



Mémoire de fin d'étude

Pour l'obtention du diplôme de Master

Filière : Génie industriel

Spécialité : Management industriel et logistique

Présenté par :

Anes BENAYAD

Hind BEKHECHI

Thème

**Utilisation des techniques d'IA pour améliorer
l'efficacité de la distribution des produits pétroliers
Cas de l'entreprise : NAFTAL**

Soutenu publiquement, 16 juin 2026, devant le jury composé de :

- | | | | |
|----------------------------|------------|---------------|--------------------------|
| - M. Mohammed BENNEKROUF | MCA | ESSA. Tlemcen | Président |
| - Mme. Zoulikha BELMEKKI | MCB | ESSA. Tlemcen | Directrice de mémoire |
| - M. Djamel BENNACER | Docteur | NAFTAL | Co- Directeur de mémoire |
| - M. Mehdi SOUIER | Professeur | Univ. Tlemcen | Examineur |
| - M. Mohamed Amine CHERIER | MCB | ESSA. Tlemcen | Examineur |

Année universitaire : 2025/2026

Résumé

Ce mémoire de master traite de l'amélioration de l'efficacité de la distribution des produits pétroliers chez Naftal à travers les techniques d'intelligence artificielle. Le système proposé repose sur une approche hybride combinant la prédiction de la demande et l'optimisation par apprentissage par renforcement. Grâce à l'analyse des données historiques, des modèles prédictifs ont été développés pour anticiper les besoins des stations-service, permettant ainsi une planification proactive. Couplé à une application de visualisation, ce moteur intelligent adapte les flux logistiques en temps réel aux variations du marché, garantissant une meilleure réactivité et une optimisation continue de la performance de la chaîne logistique.

Mots-clés : Intelligence Artificielle, Prédiction de la demande, Apprentissage par Renforcement, Naftal, Logistique pétrolière, Optimisation.

Abstract

This master's thesis addresses enhancing petroleum product distribution efficiency at Naftal using artificial intelligence. The proposed system is based on a hybrid approach combining demand forecasting and reinforcement learning optimization. By analyzing historical data, predictive models were developed to anticipate gas station requirements, enabling proactive planning. Integrated with a visualization application, this intelligent engine adjusts logistical flows in real-time to market variations, ensuring increased responsiveness and continuous optimization of the supply chain performance.

Keywords: Intelligence Artificielle, Demand Forecasting, Reinforcement Learning, Naftal, Petroleum Logistics, Optimization.

ملخص

تتناول مذكرة الماستر هذه تحسين كفاءة توزيع المنتجات النفطية في شركة نفطال باستخدام تقنيات الذكاء الاصطناعي. يعتمد النظام المقترح على نهج هجين يجمع بين التنبؤ بالطلب والتحسين عن طريق التعلم المعزز. من خلال تحليل البيانات التاريخية، تم تطوير نماذج تنبؤية لاستباق احتياجات محطات الخدمة، مما يسمح بخطط استباقية. يشمل هذا المحرك الذكي، المرتبط بتطبيق للعرض المرئي، على تعديل التدفقات اللوجستية في الوقت الفعلي وفقاً لتقلبات السوق، مما يضمن استجابة أفضل وتحسيناً مستمراً لأداء سلسلة التوريد.

كلمات مفتاحية : الذكاء الاصطناعي، التنبؤ بالطلب، التعلم المعزز، نفطال، اللوجستيات النفطية، التحسين.

Remerciements

Avant tout, nous remercions Dieu, le Tout-Puissant, pour nous avoir donné la force, la patience et la volonté nécessaires à l'accomplissement de ce travail.

Nous tenons à exprimer notre profonde reconnaissance à Monsieur Fouad MALIKI, chef de filière, pour son accompagnement constant et son encadrement de qualité tout au long de ce projet. Sa disponibilité exemplaire, sa patience, et la pertinence de ses remarques nous ont permis de progresser avec confiance dans la réalisation de ce travail. Sa bienveillance, son exigence constructive et son sens pédagogique ont constitué un véritable soutien, tant sur le plan technique que personnel. Ce mémoire n'aurait pu aboutir sans ses conseils avisés et son engagement sincère dans notre évolution académique.

Nos remerciements les plus chaleureux s'adressent également à notre encadrante et directrice de ce mémoire, Madame Zoulikha KOUDAD, pour son suivi rigoureux et sa précieuse contribution à la réussite de ce projet. Nous la remercions pour la générosité de son encadrement, sa grande disponibilité et pour nous avoir transmis son savoir avec autant de bienveillance. Ses encouragements et sa vision éclairée ont été des éléments moteurs dans l'avancement de nos recherches.

Nous tenons à exprimer notre gratitude particulière à Monsieur Djamel BENNACER, notre encadrant au sein de l'entreprise NAFTAL, qui nous a ouvert les portes de l'institution et a grandement facilité notre intégration. Sa rigueur professionnelle, son sens de l'analyse et les éclairages concrets qu'il nous a apportés sur le terrain nous ont permis de mieux adapter notre démarche aux exigences réelles de l'entreprise. Qu'il trouve ici l'expression de notre profond respect pour son accueil et son soutien technique inestimable.

Nous adressons nos sincères remerciements aux membres du jury pour l'honneur qu'ils nous font en acceptant d'évaluer notre travail. Nous tenons à exprimer notre profonde gratitude à Monsieur Mohammed BENNEKROUF, qui nous fait l'honneur de présider ce jury. Nos remerciements s'adressent également à Messieurs Mehdi SOUIER et Mohamed Amine CHERIER, pour avoir accepté de siéger en qualité d'examineurs. Nous leur sommes particulièrement reconnaissants pour le temps qu'ils ont consacré à la lecture de ce mémoire, ainsi que pour l'intérêt qu'ils y ont porté. Leurs remarques et leurs critiques constructives contribueront sans doute à enrichir notre réflexion et à améliorer la qualité de ce travail.

Nous remercions nos familles respectives et nos amis les plus proches pour leur amour inconditionnel et leur soutien indéfectible. Leur présence bienveillante et leurs encouragements

ont été une source de force précieuse, notamment durant les périodes de doute et de fatigue. Ce mémoire leur est dédié, en reconnaissance de tout ce qu'ils ont fait pour nous permettre d'arriver jusqu'ici.

Enfin, nous tenons à témoigner notre profonde gratitude à l'ensemble du département de Génie Industriel pour l'excellence de la formation dispensée et la qualité de l'encadrement dont nous avons bénéficié tout au long de notre parcours. Nous saluons l'engagement constant du corps enseignant, qui, par la diversité des enseignements et une approche résolument tournée vers le monde professionnel, nous a permis d'acquérir des compétences solides tout en forgeant notre esprit critique et notre rigueur scientifique. La bienveillance et le soutien de l'ensemble du personnel administratif et pédagogique ont été des piliers essentiels de notre progression, transformant ces années d'études en une expérience humaine et intellectuelle profondément formatrice et mémorable.

Hind & Anes

Dédicace

C'est avec une émotion profonde et une immense fierté que je dédie ce modeste fruit de plusieurs années d'études à ceux qui ont été les piliers de ma vie :

À mes très chers parents,

Aucune dédicace ne saurait exprimer l'amour et la reconnaissance que je vous porte. Vous avez été mon premier soutien, sacrifiant tant pour ma réussite et veillant toujours à ce que je ne manque de rien. Vos prières ont été mon bouclier, et votre confiance inébranlable a été le moteur de ma persévérance. Que ce travail soit le reflet de votre éducation et le témoignage de ma gratitude éternelle.

À mon cher frère et ma chère sœur,

Pour votre complicité irremplaçable et votre soutien sans faille au quotidien. Vous avez su partager mes moments de doute comme mes plus grandes joies, m'encourageant toujours à aller au bout de mes rêves. Ce mémoire est aussi le vôtre, symbole de notre lien fraternel indéfectible et de l'affection infinie que je vous porte.

À ma famille adorée,

À mes grands-parents, mes oncles, mes tantes et mes cousins, qui forment le socle de mes racines. Merci pour votre affection, vos conseils précieux et pour avoir toujours célébré mes réussites avec autant de ferveur. Votre présence chaleureuse et constante à mes côtés est une source de motivation et une immense fierté pour moi.

À mon binôme, Anes,

Avec qui j'ai partagé les longues heures de travail, les défis techniques et les réussites de ce projet. Merci pour ta patience, ton esprit d'équipe et pour avoir transformé ce défi académique en une aventure humaine inoubliable. Ta rigueur et ton amitié ont rendu ce parcours bien plus enrichissant et merveilleux.

À mes amis les plus proches,

À vous qui avez rendu ce parcours universitaire plus léger, plus beau et rempli de souvenirs précieux. Merci pour vos rires, votre écoute et votre fidélité dans les moments les plus intenses de ce projet. Je suis chanceuse de vous avoir à mes côtés pour célébrer aujourd'hui cette étape cruciale de ma vie.

Hind BEKHECHI

Dédicace

À mes très chers parents,

Vous êtes la raison pour laquelle je me suis relevée à chaque fois que je tombais. Sans un mot de trop, sans une plainte, vous avez porté mes rêves comme s'ils étaient les vôtres. Ce diplôme, c'est autant le vôtre que le mien — vous en avez payé le prix en silence, avec amour et dignité. Je vous dois tout.

À mon cher frère et ma chère sœur,

Vous avez été mes premiers confidents et mes plus fidèles alliés. Dans les moments où tout semblait lourd, votre simple présence suffisait à alléger le poids. Grandir à vos côtés est la chance la plus belle que la vie m'ait offerte. Cette réussite, je la partage avec vous de tout cœur.

À ma famille,

À ceux qui ont toujours gardé une place pour moi autour de la table, une pensée douce dans leurs prières et un mot d'encouragement au bon moment. Vous m'avez appris que les racines ne retiennent pas — elles donnent la force de s'élever. Je vous porte en moi où que j'aille.

À ma binôme, Hind,

Tu es bien plus qu'une coéquipière — une amie sincère et un miroir de passion pour le travail bien fait. Ta rigueur et ton esprit collaboratif ont été une source de motivation tout au long de ce parcours. C'est avec respect et fierté que j'ai partagé cette aventure à tes côtés. Merci pour ton amitié et pour avoir rendu ce parcours inoubliable.

À mes amis,

À vous qui avez su être là sans que j'aie besoin de demander. Vous avez coloré ces années d'université de moments que je n'échangerais pour rien au monde. Merci d'avoir été mon équilibre, ma légèreté et ma joie dans ce long chemin. Cette page se tourne avec vous, et les prochaines aussi.

Anes BENAYAD

Table des matières

Table des figures	v
Liste des tableaux	vi
Liste des sigles et acronymes	vii
Introduction générale	1
1 Intelligence artificielle et Apprentissage Multi-Agent	2
1.1 Introduction à l'Intelligence Artificielle	2
1.1.1 Définition de l'Intelligence Artificielle	2
1.2 Types d'apprentissage en Intelligence Artificielle	2
1.2.1 Apprentissage supervisé	2
1.2.2 Apprentissage non-supervisé	4
1.2.3 Apprentissage par renforcement	6
1.3 Apprentissage supervisé et réseaux de neurones	7
1.3.1 Introduction aux réseaux de neurones artificiels	7
1.3.2 Le perceptron simple	8
1.3.3 Algorithme du perceptron	9
1.3.4 Perceptron multicouche (MLP) et rétropropagation du gradient	10
1.4 Apprentissage profond (Deep learning)	17
1.4.1 Réseaux de neurones convolutifs (CNN)	18
1.4.2 Réseaux de neurones récurrents (RNN)	18
1.5 Apprentissage par renforcement	19
1.5.1 Définition et concepts fondamentaux	19
1.5.2 Politique et fonction de valeur	20
1.5.3 Programmation dynamique	21
1.5.4 Apprentissage par difference temporelle	22
1.6 Apprentissage par renforcement profond (Deep RL)	26
1.6.1 Deep Q-Network (DQN)	26
1.6.2 DDPG	27
1.7 Systèmes Multi-Agents	28

1.7.1	Définition d'un système multi-agents	28
1.7.2	Les éléments d'un S.M.A	29
1.7.3	Architecture des systèmes multiagents	30
1.7.4	Coopération vs compétition	31
1.8	Apprentissage par renforcement multi-agent (MARL)	31
1.8.1	Formalisation mathématique (Dec-MDP)	31
1.8.2	Propriété d'Observabilité Collective	32
1.8.3	Paradigmes d'entraînement en MARL	32
1.8.4	Méthodes coopératives centralisées	34
1.9	Les Modèles de Prédiction Séquentielle : LSTM et GRU	36
1.9.1	Motivation : Le problème du gradient disparaissant	36
1.9.2	Le réseau LSTM (Long Short-Term Memory)	36
1.9.3	Le réseau GRU (Gated Recurrent Unit)	38
1.9.4	Comparaison LSTM et GRU	38
1.9.5	Choix de l'architecture pour la prédiction de séries temporelles	38
1.10	Conclusion	39
2	Présentation de l'entreprise	40
2.1	Introduction	40
2.2	Historique et évolution de l'entreprise NAFTAL	40
2.3	Principales tâches et responsabilités de NAFTAL	41
2.3.1	Identification et Organisation des Moyens	41
2.3.2	Gestion Opérationnelle et Approvisionnement	42
2.3.3	Maintenance, Sécurité et Liaisons Externes	42
2.3.4	Organisation de naftal	42
2.3.5	Réseau de distribution et moyens de transport	47
2.4	Présentation du centre « Enfuteur » NAFTAL Tlemcen	48
2.4.1	Présentation de la canalisation multiproduit ASR	48
2.4.2	Process du Terminal d'Arrivée REMCHI	50
2.4.3	Gestion des Stocks de Carburants	52
2.4.4	Les opérations de chargement	53
2.4.5	Laboratoire de contrôle de qualité	53
2.5	Conclusion	54
3	Prédiction de la demande et planification optimisée de la distribution des carburants par systèmes multi-agents à apprentissage par renforcement	55
3.1	Introduction	55
3.2	Problématique	56
3.3	Objectifs du projet	57
3.4	Présentation des données de Naftal	57

3.4.1	Caractérisation de la flotte de camions-citernes	57
3.4.2	Prétraitement et nettoyage des données	58
3.5	Configuration du réseau de distribution	60
3.6	Formalisation du modèle d'Apprentissage par Renforcement Multi-Agents . .	66
3.6.1	Fonction de Récompense	67
3.6.2	Modélisation des contraintes métier	67
3.7	Approche de résolution par Apprentissage par Renforcement Multi-Agents . .	68
3.7.1	L'algorithme de résolution : Q-Learning Tabulaire	68
3.7.2	Composants clés de l'implémentation	69
3.7.3	Mécanisme de coordination par Vagues	69
3.8	Mise en œuvre du moteur MARL au sein de l'application	69
3.8.1	Évolution du système : Du MILP au MARL	70
3.8.2	L'interface comme support de décision	71
3.8.3	Protocole d'analyse des résultats	72
3.9	Analyse détaillée des cycles journaliers	74
3.10	Synthèse et Analyse Hebdomadaire des Performances	84
3.11	Analyse comparative : Approche Classique (MILP) vs Intelligence Artificielle (MARL)	85
3.12	Bilan final et recommandation	88
3.13	Moteur de Prédiction IA : Anticipation de la Demande par Réseaux de Neurones LSTM	88
3.13.1	Stratégie de Segmentation et Classification des Stations	89
3.13.2	Ingénierie des Données et Encodage Temporel	89
3.13.3	Architecture du Modèle LSTM et Entraînement	90
3.13.4	Logique de Post-traitement Métier	91
3.13.5	Entraînement et Fenêtre Temporelle	91
3.14	Optimisation des tournées des demandes prédites	92
3.14.1	Approche par Programmation Linéaire en Nombres Entiers (MILP) . .	93
3.14.2	Approche par Apprentissage par Renforcement Multi-Agents (MARL) .	93
3.15	Conclusion	95
Conclusion générale		96

Table des figures

1.1	Illustration de deux problèmes de classification en 2 dimensions	4
1.2	Illustration d'un problème de régression en une dimension	4
1.3	Graphiques représentant des données linéairement séparables et non linéairement séparables.	8
1.4	Fonctionnement de l'algorithme du perceptron.	9
1.5	Exemple d'un réseau de type perceptron multicouche	11
1.6	Modèle du neurone j.	12
1.7	Gradient de l'erreur totale.	13
1.8	Architecture générale d'un réseau de neurones convolutif	18
1.9	Couche de neurones récurrents devant une couche de neurones traditionnels	19
1.10	Les trois principales approches de l'apprentissage par renforcement	26
1.11	L'architecture DQN	27
1.12	L'architecture DDPG	28
1.13	Organigramme de l'algorithme de gradient de politique déterministe profond (DDPG)	29
1.14	Système multi-agents.	29
1.15	L'architecture d'un système multi-agent fonctionnant sur réseau.	30
1.16	Le cadre d'entraînement de l'algorithme MADDPG pour le problème OPEG	36
1.17	Structure Détaillée D'une Cellule LSTM	37
2.1	Logo NAFTAL	41
2.2	Organigramme	48
2.3	Pipelines de canalisation	48
2.4	Chaîne de valeur du carburant	49
2.5	Carburant arrivé par le pipeline	52
3.1	Architecture visuelle et opérationnelle de l'application.	70
3.2	a) Sélection des stations	71
3.3	b) Choix de la journée	71
3.4	Tableau de bord de synthèse et indicateurs de performance globale.	72
3.5	Plan de distribution détaillé par point de vente et compartimentage.	72
3.6	Identification et gestion capacitive de la flotte mobilisée.	73

3.7	Structuration des rotations et indicateurs de performance des tournées.	73
3.8	Visualisation spatiale des points de livraison et des trajectoires sous OpenStreetMap.	74
3.9	Plan de livraison carburant extrait de l'application (dimanche 15/02).	74
3.10	Plan de livraison carburant extrait de l'application (lundi 16/02).	76
3.11	Plan de livraison carburant extrait de l'application (mardi 17/02).	77
3.12	Plan de livraison carburant extrait de l'application (mercredi 18/02).	79
3.13	Plan de livraison carburant extrait de l'application (jeudi 19/02).	81
3.14	Plan de livraison carburant extrait de l'application (vendredi 20/02).	82
3.15	Plan de livraison carburant extrait de l'application (samedi 21/02).	84
3.16	Résumé hebdomadaire consolidé des performances de livraison.	85
3.17	Comparaison MILP vs MARL - Lundi 16/02/2026.	86
3.18	Comparaison MILP vs MARL - Mercredi 18/02/2026.	86
3.19	Comparaison MILP vs MARL - Jeudi 19/02/2026.	87
3.20	Comparaison MILP vs MARL - Vendredi 20/02/2026.	87
3.21	Architecture du Réseau LSTM pour la prédiction de la demande	90
3.22	Interface de paramétrage et exécution de l'IA.	91
3.23	Résultats des prédictions par journée.	91
3.24	Aperçu des demandes générée	92
3.25	Plan de livraison optimisé par MILP (Vendredi 01 Mai 2026).	93
3.26	Plan de livraison optimisé par MARL (Vendredi 01 Mai 2026).	94

Liste des tableaux

1.1	Comparaison entre les architectures LSTM et GRU	38
3.1	Configuration technique de la flotte de transport Naftal pour l'apprentissage.	57
3.2	Demandes enregistrées le Dimanche 15/02/2026.	60
3.3	Demandes enregistrées le Lundi 16/02/2026.	61
3.4	Demandes enregistrées le Mardi 17/02/2026.	62
3.5	Demandes enregistrées le Mercredi 18/02/2026.	63
3.6	Demandes enregistrées le Jeudi 19/02/2026.	64
3.7	Demandes enregistrées le Vendredi 20/02/2026.	65
3.8	Demandes enregistrées le Samedi 21/02/2026.	65
3.9	Bilan comparatif des deux méthodes d'optimisation.	88

Liste des sigles et acronymes

Abréviation	Signification
CNN	Réseaux de Neurones Convolutifs (<i>Convolutional Neural Network</i>)
DQN	<i>Deep Q-Network</i>
KQML	<i>Knowledge Query and Manipulation Language</i>
MADDPG	<i>Multi-Agent Deep Deterministic Policy Gradient</i>
MARL	Multi-Agent Reinforcement Learning
MDP	Processus de Décision Markovien (<i>Markov Decision Process</i>)
ML	Machine Learning (Apprentissage automatique)
MSE	<i>Mean Squared Error</i>
OPEG	<i>Optimal Power Economic Game</i>
POMDP	Processus de Décision Markovien Partiellement Observable
PuLP	<i>Python Linear Programming library</i>
REINFORCE	Algorithme de gradient de politique (Williams, 1992)
RL	Apprentissage par Renforcement (<i>Reinforcement Learning</i>)
RNN	Réseaux de Neurones Récurents (<i>Recurrent Neural Network</i>)
SEARN	<i>Search and Learn</i> (algorithme d'apprentissage)
SMA	Système Multi-Agents
TD	Différence Temporelle (<i>Temporal Difference</i>)
API	<i>Application Programming Interface</i>
ISO	<i>International Organization for Standardization</i>
OPC DA	OLE for Process Control – Data Access
TCP	<i>Transmission Control Protocol</i>
ASR	Arzew – Sidi Bel Abbès – Remchi (canalisation multiproduit)
GD	Gestion Directe (station gérée par NAFTAL)
PVA	Points de Vente Agréés
SARL	Société à Responsabilité Limitée

Introduction générale

À l'ère de l'industrie 4.0, la performance d'une organisation ne se mesure plus seulement à sa capacité de production, mais de plus en plus à l'agilité et à l'intelligence de son système de distribution. Dans un paysage économique marqué par une volatilité croissante et des exigences de rentabilité strictes, la logistique est devenue un levier stratégique majeur. Pour un acteur central de l'économie nationale tel que Naftal, la maîtrise des flux de produits pétroliers représente un défi de souveraineté et d'efficacité opérationnelle. Il s'agit d'une mission complexe où la satisfaction de la demande doit impérativement s'aligner sur des protocoles de sécurité rigoureux et une gestion optimale des ressources matérielles.

Cependant, le secteur de la distribution d'hydrocarbures fait face à des contraintes structurelles lourdes. La variabilité des besoins au niveau des stations-service, la densité des réseaux routiers et les spécificités propres au transport de matières dangereuses — notamment la gestion des capacités de stockage et le compartimentage des véhicules — rendent la planification manuelle ou traditionnelle de plus en plus obsolète. Les méthodes classiques, bien qu'éprouvées par le passé, peinent aujourd'hui à absorber cette complexité et ne permettent pas toujours de réagir avec la célérité nécessaire aux aléas du terrain.

Face à ces limites, l'intégration de technologies avancées issues du numérique s'impose comme une nécessité pour transformer une logistique souvent subie en une logistique proactive. L'enjeu de ce travail de recherche est d'explorer comment l'exploitation intelligente des données peut révolutionner le processus décisionnel au sein de l'entreprise. En passant d'une approche statique à une approche dynamique, l'objectif est de concevoir un système capable d'anticiper les ruptures de stock tout en orchestrant les mouvements de la flotte de manière fluide et coordonnée.

L'ambition de ce mémoire est donc de démontrer que la modernisation de la distribution pétrolière repose sur la capacité à traiter l'information en temps réel pour optimiser chaque kilomètre parcouru. Au-delà de la simple réduction des coûts de transport, notre démarche vise à valider un véritable outil d'aide à la décision. Ce système doit permettre à Naftal de garantir un taux de service irréprochable, de renforcer la sécurité des flux sur le réseau routier et d'optimiser l'utilisation de son parc de véhicules. En s'appuyant sur des scénarios réels, nous illustrerons comment une vision innovante de la logistique peut bâtir une chaîne d'approvisionnement plus résiliente, plus intelligente et résolument tournée vers les défis futurs du secteur énergétique.

