticipates Human Beliefs and Decision » par Rachid ALAMI.
- 11h10 « Apprentissage de décisions alignées sur des valeurs humaines, et humains dans la boucle » par Rémi CHAPUT.
- 11h35 « A Computationally Grounded Framework for Cognitive Attitudes » par Emiliano LORINI.
- 12h00 Repas.
- 13h45 **Réunion organisation.**

Organisation

Cet événement est organisé par Damien PELLIER (IMAG, Université de Grenoble Alpes).

GAS 2025

L'Association Française pour l'Intelligence Artificielle ([Afia](#)) et l'association internationale francophone d'Extraction et de Gestion des Connaissances ([Association EGC](#)) organisent, avec l'aide du groupe de travail « Gestion et Analyse des données Spatiales et Temporelles » ([GAST](#)), une journée commune « Gestion et Analyse des données Aériennes et Satellitaires » ([G2AS' 24](#)) sur le thème de la représentation, de la gestion, de l'analyse et du stockage des données sportives.

À travers ces thématiques, cette journée offrira une vision globale des enjeux et des opportunités qu'apporte l'analyse des données sportives, tant pour les chercheurs et analystes que pour les entraîneurs et pratiquants. Elle permettra d'échanger sur les avancées récentes et d'ouvrir des perspectives pour une meilleure exploitation des données au service de la performance et de la compréhension du sport.

Date et lieu

- Date : 13 mai 2025
- Lieu : Université de Caen, Campus 2, Avenue de la Côte de Nacre

Programme

- 09h00 Accueil.
- 09h30 Introduction.
- 10h00 « Surpoids, activité physique » par Guillaume WATTELEZ.
- 10h45 Pause.
- 11h00 « Tactiques en (e-)sport collectif » par Alexis MORTELIER.
- 11h45 « Graphes de phénomènes spatio-temporels, sports collectifs » par Camille GRANGE.
- 12h30 Repas.
- 14h00 Posters et démos.
- 14h15 « Analyse des mouvements de rééducation » par Maxime DEVANNE.
- 15h00 « Modèles markoviens, natation, escalade » par Nicolas VERGNE.
- 15h45 Pause.
- 16h00 « Tennis » par Nicolas BENGUIGUI.
- 16h45 « Tactiques à partir des données des sports d'équipe » par Albrecht ZIMMERMANN.
- 17h00 Mot de la fin.

Organisation

Cet événement est organisé par

- Clément Iphar, LETG/UBO (Brest) ;
- Albrecht Zimmermann, GREYC/UniCaen (Caen) ;
- Aurélie Leborgne, ICube/UNISTRA (Strasbourg) ;
- Loïc Salmon, ISEA/UNC (Nouméa – Nouvelle-Calédonie) ;
- Nida Meddouri, LRE/EPITA (Kremlin-Bicêtre).

Inscriptions

Si vous êtes intéressés pour participer (juste par votre présence ou par la présentation d'un poster) à cette journée, veuillez vous inscrire via le lien <https://forms.gle/dcRcEUJhXGob46yM7>.

Connaissances, Décisions & IA 2025

L'Association Française pour l'Intelligence Artificielle ([AFIA](#)), la Société d'Automatique, de Génie Industriel et Productique (SAGIP), et le GDR MACS organisent conjointement une journée commune « Connaissances, Décisions & IA ».

Cette journée a pour ambition de rassembler les communautés scientifiques, industrielles et académiques qui s'intéressent aux enjeux liés à la gestion des connaissances, à la prise de décision et à l'Intelligence Artificielle. Dans un esprit d'ouverture, nous souhaitons favoriser les échanges entre chercheurs, ingénieurs, praticiens et non-spécialistes autour de ces thématiques, afin de croiser les regards et d'explorer l'ensemble des défis actuels et futurs.

Nous encourageons la participation et les contributions portant sur tous les aspects des connaissances et de la décision en lien avec l'IA : approches théoriques, méthodologiques, algorithmiques, applications industrielles, questions organisationnelles, ou encore perspectives éthiques, juridiques, philosophiques ou sociétales. L'objectif est de créer un espace de dialogue interdisciplinaire pour mieux comprendre l'apport, les limites et les évolutions des systèmes intelligents dans les processus de décision et de gestion des savoirs, au service des entreprises comme de la société.