Chapitre 1

Intelligence artificielle et Apprentissage Multi-Agent

1.1 Introduction à l'Intelligence Artificielle

L'intelligence artificielle est un domaine en pleine croissance qui a le potentiel de transformer de nombreux secteurs, notamment l'industrie. En tant qu'ingénieur en génie industriel, il est essentiel de comprendre les bases de l'IA ainsi que ses applications potentielles dans le management industriel et la logistique.

L'IA fait référence au développement de systèmes informatiques capables d'effectuer des tâches qui nécessitent généralement l'intelligence humaine, telles que la reconnaissance de modèles, l'apprentissage à partir de données, l'optimisation des processus et la prise de décision.

1.1.1 Définition de l'Intelligence Artificielle

La construction de programmes informatiques qui s'adonnent à des tâches qui sont, pour l'instant, accomplies de façon plus satisfaisante par des êtres humains car elles demandent des processus mentaux de haut niveau tels que : l'apprentissage perceptuel, l'organisation de la mémoire et le raisonnement critique” par John McCarthy et Marvin Minsky en 1956.

1.2 Types d'apprentissage en Intelligence Artificielle

1.2.1 Apprentissage supervisé

L'apprentissage supervisé correspond au cas où l'objectif de l'apprentissage est déterminé explicitement via la définition d'une cible à prédire. Dans ce cas, $\mathcal{D}$ correspond à un ensemble de n paires **d'entrées** x_t et de **cibles** associées y_t :

$$\mathcal{D} = \{(x_t, y_t) \mid x_t \in \mathcal{X}, y_t \in \mathcal{Y}\}_{t=1}^n$$

Typiquement, un tel ensemble est récolté en fournissant l'ensemble des entrées à un groupe de personnes et en leur demandant d'associer à chacune de ces entrées une cible appropriée dans le contexte du problème à résoudre. La tâche d'un algorithme d'apprentissage est alors d'entraîner un modèle qui puisse imiter ce processus d'étiquetage par un humain, i.e., qui puisse prédire pour une entrée x quelconque la valeur de la cible y qui aurait normalement été donnée par un humain. Cependant, les algorithmes d'apprentissage ne se limitent pas à la modélisation du comportement de l'humain, et peuvent être utilisés pour modéliser la relation liant des paires d'entrées et de cibles provenant d'un autre phénomène (e.g., la relation entre une action et son prix à la bourse telle que générée par les marchés financiers). La nature de l'ensemble $\mathcal{Y}$ d'où proviennent les cibles dépendra du type de problème à résoudre. Deux types de problèmes fréquents sont les problèmes de **classification** et de **régression**.

Classification

Dans le cadre d'un problème de classification, la variable y correspond à un ensemble fini de classes auxquelles peuvent appartenir les différentes entrées possibles $x \in \mathcal{X}$. Au cas où x c'est une séquence, $\mathcal{Y}$ peut également être un ensemble de séquences de classes. Voici d'ailleurs quelques exemples de problèmes de classification :

Reconnaissance de caractère : Dans cet exemple, $x \in \mathcal{X}$ correspond à la représentation vectorielle des valeurs des pixels d'une image, et $y \in \mathcal{Y}$ correspond au caractère associé, par exemple un chiffre compris entre 0 et 9.

Classification syntaxique : Dans cet exemple, $x \in \mathcal{X}$ correspond à une phrase dans une langue donnée, et $y \in \mathcal{Y}$ est la séquence des classes syntaxiques (*part-of-speech tags*) de chacun des mots de x . La figure I.1 présente une illustration de problèmes de classification en deux dimensions. Afin de mesurer la performance d'un modèle de classification, on utilise normalement le taux d'erreur de classification, i.e., la proportion du nombre de cas où la sortie donnée par le modèle ne correspond pas à la cible attendue. Lorsque le modèle f donne une probabilité pour chaque classe possible, la log-vraisemblance négative de la classe cible peut aussi être utilisée. Comme on le verra plus tard, cette mesure de performance a l'avantage d'être dérivable par rapport à la sortie du modèle, et peut ainsi être optimisée à l'aide d'un algorithme de descente de gradient.

Régression

Dans un problème de régression, la variable y correspond à un ensemble de valeurs continues ou de vecteurs de valeurs continues. Voici quelques exemples de tels problèmes :

Prédiction du poids : $x \in \mathcal{X}$ correspond aux caractéristiques physiques d'une personne (par exemple, l'âge, le sexe, etc.), et $y \in \mathcal{Y}$ correspond à son poids.

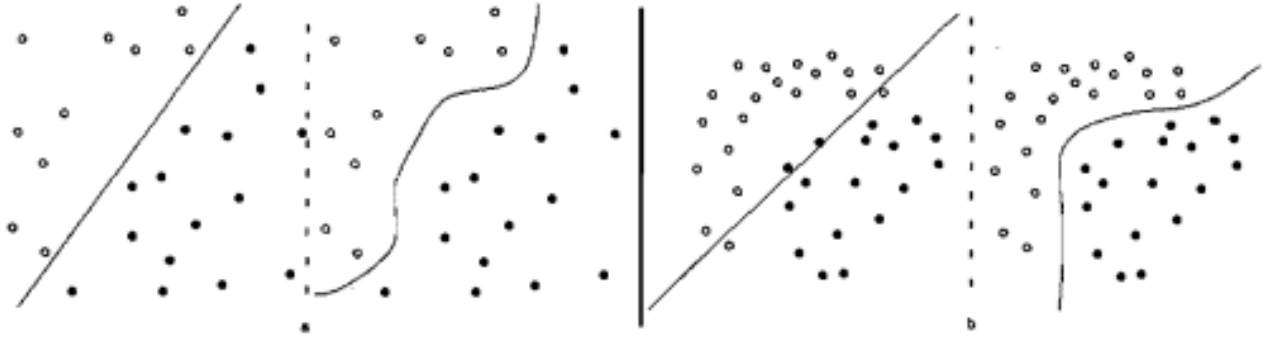


Fig. 1.1 : Illustration de deux problèmes de classification en 2 dimensions

Prédiction de la valeur d'une action : $x \in \mathcal{X}$ correspond à une séquence d'indicateurs du marché observés sur différentes journées, et $y \in \mathcal{Y}$ correspond à l'évolution de la valeur d'un action.

La figure I.2 présente un exemple de problème de régression en une dimension. La performance d'un modèle de régression est typiquement mesurée en calculant la différence au carré entre la sortie du modèle et la cible attendue, bien que d'autres mesures d'évaluation soient également possibles.

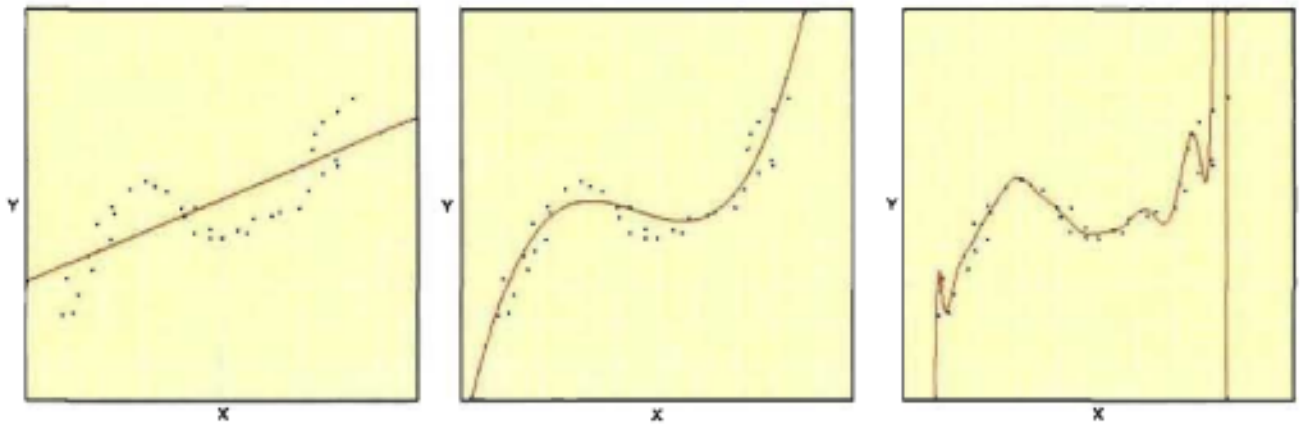


Fig. 1.2 : Illustration d'un problème de régression en une dimension

1.2.2 Apprentissage non-supervisé

L'apprentissage non-supervisé correspond au cas où aucune cible n'est prédéterminée. Ainsi, l'ensemble d'entraînement ne contient que des entrées :

$$\mathcal{D} = \{x_t \mid x_t \in \mathcal{X}\}_{t=1}^n$$

et ne définit pas explicitement la nature de la fonction f qui doit être retournée par l'algo-

rythme d'apprentissage. Ainsi, c'est plutôt l'utilisateur qui doit spécifier le problème à résoudre. Par contre, dans tous les cas, le modèle capture certains éléments de la véritable distribution ayant généré $\mathcal{D}$. Voici un bref survol des problèmes d'apprentissage non-supervisé les plus répandus.

- **Estimation de densité :** Pour ce problème, J doit fournir une estimation de la fonction de densité (ou de probabilité) de la distribution ayant généré les éléments de D . Afin de résoudre ce problème, une approche possible pour un algorithme d'apprentissage consiste à spécifier une forme précise de fonction de densité p valide pour le modèle J , puis à ajuster les paramètres de cette fonction afin d'obtenir une valeur élevée de

$$J(X_t) = p(X_t)$$

pour tous les exemples de l'ensemble D .

Afin d'être une fonction de densité valide, la fonction $p(x)$ doit satisfaire les deux conditions suivantes :

$$p(x) \geq 0 \quad \forall x \in \mathcal{X},$$

$$\int_{\mathcal{X}} p(x) dx = 1.$$

Ainsi, après l'entraînement, le modèle contient une information importante sur la structure statistique des données d'entrée. Par exemple, lorsqu'un des éléments X_i d'une entrée est manquant, il est possible d'utiliser un tel modèle d'estimation de densité afin de trouver une valeur vraisemblable pour cet élément, en exploitant le fait que $X_i \in \mathcal{X}_i$, où $\mathcal{X}_i$ représente l'ensemble des valeurs possibles de la variable X_i .

$$p(x_i \mid x_1, \dots, x_{i-1}, \dots, x_{i+1}, \dots, x_d) = \frac{p(x)}{\sum_{x'_i \in \mathcal{X}_i} p(x_1, \dots, x'_i, \dots, x_d)}$$

- **Réduction de dimensionnalité :**

L'objectif de cette tâche est d'apprendre une nouvelle représentation $f(x)$ de l'entrée, qui soit plus utile que la représentation vectorielle originale x . Cette représentation prend généralement la forme d'un vecteur, dont les éléments (appelés *caractéristiques* ou *features*) sont calculés par le modèle f .

En général, l'utilité de cette représentation dépend de la tâche que l'on souhaite réellement résoudre. En d'autres termes, l'extraction de caractéristiques n'est normalement pas une fin en soi, mais plutôt un moyen d'améliorer un autre algorithme d'apprentissage, dont le modèle f^* utiliserait alors cette nouvelle représentation et serait entraîné sur le nouvel ensemble de données

$$\mathcal{D}' = \{(f(X_t), Y_t)\}_{t=1}^m.$$

La qualité de l'extraction de caractéristiques peut être mesurée de différentes façons, en fonction du contexte de l'application. Le critère le plus souvent considéré est l'amélioration de la performance de cet autre algorithme. On souhaite alors que la prédiction $f^*(f(x))$ d'un modèle entraîné à partir des caractéristiques extraites soit meilleure que celle d'un modèle entraîné à partir de la représentation originale des entrées.

Un autre critère est le gain en efficacité de la prédiction $f^*(f(x))$. Ce gain peut être dû à la réduction du calcul nécessaire pour effectuer une prédiction ou à la diminution du coût réel du calcul, dans le cas où un coût (par exemple financier) est associé à l'obtention des éléments de l'entrée.

- **Groupeage :**

Le problème du groupeage (*clustering*) correspond à la recherche d'un partitionnement de l'espace $\mathcal{X}$ en g sous-ensembles (ou groupes) notés $Q_1, \dots, Q_g$. Chacun de ces sous-ensembles Q_i peut éventuellement être associé à un prototype X_i qui résume de manière représentative les données de $\mathcal{D}$ contenues dans Q_i .

Ainsi, la fonction f doit fournir, pour chaque vecteur d'entrée x , l'indice i du sous-ensemble Q_i auquel il appartient. Dans le domaine du forage de données, le groupeage est souvent utilisé comme outil de visualisation d'un ensemble de données, dans le but de découvrir une structure de groupes latente.

Le groupeage peut également être utilisé pour compresser un ensemble de données en le remplaçant par l'ensemble de ses prototypes. Il est même possible d'extraire des caractéristiques à l'aide du groupeage, par exemple en considérant le vecteur binaire d'association aux différents sous-ensembles

$$(I_{f(x)=1}, \dots, I_{f(x)=g})$$

comme vecteur de caractéristiques, où I désigne la fonction indicatrice.

1.2.3 Apprentissage par renforcement

L'apprentissage par renforcement, c'est apprendre à agir par essai et erreur. Dans ce paradigme, un agent peut percevoir son état et effectuer des actions. Après chaque action, une récompense numérique est donnée. Le but de l'agent est de maximiser la récompense totale qu'il reçoit au cours du temps.

1.3 Apprentissage supervisé et réseaux de neurones

1.3.1 Introduction aux réseaux de neurones artificiels

Le réseau de neurones artificiels représente un sous-domaine de l'apprentissage automatique, il s'agit d'un système inspiré des réseaux neuronaux biologiques du cerveau humain dans le but de reproduire le comportement de ce dernier. C'est l'algorithme le plus approprié pour effectuer des tâches telles que la reconnaissance d'image, le traitement de langage naturel et la reconnaissance vocale.

Le réseau neuronal se compose d'une collection d'unités connectées ou nœuds appelés neurones artificiels : chaque connexion, appelée synapse comme celui d'un cerveau biologique, peut transmettre un signal d'un neurone à un autre. Un neurone artificiel reçoit un signal, le traite puis en transmet un autre à son tour vers d'autres neurones qui sont liés à lui. Chaque synapse est associée à un poids synaptique qui augmente ou réduit l'ampleur du signal. Le signal est un nombre réel et la sortie de chaque neurone est traitée en calculant la somme des signaux d'entrées pondérés puis en faisant passer cette dernière par une fonction d'activation ⁵. La somme peut être mathématiquement représentée comme suit :

$$\sum_{i=1}^n E_i \times P_i$$

où E_i représente une entrée et P_i le poids associé. La fonction d'activation introduit la non-linéarité dans le résultat, c'est-à-dire la sortie du neurone, ce qui permet au réseau d'apprendre des relations complexes entre entrées et sorties. Différentes fonctions d'activation peuvent être utilisées, et la plus commune est la fonction sigmoïde. La fonction sigmoïde se représente graphiquement par une courbe de la forme de la lettre S qui associe toute valeur à un nombre compris entre 0 et 1. Sa formule est la suivante :

$$f(z) = \frac{1}{1 + e^{-z}}$$

Un neurone peut aussi avoir un seuil, de sorte qu'un signal pondéré ne sera transmis que s'il est supérieur ou égale au mentionné seuil.

Avant d'aller plus loin, nous allons tout d'abord expliquer d'une manière générale comment se déroule l'entraînement d'un réseau neuronal. En second lieu, pour une meilleure idée de son fonctionnement et de ses cas d'utilisations, nous allons naviguer les intentions des premiers chercheurs qui ont rendu possible ce potentiel déterminant du futur de la technologie moderne. Troisièmement, nous ferons passer sous la loupe, l'un des plus communs types de réseaux neuronaux dont l'implémentation est devenue aujourd'hui le premier choix de toutes les applications d'intelligence artificielle, le réseau neuronal convolutif. En quatrième et dernier lieu, nous parlerons d'un débouché révolutionnaire des réseaux neuronaux, le prototype d'agent conver-

sationnel ChatGPT qui fait ravage dans tous les secteurs constituant notre société et qui pose un sérieux risque à cette dernière.

1.3.2 Le perceptron simple

Un perceptron est un modèle simple de neurone biologique utilisé dans un réseau neuronal artificiel. Frank Rosenblatt a introduit ce concept en 1957, lorsqu'il a démontré comment il pouvait constituer un élément de base dans un réseau neuronal à couche unique. Le perceptron est considéré comme l'un des premiers algorithmes créés pour l'apprentissage supervisé des classificateurs binaires.

L'algorithme perceptron a été conçu pour classer les entrées visuelles, en les regroupant dans l'une des deux catégories. L'algorithme suppose que les données sont linéairement séparables, c'est-à-dire qu'elles peuvent être naturellement séparées en deux catégories distinctes. Ce concept peut être visualisé comme un plan bidimensionnel avec deux ensembles de points de données. Si ces ensembles de données peuvent être séparés par une ligne droite, ils sont dits linéairement séparables. Sinon, elles sont non linéairement séparables. La classification est un

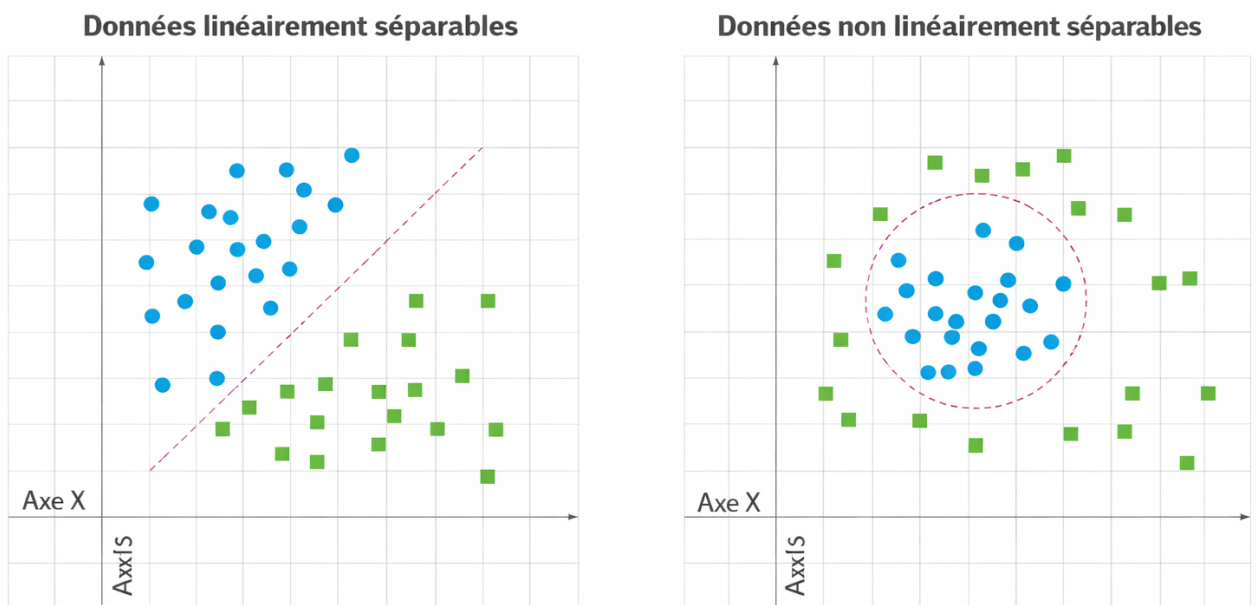


Fig. 1.3 : Graphiques représentant des données linéairement séparables et non linéairement séparables.

élément important de l'apprentissage automatique et du traitement d'images. Les algorithmes d'apprentissage automatique utilisent différentes méthodes pour trouver et catégoriser les modèles de jeux de données. L'une de ces méthodes est l'algorithme perceptron. Il effectue une classification binaire en trouvant la séparation linéaire entre les points de données reçus via l'entrée du perceptron.

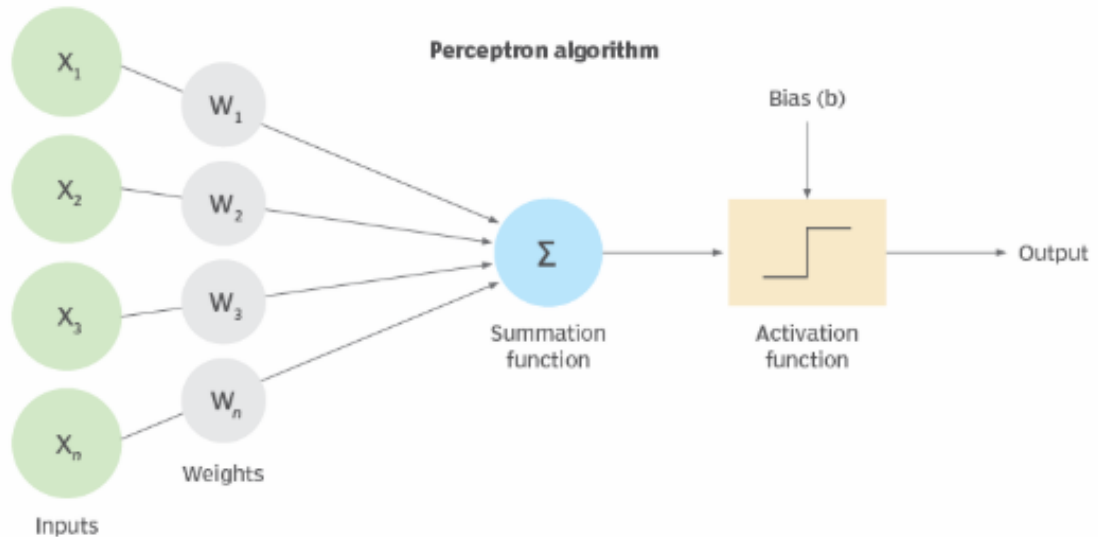


Fig. 1.4 : Fonctionnement de l'algorithme du perceptron.

1.3.3 Algorithme du perceptron

L'algorithme du perceptron peut être représenté de différentes manières, souvent à l'aide de conventions de nommage différentes, mais les concepts de base restent les mêmes. Le perceptron est composé des éléments suivants :

- **Entrées** : Le perceptron reçoit des données provenant d'une ou plusieurs entrées. Chaque entrée possède une valeur numérique représentant un attribut des données.
- **Poids** : Chaque valeur d'entrée se voit attribuer un poids numérique. Le poids détermine l'importance relative de l'entrée dans le modèle. La valeur d'entrée est multipliée par son poids afin d'obtenir une valeur pondérée.
- **Fonction de sommation** : Cette fonction additionne les valeurs pondérées de toutes les entrées :

$$z = \sum_{i=1}^n w_i x_i$$

Elle est parfois appelée *fonction d'entrée nette*.

- **Biais (seuil)** : Une valeur numérique supplémentaire, notée b , est ajoutée afin de contrôler la sortie indépendamment des entrées, ce qui augmente la flexibilité du modèle :

$$z = \sum_{i=1}^n w_i x_i + b$$

- **Fonction d'activation** : La fonction d'activation applique un calcul sur la valeur z afin de produire une sortie binaire. Une fonction en escalier (est appelée fonction seuil) est

souvent utilisée :

$$f(z) = \begin{cases} 1 & \text{si } z \geq 0 \\ 0 & \text{sinon} \end{cases}$$

- **Sortie** : Le résultat binaire produit par la fonction d'activation.

1.3.4 Perceptron multicouche (MLP) et rétropropagation du gradient

Le perceptron multicouche est un réseau orienté de neurones artificiels organisé en couches et où l'information voyage dans un seul sens, de la couche d'entrée vers la couche de sortie. La figure I.5 donne l'exemple d'un réseau contenant une couche d'entrée, deux couches cachées et une couche de sortie. La couche d'entrée représente toujours une couche virtuelle associée aux entrées du système. Elle ne contient aucun neurone. Les couches suivantes sont des couches de neurones. Dans l'exemple illustrée, il y a 3 entrées, 4 neurones sur la première couche cachée, trois neurones sur la deuxième et quatre neurones sur la couche de sortie. Les sorties des neurones de la dernière couche correspondent toujours aux sorties du système. Dans le cas général, un perceptron multicouche peut posséder un nombre de couches quelconque et un nombre de neurones (ou d'entrées) par couche également quelconque.

Les neurones sont reliés entre eux par des connexions pondérées. Ce sont les poids de ces connexions qui gouvernent le fonctionnement du réseau et "programment" une application de l'espace des entrées vers l'espace des sorties 'à l'aide d'une transformation non linéaire. La création d'un perceptron multicouche pour résoudre un problème donné passe donc par l'inférence de la meilleure application possible telle que définie par un ensemble de données d'apprentissage constituées de paires de vecteurs d'entrées et de sorties désirées. Cette inférence peut se faire, entre autre, par l'algorithme dit de rétropropagation.

Algorithme de rétropropagation

Soit le couple $(\vec{x}(n), \vec{d}(n))$ désignant la n -ième donnée d'entraînement du réseau où :

$$\vec{x}(n) = \langle x_1(n), \dots, x_p(n) \rangle$$

et

$$\vec{d}(n) = \langle d_1(n), \dots, d_q(n) \rangle,$$

correspondent respectivement aux p entrées et aux q sorties désirées du système.

L'algorithme de rétropropagation consiste alors à mesurer l'erreur entre les sorties désirées $\vec{d}(n)$ et les sorties observées $\vec{y}(n)$:

$$\vec{y}(n) = \langle y_1(n), \dots, y_q(n) \rangle$$

résultant de la propagation vers l'avant des entrées $\vec{x}(n)$, et à rétropropager cette erreur à travers les couches du réseau en allant des sorties vers les entrées.

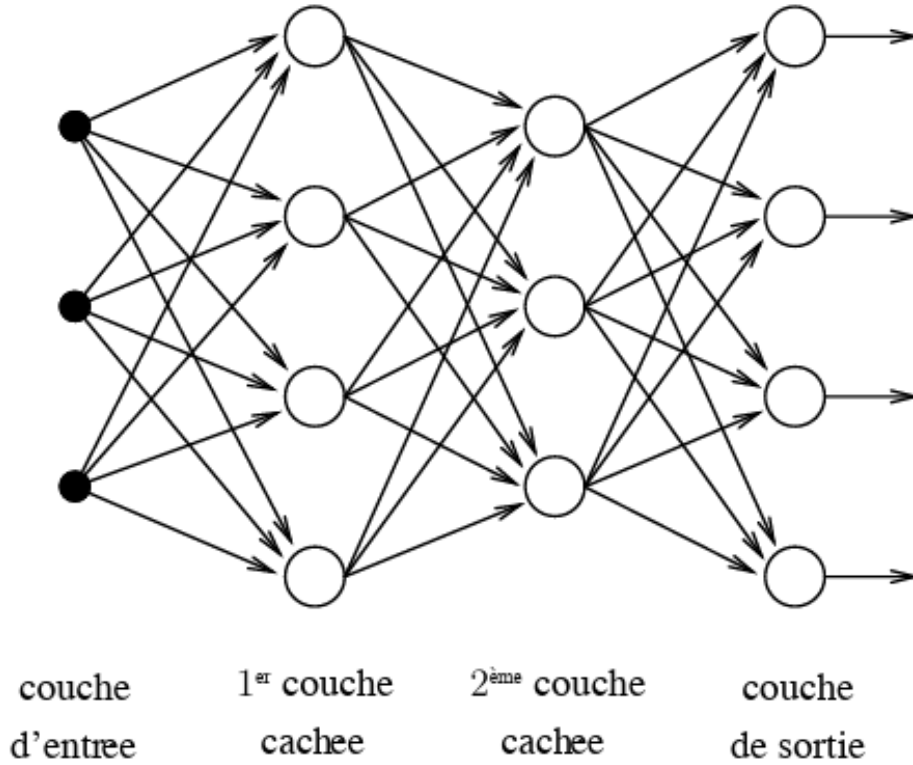


Fig. 1.5 : Exemple d'un réseau de type perceptron multicouche

Cas de la couche de sortie

L'algorithme de rétropropagation procède à l'adaptation des poids neurone par neurone en commençant par la couche de sortie. Soit l'erreur observée $e_j(n)$ pour le neurone de sortie j et la donnée d'entraînement n :

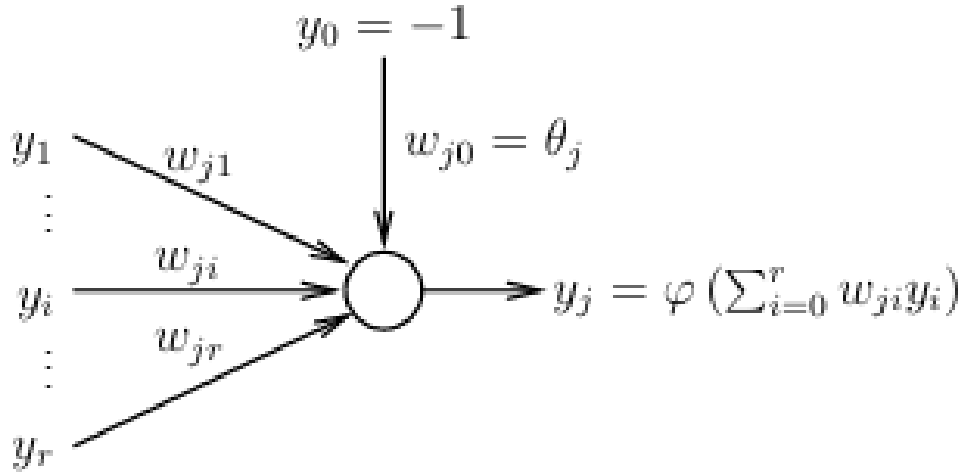
$$e_j(n) = d_j(n) - y_j(n) \quad (1.1)$$

où $d_j(n)$ correspond à la sortie désirée du neurone j et $y_j(n)$ à sa sortie observée.

- La variable n représentera toujours la donnée d'entraînement, c'est-à-dire le couple contenant un vecteur d'entrées et un vecteur de sorties désirées.
- L'objectif de l'algorithme est d'adapter les poids des connexions du réseau de manière à minimiser la somme des erreurs sur tous les neurones de sortie.
- L'indice j représentera toujours le neurone pour lequel on veut adapter les poids.

Soit $E(n)$ la somme des erreurs quadratiques observées sur l'ensemble C des neurones de sortie :

$$E(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n) \quad (1.2)$$


 Fig. 1.6 : Modèle du neurone j .

La sortie $y_j(n)$ du neurone j est définie par :

$$y_j(n) = \phi[v_j(n)] = \phi \left[\sum_{i=0}^r w_{ji}(n) y_i(n) \right] \quad (1.3)$$

où :

- $\phi[.]$ est la fonction d'activation du neurone ;
- $v_j(n)$ est la somme pondérée des entrées du neurone j ;
- $w_{ji}(n)$ est le poids de la connexion entre le neurone i de la couche précédente et le neurone j de la couche courante ;
- $y_i(n)$ est la sortie du neurone i .

On suppose ici que la couche précédente contient r neurones numérotés de 1 à r , que le poids $w_{j0}(n)$ correspond au biais du neurone j et que l'entrée $y_0(n) = -1$.

-L'indice i représentera toujours un neurone sur la couche précédente par rapport au neurone j .

Pour corriger l'erreur observée, il s'agit de modifier le poids $w_{ji}(n)$ dans le sens opposé au gradient $\frac{\partial E(n)}{\partial w_{ji}(n)}$ de l'erreur.

-Cette dérivée partielle représente un facteur de sensibilité : si je change un peu $w_{ji}(n)$, est-ce que cela change beaucoup $E(n)$?

- Si oui, alors je vais changer beaucoup $w_{ji}(n)$ dans le sens inverse de cette dérivée car cela devrait me rapprocher du minimum local.
- Sinon, je dois changer seulement un peu $w_{ji}(n)$ pour corriger l'erreur car je suis tout près de ce minimum !

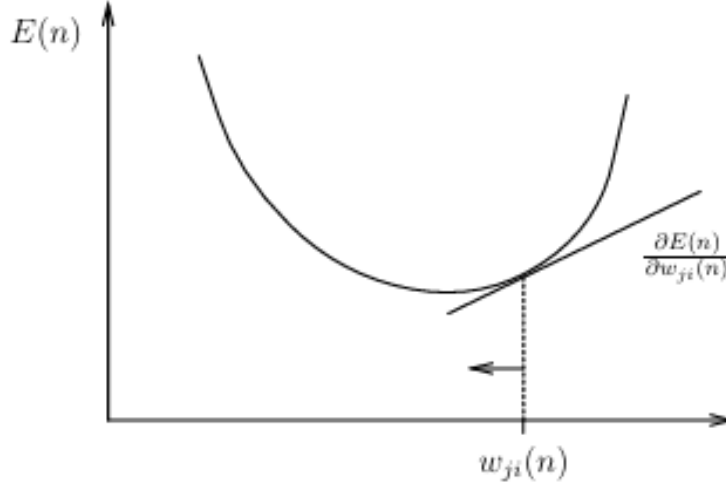


Fig. 1.7 : Gradient de l'erreur totale.

Remarque sur la dimensionnalité : Puisqu'il y a r neurones sur la couche précédant la couche de sortie, il y a aussi r poids à adapter. Il importe donc de remarquer que la courbe de l'erreur correspond en fait à une hyper-surface de $r + 1$ dimensions.

Calcul du gradient par la règle de la chaîne : Par la règle de la chaîne (chain rule) des dérivées partielles, qui nous dit que $\frac{\partial f(y)}{\partial x} = \frac{\partial f(y)}{\partial y} \cdot \frac{\partial y}{\partial x}$, on obtient :

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = \frac{\partial E(n)}{\partial e_j(n)} \cdot \frac{\partial e_j(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial v_j(n)} \cdot \frac{\partial v_j(n)}{\partial w_{ji}(n)} \quad (1.4)$$

On exprime alors la variation de poids $\Delta w_{ji}(n)$ sous la forme suivante :

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ji}(n)} \quad (1.5)$$

avec $0 \leq \eta \leq 1$ représentant le taux d'apprentissage ou gain de l'algorithme.

Évaluation des termes du gradient Évaluons maintenant chacun des termes du gradient :

$$\frac{\partial E(n)}{\partial e_j(n)} = \frac{\partial \left[\frac{1}{2} \sum_{k \in C} e_k^2(n) \right]}{\partial e_j(n)} = \frac{1}{2} \cdot \frac{\partial e_j^2(n)}{\partial e_j(n)} = e_j(n) \quad (1.6)$$

Suite de l'évaluation des termes du gradient Continuons l'évaluation des dérivées partielles pour obtenir le gradient local :

1. Dérivée de l'erreur par rapport à la sortie :

$$\frac{\partial e_j(n)}{\partial y_j(n)} = \frac{\partial [d_j(n) - y_j(n)]}{\partial y_j(n)} = -1 \quad (1.7)$$

2. Dérivée de la sortie par rapport à la somme pondérée (cas de la fonction sigmoïde) :

$$\frac{\partial y_j(n)}{\partial v_j(n)} = \frac{\partial \left[\frac{1}{1+e^{-v_j(n)}} \right]}{\partial v_j(n)} = \frac{e^{-v_j(n)}}{[1 + e^{-v_j(n)}]^2} \quad (1.8)$$

En simplifiant, nous obtenons :

$$\frac{\partial y_j(n)}{\partial v_j(n)} = y_j(n) \left[\frac{e^{-v_j(n)}}{1 + e^{-v_j(n)}} \right] = y_j(n) \left[\frac{e^{-v_j(n)} + 1}{1 + e^{-v_j(n)}} - \frac{1}{1 + e^{-v_j(n)}} \right] \quad (1.9)$$

Ce qui nous donne la propriété classique de la sigmoïde :

$$\frac{\partial y_j(n)}{\partial v_j(n)} = y_j(n)[1 - y_j(n)] \quad (1.10)$$

3. Dérivée de la somme pondérée par rapport au poids :

$$\frac{\partial v_j(n)}{\partial w_{ji}(n)} = \frac{\partial [\sum_{l=0}^r w_{jl}(n)y_l(n)]}{\partial w_{ji}(n)} = \frac{\partial [w_{ji}(n)y_i(n)]}{\partial w_{ji}(n)} = y_i(n) \quad (1.11)$$

Synthèse du gradient local En regroupant tous ces termes, nous obtenons finalement :

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = -e_j(n)y_j(n)[1 - y_j(n)]y_i(n) \quad (1.12)$$

La règle dite du «**delta**» pour la couche de sortie s'exprime par :

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ji}(n)} = \eta \delta_j(n)y_i(n) \quad (1.13)$$

où $\delta_j(n)$ correspond au «**gradient local**» défini par :

$$\delta_j(n) = e_j(n)y_j(n)[1 - y_j(n)] \quad (1.14)$$

Note : Jusqu'ici, nous avons traité seulement le cas de la couche de sortie. L'étape suivante consiste à adapter les poids sur les couches cachées, où l'erreur observée n'est pas directement disponible.

Cas d'une couche cachée

Considérons maintenant le cas des neurones sur la dernière couche cachée (le cas des autres couches cachées est semblable).

- La variable n désignera toujours la donnée d'entraînement (couple de vecteurs d'entrées et de sorties désirées).

- L'objectif est d'adapter les poids de la couche courante en minimisant la somme des erreurs sur les neurones de la couche de sortie.
- Les indices i et j désigneront respectivement un neurone sur la couche précédente et un neurone sur la couche courante.
- L'indice k servira à désigner un neurone sur la couche suivante (couche de sortie dans ce cas).

Reprenons l'expression de la dérivée partielle de l'erreur totale $E(n)$ par rapport à w_{ji} , en notant que l'erreur $e_j(n)$ est ici inconnue :

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = \frac{\partial E(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial v_j(n)} \cdot \frac{\partial v_j(n)}{\partial w_{ji}(n)} \quad (1.15)$$

Par rapport aux résultats obtenus pour la couche de sortie, les deux derniers termes restent inchangés. Seul le premier terme requiert d'être évalué :

$$\frac{\partial E(n)}{\partial y_j(n)} = \frac{\partial \left[\frac{1}{2} \sum_{k \in C} e_k^2(n) \right]}{\partial y_j(n)} \quad (1.16)$$

Contrairement au cas de la couche de sortie, tous les termes $e_k(n)$ de la somme dépendent de $y_j(n)$. Nous devons donc conserver la sommation et utiliser à nouveau la règle de la chaîne :

Reprenons l'expression de la dérivée partielle de l'erreur totale $E(n)$ par rapport à w_{ji} . En substituant les termes, nous obtenons la règle de mise à jour des poids pour les couches cachées :

$$\Delta w_{kj}(n) = \eta \cdot \delta_k(n) \cdot y_j(n) \quad (1.17)$$

Où :

- $\Delta w_{kj}(n)$ est la modification du poids synaptique.
- η est le taux d'apprentissage.
- $\delta_k(n)$ est le gradient local (signal d'erreur).
- $y_j(n)$ est le signal d'entrée du neurone précédent.

En substituant la définition du gradient local de la couche de sortie ($\delta_k(n)$) établie précédemment (Eq. 10), nous obtenons finalement :

$$\frac{\partial E(n)}{\partial y_j(n)} = - \sum_{k \in C} \delta_k(n) w_{kj}(n) \quad (1.18)$$

Formulation finale pour les couches cachées En substituant l'expression obtenue précédemment (Eq. 13) dans l'équation du gradient (Eq. 11), nous parvenons à la forme finale de la dérivée de l'erreur par rapport aux poids de la couche cachée :

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = -y_j(n)[1 - y_j(n)] \left[\sum_{k \in C} \delta_k(n) w_{kj}(n) \right] y_i(n) \quad (1.19)$$

Dès lors, la variation du poids $\Delta w_{ji}(n)$ et le gradient local $\delta_j(n)$ pour une couche cachée s'expriment comme suit :

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ji}(n)} = \eta \delta_j(n) y_i(n) \quad (1.20)$$

Avec :

$$\delta_j(n) = y_j(n)[1 - y_j(n)] \sum_{k \in C} \delta_k(n) w_{kj}(n) \quad (1.21)$$

Remarques importantes

- On peut démontrer que les équations (15) et (16) sont valides pour toutes les couches cachées du réseau, quelle que soit leur profondeur.
- **Cas particulier** : Pour la première couche cachée du réseau, puisqu'il n'existe pas de couche précédente de neurones, il convient de substituer la variable $y_i(n)$ par l'entrée directe du réseau $x_i(n)$.

Sommaire de l'algorithme (règle du “delta”)

L'algorithme de rétropropagation standard se résume à la série d'étapes suivantes :

1. **Initialisation** : Initialiser tous les poids à de petites valeurs aléatoires dans l'intervalle $[-0,5; 0,5]$.
2. **Normalisation** : Normaliser les données d'entraînement.
3. **Permutation** : Permuter aléatoirement les données d'entraînement (pour éviter les biais d'ordre).
4. **Apprentissage** : Pour chaque donnée d'entraînement n :
 - (a) **Passe avant (Forward pass)** : Calculer les sorties observées en propageant les entrées vers l'avant à travers les couches du réseau.
 - (b) **Passe arrière (Backward pass)** : Ajuster les poids en rétropropageant l'erreur observée :

$$w_{ji}(n) = w_{ji}(n-1) + \Delta w_{ji}(n) = w_{ji}(n-1) + \eta \delta_j(n) y_i(n) \quad (1.22)$$

où le « gradient local » $\delta_j(n)$ est défini par :

$$\delta_j(n) = \begin{cases} e_j(n) y_j(n) [1 - y_j(n)] & \text{si } j \in \text{couche de sortie} \\ y_j(n) [1 - y_j(n)] \sum_k \delta_k(n) w_{kj}(n) & \text{si } j \in \text{couche cachée} \end{cases}$$

avec $0 \leq \eta \leq 1$ représentant le taux d'apprentissage et $y_i(n)$ représentant soit la sortie du neurone i sur la couche précédente, soit l'entrée $x_i(n)$ si aucune couche ne précède.

5. **Condition d'arrêt :** Répéter les étapes 3 et 4 jusqu'à atteindre un nombre maximum d'itérations ou jusqu'à ce que la racine de l'erreur quadratique moyenne (EQM) soit inférieure à un seuil prédéfini.

Règle du “delta généralisé”

Règle du delta généralisée (Momentum) L'équation (17) décrit ce qu'on appelle la règle du «delta» pour l'algorithme de rétropropagation des erreurs. L'équation suivante, nommée règle du «delta généralisée», décrit une variante de l'algorithme intégrant un terme de quantité de mouvement :

$$w_{ji}(n) = w_{ji}(n-1) + \eta \delta_j(n) y_i(n) + \alpha \Delta w_{ji}(n-1) \quad (1.23)$$

où $0 \leq \alpha \leq 1$ est un paramètre nommé **momentum** (ou moment). Ce paramètre représente une espèce d'inertie dans le changement de poids, permettant d'accélérer la convergence dans les zones de faible pente et de réduire les oscillations dans les vallées étroites de la surface d'erreur.

1.4 Apprentissage profond (Deep learning)

L'apprentissage profond permet aux modèles informatiques, composés de plusieurs couches de traitement, d'apprendre des représentations de données à différents niveaux d'abstraction. Ces méthodes ont considérablement amélioré l'état de l'art en reconnaissance vocale, reconnaissance visuelle d'objets, détection d'objets et dans de nombreux autres domaines tels que la découverte de médicaments et la génomique.

L'apprentissage profond découvre la structure complexe de vastes ensembles de données grâce à l'algorithme de rétropropagation, qui indique comment un modèle doit modifier ses paramètres internes. Ces paramètres servent à calculer la représentation de chaque couche à partir de celle de la couche précédente. Les réseaux de neurones convolutifs profonds ont permis des avancées majeures dans le traitement des images, des vidéos, de la parole et de l'audio, tandis que les réseaux récurrents ont particulièrement brillé dans le traitement des données séquentielles telles

que le texte et la parole. Les types de réseaux de neurones comprennent une variété d'architectures spécialisées pour des tâches spécifiques en intelligence artificielle. Voici une liste des principaux types de réseaux de neurones :

1.4.1 Réseaux de neurones convolutifs (CNN)

Les réseaux de neurones à convolution, également connus sous le nom de CNN (Convolutional Neural Networks), sont des architectures spécifiques de réseaux de neurones qui sont particulièrement efficaces pour l'analyse et la reconnaissance d'images. Ces réseaux sont conçus pour apprendre directement à partir des données et sont capables de détecter des motifs et des caractéristiques complexes dans les images pour effectuer des tâches de classification et de segmentation. Les CNN peuvent être utilisés pour diverses tâches telles que la classification d'images, la détection d'objets, la segmentation d'images médicales, et bien d'autres applications où l'analyse d'images est nécessaire.

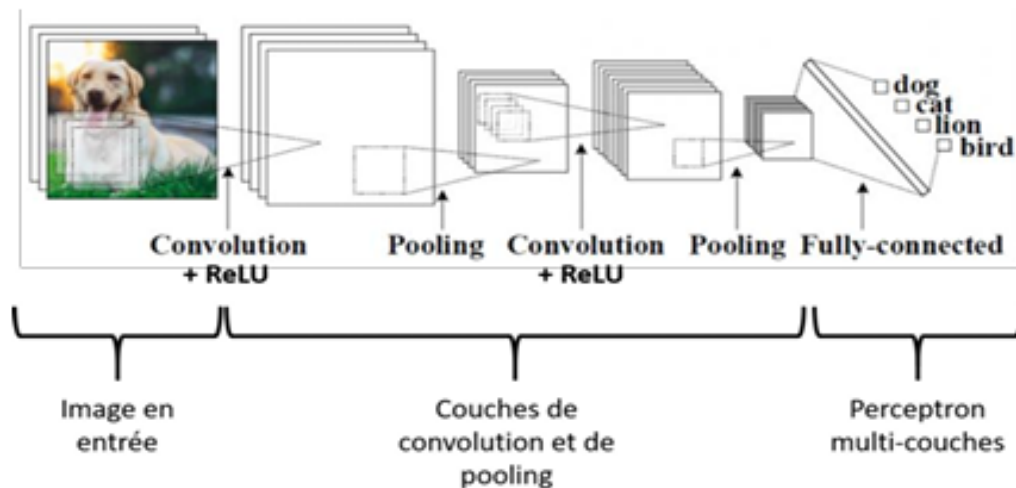


Fig. 1.8 : Architecture générale d'un réseau de neurones convolutif

1.4.2 Réseaux de neurones récurrents (RNN)

Un réseau de neurones récurrent (RNN) est un type de réseau de neurones artificiel qui utilise des données séquentielles ou des données de séries temporelles. Ces algorithmes d'apprentissage en profondeur sont couramment utilisés pour des problèmes ordinaux ou temporels, tels que la traduction linguistique, le traitement du langage naturel, la reconnaissance vocale et le sous-titrage d'images ; ils sont incorporés dans des applications populaires telles que Siri, la recherche vocale et Google Translate. Comme les réseaux de neurones convolutifs (CNN) à propagation avant, les réseaux de neurones récurrents utilisent des données d'entraînement pour apprendre.