Date et lieu

- Date : 6 octobre 2025
- Lieu : Conservatoire National des Arts et Métiers (CNAM), Salle 31.2.85, 2, rue Conté, 75003 PARIS FRANCE

Programme (Session matinale)

- 08h30 Café d'accueil.
- 09h00 Ouverture de Benoit LE BLANC (Président de l'AFIA) et Laurent Geneste (de la SAGIP).
- 09h10 « Introduction de la journée » par Davy MONTICOLO (Université de Lorraine) et Nada MATTA (AFFILIATION).
- 09h20 « Intelligence Artificielle et aide à la décision, miroir (déformant) de nos usages ? » par Florence SEDES (IRIT, Université de Toulouse).
- 10h05 « NutriKG - un graphe de connaissances pour modéliser les préférences et les besoins nutritionnels » par Alexandre COMBEAU et Fatiha SAIS (LISN, Université Paris Saclay).
- 10h50 Pause.
- 11h15 « L'expertise dans une base de connaissances du système expert au système d'information et de connaissance : Rex de l'implantation dans l'industrie sur les 35 dernières années » par Alain BERGER (Directeur Général, société ARDANS – Knowledge Consulting & Software Solutions).

Programme (Session post-méridienne)

- 13h30 « Approche d'ingénierie des connaissances à base de systèmes "personnels" et partagés / LabCom Remind 4.0 » par Laurent GENESTE (ENI de Tarbes).
- 14h00 « Alignement de données hétérogènes et dynamiques pour la prise de décision : un défi toujours ouvert ? » par Oumaima EL HADDADI (IRIT, Université de Toulouse).
- 14h30 « Intégration des LLM en entreprise : valoriser les connaissances internes sans compromettre la confidentialité » par Pierre BOURGUEIL (Université de Technologie de Troyes).
- 15h00 « Lien entre connaissances et décisions, l'apport de l'Intelligence Artificielle en génie Industriel » par Bernard KAMSU-FOGUEM (ENI de Tarbes).
- 15h30 « Table ronde » par l'ensemble des conférenciers.
- 16h00 Clôture.



Afia

Association française
pour l'Intelligence Artificielle



Organisation

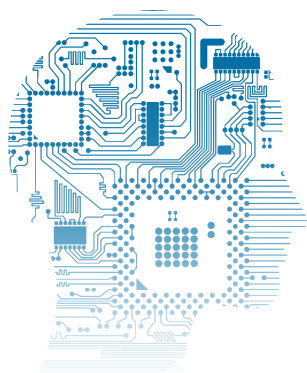
Cet événement est organisé par Davy MONTICOLO (Université de Lorraine) pour le Conseil d'Administration de l'AFIA et Nada MATTA (AFFILIATION) pour la SAGIP.

Inscriptions

PFIA 2024

Plate-forme
Intelligence
Artificielle

La rencontre annuelle de la communauté
Intelligence Artificielle autour de conférences
et ateliers thématiques



LA ROCHELLE
01-05 JUILLET
2024

pfia2024.univ-lr.fr

PROGRAMME

CONFÉRENCES

APIA - Applications Pratiques de
l'Intelligence Artificielle

CNIA - Conférence Nationale en
Intelligence Artificielle

IC - Journées francophones d'Ingénierie
des Connaissances

JIAF - Journées d'Intelligence Artificielle
Fondamentale

RJCIA - Rencontres des Jeunes
Chercheurs en Intelligence Artificielle

JOURNÉES

Agents & IA

Santé & IA

Société & IA

ATELIERS

Défense et IA

Jeux et IA

MAFTEC

SOSEM

CÉCILIA

IA en Nouvelle-Aquitaine

GdR RADIA

TUTORIELS

8 tutoriels répartis sur 5 jours

PFIA 2025

PLATEFORME
INTELLIGENCE
ARTIFICIELLE





Afia

Association française
pour l'Intelligence Artificielle



En partenariat avec



L'AVENIR EST AUX VALEURS SÛRES