Ils se distinguent par leur «mémoire» car ils prennent des informations d'entrées antérieures

pour influencer l'entrée et la sortie en cours. Alors que les réseaux de neurones profonds traditionnels supposent que les entrées et les sorties sont indépendantes les unes des autres, la sortie des réseaux de neurones récurrents dépend des éléments antérieurs au sein de la séquence. Alors que les événements futurs seraient également utiles pour déterminer la sortie d'une séquence donnée, les réseaux de neurones récurrents unidirectionnels ne peuvent pas rendre compte de ces événements dans leurs prédictions.

Une autre particularité des réseaux récurrents est qu'ils partagent des paramètres à chaque couche du réseau. Alors que les réseaux de propagation vers l'avant ont des poids différents à chaque nœud, les réseaux de neurones récurrents partagent les mêmes paramètres de poids dans chaque couche du réseau. Autrement dit, ces poids sont toujours ajustés via des processus de rétropropagation et de descente de gradient pour faciliter l'apprentissage par renforcement.

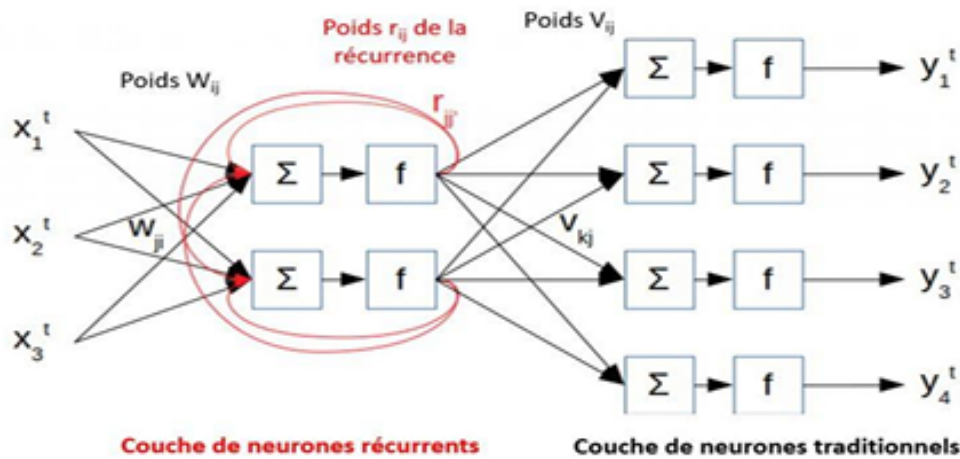


Fig. 1.9 : Couche de neurones récurrents devant une couche de neurones traditionnels

1.5 Apprentissage par renforcement

1.5.1 Définition et concepts fondamentaux

L'apprentissage par renforcement a comme objectif d'entraîner un agent à se comporter de façon intelligente dans un environnement donné. Un agent interagit avec l'environnement en choisissant, à chaque temps donné, d'exécuter une action parmi un ensemble d'actions permises. Le comportement intelligent que doit apprendre cet agent est donné implicitement via un signal de renforcement qui, après chaque décision de l'agent, indique s'il a bien ou mal agi. L'agent doit donc se baser sur ce signal afin d'améliorer son comportement, qui est dicté par sa politique d'actions. À chaque temps donné, l'agent a normalement à sa disposition un ensemble de caractéristiques ou indicateurs d'entrée, décrivant l'environnement. Par exemple, si l'agent correspond à un robot, ces indicateurs pourraient être obtenus à partir de capteurs sensoriels brossant un portrait de l'endroit où il se trouve.

Les concepts d'action et de signal de renforcement sont probablement ceux qui distinguent le plus l'apprentissage par renforcement des apprentissages supervisé et non-supervisé. Contrairement à l'apprentissage supervisé, le comportement intelligent à apprendre n'est pas explicitement donné par une cible à prédire mais doit plutôt être défini par l'utilisateur à l'aide d'un signal de renforcement. Cependant, contrairement à l'apprentissage non-supervisé dont l'objectif est de modéliser la structure statistique des entrées, l'apprentissage par renforcement s'intéresse à la notion d'action qui est différente de celle l'entrée, de la même façon que le concept de cible de l'apprentissage supervisé est différent du concept d'entrée. Ceci étant dit, certains concepts de l'apprentissage par renforcement peuvent être utiles dans un contexte d'apprentissage supervisé, comme le démontre l'algorithme SEARN de Daumé III, un algorithme de classification pour cibles structurées qui utilise la notion de politique d'actions.

Processus de décision de Markov – MDP

Une chaîne de Markov est constituée d'un ensemble d'états S et d'une fonction de transition P indiquant la probabilité $P(s'|s)$ de passer d'un état s à un état s' . Nous limiterons notre description aux chaînes de Markov à espace d'états nis. Il est alors possible de représenter ces données par un graphe ou bien par une matrice de transitions. De ce fait, l'étude des chaînes de Markov à espaces d'états nis est étroitement liée à la théorie des graphes orientés ainsi qu'au calcul matriciel.

Les chaînes de Markov permettent de décrire la dynamique de systèmes stochastiques dans lesquels l'observateur n'a aucun moyen d'in uencer l'évolution du système. Nous quali erons cet observateur de passif , il observe le système mais n'intervient pas sur celui-ci. Dans notre exemple, le joueur passe d'une urne à une autre en suivant des règles pré-établies. Il n'a aucun moyen de décider par lui même dans quelle urne tirer la prochaine boule. Les chaînes de Markov s'avèrent donc peu adaptées pour la représentation de problèmes dans lesquels un agent peut intervenir sur le système.

L'agent est dans ce cas actif et peut choisir d'exécuter une action plutôt qu'une autre a n d'in uencer l'évolution du système. Les processus décisionnels de Markov permettent de formaliser un tel système. Ils constituent une extension des chaînes de Markov dans lesquelles sont ajoutés un ensemble d'actions et une fonction de récompense.

1.5.2 Politique et fonction de valeur

La fonction de transition d'un MDP formalise la dynamique du système. Elle véri e la propriété de Markov. Étant donné que l'action exécutée par l'agent in uence l'évolution du système, la propriété de Markov doit tenir compte des actions. Elle se résume alors par l'équation suivante :

$$P(s_{t+1} = s' \mid s_0, a_0, s_1, a_1, \dots, s_t, a_t) = P(s_{t+1} = s' \mid s_t, a_t) \quad (1.24)$$

Comme pour les chaînes de Markov, chaque état doit être défini de façon à résumer toute l'information nécessaire à la prédiction de l'évolution du système, et par conséquent à la décision.

Stationnarité du système Lorsque la dynamique du système est indépendante du temps, le système est dit **stationnaire**. Dans ce cas, la probabilité de passer d'un état s à un instant t à un état s' à un instant $t + 1$ est indépendante de t . Nous pouvons donc écrire :

$$P(s'_{t+1} | s_t, a_t) = P(s' | s, a), \quad \forall t \quad (1.25)$$

Par la suite, nous restreindrons nos propos aux processus décisionnels de Markov stationnaires.

1.5.3 Programmation dynamique

La programmation dynamique est une méthode d'optimisation des processus de décisions séquentielles. Elle s'appuie sur l'algorithme de Bellman, dont la traduction sur le graphe associé au processus (supposé discret) permet de rechercher le chemin de valeur optimale (1^{re} partie). Les programmes les plus simples sont ceux où l'avenir est déterminé : l'intérêt de l'algorithme est évident quand le caractère combinatoire du problème amène à comparer un très grand nombre de politiques (2^e partie). Mais dans beaucoup de cas concrets l'avenir est incertain. Dans le cas où il est probabilisable, il est possible de comparer les stratégies en les évaluant d'après leur espérance mathématique. Ce critère n'est cependant pas suffisant où les distributions de probabilité sont trop dispersées. Il est inapplicable dans le cas d'un avenir non probabilisable : les différents critères de choix qu'on peut alors utiliser sont parfois contradictoires, donc peu satisfaisants (3^e partie).

La programmation dynamique regroupe des méthodes permettant de résoudre des MDP en utilisant les équations de Bellman.

Évaluation de politique (Prediction)

L'évaluation consiste à calculer la fonction de valeur V^π pour une politique arbitraire π . Elle utilise l'équation de Bellman comme règle de mise à jour itérative :

$$V_{k+1}(s) = \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a) [r + \gamma V_k(s')] \quad (1.26)$$

[Convergence de l'évaluation] La suite $\{V_k\}$ converge vers V^π quand $k \rightarrow \infty$ sous les mêmes conditions qui garantissent l'existence de V^π (existence d'un état terminal ou $\gamma < 1$).

Amélioration de la politique

Une fois V^π connue, on cherche à savoir s'il faut changer de politique.

[Théorème d'amélioration de la politique] Soient π et π' deux politiques déterministes telles que pour tout $s \in S$:

$$q_\pi(s, \pi'(s)) \geq v_\pi(s) \quad (1.27)$$

Alors la politique π' doit être meilleure ou égale à π , soit $v_{\pi'}(s) \geq v_\pi(s)$.

L'amélioration consiste à rendre la politique *greedy* (gloutonne) par rapport à la fonction de valeur actuelle :

$$\pi'(s) = \arg \max_a q_\pi(s, a) = \arg \max_a \sum_{s', r} p(s', r | s, a) [r + \gamma V^\pi(s')] \quad (1.28)$$

Itération de Politique (Policy Iteration)

L'algorithme d'itération de politique alterne entre l'évaluation complète et l'amélioration. Puisqu'un MDP fini ne possède qu'un nombre fini de politiques, ce processus converge vers une politique optimale π^* en un nombre fini d'étapes.

Itération de Valeur (Value Iteration)

L'un des inconvénients de l'itération de politique est que chaque étape d'évaluation nécessite elle-même un processus itératif. L'itération de valeur simplifie cela en combinant l'amélioration et l'évaluation en une seule mise à jour basée sur l'équation d'optimalité de Bellman :

$$V_{k+1}(s) = \max_a \sum_{s', r} p(s', r | s, a) [r + \gamma V_k(s')] \quad (1.29)$$

[Convergence de Value Iteration] Pour tout V_0 , la suite $\{V_k\}$ converge vers la fonction de valeur optimale V^* quand $k \rightarrow \infty$.

1.5.4 Apprentissage par difference temporelle

L'algorithme le plus général de différences temporelles de Kalman, qui se subdivise en trois parties, peut maintenant être obtenu. La première étape consiste à calculer les prédictions $\hat{\theta}_{i|i-1}$ et $P_{i|i-1}$. Rappelons que pour un modèle de marche aléatoire, la prédiction du vecteur de paramètres est égale à son estimation précédente : $\hat{\theta}_{i|i-1} = \hat{\theta}_{i-1|i-1}$. La covariance prédite peut également être calculée analytiquement :

$$\begin{aligned} P_{i|i-1} &= \text{cov} \left(\tilde{\theta}_{i|i-1} | r_{1:i-1} \right) \\ &= \text{cov} \left(\tilde{\theta}_{i-1|i-1} + v_{i-1} | r_{1:i-1} \right) \\ &= P_{i-1|i-1} + P_{v_{i-1}} \end{aligned} \quad (1.30)$$

La seconde étape consiste à calculer quelques statistiques d'intérêt. C'est principalement cette partie qui sera spécialisée dans la section suivante. La première statistique à calculer est la prédiction de la récompense $\hat{r}_{i|i-1}$ (29). La seconde est la covariance entre l'erreur sur les paramètres et l'innovation :

$$P_{\theta r_i} = E \left[(\theta_i - \hat{\theta}_{i|i-1})(r_i - \hat{r}_{i|i-1}) | r_{1:i-1} \right] \quad (1.31)$$

Étant donné la forme de l'équation d'observation ($r_i = g_t^i(\theta_i) + n_i$) et l'indépendance du bruit d'observation qui est centré, cette statistique peut se réécrire :

$$P_{\theta r_i} = E \left[(\theta_i - \hat{\theta}_{i|i-1})(g_t^i(\theta_i) - \hat{r}_{i|i-1}) | r_{1:i-1} \right] \quad (1.32)$$

Enfin, la dernière statistique à calculer est la variance de l'innovation, qui peut être écrite (en utilisant à nouveau les caractéristiques du bruit d'observation) :

$$\begin{aligned} P_{r_i} &= E \left[(r_i - \hat{r}_{i|i-1})^2 | r_{1:i-1} \right] \\ &= E \left[(g_t^i(\theta_i) - \hat{r}_{i|i-1} + n_i)^2 | r_{1:i-1} \right] \\ &= E \left[(g_t^i(\theta_i) - \hat{r}_{i|i-1})^2 | r_{1:i-1} \right] + P_{n_i} \end{aligned} \quad (1.33)$$

La dernière étape de l'algorithme est la phase de correction qui consiste à calculer le gain et à mettre à jour le vecteur de paramètres ainsi que la matrice de covariance associée. Notons que la méthode proposée étant en ligne, elle se doit d'être initialisée avec un a priori sur l'espérance $\hat{\theta}_{0|0}$ et la covariance $P_{0|0}$ des paramètres. L'approche générale proposée est résumée dans l'algorithme 1. Le calcul de la variance $P_{i|i}$ peut sembler à première vue inutile, mais elle est utilisée dans tous les cas (linéaire, linéarisé, non-linéaire) pour calculer les statistiques d'intérêt.

Algorithme Q-learning

Le principe fondamental du *Q-Learning* réside dans la mise à jour itérative des valeurs d'action $Q(s, a)$ en exploitant les récompenses immédiates et les estimations des valeurs d'action futures. Cette méthode s'appuie sur l'équation d'optimalité de Bellman, qui établit une relation récursive pour la valeur d'une action. La règle de mise à jour est définie par l'expression suivante :

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t) \right] \quad (1.34)$$

Où les paramètres sont définis comme suit :

- $Q(s_t, a_t)$: Représente la valeur actuelle de l'action a_t exécutée dans l'état s_t .
- r_{t+1} : Désigne la récompense immédiate perçue après avoir effectué l'action a_t dans l'état s_t .

- $\gamma \max_a Q(s_{t+1}, a)$: Correspond à la valeur d'action maximale possible pour l'état suivant s_{t+1} , pondérée par le facteur de réduction γ (*discount factor*). Ce terme estime les récompenses futures cumulées maximales que l'agent peut espérer obtenir en suivant une politique optimale à partir de l'état s_{t+1} .
- $\delta_t = r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)$: Représente l'erreur de différence temporelle (*TD-Error*). Il s'agit de l'écart entre l'estimation révisée (récompense réelle augmentée du futur estimé) et la valeur actuelle.
- α : Est le taux d'apprentissage (*learning rate*), régulant l'influence des nouvelles informations sur la mise à jour des connaissances acquises.

Cette règle de mise à jour privilégie la meilleure action possible dans l'état suivant, ce qui confère au *Q-Learning* sa nature *Off-Policy*. Ainsi, l'agent est capable d'apprendre une politique optimale tout en suivant une stratégie comportementale différente (généralement plus exploratoire).

Étapes de mise en œuvre et convergence

La mise en place de l'algorithme s'articule autour de trois étapes distinctes :

1. **Initialisation** : La fonction $Q(s, a)$ est initialisée de manière arbitraire pour l'ensemble des paires états-actions, et un état initial s est défini.
2. **Processus itératif** : Cette étape est répétée pour chaque transition :
 - Sélectionner et exécuter une action a (via une stratégie telle que ϵ -greedy).
 - Observer la récompense r et le nouvel état s' .
 - Mettre à jour la valeur $Q(s, a)$ selon la règle de Bellman.
 - Passer à l'état suivant ($s \leftarrow s'$).

3. **Convergence** : Le cycle doit être répété jusqu'à la stabilisation des valeurs de Q .

Pour garantir la convergence vers l'optimalité, deux conditions sont impératives :

- **Exploration exhaustive** : Chaque paire (s, a) doit être visitée suffisamment de fois pour assurer une couverture complète de l'espace d'états.
- **Paramétrage du facteur γ** : Le *discount factor* doit être ajusté selon l'horizon de décision. Un γ proche de 1 valorise les récompenses lointaines (stratégie prévoyante), tandis qu'un γ faible favorise la gratification immédiate (stratégie court-termiste).

Une fois ces conditions réunies, le *Q-Learning* converge vers la fonction de valeur d'action optimale $Q^*(s, a)$. La politique optimale π^* est alors dérivée en sélectionnant l'action maximisant Q pour chaque état : $\pi^*(s) = \arg \max_a Q^*(s, a)$.

Algorithme acteur-critique

L'approche *Actor-Critic* est une méthode d'apprentissage par renforcement hybride qui combine les avantages des méthodes basées sur les valeurs et de celles basées sur les politiques.

L'Actor (Acteur) : Il est responsable de la sélection des actions en fonction de l'état actuel selon une politique paramétrée $\pi(a|s; \theta)$. L'acteur ajuste les paramètres θ de sa politique en fonction des retours (feedbacks) qu'il reçoit du critique.

Le Critic (Critique) : Cet élément est chargé de l'évaluation des actions choisies par l'acteur en estimant la valeur des états. Il utilise une fonction de valeur $V_w(s)$ ou $Q_w(s, a)$, paramétrée par w , pour estimer la qualité des actions. Le critique fournit un signal d'erreur de différence temporelle (TD-Error) à l'acteur, indiquant si les résultats obtenus sont meilleurs ou pires que les prédictions initiales.

Mise à jour et formulations mathématiques

La mise à jour de la politique repose sur le gradient de politique. Contrairement à l'algorithme REINFORCE qui utilise les récompenses cumulées totales, l'approche *Actor-Critic* utilise la *Q-value* fournie par le critique. La mise à jour de l'acteur s'effectue selon la règle suivante :

$$\theta \leftarrow \theta + \alpha \nabla_{\theta} [\log \pi(a|s; \theta) Q_w(s, a)] \quad (1.35)$$

Où :

- α est le taux d'apprentissage de l'acteur.
- $Q_w(s, a)$ est l'estimation de la *Q-value* pour le couple (s, a) , paramétrée par w .

Parallèlement, le critique met à jour ses propres paramètres w en minimisant l'erreur de TD définie par $\delta = r + \gamma Q_w(s'; a') - Q_w(s; a)$. La mise à jour est donnée par :

$$w \leftarrow w + \beta \delta \nabla_w Q_w(s, a) \quad (1.36)$$

Où :

- $\nabla_w Q_w(s, a)$ représente le gradient de la *Q-value* par rapport aux paramètres w , indiquant la direction de l'ajustement pour réduire l'erreur δ .
- β est le taux d'apprentissage du critique.

L'architecture *Actor-Critic* présente plusieurs points forts. Elle permet une convergence souvent plus rapide et des mises à jour plus stables que les méthodes isolées. Grâce au critique, l'apprentissage est mieux informé, ce qui réduit la variance du gradient. De plus, sa flexibilité

lui permet de traiter des espaces d'états et d'actions aussi bien discrets que continus, un atout majeur pour la simulation du trafic routier complexe.

Cependant, cette méthode est plus complexe à implémenter et à stabiliser que les méthodes classiques, car elle nécessite l'optimisation simultanée de deux réseaux de neurones (ou fonctions paramétrées) qui interagissent entre eux.

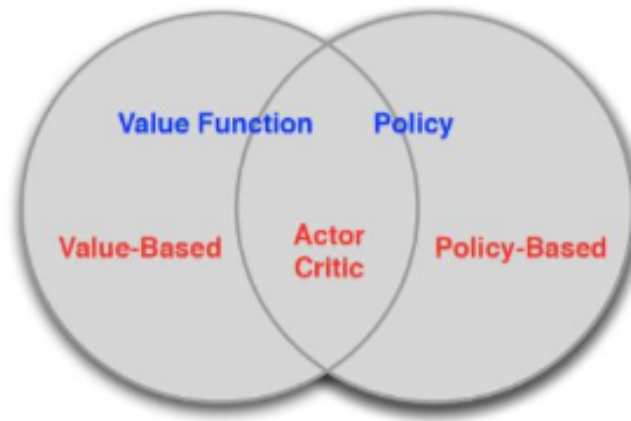


Fig. 1.10 : Les trois principales approches de l'apprentissage par renforcement

Pour résumé :

Les méthodes Value-Based, telles que Q-Learning, se concentrent sur l'estimation des fonctions de valeur pour déterminer les meilleures actions à prendre. Les méthodes Policy-Based, comme celles utilisant les gradients de politique, optimisent directement la politique de l'agent sans passer par une fonction de valeur explicite. L'approche Actor-Critic combine ces deux méthodes en utilisant un acteur pour choisir les actions et un critique pour évaluer la fonction de valeur.

1.6 Apprentissage par renforcement profond (Deep RL)

1.6.1 Deep Q-Network (DQN)

Le *Deep Q-Network* (DQN) est un algorithme d'apprentissage par renforcement (*Reinforcement Learning*, RL) basé sur la valeur. Il utilise un réseau de neurones profonds pour approximer la fonction de valeur d'action optimale $Q^*(s_t, a_t)$, qui représente la récompense cumulative attendue obtenue en exécutant l'action a_t dans l'état s_t , puis en suivant la politique optimale par la suite.

Afin d'améliorer la stabilité et la convergence de l'apprentissage, la méthode DQN introduit deux mécanismes principaux : le réseau cible et la mémoire de relecture (*experience replay*).

La mémoire de relecture stocke les transitions (s_t, a_t, r_t, s_{t+1}) dans un tampon, à partir duquel des lots de transitions sont échantillonnés de manière uniforme et aléatoire pour mettre à jour le réseau Q .

Le réseau cible est une copie distincte du réseau principal Q . Il est utilisé pour fournir des cibles plus stables lors des mises à jour de l'algorithme de Q-learning. Les paramètres du réseau cible sont mis à jour périodiquement en copiant les poids du réseau Q principal.

Grâce à ces améliorations, le DQN a été appliqué avec succès à une large gamme de tâches, notamment le jeu sur les environnements Atari ainsi que le contrôle de systèmes robotiques.

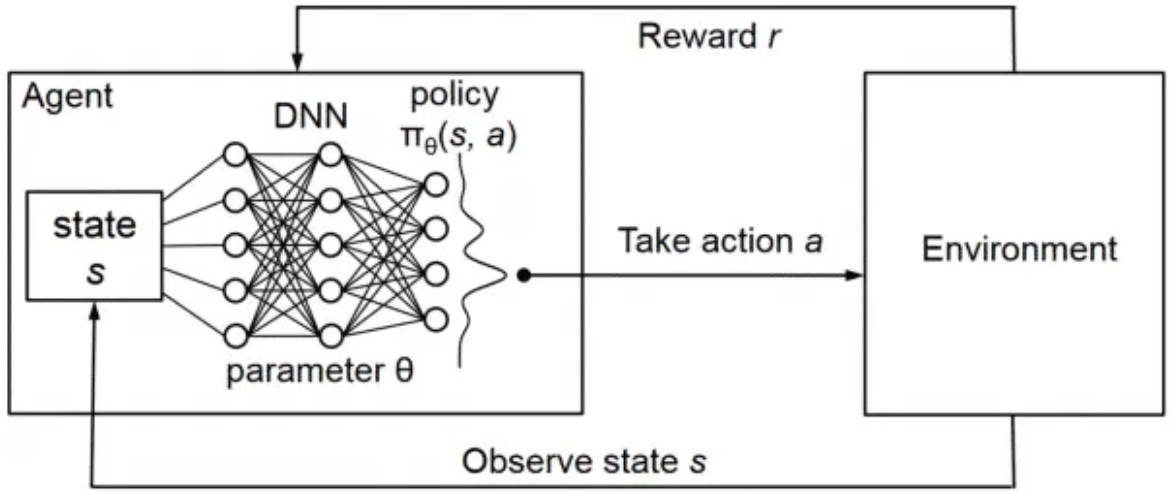


Fig. 1.11 : L'architecture DQN

1.6.2 DDPG

La structure interne et le flux de données de l'algorithme DDPG sont illustrés par le schéma suivant :

Le réseau de politique de l'Actor de l'algorithme DDPG est utilisé pour explorer l'environnement et prendre des décisions d'action, tandis que le réseau d'évaluation du Critic sert à juger la qualité de chaque action, ce qui guide la direction de la mise à jour du gradient du réseau de politique de l'Actor. La structure du cadre et le processus spécifique de l'algorithme DDPG sont illustrés dans la figure correspondante.

Tant l'Actor que le Critic contiennent deux réseaux de neurones ayant la même architecture : le réseau en ligne (*online network*) et le réseau cible (*target network*). Les paramètres du réseau Q en ligne $Q(s, a|\theta_Q)$ et du réseau de politique en ligne $\mu(s|\theta_\mu)$ sont initialisés de manière aléatoire. Les paramètres du réseau cible sont initialisés comme suit : $\theta'_Q \leftarrow \theta_Q$ et $\theta'_\mu \leftarrow \theta_\mu$.

L'action est sélectionnée selon la politique courante μ comme indiqué dans l'équation (1.37) :

$$a_t = \mu(s_t | \theta_\mu) + N_t \quad (1.37)$$

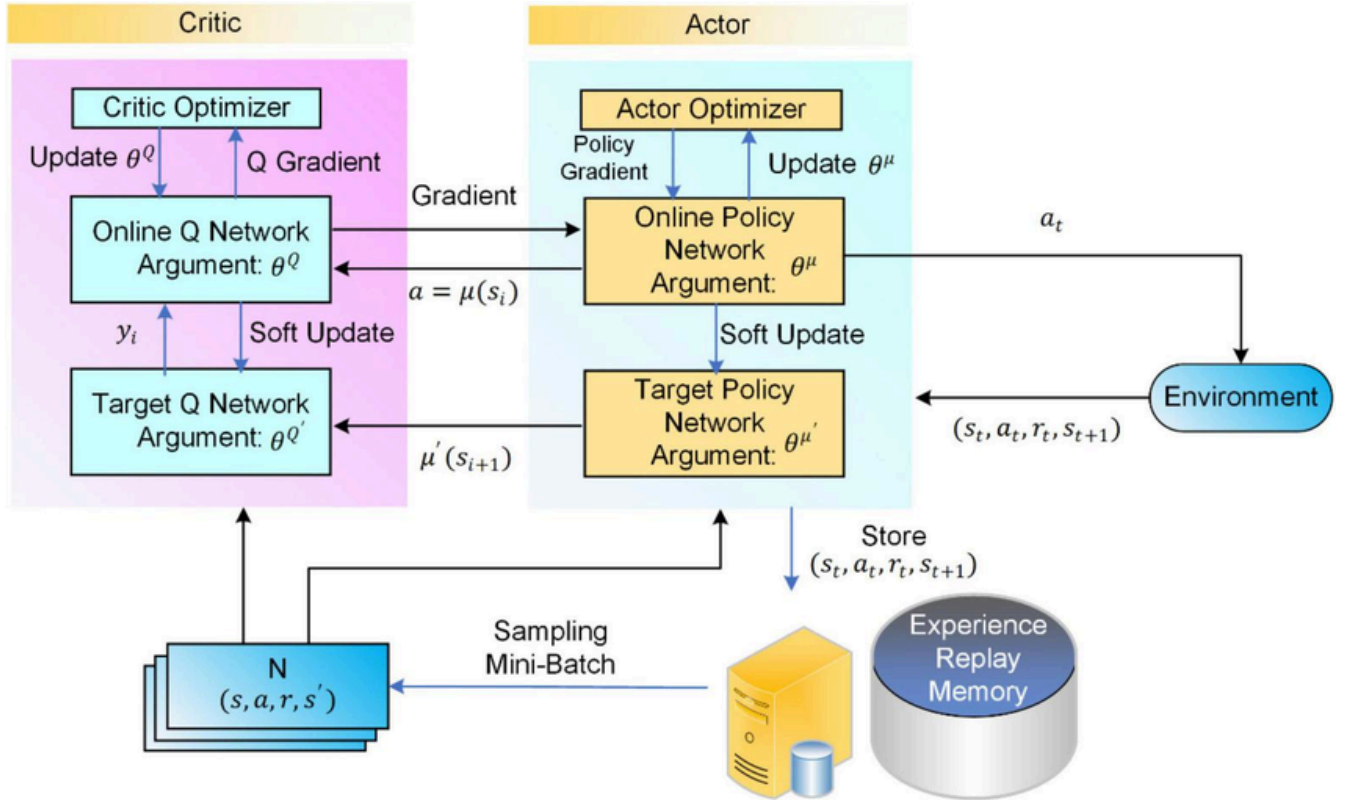


Fig. 1.12 : L'architecture DDPG

où a_t est l'action exécutée selon la politique actuelle, s_t est l'état observé courant et N_t représente le bruit d'exploration.

1.7 Systèmes Multi-Agents

1.7.1 Définition d'un système multi-agents

On appelle **système multi-agents (ou SMA)** un système composé des éléments suivants :

- Un environnement E , c'est-à-dire un espace disposant généralement d'une métrique.
- Un ensemble d'objets O . Ces objets sont situés, c'est-à-dire que, pour tout objet, il est possible, à un moment donné, d'associer une position dans E . Ces objets sont passifs, c'est-à-dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents.
- Un ensemble A d'agents, qui sont des objets particuliers ($A \subseteq O$), et qui représentent les entités actives du système.
- Un ensemble de relations R qui unissent les objets (et donc les agents) entre eux.
- Un ensemble d'opérations Op permettant aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O .

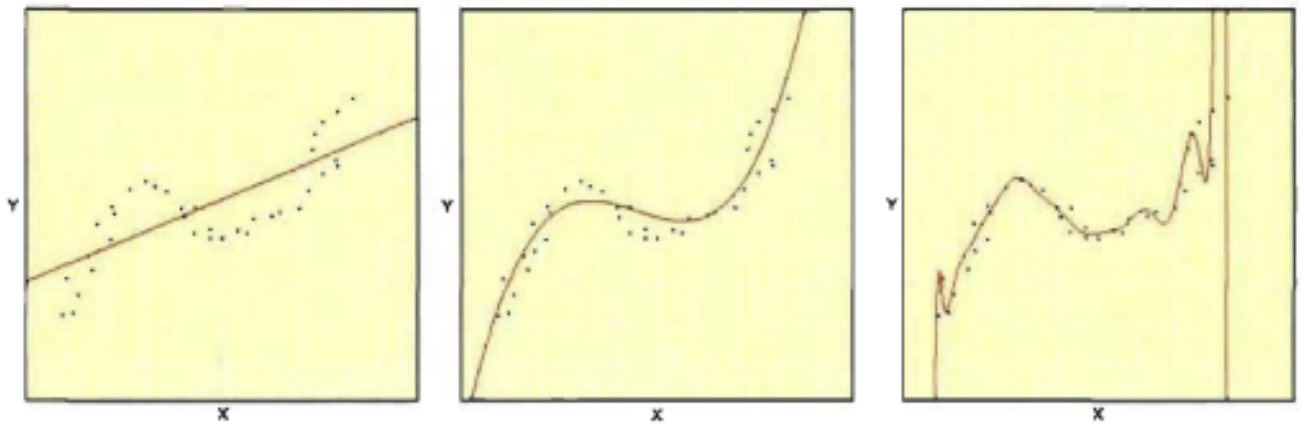


Fig. 1.13 : Organigramme de l'algorithme de gradient de politique déterministe profond (DDPG)

- Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, appelés les *lois de l'univers*.

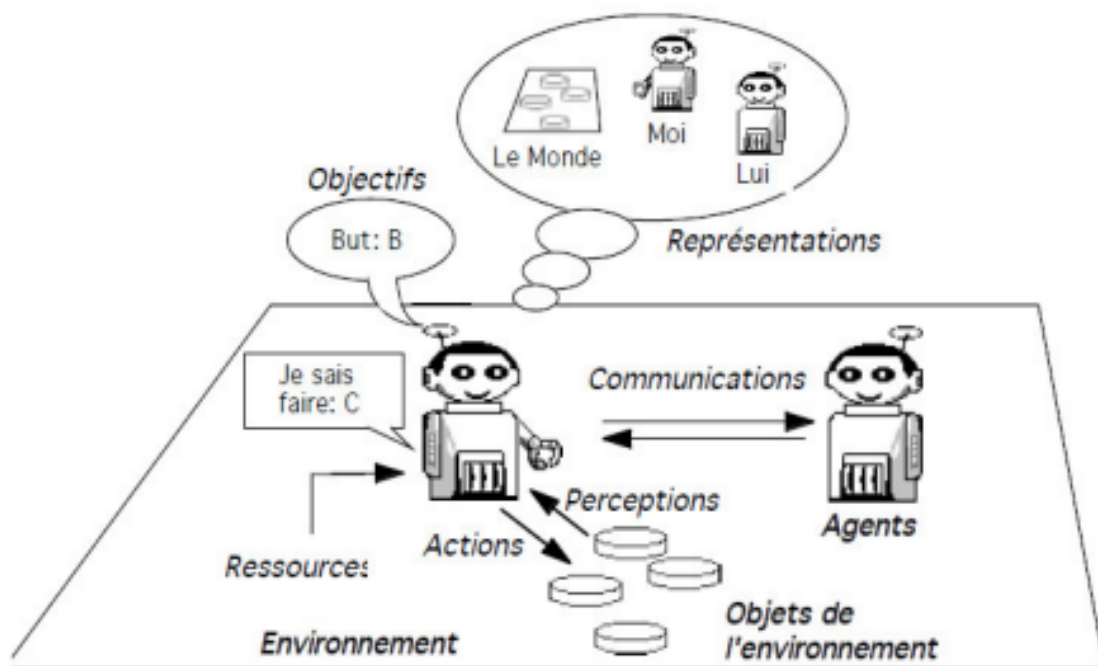


Fig. 1.14 : Système multi-agents.

1.7.2 Les éléments d'un S.M.A

- Un environnement : c'est-à-dire un espace disposant généralement d'une métrique.
- Un ensemble d'objets situés : c'est-à-dire qu'il est possible d'associer à chaque objet, à un moment donné, une position dans l'environnement.

- Un ensemble d'agents représentant des entités actives du système.

1.7.3 Architecture des systèmes multiagents

Un système multi-agent conçu autour d'agents communicants possède une architecture caractéristique qui est illustrée Fig I.12

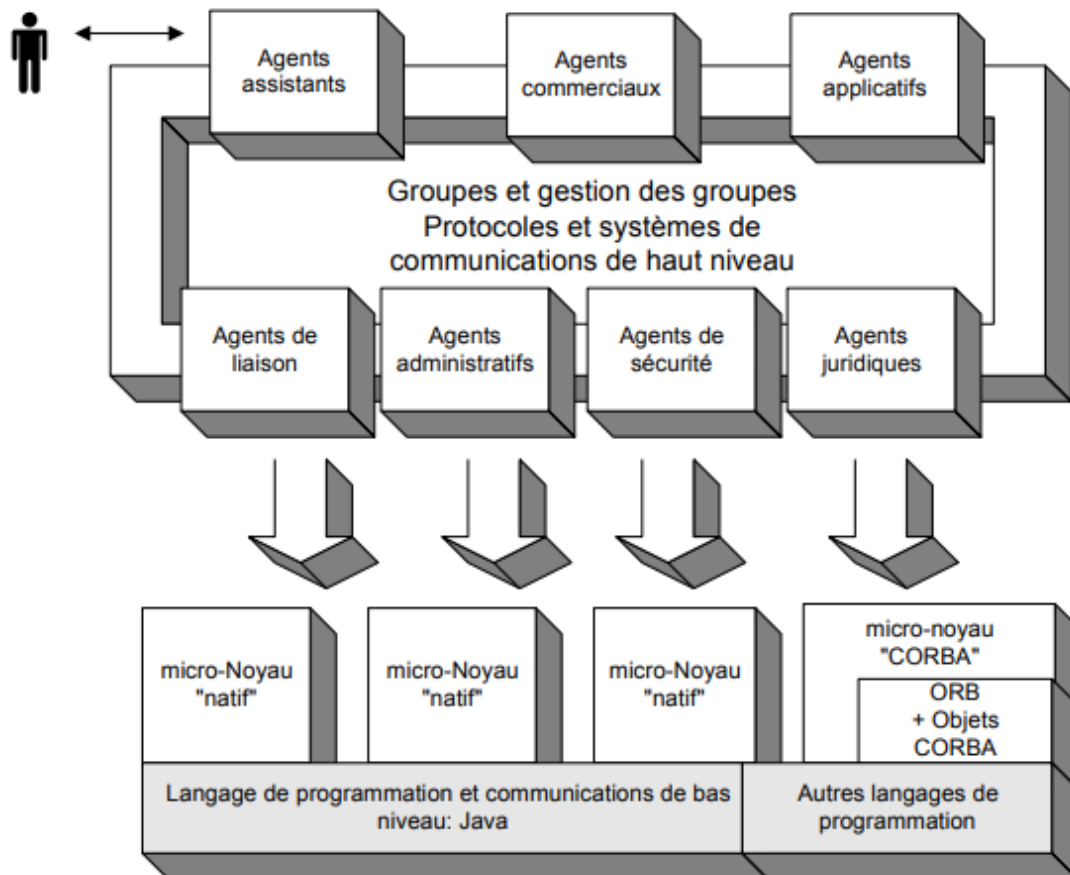


Fig. 1.15 : L'architecture d'un système multi-agent fonctionnant sur réseau.

Le niveau 0 correspond à l'ensemble des ressources disponibles telles que les mécanismes de communication de bas niveau (sockets Unix, protocoles TCP/IP ou http, etc.) ainsi que les mécanismes d'exécution parallèles tels que les «threads». C'est au dessus de cette couche, qui est utilisée comme un existant, que vient s'intégrer un système multi-agent proprement dit. Le niveau 1 décrit les couches de bas niveau d'un système multi-agent : les primitives de communication entre agents distants (primitives de type KQML), les serveurs de noms permettant à des agents d'entrer et de sortir du système (procédures de check-in check-out), ainsi que les moteurs qui implémentent le cycle de base de fonctionnement d'un agent. Ce cycle est lié à un processus en boucle de type perception/délibération/action. Le niveau 2 correspond à l'ensemble des mécanismes génériques qui sont mis en œuvre dans un système multi-agent. Il s'agit du niveau

fondamental sur lequel porte la plupart des recherches décrites dans la section précédente : définition de protocoles génériques de coopération, description d’agents administratifs et de liaison tels que les « courtiers » (ou brokers) qui mettent en rapport les agents qui demandent et ceux qui offrent des services, description de langage de contenu généraux, comportements génériques d’agents, etc. Le niveau 3 enfin traite des applications spécifiques et des domaines particuliers auxquels on dédie un système multi-agent particulier.

1.7.4 Coopération vs compétition

Les systèmes multi-agents apparaissent dans de nombreuses situations sociales, économiques et politiques. Dans leur travail, Tampuu2015 étendent l’architecture du *Deep Q-Learning Network* proposée par :contentReference[oaicite :0]index=0 aux environnements multi-agents et étudient l’interaction de deux agents contrôlés par des réseaux de neurones profonds indépendants dans le jeu vidéo classique :contentReference[oaicite :1]index=1.

En modifiant le schéma de récompense classique du jeu, les auteurs montrent comment des comportements compétitifs ou coopératifs peuvent émerger automatiquement. Dans un contexte compétitif, les agents apprennent à jouer efficacement et à marquer des points contre leur adversaire. En revanche, lorsqu’ils sont entraînés avec un schéma de récompense coopératif, les agents développent une stratégie optimale consistant à maintenir la balle en jeu le plus longtemps possible.

Les auteurs décrivent également la transition progressive entre des comportements compétitifs et coopératifs en fonction de la structure de la récompense. Ce travail montre que les réseaux de neurones profonds de type DQN constituent un outil pertinent pour l’étude de l’apprentissage décentralisé dans les systèmes multi-agents évoluant dans des environnements hautement complexes.

1.8 Apprentissage par renforcement multi-agent (MARL)

1.8.1 Formalisation mathématique (Dec-MDP)

Un processus de décision markovien décentralisé (Dec-MDP) est formellement défini par un tuple $\mathcal{M} = \langle I, S, \{A_i\}, P, R, \{\Omega_i\}, O, T \rangle$ où :

- $I = \{1, \dots, n\}$ est un ensemble fini d’agents.
- S représente l’espace d’état fini du système.
- $A = \alpha_1 \times \dots \times \alpha_n$ est l’ensemble des actions jointes, où A_i est l’ensemble fini des actions disponibles pour l’agent i .

- $P : S \times A \times S \rightarrow [0, 1]$ est la fonction de transition, notée $P(s'|s, a)$, qui définit la probabilité de transition vers l'état s' sachant que l'action jointe a a été effectuée dans l'état s .
- $R : S \times A \rightarrow R$ est la fonction de récompense immédiate perçue collectivement par les agents.
- $\Omega = \Omega_1 \times \dots \times \Omega_n$ est l'ensemble fini des observations jointes.
- $O : \Omega \times A \times S \rightarrow [0, 1]$ est la fonction d'observation, notée $O(o|a, s')$, décrivant la probabilité d'obtenir l'observation jointe $o = (o_1, \dots, o_n)$ après avoir effectué l'action a et atteint l'état s' .

1.8.2 Propriété d'Observabilité Collective

La particularité du Dec-MDP par rapport au Dec-POMDP réside dans l'observabilité jointe. Formellement, il existe une fonction f telle que pour chaque observation jointe $o = (o_1, \dots, o_n)$ ayant une probabilité non nulle d'apparaître dans l'état s , l'état est identifié de manière unique :

$$\forall o \in \Omega, \exists s \in S : P(s|o) = 1 \quad (1.38)$$

L'objectif des agents est de trouver une politique jointe $\pi = (\pi_1, \dots, \pi_n)$ qui maximise l'espérance de la somme des récompenses actualisées sur un horizon T :

$$V^\pi(s_0) = E \left[\sum_{t=0}^{T-1} \gamma^t R(s_t, a_t) \middle| s_0, \pi \right] \quad (1.39)$$

où $\gamma \in [0, 1[$ est le facteur d'atténuation.

1.8.3 Paradigmes d'entraînement en MARL

Apprentissage indépendant

L'une des façons les plus simples de construire un algorithme MARL consiste à utiliser un algorithme qui ne prend pas en compte le comportement des autres agents. Il s'agit donc de faire l'hypothèse qu'on peut faire abstraction du problème de non stationnarité.

Apprentissage centralisé

Dans l'apprentissage centralisé (CL), le modèle a accès à l'ensemble complet du jeu de données $\mathcal{D}$, collecté par le serveur central (PS) à partir des clients. Soit $\mathcal{D}_k$ le jeu de données local du $k^{\text{ème}}$ client, de taille D_k , tel que :

$$\mathcal{D} = \bigcup_{k \in \mathcal{K}} \mathcal{D}_k \quad \text{et} \quad D = \sum_{k \in \mathcal{K}} D_k,$$

où $\mathcal{K} = \{1, \dots, K\}$ représente l'ensemble des clients.

Dans ce contexte, l'entraînement du modèle basé sur l'apprentissage centralisé est effectué au niveau du serveur central en résolvant le problème d'optimisation suivant sur les paramètres apprenables θ :

$$\theta^* = \arg \min_{\theta} F(\theta) = \sum_{k=1}^K \frac{D_k}{D} F_k(\theta),$$

sous la contrainte :

$$f\left(X_k^{(i)} \mid \theta\right) = Y_k^{(i)}, \quad i = 1, \dots, D_k. \quad (1)$$

Le vecteur θ^* représente les paramètres du modèle après l'entraînement. Les variables $X_k^{(i)}$ et $Y_k^{(i)}$ désignent respectivement l'entrée et la sortie du $i^{\text{ème}}$ échantillon du jeu de données $\mathcal{D}_k$, avec $\mathcal{D}_k^{(i)} = (X_k^{(i)}, Y_k^{(i)})$. La fonction $F_k(\theta)$ correspond à la fonction de perte associée au client k .

Pour une tâche de régression, la fonction de perte est définie par :

$$F_k(\theta) = \frac{1}{D_k} \sum_{i=1}^{D_k} \left\| f\left(X_k^{(i)} \mid \theta\right) - Y_k^{(i)} \right\|^2. \quad (2)$$

Pour une tâche de classification binaire, la fonction de perte est donnée par :

$$F_k(\theta) = -\frac{1}{D_k} \sum_{i=1}^{D_k} \left[Y_k^{(i)} \ln f\left(X_k^{(i)} \mid \theta\right) + \left(1 - Y_k^{(i)}\right) \ln \left(1 - f\left(X_k^{(i)} \mid \theta\right)\right) \right]. \quad (3)$$

Apprentissage décentralisé

L'apprentissage décentralisé retire le serveur centralisé d'agrégation et connecte les nœuds à l'aide d'un réseau pair-à-pair ou d'une chaîne de bloc. Ainsi, l'apprentissage décentralisé hérite généralement des propriétés de l'apprentissage fédéré. De plus, il est possible d'y appliquer des stratégies d'agrégation similaires à l'apprentissage fédéré. Cependant, le retrait du serveur centralisé implique généralement plus de communication, puisque tous les nœuds doivent connaître tous les modèles, afin d'en faire l'agrégation et vérifier que l'agrégation soit correcte. Par exemple, dans un réseau d'apprentissage décentralisé dans lequel tous les K nœuds font l'entraînement et l'agrégation, alors K^2 modèles seront envoyés sur le réseau. L'apprentissage fédéré et décentralisé peut être exécuté dans deux variantes : entre appareils ou entre silos. Dans la première variante, il s'agit d'un entraînement entre appareils dans laquelle on attend plusieurs milliers d'appareils. Dans la deuxième variante, c'est plutôt un entraînement entre organisations, ce qui réduit considérablement le nombre de participants. De plus, on considère que les

participants ont des connexions stables et serveurs d'entraînement puissants.

Apprentissage centralisé avec exécution décentralisée (CTDE)

Dans un SMA avec un très grand nombre d'agents, il est impossible pour les agents d'avoir une connaissance parfaite du système, ce qui induit le problème d'observation partielle. Par exemple, dans le cas de voitures autonomes circulant dans une ville, il est irréalisable d'avoir des voitures avec une connaissance parfaite de l'environnement et de toutes les autres voitures. Pour pallier ce problème, cette approche repose sur le fait de pouvoir séparer la simulation en 2 phases : l'apprentissage et l'exécution. L'idée est de permettre aux agents d'utiliser pendant l'apprentissage des informations auxquelles ils n'auront plus accès pendant la phase d'exécution.

1.8.4 Méthodes coopératives centralisées

De nombreuses variantes d'un système de positionnement coopératif sont issues essentiellement des systèmes centralisés, où le traitement de l'estimation d'état est effectué par une unité centrale qui communique avec tous les robots du groupe. Le principe de l'estimation centralisée est de contenir l'estimation de l'état de chaque robot dans un seul vecteur d'état du groupe et de le mettre à jour en traitant les données de capteurs issues de chaque membre. Ces données peuvent être égocentriques, telles que les déplacements des robots et la pose absolue, ou bien des observations inter-robots, comme la pose relative entre deux robots.

L'un des progrès les plus notables dans le positionnement coopératif a été l'utilisation d'un système coopératif pour le positionnement d'un groupe de robots mobiles par une approche centralisée, qui emploie un Filtre de Kalman Étendu pour mettre à jour l'état du groupe. Plusieurs approches coopératives ont étudié les effets de l'utilisation des informations relatives, comme les mesures d'orientation et les distances relatives dans la mise à jour d'état du groupe, ainsi que la propagation des incertitudes.

Les méthodes de fusion de données ne se limitent pas au filtre de Kalman. D'autres outils sont utilisés, tels que l'estimateur par maximum a posteriori et l'estimation par maximum de vraisemblance. Certaines applications utilisent également différents types de capteurs, comme des capteurs de vision ou des ondes acoustiques, pour mesurer la pose relative entre les robots et mettre à jour l'état du groupe.

Algorithme MADDPG

Le but de l'algorithme MADDPG est d'apporter une série d'améliorations basées sur le cadre Actor-Critic et l'algorithme DDPG en utilisant le cadre d'entraînement de *Centralized Training and Decentralized Execution* (CTDE), afin qu'il puisse être appliqué à des environnements multi-agents complexes où l'apprentissage par renforcement pour un seul agent n'est pas approprié. Par conséquent, cet article étudie le problème impulsif OPEG basé sur l'algorithme MADDPG.

Pour une meilleure compréhension, cette section introduit d'abord le cadre et le principe de l'algorithme MADDPG pour le problème OPEG impulsif.

Le cadre du réseau dans l'algorithme peut être divisé en deux parties : le réseau Actor et le réseau Critic. Le réseau Actor adopte une conception distribuée, ce qui signifie que l'entrée du réseau Actor de chaque vaisseau spatial est son état environnemental observé actuel, s_P ou s_E , et la sortie est l'action que le vaisseau spatial doit choisir dans cette situation. Le réseau Actor est équivalent à un contrôleur en rétroaction du vaisseau spatial, et nous utilisons $\phi_{a_i}^{h_i}$ pour représenter le réseau Actor du vaisseau spatial i , où h_i représente les paramètres du réseau Actor, $i \in \{P, E\}$.

Le réseau Critic adopte une structure centralisée. Les entrées du réseau Critic incluent les états (s_P, s_E) et les actions (a_P, a_E), et la sortie est la valeur Q de chaque vaisseau spatial. Le réseau Critic du vaisseau spatial i est représenté par $Q_i^\#(s_P, s_E, a_P, a_E)$, où $\#_i$ désigne le paramètre du réseau Critic.

Le cadre d'entraînement de l'algorithme MADDPG pour le problème OPEG est montré dans la Fig. I.11. Le réseau Critic évalue les actions en fonction des informations d'état et d'action à la fois du poursuivant et de l'évadé, tandis que le réseau Actor ne repose que sur les informations d'observation du vaisseau spatial lui-même pour choisir ses actions. Ainsi, après l'entraînement de l'algorithme, toutes les parties représentées par des flèches rouges dans la Fig. 5 ne sont plus nécessaires. Chaque vaisseau spatial peut uniquement s'appuyer sur son propre réseau Actor pour prendre des décisions et interagir avec l'environnement (c'est-à-dire la partie à l'intérieur de la ligne pointillée verte sur la figure).

Ensuite, nous introduirons les réseaux inclus dans l'algorithme MADDPG et leurs formules de mise à jour par optimisation de gradient. Le retour attendu et son gradient pour le vaisseau spatial i peuvent s'écrire comme suit :

$$J(\theta) = E_{R_i} [\phi_{a_i}^{h_i}]$$

$$\nabla_\theta J(\theta) = E_{\phi_{a_i}^{h_i}} [\nabla_{h_i} \phi_{a_i}^{h_i} \nabla_{a_i} Q_i^\#(s_P, s_E, a_P, a_E)]$$

Dans le processus d'entraînement, le *replay buffer* et le *target network* doivent être ajoutés. La fonction principale du *replay buffer* est de stocker les échantillons d'expérience et de rompre la corrélation entre les échantillons. Nous empaquetons l'état joint actuel, l'action jointe, la récompense et l'état joint du pas suivant dans un échantillon $(s_P, s_E, a_P, a_E, r_P, r_E, s'_P, s'_E)$.

Le *target network* peut aider à améliorer la stabilité et la convergence du processus d'entraînement de l'algorithme. Soit Q'_i le *critic target network* du vaisseau spatial i , et de la même manière, le *target actor network* du vaisseau spatial i est ϕ'_{h_i} . Le réseau Critic peut être mis à jour en minimisant la perte $L(\#_i)$:

$$L(\#_i) = E_{s_P, s_E, a_P, a_E, r_P, r_E, s'_P, s'_E} [Q_i^\#(s_P, s_E, a_P, a_E) - y_i]^2$$

où

$$y_i = r_i + \gamma Q'_i(s'_P, s'_E, a'_P, a'_E), \quad a'_i = \phi'_{h_i}(s'_i)$$

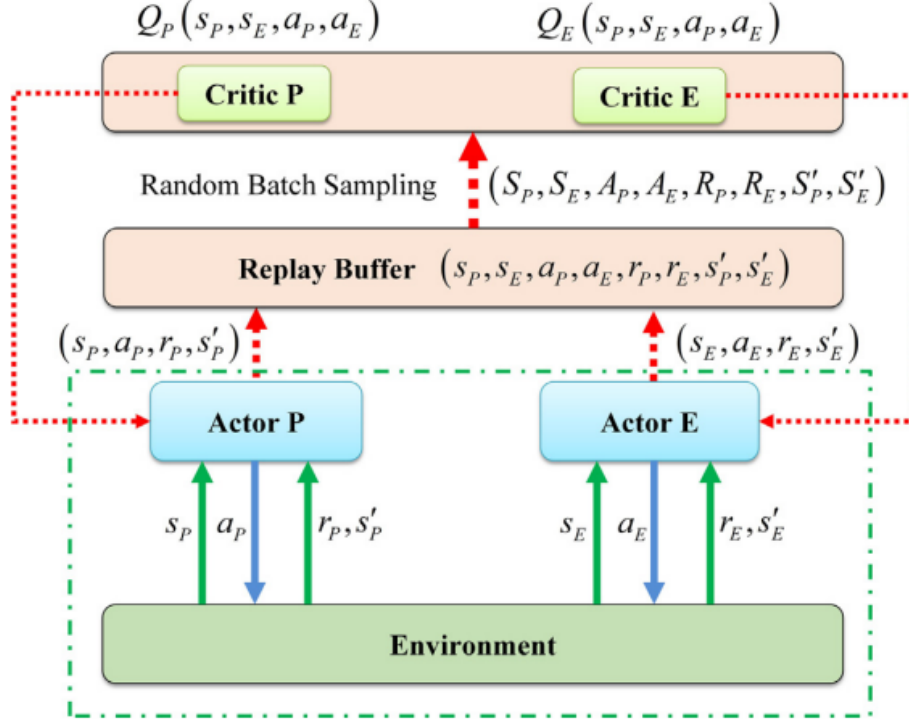


Fig. 1.16 : Le cadre d'entraînement de l'algorithme MADDPG pour le problème OPEG

1.9 Les Modèles de Prédiction Séquentielle : LSTM et GRU

1.9.1 Motivation : Le problème du gradient disparaissant

Les réseaux de neurones récurrents (RNN) classiques souffrent d'une limitation fondamentale lors du traitement de séquences longues : le **problème du gradient disparaissant**. Lors de la rétropropagation à travers le temps (BPTT), le gradient devient exponentiellement petit au fil des étapes temporelles, empêchant le réseau d'établir des dépendances entre des événements éloignés dans la séquence. Les architectures LSTM et GRU ont été conçues spécifiquement pour surmonter cette limitation.

1.9.2 Le réseau LSTM (Long Short-Term Memory)

Introduit par Hochreiter et Schmidhuber (1997), le LSTM enrichit le RNN classique d'un mécanisme de **portes** (*gates*) permettant de contrôler explicitement le flux d'information à travers le temps. Il repose sur deux états internes :

- **L'état caché** h_t : représente la mémoire à court terme.
- **L'état de cellule** c_t : représente la mémoire à long terme, protégée des modifications brutales.

Le LSTM régule ces deux états via trois portes :

a) La porte d'oubli (*Forget gate*) Elle décide quelle partie de la mémoire passée doit être effacée :

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1.40)$$

b) La porte d'entrée (*Input gate*) Elle décide quelles nouvelles informations sont intégrées à la cellule :

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (1.41)$$

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (1.42)$$

La mise à jour de la cellule s'écrit alors :

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (1.43)$$

c) La porte de sortie (*Output gate*) Elle détermine quelle information est transmise en sortie :

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (1.44)$$

$$h_t = o_t \odot \tanh(c_t) \quad (1.45)$$

Grâce à cette architecture, le LSTM est capable de mémoriser des dépendances sur plusieurs dizaines de pas de temps, ce qui le rend particulièrement adapté à la modélisation de séries temporelles complexes.

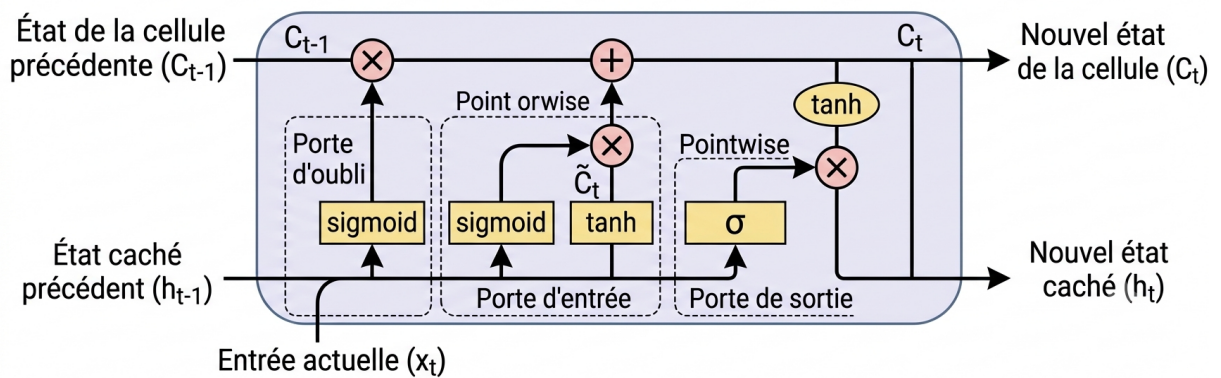


Fig. 1.17 : Structure Détaillée D'une Cellule LSTM

1.9.3 Le réseau GRU (Gated Recurrent Unit)

Proposé par Cho et al. (2014), le GRU est une version **simplifiée** du LSTM qui fusionne certaines portes afin de réduire la complexité computationnelle. Il ne maintient qu'un seul état caché h_t et opère à travers deux portes :

a) La porte de mise à jour (*Update gate*) Elle fusionne les fonctions d'oubli et d'entrée du LSTM :

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \quad (1.46)$$

b) La porte de réinitialisation (*Reset gate*) Elle contrôle l'influence du passé sur le calcul de l'état candidat :

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \quad (1.47)$$

Le nouvel état caché candidat et la mise à jour finale s'expriment comme suit :

$$\tilde{h}_t = \tanh(W \cdot [r_t \odot h_{t-1}, x_t]) \quad (1.48)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \quad (1.49)$$

Le GRU converge plus rapidement et nécessite moins de données d'entraînement, au prix d'une capacité de mémorisation légèrement inférieure sur des séquences très longues.

1.9.4 Comparaison LSTM et GRU

Le tableau suivant synthétise les principales différences entre les deux architectures :

Critère	LSTM	GRU
Nombre de portes	3 (Forget, Input, Output)	2 (Update, Reset)
États internes	h_t et c_t	h_t uniquement
Complexité	Élevée	Réduite
Vitesse d'entraînement	Plus lente	Plus rapide
Performances (longues séquences)	Supérieures	Légèrement inférieures
Quantité de données requises	Volumineuse	Modérée

Tab. 1.1 : Comparaison entre le LSTM et le GRU

1.9.5 Choix de l'architecture pour la prédiction de séries temporelles

Le choix entre LSTM et GRU dépend principalement de trois facteurs :

1. **La longueur des dépendances temporelles** : lorsque les séquences présentent des cycles complexes s'étendant sur plusieurs périodes, le LSTM offre une meilleure capacité de mémorisation grâce à son état de cellule c_t dédié.

2. **Le volume de données disponible** : avec un volume important, le LSTM exploite pleinement sa capacité de modélisation. En revanche, sur des jeux de données plus réduits, le GRU, moins sujet au surapprentissage, peut s'avérer plus performant.
3. **Les contraintes de temps de calcul** : le GRU, de par son architecture allégée, est préférable dans les contextes où la rapidité d'entraînement est une priorité.

De manière générale, le LSTM est privilégié lorsque la précision de la mémoire à long terme est critique, tandis que le GRU représente un bon compromis entre performance et efficacité computationnelle.

1.10 Conclusion

L'intelligence artificielle repose sur différents paradigmes d'apprentissage qui permettent aux systèmes de traiter, analyser et exploiter les données de manière autonome. Les réseaux de neurones et le deep learning ont considérablement renforcé les capacités de modélisation et de résolution de problèmes complexes.

L'apprentissage par renforcement, ainsi que son extension aux systèmes multi-agents, introduit une dimension décisionnelle et interactive, où plusieurs agents peuvent apprendre, coopérer ou entrer en compétition dans un environnement partagé. Ces approches constituent aujourd'hui un axe majeur dans le développement de systèmes intelligents, adaptatifs et performants.

Chapitre 2

Présentation de l'entreprise

2.1 Introduction

L'entreprise NAFTAL est une filiale de la compagnie nationale algérienne des hydrocarbures SONATRACH. Elle a été créée en 1981 et est spécialisée dans la distribution et la commercialisation de produits pétroliers et gaziers en Algérie.

2.2 Historique et évolution de l'entreprise NAFTAL

Issue de la sonatrach (Société Nationale pour la recherche, le transport, la production, la transformation et la commercialisation des hydrocarbures), l'Entreprise Nationale de Raffinage et de Distribution de Produits Pétroliers (erdp) a été créée par le décret N°80-101 datant du 06 avril 1980.

Entrée en activité le 01^{er} janvier 1982, elle fut chargée de l'industrie de raffinage et de la distribution de produits pétroliers. Le 04 mars 1985, les districts suivants : carburants, lubrifiants, pneumatiques et bitumes, ont été regroupés sous le nom und (Unité naftal de Distribution).

Durant l'année 1987, l'activité de raffinage est séparée de la distribution, conformément au Décret N°87-189 du 25 août 1987. Modifiant ainsi le décret N°80-101 du 06 avril 1980, ce texte donne naissance à une nouvelle entreprise nationale dénommée : « Entreprise nationale de commercialisation et de distribution de produits pétroliers » sous le sigle de “naftal”.

Dès l'année 1998, elle change de statut et devient une société par action (spa), filiale de la sonatrach à 100 %. Elle interviendra par la suite dans les domaines suivants :

- Dans l'enfûtage gpl.
- Dans la formulation des bitumes.
- Dans la distribution, le stockage et la commercialisation des carburants, gpl, lubrifiants, bitumes, pneumatiques et gpl/produits spéciaux.

- Dans le transport des produits pétroliers.

Le 01^{er} janvier 2000, l'activité gpl enfûtage est séparée de l'activité clp. Par décision N°S554 du 29 mars 2000, il a été procédé à l'organisation générale de la division clp et l'identification des zones de distribution "clp"(Carburants, Lubrifiants et Pneumatiques).

Par décision N°S555 du 29 mars 2000, il a été procédé à la création des zones de distribution clp. Par décision N°S606 du 10 février 2001, il a été procédé à l'organisation et la classification des centres Bitumes de la division Bitume.

Par décision N°S705 du 17 juin 2002, il a été procédé à la renominations des zones de distribution clp et gpl en "District". Par décision N°S766 du 22 décembre 2003, il a été procédé à la dissolution de la branche clpb. Par la décision N°S770 du 03 janvier 2004, il a été procédé à la dissolution des districts clp et à la création des districts commercialisation.

À partir du 01^{er} décembre 2006, l'activité carburant est séparée de l'activité commercialisation. L'activité carburants est chargée du stockage et du déstockage des carburants, tandis que l'activité commercialisation s'occupe essentiellement des achats, des ventes, du bilan annuel et des projets.



Fig. 2.1 : Logo NAFTAL

2.3 Principales tâches et responsabilités de NAFTAL

L'activité carburants du district repose sur une gestion rigoureuse des moyens matériels, humains et organisationnels. Ses principales missions se déclinent comme suit :

2.3.1 Identification et Organisation des Moyens

- **Moyens matériels et infrastructures :** Identification des équipements relevant de l'activité (camions-citernes, réseaux de canalisations), ainsi que des installations fixes de stockage.
- **Structures d'organisation :** Mise en place et suivi des services de maintenance, de la surveillance et de l'entretien des canalisations, ainsi que des laboratoires de reconnaissance des produits.

- **Moyens humains** : Gestion et coordination du personnel œuvrant spécifiquement pour l'activité carburants au sein du district.

2.3.2 Gestion Opérationnelle et Approvisionnement

- **Suivi des flux** : Suivre les plans établis par la branche carburant pour l'approvisionnement et le ravitaillement des dépôts. Un compte-rendu régulier des états d'exécution est transmis aux structures concernées.
- **Distribution et livraison** : Exécuter les programmes de distribution définis par les districts commercialisation afin d'assurer la livraison fluide de la clientèle.
- **Gestion des stocks** : Administrer les niveaux de carburants dans les dépôts et communiquer les points de situation périodiques à la branche.

2.3.3 Maintenance, Sécurité et Liaisons Externes

- **Exploitation et entretien** : Assurer le suivi technique des infrastructures de stockage, des parcs de camions et des canalisations rattachés au district.
- **Sûreté interne** : naftal assure, en concertation avec les responsables des centres et des canalisations, la protection et la sécurité des installations et des moyens de transport.
- **Relations inter-entreprises** : Gérer les relations avec les partenaires stratégiques tels que les raffineries naftec, ainsi que les prestataires de transport (sntr), et assurer la transmission des dossiers aux structures de la branche pour règlement.
- **Gestion administrative** : Approuver les Bordereaux Inter-Unités (biu) émis par les districts commercialisation vers le district carburant.

2.3.4 Organisation de naftal

Comme toute grande entreprise, naftal possède sa propre structure organisationnelle. Nous illustrons ici les principaux services et leurs missions respectives :

Service Informations de Gestion

Sa mission consiste à :

- Collecter, vérifier et analyser les informations de gestion du district.
- Élaborer les tableaux de bord et les rapports d'activité du District.
- Assurer l'installation, l'exploitation et la sauvegarde des logiciels de gestion et des données afférentes.
- Prêter assistance aux structures en matière d'exploitation des applications informatiques opérationnelles.

Département Administration et Moyens Généraux

Les missions du département amg sont :

- Assurer la gestion des moyens généraux du district.
- Assurer la gestion des ressources humaines .
- Assurer la gestion de l'administration générale.
- Assurer la gestion des œuvres sociales et culturelles.

Département Finance et Comptabilité

Ce département a pour mission de :

- Coordonner et suivre toutes les activités de comptabilité, de trésorerie, de budget et de patrimoine.
- Consolider, analyser les états comptables et veiller à la sincérité des comptes du District.
- Veiller à la concordance des écritures comptables avec les flux physiques et financiers.

Service Trésorerie

Il est composé de deux sections : la **Section Recettes** et la **Section Dépenses**. Sa mission est de :

- Suivre et contrôler les flux, les recettes et les dépenses de trésorerie.
- Traiter les dossiers de paiement d'investissement, des fournisseurs et autres dépenses.
- Établir les situations de rapprochement des comptes (recettes et dépenses).
- Contrôler et effectuer les comptabilisations des comptes et grands livres de trésorerie.
- Établir des rapports d'activités périodiques.

Service Comptabilité Générale

Il est composé de deux sections : la **Section Vérification et Contrôle des Dossiers** et la **Section Comptabilité**. Sa mission est de :

- Procéder aux écritures comptables conformément aux préconisations du Plan Comptable National.
- Élaborer les documents comptables réglementaires (Bilans, balances et livres).
- Contrôler les arrêtés de comptes et préparer les inventaires et bilans.
- Élaborer les analyses et synthèses comptables.
- Procéder aux opérations de clôtures et de réouvertures des comptes.

Service Budgets et Coûts

Ses diverses missions sont :

- Élaborer les budgets prévisionnels d'investissement et de fonctionnement du District.
- Consolider l'ensemble des charges nécessaires à la détermination des coûts.
- Contrôler et traiter les situations financières du District.
- Procéder aux ajustements des budgets et des crédits.
- Assurer le suivi régulier de la comptabilité analytique.

Département Transport & Technique

Il a pour mission :

- Élaborer les plans de maintenance préventive et curative des équipements, des dépôts et des canalisations, et en suivre l'exécution.
- Élaborer les plans annuels et pluriannuels de transport, en prenant en charge les besoins de distribution et de ravitaillement des produits commercialisés.
- Assurer le suivi de la réalisation des travaux sur site.
- Élaborer les plans et budgets d'investissement (rénovation, extension, remise à niveau, remplacement) des installations fixes, des canalisations et du réseau de stations-service.
- Établir un rapport d'activité périodique.

Ce département comporte les services suivants :

1. Service Exploitation et Maintenance :

- Vérifier l'application des prescriptions du règlement d'exploitation et de sécurité des équipements.
- Établir les indicateurs de performance de maintenance.
- Assurer la maintenance des installations au niveau des dépôts carburants.

2. Service Études et Réalisation :

- Établir la partie technique des cahiers des charges.
- Contrôler et diriger les différents travaux sur le terrain.
- Suivre les travaux programmés ayant trait aux projets de développement.

Les produits commercialisés par l'entreprise naftal

L'entreprise naftal propose une large gamme de produits pétroliers et dérivés adaptés aux besoins du marché national :

1. Les carburants :

- **Essence Normale** : Carburant avec plomb utilisé pour les anciens véhicules. Sa teneur en octane est inférieure à celle du super. Étant très polluant, sa commercialisation a fortement diminué.
- **Essence Super** : Utilisée pour les moteurs automobiles, elle se caractérise par un indice d'octane plus élevé que l'essence normale.
- **Essence sans Plomb** : Combustible issu du raffinage, exempt de plomb pour limiter l'impact environnemental et préserver la santé publique.
- **Gasoil** : Aussi appelé Diesel, ce carburant issu du raffinage est destiné aux moteurs diesel routiers et non routiers.

2. Les lubrifiants (huiles et graisses) : Un lubrifiant est une matière onctueuse (liquide ou solide) ayant pour propriété d'atténuer le frottement et l'usure des pièces mécaniques, d'évacuer la chaleur et de prévenir la corrosion . On distingue :

- Huiles pour moteurs.
- Huiles isolantes pour transformateurs électriques.
- Huiles pour transmissions hydrauliques.
- Graisses pour les rouages.

3. Les pneumatiques : Accessoires indispensables assurant le contact entre le véhicule et le sol. Formés d'un tube de caoutchouc gonflé d'air, ils amortissent les chocs et sont des éléments capitaux de sécurité.

4. Le bitume : Substance hydrocarburée très visqueuse, solide à température ambiante. Il est obtenu par distillation du pétrole brut. C'est le produit le plus lourd, principalement utilisé dans les travaux publics (routes, autoroutes, pistes d'aéroports).

5. Fuel (léger et lourd) : Huile industrielle dérivée du pétrole (distillation au-dessus de 275 °C). Utilisé dans les chaudières et certains moteurs diesel, il est classé parmi les ressources énergétiques fossiles.

Le marché de l'entreprise

La clientèle de naftal est classée en deux grandes catégories :

- **Clientèle Réseaux** : Elle se divise selon le lien juridique avec l'entreprise :
 - **Gestion Directe (gd)** : Stations gérées directement par naftal avec son propre personnel.
 - **Gestion Libre (gl)** : Stations appartenant à naftal mais dont l'exploitation est confiée à des gérants privés.
 - **Revendeurs Ordinaires (ro)** : Points de vente de faible capacité liés par contrat commercial.
 - **Points de Vente Agréés (pva)** : Stations-service réalisées par des investisseurs privés.
- **Gros consommateurs** : Clients disposant de leurs propres capacités de stockage (bacs ou réservoirs) :
 - Entreprises publiques et privées.
 - Administrations et collectivités locales.
 - Hôpitaux et Défense Nationale.
 - Secteurs agricole et domestique.

Moyens et infrastructures de Naftal

Avec un effectif de 30 000 agents, naftal est le premier distributeur en Algérie .

Elle fournit 10 millions de tonnes de produits par an (carburants, gpl, bitumes et lubrifiants).

Pour assurer cette mission, elle dispose d'un patrimoine matériel important :

1. Dépôts et centres

- **47** dépôts carburants terrestres.
- **42** centres et minicentres GPL.
- **9** centres GPL vrac.
- **47** dépôts relais.
- **30** dépôts aviation.

- **6** centres marins.
- **15** centres bitumes.
- **24** centres lubrifiants & pneumatiques.

2. Stations-service

- Environ **2 010** stations-service à l'échelle nationale.
- Selon certaines sources, ce chiffre peut varier légèrement (jusqu'à **2 300** stations), notamment en fonction des stations en gestion directe ou privée.

3. Parc roulant et logistique

- Environ **3 300** véhicules pour assurer le transport et la distribution.
- Un réseau de dépôts et relais bien maillé pour garantir la distribution finale des produits.

4. Modernisation et Digitalisation

- Naftal développe activement le paiement électronique (**e-paiement**) dans ses stations.
- On enregistre environ **42 000** transactions par mois, marquant un progrès significatif dans la modernisation des services clients.

2.3.5 Réseau de distribution et moyens de transport

Le réseau national de distribution repose sur trois étapes essentielles :

- **L'approvisionnement** : Acheminement des produits depuis la raffinerie vers un **centre primaire** (ex : centre de stockage de Remchi). Ce transfert s'effectue par pipeline ou par cabotage (navire).
- **Le ravitaillement** : Transport des produits depuis les centres primaires vers les **centres secondaires** (dépôts). Ce transfert se fait par rail (train) ou par camions citerne (wagon-citerne).
- **La livraison** : Phase finale assurant la disponibilité des produits en zones de consommation (stations-service). Elle s'effectue de deux manières :
 - **Livraison directe** : Utilisation des camions propres à naftal pour livrer les clients depuis un centre secondaire.
 - **Livraison en droiture** : Livraison directe depuis un centre primaire vers les gros consommateurs (industries, aéroports, chantiers) via des camions privés loués par l'entreprise.

2.4 Présentation du centre « Enfuteur » NAFTAL Tlemcen

Le centre de stockage et de distribution des carburants de remchi est implanté au sein de la zone industrielle de Tlemcen sur une superficie de 10 hectares. Il constitue un pôle stratégique pour le stockage et le conditionnement.

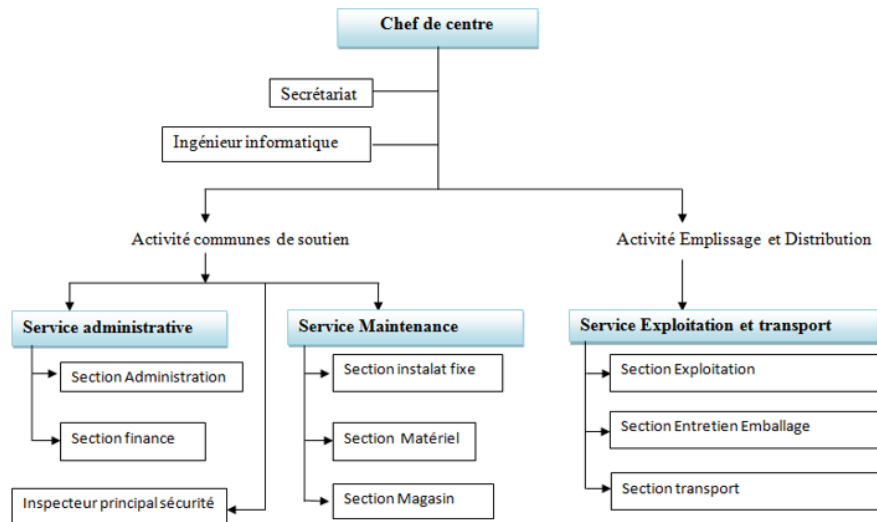


Fig. 2.2 : Organigramme

2.4.1 Présentation de la canalisation multiproduit ASR

Description générale de l'oléoduc multiproduit ASR :

La canalisation multiproduit ASR relie la raffinerie d'Arzew à la station finale de Remchi, en passant par Sidi Bel Abbés.



Fig. 2.3 : Pipelines de canalisation

Elle alimente les centres de stockage de Sidi Bel Abbès et de Remchi et constitue un maillon logistique central pour l'approvisionnement régional.

La longueur totale du tracé est d'environ 168 km :
(Arzew → Sidi Bel Abbès → Sidi Abdelli → Remchi).

La capacité totale de transport du pipeline est d'environ 10 100 m³, répartie comme suit : 6 200 m³ sur le tronçon Arzew → Sidi Bel Abbès, Et 3 900 m³ sur le tronçon Sidi Bel Abbès → Remchi. Cette capacité permet d'assurer un flux régulier et sécurisé des produits pétroliers vers les zones desservies.

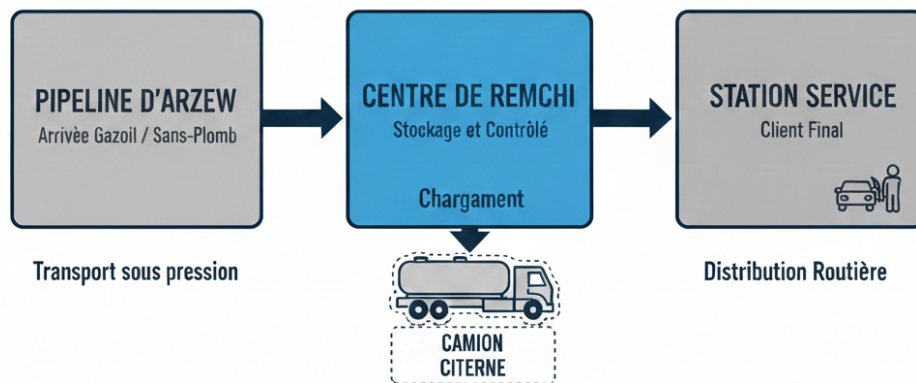


Fig. 2.4 : Chaîne de valeur du carburant

Configuration et caractéristiques techniques :

Le pipeline ASR est composé de trois tronçons aux diamètres et épaisseurs différenciés, ce qui permet d'optimiser le transport multiproduit selon les débits et les pressions :

- **Tronçon 1** : 12", épaisseur 9,52 mm : Raffinerie d'Arzew → Sidi Bel Abbès (~ 85 km).
- **Tronçon 2** : 10", épaisseur 6,35 mm : Sidi Bel Abbès → Sidi Abdelli (~ 56 km).
- **Tronçon 3** : 8", épaisseur 6,40 mm : Sidi Abdelli → Remchi (~ 27 km).

Analyse scientifique et hydraulique

D'un point de vue scientifique, la diminution progressive du diamètre le long du tracé permet de maintenir une pression et une vitesse d'écoulement adaptées, malgré les pertes de charge naturelles dues à la longueur du pipeline. En réduisant le diamètre, on compense la baisse de pression et on évite la décélération excessive du fluide, ce qui améliore la stabilité hydraulique et l'efficacité énergétique du transport.

Ces caractéristiques techniques permettent la **circulation séquentielle** et sécurisée de différents produits (*Essence Sans Plomb, Gazoil*), avec des dispositifs de séparation et d'exploitation adaptés aux profils des produits transportés.

Le parc et le centre de stockage (site de Remchi) :

Le centre carburant de Remchi constitue un maillon terminal du système ASR. Il reçoit les produits via l'oléoduc multiproduit et assure le stockage, la décantation, le contrôle qualité et l'expédition du Gasoil (GS) et de l'Essence Sans Plomb (SP).

Le site comprend généralement plusieurs zones fonctionnelles :

- **Réception Pipeline :** Contrôle de pression par vannes et débitmètres, avec suivi rigoureux des « batches » (lots) pour garantir l'intégrité des produits.
- **Parc de Cuves :** Réservoirs cylindriques équipés de détecteurs de niveau (radar/ultrason), de dispositifs anti-débordement et d'évents anti-surpression.
- **Chargement Camion :** Expédition via des bras de chargement avec compteurs volumétriques, filtres anti-pollution et systèmes de mise à la terre.
- **Sécurité & HSE :** Protection garantie par un réseau incendie (eau/mousse), des détecteurs de gaz et un plan d'intervention conforme aux normes NAFTAL.

2.4.2 Process du Terminal d'Arrivée REMCHI

Le terminal d'arrivée est le point où le pétrole raffiné (Gasoil ou Sans-plomb) arrive depuis le pipeline principal. Le rôle de ce terminal est de recevoir le produit, le mesurer, le contrôler, puis l'orienter vers les cuves de stockage.

Arrivée du produit depuis le pipeline

Le produit est acheminé sous pression depuis l'axe *Arzew → Sidi Bel Abbès* avec un suivi rigoureux de la pression (env. 3,5 bars) et de la température (env. 25 °C). Ces paramètres sont fondamentaux pour l'exploitation car ils déterminent la densité, la précision du calcul du volume réel et permettent la détection d'éventuels mélanges entre les lots. Enfin, l'intégrité du système est assurée par une gare de racleur (*Pigging Station*) située en amont de l'entrée du terminal.

Traversée des organes de sécurité

Avant son admission dans les unités du terminal, le carburant transite par un système de contrôle et de sécurité multicouche. Ce dispositif comprend des vannes manuelles (HV) et motorisées permettant l'isolation de la conduite en cas d'urgence ou l'orientation du flux selon le produit (Gasoil, Sans Plomb ou retour). La stabilité du débit est assurée par des vannes de régulation de pression (PCV), tandis que l'intégrité structurelle des équipements est garantie par des soupapes de sécurité (PSV). Ces dernières interviennent pour évacuer tout excès de pression, prévenant ainsi les risques de surpression avant le traitement final du produit.

Système de comptage

traverse un dispositif de sécurité et de contrôle essentiel. Ce système repose sur des vannes manuelles (HV) et motorisées permettant l'isolation de la conduite ou l'aiguillage du flux (Gasoil, Sans Plomb ou retour). La régulation du débit et la stabilité opérationnelle sont assurées par des vannes de contrôle de pression (PCV), tandis que des soupapes de sécurité (PSV) protègent les installations contre tout risque de surpression accidentelle. Cet ensemble garantit une gestion sécurisée et un contrôle optimal du produit avant son admission définitive dans le terminal.

Séparation, direction et contrôle des lignes

Une fois les mesures effectuées, le produit est orienté vers sa ligne de destination spécifique (Gasoil ou Sans-plomb) via un réseau de vannes manuelles (HV). Ce système d'aiguillage permet de sélectionner avec précision la conduite adéquate, d'éviter toute erreur d'orientation et de basculer le flux entre les différentes cuves de stockage (A ou B). En complément de l'aiguillage, l'installation assure un monitoring continu de la pression interne, de la température, ainsi que de l'état d'encrassement des filtres et de la performance des pompes.

D'un point de vue historique, bien que le centre traitait initialement quatre produits (Gasoil, Sans-plomb, Kérosène et Super), le transport par pipeline est aujourd'hui rationalisé autour du Gasoil et du Sans-plomb. Les infrastructures et cuves autrefois dédiées au Kérosène et au Super ont été réaffectées pour augmenter la capacité de stockage du Sans-plomb.

Stockage dans les cuves

La zone de stockage du terminal se concentre sur la gestion des mélanges (interfaces) via les cuves A et B, équipées d'une instrumentation complète (niveaux, températures, sécurité trop-plein). La séparation s'effectue par analyse de densité : la cuve A reçoit les contaminations lourdes et la cuve B les légères.

Historiquement, les capacités autrefois dédiées au Kérosène et au Super ont été réaffectées au Sans-plomb pour répondre à l'évolution de la demande. Il est à noter que les cuves de produits purs ne sont pas représentées sur ce schéma technique simplifié.

Gare de Racleur (Pigging Station)

La gare de racleur se situe avant le terminal, sur la ligne d'arrivée. Elle est indispensable pour nettoyer le pipeline, séparer les produits et inspecter la conduite.

Citerne de purge

La citerne de purge, reliée à la gare de racleur par des lignes de collecte (représentées en orange), recueille les excédents de produit évacués lors des ouvertures de soupapes. Cet équipement

assure la protection de l'environnement, la sécurité des opérateurs lors des opérations de raclage et le maintien de la propreté du système. Une fois collectés, ces résidus sont réorientés par pipeline vers les cuves de contamination A ou B en fonction de leur densité (lourde ou légère).

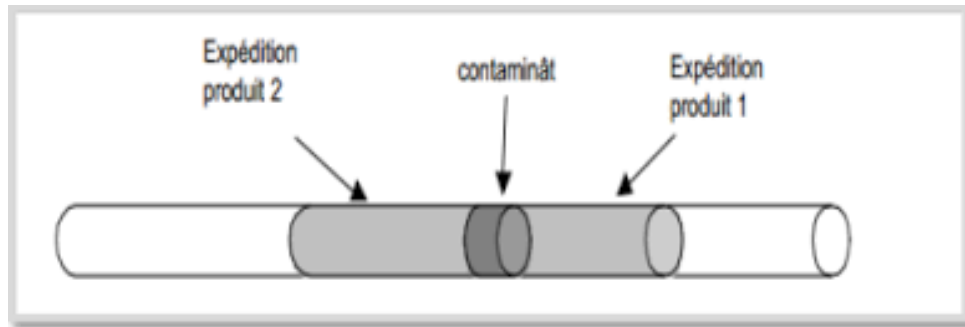


Fig. 2.5 : Carburant arrivé par le pipeline

2.4.3 Gestion des Stocks de Carburants

La gestion des stocks de carburants est un aspect essentiel du fonctionnement de l'entreprise, car elle permet d'assurer la disponibilité des produits, de préserver leur qualité et d'optimiser l'utilisation des capacités de stockage. Durant mon stage, j'ai pu observer le système mis en place pour le stockage, le suivi et la gestion des différents produits pétroliers au sein du centre de stockage.

- **Zones de stockage :** Le centre comporte deux zones de stockage du carburant, chacune équipée de neuf cuves destinées au Gasoil (GS) et à l'Essence sans plomb (SP). La capacité de ces cuves varie entre 500 m^3 et $6\,450\text{ m}^3$.
- **Gestion du contaminat :** Il existe deux cuves dédiées au stockage de « contaminat », mélange de deux produits au point de contact lors du transport par pipeline, avec une capacité d'environ 500 m^3 chacune.
- **Réinjection et classification :** Le produit contaminé est stocké temporairement, puis réinjecté dans les cuves de produit pur selon sa densité (ρ) :
 - $725 \leq \rho \leq 775\text{ kg/m}^3$: Essence sans plomb (SP)
 - $810 \leq \rho \leq 860\text{ kg/m}^3$: Gasoil (GS)
 - $775 \leq \rho < 790\text{ kg/m}^3$: Contaminat léger (Fraction légère)
 - $790 \leq \rho < 810\text{ kg/m}^3$: Contaminat lourd (Fraction lourde)

Pour connaître en temps réel l'état des cuves, deux méthodes sont utilisées :

Méthode 1 : Système de centralisation des données (Logiciel de supervision)

Ce logiciel centralise les données de terrain (Tank Scanner, Multi Scan, API) via les protocoles Modbus et OPC DA. Conçu pour la gestion à grande échelle (jusqu'à 400 réservoirs), il offre une visualisation en temps réel et des calculs de stock précis. Sa structure modulaire permet une gestion évolutive du site tout en garantissant des mesures certifiées, conformes aux exigences des transactions commerciales.

Méthode 2 : Manuelle

En complément du système automatique, un suivi physique est réalisé via des jauges graduées. Le technicien relève la hauteur du produit (en mètres) et la convertit en volume (m^3) à l'aide de la **fiche de barémage** spécifique à chaque cuve (par exemple : $1,33\text{ m} \rightarrow 74,914\text{ m}^3$).

- **Stratégie de gestion (FIFO) :** Le centre applique la méthode *First In, First Out* (Premier Entré, Premier Sorti). Cette approche garantit une rotation régulière des stocks, évitant ainsi la stagnation du carburant et préservant ses propriétés physico-chimiques avant la livraison.

2.4.4 Les opérations de chargement

Le cycle de chargement au dépôt de Remchi repose sur un protocole rigoureux alliant sécurité opérationnelle et supervision informatique :

- **Accueil et Identification :** À son arrivée, le bon d'enlèvement du camion est scanné pour vérifier les informations critiques (capacité, compartiments, destination). Un numéro d'ordre est attribué pour réguler la file d'attente.
- **Supervision par ALMA Ingénierie :** Ce logiciel automatise et sécurise l'opération. L'opérateur y configure la mission (quantité, produit, destination), tandis que le système surveille en temps réel les équipements (bras, vannes, débitmètres) et archive les données de traçabilité (volumes, densités).
- **Sécurité Incendie :** Avant tout transfert, la **mise à la terre** du véhicule est une étape obligatoire. Elle permet de dissiper l'électricité statique et d'éliminer tout risque d'étincelle en présence de vapeurs inflammables, protégeant ainsi le personnel et les installations.

2.4.5 Laboratoire de contrôle de qualité

Dans ce centre de distribution, les produits stockés sont le gasoil et l'essence sans plomb. Le gasoil est un carburant plus lourd, moins volatil et utilisé principalement dans les moteurs diesel, tandis que l'essence sans plomb est plus légère, plus volatile et destinée aux moteurs à allumage.

Le laboratoire du site réalise des opérations de recertification, bien que les carburants soient déjà certifiés à leur sortie de la raffinerie.

L'objectif est de vérifier que leur qualité n'a pas été altérée pendant le transport par pipeline ou le stockage. Les essais réalisés sont conformes aux exigences de la norme ISO 17025, qui garantit la compétence technique du laboratoire, la fiabilité des méthodes d'essai et la traçabilité des mesures selon un référentiel reconnu internationalement.

2.5 Conclusion

Ce deuxième chapitre nous a permis de situer le cadre opérationnel de notre étude en présentant le centre de distribution de Naftal Remchi. À travers l'analyse de son infrastructure, de sa capacité de stockage et de la gestion de sa flotte de camions-citernes, nous avons pu cerner les enjeux logistiques majeurs auxquels ce centre est confronté quotidiennement.

L'étude de l'organisation actuelle a mis en évidence la complexité de la chaîne de distribution, allant de la réception jusqu'à la livraison finale vers les stations-service. Nous avons constaté que, malgré une infrastructure robuste, l'efficacité globale du centre repose sur une coordination fine entre la disponibilité des ressources mobiles et le respect de contraintes de sécurité strictes.

En somme, cette immersion au cœur de Naftal Remchi constitue une étape charnière. Elle nous a permis de collecter les données réelles et de comprendre les spécificités du terrain indispensables à la phase de modélisation. Ce diagnostic exhaustif prépare ainsi le terrain pour le chapitre suivant, consacré à la conception et à l'implémentation de notre solution d'optimisation, visant à transformer ces défis logistiques en opportunités de performance grâce aux outils de simulation et d'intelligence artificielle.

Chapitre 3

Prédiction de la demande et planification optimisée de la distribution des carburants par systèmes multi-agents à apprentissage par renforcement

3.1 Introduction

L'intégration de l'Intelligence Artificielle dans le secteur de la logistique pétrolière représente aujourd'hui un levier stratégique pour répondre aux défis de la complexité et de l'incertitude. Dans un environnement où les flux de distribution sont soumis à des variations constantes, l'IA permet de dépasser les modèles rigides pour offrir une réactivité et une précision accrues dans la prise de décision.

Ce troisième chapitre est dédié à la mise en œuvre pratique et à l'expérimentation de ces solutions d'intelligence artificielle au sein du système de distribution de Naftal. Après avoir établi les fondements théoriques des réseaux de neurones et de l'apprentissage par renforcement, il s'agit désormais de tester la capacité de ces technologies à transformer une gestion logistique conventionnelle en un système dynamique et prédictif. Cette phase expérimentale s'appuie sur l'utilisation de données historiques réelles pour confronter les méthodes d'optimisation par apprentissage aux approches de calcul exactes. À travers une série de simulations et de tests de robustesse, ce chapitre expose la transition vers une planification autonome, capable de s'adapter aux fluctuations de la demande tout en garantissant l'efficacité opérationnelle des tournées de livraison.

3.2 Problématique

La gestion de la distribution chez Naftal, bien qu'optimisée par des méthodes mathématiques exactes (MILP), reste confrontée à des défis de flexibilité et d'anticipation que l'approche classique ne peut totalement résoudre.

Le problème central de cette étude de Master s'articule autour de trois axes liés à l'intelligence artificielle :

- **L'incertitude et la volatilité de la demande :** Si le modèle d'ingénierie traitait des données de demandes figées, la réalité opérationnelle est marquée par des fluctuations imprévisibles. Le défi n'est plus seulement de livrer une commande connue, mais d'anticiper la consommation des stations pour transformer une logistique réactive en une planification proactive. Sans cette capacité de prédiction, le système reste dépendant de saisies manuelles tardives qui limitent la réactivité globale.
- **L'apprentissage de la décision en environnement complexe :** La résolution mathématique exacte, bien que performante, devient très coûteuse en temps de calcul dès que les contraintes se multiplient. En remplaçant l'optimisation statique par un système Multi-Agents, nous cherchons à doter chaque camion d'une intelligence capable d'apprendre les meilleures stratégies de livraison par expérience. L'enjeu est de passer d'un calcul rigide à un apprentissage dynamique capable de maximiser les livraisons quotidiennes tout en gérant les priorités.
- **La robustesse face aux scénarios variables :** Le modèle doit prouver sa validité non plus sur une seule semaine type, mais sur des horizons temporels futurs et incertains. La difficulté réside dans l'intégration de données prédites dans un moteur d'exécution qui doit maintenir un respect strict des contraintes de compartimentage et de sécurité de Naftal, quel que soit le scénario généré.

Comment l'hybridation des techniques d'apprentissage profond et d'apprentissage par renforcement permet-elle de transformer une application d'optimisation statique en un système expert capable d'anticiper la demande et d'automatiser les décisions de transport pour minimiser durablement les reliquats ?

La problématique de ce projet de Master est donc de concevoir une architecture intelligente capable de s'auto-adapter à n'importe quel scénario de demande, garantissant ainsi une efficacité logistique maximale pour Naftal dans un contexte de distribution en flux tendus.

3.3 Objectifs du projet

L'objectif principal de ce travail est de concevoir un système expert capable de transformer la planification de la distribution chez Naftal en une opération proactive et intelligente. Pour y parvenir, notre démarche se décline en trois objectifs opérationnels :

- **Anticipation de la demande** : Intégrer un module de prédiction basé sur l'architecture de réseaux de neurones récurrents *Long Short-Term Memory* (LSTM) afin d'anticiper avec précision les besoins réels des stations et d'affranchir le système des limites liées aux données historiques figées.
- **Optimisation par apprentissage** : Implémenter une approche d'optimisation par apprentissage par renforcement multi-agents (MARL) permettant à la flotte de camions d'apprendre dynamiquement les meilleures stratégies de tournée, avec pour priorité de maximiser le volume de produits pétroliers distribués quotidiennement.
- **Validation et Robustesse** : Démontrer la fiabilité de l'application en la testant sur des scénarios de demandes prédites, prouvant ainsi que l'hybridation de ces techniques d'IA garantit une réduction significative des reliquats et une adaptation totale aux variations imprévisibles du marché.

3.4 Présentation des données de Naftal

La performance du modèle d'Intelligence Artificielle repose sur la précision des données d'entrée. Ces dernières constituent l'environnement dans lequel les agents vont évoluer et apprendre à optimiser la distribution.

3.4.1 Caractérisation de la flotte de camions-citernes

Le parc de camions utilisé pour l'entraînement et le test de nos algorithmes se compose de 26 véhicules appartenant à Naftal. L'hétérogénéité de cette flotte est un paramètre majeur : elle définit les capacités d'action de chaque agent dans notre modèle d'apprentissage par renforcement multi-agents (MARL).

Identification	Nombre	Capacité Totale	Détail des Compartiments (m^3)
Camion 01 à 04	4	$12m^3$	3 – 3 – 3 – 3
Camion 05 à 06	2	$15m^3$	4 – 2 – 3 – 6
Camion 07 à 16	10	$27m^3$	6 – 7 – 7 – 7
Camion 17 à 26	10	$30m^3$	7 – 7 – 8 – 8

Tab. 3.1 : Configuration technique de la flotte de transport Naftal pour l'apprentissage.

Cette hétérogénéité impose une contrainte de sélection rigoureuse au système intelligent. Le moteur d'optimisation doit apprendre à affecter le véhicule dont la configuration de compartiments permet de satisfaire au mieux la demande prédite en Gazole (GS) et Essence (SP), tout en respectant la contrainte de remplissage à 100% de chaque compartiment utilisé pour des raisons de sécurité.

3.4.2 Prétraitement et nettoyage des données

Avant d'appliquer tout modèle d'intelligence artificielle, une étape rigoureuse de prétraitement des données brutes de NAFTAL a été nécessaire afin de garantir la qualité et la cohérence des entrées fournies aux algorithmes.

Collecte et inspection des données

Les données historiques de consommation journalière en carburant (Gasoil GS et Essence SP) ont été extraites depuis les fichiers Excel de gestion du dépôt de Remchi, couvrant l'ensemble des stations-service desservies sur la région de Tlemcen. Une première inspection a permis d'identifier les principales anomalies :

- Valeurs manquantes (jours sans livraison ou sans enregistrement).
- Valeurs aberrantes (*outliers*) dues à des erreurs de saisie ou à des livraisons exceptionnelles.
- Irrégularité temporelle (jours non ouverts, fermetures ponctuelles).

Traitement des valeurs manquantes

Les valeurs manquantes ont été traitées selon deux règles :

- Pour les jours fériés ou les jours sans activité recensés, la consommation a été remplacée par zéro.
- Pour les valeurs manquantes isolées (moins de trois jours consécutifs), une interpolation linéaire a été appliquée afin de préserver la continuité de la série temporelle.

Détection et suppression des valeurs aberrantes

Les valeurs aberrantes ont été détectées par la méthode de l'écart interquartile (*IQR*) : toute valeur supérieure à $Q_3 + 1,5 \times IQR$ ou inférieure à $Q_1 - 1,5 \times IQR$ a été considérée comme aberrante et remplacée par la médiane glissante sur une fenêtre de sept jours.

Classification des stations et filtrage

Les stations ont été classifiées selon leur nombre de jours actifs de consommation sur la période d'observation :

- **Catégorie A** (≥ 60 jours actifs) : données suffisantes pour entraîner un modèle LSTM par station.
- **Catégorie B** (15 à 59 jours actifs) : prédiction par moyenne mobile exponentielle (EWMA).
- **Catégorie C** (< 15 jours actifs) : stations exclues de la prédiction automatique.

Normalisation

Afin d'accélérer la convergence des réseaux de neurones et d'éviter les biais liés à l'amplitude des valeurs, chaque série temporelle a été normalisée individuellement par un scaler *Min-Max* propre à chaque station :

$$x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

Les scalers sont sauvegardés et réappliqués lors de l'inférence pour inverser la normalisation et obtenir la prédiction en litres réels.

Construction des séquences d'entrée

Pour l'apprentissage supervisé du modèle LSTM, les séries temporelles ont été transformées en séquences glissantes de longueur $T = 7$ jours, chaque séquence étant associée à la consommation du jour suivant :

$$(x_{t-6}, x_{t-5}, \dots, x_t) \longrightarrow \hat{x}_{t+1}$$

Des variables exogènes ont également été intégrées : le jour de la semaine (encodé par fréquence de consommation) et le type de carburant (GS ou SP).

3.5 Configuration du réseau de distribution

Le réseau logistique étudié s'articule autour d'un point central unique, le dépôt de Remchi, qui assure l'approvisionnement d'un ensemble de stations-service réparties sur un large périmètre géographique. Dans le cadre de notre approche par MARL, ce réseau est modélisé comme l'environnement d'évolution des agents. La complexité de cet environnement réside dans l'hétérogénéité des distances à parcourir, lesquelles constituent des variables d'état critiques influençant directement le temps de rotation des camions et les coûts de transport.

Présentation des données de demande quotidienne

L'efficacité de notre système hybride (prédiction et optimisation) est testée sur une période réelle d'exploitation au niveau du dépôt de Remchi. Cette phase expérimentale est cruciale car elle permet de confronter l'apprentissage par renforcement multi-agents (MARL) aux réalités du terrain.

Pour chaque journée de la période de référence (semaine de février), les demandes des stations-service en Essence Sans Plomb (SP) et en Gazole (GS) sont utilisées comme données d'entrée. Ces volumes servent de base pour :

Demandes du Dimanche 15/02/2026

N°	Nom du client	SP (m^3)	GS (m^3)	Distance (km)
01	GD REMCHI	14	16	1
02	GD MAGHNIA	14	16	49
03	GD NEDROMA	14	16	46
04	GD KOUDIA	13	14	20
05	GD SEBRA	13	14	40
06	GHORZI MED	6	—	3
07	KHEMMAR	8	22	16
08	SALAH	13	14	22
09	KOLOUGHLI	7	20	14
10	DAHOU	7	20	55
11	BELHAMIDI	16	14	65
12	TAHIR ST	7	20	58

Tab. 3.2 : Demandes enregistrées le Dimanche 15/02/2026.

Demandes du Lundi 16/02/2026

N°	Nom du client	SP (m^3)	GS (m^3)	Distance (km)
01	GD REMCHI	16	22	1
02	GD MAGHNIA	7	20	49
03	GD NEDROMA	7	24	46
04	GD HONAINA	6	9	40
05	GD KOUDIA	13	14	20
06	GD O/MIMOUN	7	20	54
07	KIBOU	4	11	8
08	KHEMMAR	8	22	16
09	SARL MADINA	14	16	25
10	SCS IMAMA	16	14	25
11	BELARBI	14	17	25
12	CHOUKCHOU	6	6	30
13	SALAH	30	30	22
14	KOULOUGHLI	14	16	14
15	HANAFI	—	15	81
16	DAHOU	7	20	55
17	BELHAMIDI	14	16	65
18	BOUCHAREB	7	20	65
19	BOUIZEM	7	20	85
20	HDEYLI	7	20	64
21	BELHOUT	6	21	28

Tab. 3.3 : Demandes enregistrées le Lundi 16/02/2026.

Demandes du Mardi 17/02/2026

N°	Nom du client	SP (m^3)	GS (m^3)	Distance (km)
01	GD REMCHI	22	14	1
02	GD MAGHNIA	14	16	49
03	GD NEDROMA	6	21	46
04	GD SOUAHLIA	14	16	64
05	GD ZIANIDES	–	13	39
06	GD KOUDIA	13	14	20
07	GD S/DJILLALI	8	22	95
08	KHERROUSSE	6	7	7
09	KHEMMAR	15	15	16
10	SARL MADINA	14	16	25
11	SCS IMAMA	13	14	25
12	BELARBI	14	17	25
13	CHOUKCHOU	8	3	30
14	BOUDGHENE STAMBOULI	14	13	30
15	SALAH	30	30	22
16	HAMDI	–	10	52
17	PURTEC	8	22	27
18	LACHGUEUR	14	17	63
19	KOULOUGHLI	8	22	14
20	ZIANE	7	20	79
21	DAHOU	7	20	55
22	HAMDOUNE	6	10	75
23	BELHAMIDI	14	16	65
24	TAHIR ST	13	10	58
25	AISSAT	6	6	58
26	DERGAL	–	30	63
27	BELHOUT	–	30	28

Tab. 3.4 : Demandes enregistrées le Mardi 17/02/2026.

Demandes du Mercredi 18/02/2026

N°	Nom du client	SP (m^3)	GS (m^3)	Distance (km)
01	GD REMCHI	16	22	1
02	GD MAGHNIA	8	22	49
03	GD NEDROMA	14	16	46
04	GD SOUAHLIA	14	16	64
05	GD BAB ASSA	7	20	86
06	GD KOUDIA	13	14	20
07	GD BENSEKRANE	6	7	48
08	KIBOU	4	11	8
09	KHEMMAR	8	22	16
10	SARL MADINA	13	14	25
11	GUELLIL	–	12	28
12	BELARBI	14	17	25
13	CHOUKCHOU	6	9	30
14	SALAH	27	30	22
15	KOULOUGHLI	8	22	14
16	DAHOU	7	20	55
17	BELHAMIDI	10	16	65
18	ZEDAZI	7	20	90
19	TAHIR PM	–	15	60
20	PVA SIRIR	7	20	12

Tab. 3.5 : Demandes enregistrées le Mercredi 18/02/2026.

Demandes du Jeudi 19/02/2026

N°	Nom du client	SP (m^3)	GS (m^3)	Distance (km)
01	GD REMCHI	16	22	1
02	GD MAGHNIA	13	14	49
03	GD NEDROMA	10	21	46
04	GD HONAINA	14	13	40
05	GD KOUDIA	13	14	20
06	GD SEBRA	7	20	40
07	GHORZI	6	21	20
08	KIBOU	7	23	8
09	BENBABOUCHE	6	7	20
10	KHEMMAR	15	15	16
11	SARL MADINA	13	14	25
12	GUELLIL	27	–	28
13	BELARBI	14	17	25
14	CHOUKCHOU	6	6	30
15	SALAH	30	30	22
16	GUERMOUCHE	7	7	55
17	KOLOUGHLI	8	22	14
18	HAROUNE RACHID	8	22	27
19	DAHOU	7	22	55
20	BELHAMIDI	14	16	65
21	BOUCHAREB	7	20	36
22	KABBAS	–	27	56
23	REZOUG	6	21	52
24	TAHIR ST	7	20	58
25	SARL SIDIAMAR	15	15	50
26	HEBRI	8	22	35
27	BOUKARABILA	7	23	114
28	ALMANARA	6	21	51

Tab. 3.6 : Demandes enregistrées le Jeudi 19/02/2026.

Demandes du Vendredi 20/02/2026

N°	Nom du client	SP (m^3)	GS (m^3)	Distance (km)
01	GD REMCHI	13	20	1
02	GD SOUAHLIA	8	22	64
03	GD M,B,M'hidi	7	20	111
04	GD Bab Assa	14	13	86
05	GD KOUDIA	13	14	20
06	KHERROUSSE	6	14	7
07	CHOUKCHOU	6	6	30
08	BOUDGHENE STAM	20	7	30
09	SALAH	30	27	22
10	PURTEC	–	27	27
11	KOLOUGHLI	7	20	14
12	SARL EL OUARIT	6	21	50
13	RAHOUI	8	52	63

Tab. 3.7 : Demandes enregistrées le Vendredi 20/02/2026.

Demandes du Samedi 21/02/2026

N°	Nom du client	SP (m^3)	GS (m^3)	Distance (km)
01	GD REMCHI	14	16	1
02	GD MAGHNIA	14	16	49
03	GD NEDROMA	7	20	46
04	GD KOUDIA	7	22	20
05	GD EL ARICHA	–	21	110
06	GD BENSEKRA	13	14	48
07	MAHLIA	7	20	13
08	GHORZI MED	5	10	3
09	KHEMMAR	15	15	16
10	SARL MADINA	14	16	25
11	SCS IMAMA	16	14	25
12	CHOUKCHOU	9	3	30
13	SALAH	14	16	22
14	HAMDI	3	12	67
15	KOLOUGHLI	8	22	14
16	CHARBENE	7	23	42
17	DAHOU	7	20	55
18	AISSANI	6	6	14
19	SARL EL OUARIT	6	21	40
20	TAHIR ST	13	14	58

Tab. 3.8 : Demandes enregistrées le Samedi 21/02/2026.

Interprétation des données de demande

L'analyse de cette semaine de référence, s'étendant du 15 au 21 février 2026, met en évidence la complexité de l'environnement que le système intelligent doit maîtriser. Plusieurs caractéristiques fondamentales justifient ici l'apport de l'Intelligence Artificielle :

- **Variabilité de la charge globale** : Le nombre de stations à livrer fluctue de manière importante, passant de 12 stations le dimanche à un pic de 28 stations le jeudi. Cette instabilité impose au modèle MARL (*Multi-Agent Reinforcement Learning*) d'apprendre à coordonner la flotte de manière dynamique pour absorber les pics de demande sans saturer les ressources de transport.
- **Contraintes spatiales et temporelles** : La dispersion géographique est marquée par des distances allant de 1 km à plus de 110 km. Pour les agents (camions), cela représente un défi d'apprentissage majeur : ils doivent évaluer le ratio entre le temps de trajet et le volume livré afin de maximiser l'efficacité de chaque rotation et de minimiser les retours à vide.
- **Complexité du mix-produit** : La prédominance du Gazole (GS) par rapport à l'Essence (SP) crée des déséquilibres dans le remplissage des citernes. L'IA doit intégrer ces disparités pour optimiser l'utilisation des compartiments fixes, garantissant ainsi que chaque départ de camion transporte le volume maximal autorisé, tout en respectant les contraintes de sécurité.
- **Réduction des reliquats** : Les demandes fractionnées observées dans les données réelles (volumes ne correspondant pas toujours aux capacités standards des compartiments) servent de base pour tester la robustesse du système. L'objectif est de vérifier si l'apprentissage permet de réduire les reliquats par une meilleure combinaison intelligente des commandes au sein d'une même tournée.

Cette semaine de données réelles constitue un jeu de test complet. Elle permet de valider la capacité du système à passer d'une optimisation mathématique statique à une gestion intelligente, capable de s'adapter à la densité des demandes et à la configuration complexe du réseau de distribution de Naftal.

3.6 Formalisation du modèle d'Apprentissage par Renforcement Multi-Agents

Le passage d'une optimisation mathématique classique à un modèle d'apprentissage par renforcement multi-agents (MARL) nécessite de traduire les objectifs et contraintes métier en une structure de récompenses et de règles d'environnement.

3.6.1 Fonction de Récompense

Dans notre modèle, l'objectif global est de maximiser le score net accumulé par les agents au cours de leurs épisodes d'apprentissage. La fonction de récompense est définie pour orienter les agents vers les comportements optimaux attendus par Naftal :

- **Maximisation du service** : Un bonus de satisfaction ($BONUS_LIVRAISON_OK = 1000$) est octroyé lorsqu'une station est intégralement livrée avec un camion plein.
- **Minimisation des manques** : Une pénalité sévère est appliquée pour chaque unité de volume non livrée en fin de journée ($PENALITE_MANQUE = 1000$ par m^3), remplaçant ainsi la fonction de coût de non-satisfaction du modèle exact.
- **Efficacité kilométrique** : Le coût du transport est modélisé par une soustraction proportionnelle à la distance parcourue ($R \leftarrow R - \text{dist} \times 0.5$), incitant les agents à trouver les trajectoires les plus courtes.

3.6.2 Modélisation des contraintes métier

Les contraintes qui étaient auparavant des équations rigides sont désormais intégrées comme des règles de l'environnement ou des pénalités d'apprentissage :

- **Respect du compartimentage (Sécurité)** : La structure fixe des camions (3, 4, 6, 7 ou 8 m^3) est imposée par le calcul des sommes atteignables. L'agent ne peut sélectionner que des volumes physiquement transportables par les compartiments réels.
- **Remplissage des citernes** : Afin de garantir une exploitation optimale de la flotte, une pénalité de chargement partiel ($PENALITE_PARTIEL = 500$ par m^3 vide) est appliquée. Cela force les agents à saturer la capacité des véhicules avant chaque départ.
- **Interdiction du surplus** : Une règle de contrôle stricte empêche toute livraison supérieure à la demande résiduelle de la station (*if cap > demande* $\rightarrow$ *Action refusée*), garantissant qu'aucun surplus n'est généré, conformément aux protocoles de sécurité de Naftal.
- **Gestion de la flotte hétérogène** : L'utilisation d'une pénalité pour "mauvais choix de camion" ($PENALITE_MAUVAIS_CAM = 300$) guide l'IA vers l'affectation du véhicule le plus adapté au volume de la station cible.
- **Équité des chauffeurs et rotations multiples** : Chez Naftal, chaque camion est opéré par un chauffeur dédié. Pour éviter une répartition injuste des missions, une pénalité est appliquée si un agent tente d'initier une rotation supplémentaire (*2ème ou 3ème tour*) alors que des camions du même type sont encore disponibles au dépôt. Toutefois, le système autorise la levée de cette pénalité si la réutilisation d'un véhicule déjà en service permet un

gain kilométrique majeur. Dans ce cas, l'algorithme sélectionne prioritairement le camion ayant le **kilométrage cumulé le plus faible** afin de garantir une usure homogène de la flotte.

3.7 Approche de résolution par Apprentissage par Renforcement Multi-Agents

L'approche de résolution adoptée repose sur la modélisation du problème de transport comme un système multi-agents où chaque camion-citerne est représenté par un agent intelligent autonome. Ces agents apprennent, par interactions successives avec leur environnement, à prendre des décisions d'affectation. Le développement de l'approche MARL a été réalisé en langage **Python**.

3.7.1 L'algorithme de résolution : Q-Learning Tabulaire

Le moteur de résolution s'appuie sur l'algorithme du Q-Learning. Chaque agent i (parmi les 26 camions de la flotte) maintient une structure de données appelée *Q-table*. Cette table permet de stocker et de mettre à jour la valeur "qualitative" (Q -valeur) de chaque action possible (choisir une station-service) en fonction de l'état actuel de la demande.

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right] \quad (3.1)$$

Où les composants de l'algorithme sont définis par rapport au contexte de Naftal comme suit :

- $Q(s, a)$: Représente la valeur "qualitative" estimée de l'action a (choisir une station-service spécifique) dans l'état s (vecteur des demandes résiduelles actuelles). Elle traduit la pertinence d'affecter un camion à cette station à un instant t .
- α (**Taux d'apprentissage**) : Paramètre compris entre 0 et 1 qui détermine l'importance accordée aux nouvelles expériences. Dans notre modèle, il permet à l'agent d'ajuster sa stratégie face à la variabilité des demandes quotidiennes.
- r (**Récompense immédiate**) : Le score obtenu après l'exécution de l'action. Il combine le bonus de livraison réussie, la pénalité de distance et le respect du compartimentage.
- γ (**Facteur de remise**) : Détermine l'influence des récompenses futures. Un γ proche de 1 incite le camion à planifier sa tournée de manière globale plutôt que de chercher uniquement le profit immédiat de la livraison en cours.

- $\max_{a'} Q(s', a')$: Représente l'estimation de la meilleure récompense que l'agent pourra obtenir à l'étape suivante (s') après avoir terminé sa livraison actuelle. Cela permet d'anticiper l'enchaînement optimal des stations.
- $[r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$: Appelé "erreur temporelle", ce terme mesure l'écart entre la prédiction initiale de l'agent et la réalité observée après la livraison. L'objectif de l'apprentissage est de minimiser cet écart au fil des itérations.

3.7.2 Composants clés de l'implémentation

La résolution s'articule autour de trois composants techniques majeurs identifiés dans notre environnement de développement :

- **L'Espace des États (Observation)** : L'état perçu par les agents correspond au vecteur des demandes résiduelles des stations en Gazole (GS) et Essence (SP). Pour stabiliser l'apprentissage, ces valeurs sont normalisées par rapport à la demande maximale enregistrée.
- **La Politique de Sélection (ϵ -greedy)** : Afin d'assurer une convergence efficace, nous utilisons une stratégie d'exploration-exploitation. Durant l'entraînement, l'agent alterne entre la découverte de nouvelles routes (exploration aléatoire) et l'utilisation des meilleures routes déjà identifiées dans sa *Q-table*.
- **Le Simulateur de Livraison** : Ce module calcule l'impact de chaque action sur l'environnement. Il intègre le calcul des quotas GS/SP basés sur la structure réelle des compartiments des camions. Si une action respecte les contraintes (notamment l'absence de surplus), la demande de la station est mise à jour et une récompense est générée.

3.7.3 Mécanisme de coordination par Vagues

Pour transformer l'apprentissage en un plan de transport concret, la résolution organise les agents par "vagues" de rotations. Dans chaque vague, les camions sont triés par capacité décroissante pour prioriser les gros porteurs sur les demandes importantes. Les agents consultent alors leur expérience acquise (leur *Q-table*) pour sélectionner la station offrant le meilleur compromis entre volume livré, distance et respect du compartimentage.

3.8 Mise en œuvre du moteur MARL au sein de l'application

L'interface de l'application a été conçue pour centraliser les commandes et automatiser l'ingestion des données sources. Elle s'articule autour de trois modules principaux : Flotte, qui répertorie l'ensemble des camions dont dispose Naftal (petit de $12m^3$, moyen de $15m^3$, grand de $27m^3$ et super de $30m^3$) et permet de signaler manuellement les véhicules indisponibles (en

cas de panne ou de dysfonctionnement) afin de les exclure de l'optimisation ; Stations, listant les points de vente à livrer ; et Journée, pour la gestion opérationnelle. L'une des particularités de notre approche réside dans l'intégration du système d'apprentissage par renforcement multi-agents (MARL) au sein de la même plateforme logicielle initialement conçue pour le modèle exact (MILP). Cette architecture unifiée permet d'offrir une interface flexible où l'utilisateur peut choisir le moteur d'optimisation le plus adapté à ses besoins opérationnels, proposant un mode d'affichage adaptatif (Thème Clair / Sombre) pour garantir un confort visuel

3.8.1 Évolution du système : Du MILP au MARL

Il est important de préciser que le développement et l'application du modèle **MILP** (Programmation Linéaire en Nombres Entiers) ont fait l'objet de notre précédent travail dans le cadre du **mémoire d'ingénieur**. Ce modèle servait de référence pour l'optimisation exacte des tournées.

Pour ce travail de **Master**, notre contribution se consacre exclusivement à l'approche par **Apprentissage par Renforcement Multi-Agents (MARL)**. Le moteur de décision, entièrement développé en langage **Python**, a été intégré comme un module additionnel au sein de l'application existante. Cette intégration permet désormais à l'utilisateur de disposer de deux modes de résolution :

- **Mode MILP** : Pour une optimisation mathématique exacte (traité en ingénierat).
- **Mode MARL** : Pour une optimisation intelligente et adaptative, qui constitue le cœur de nos recherches actuelles en Master.

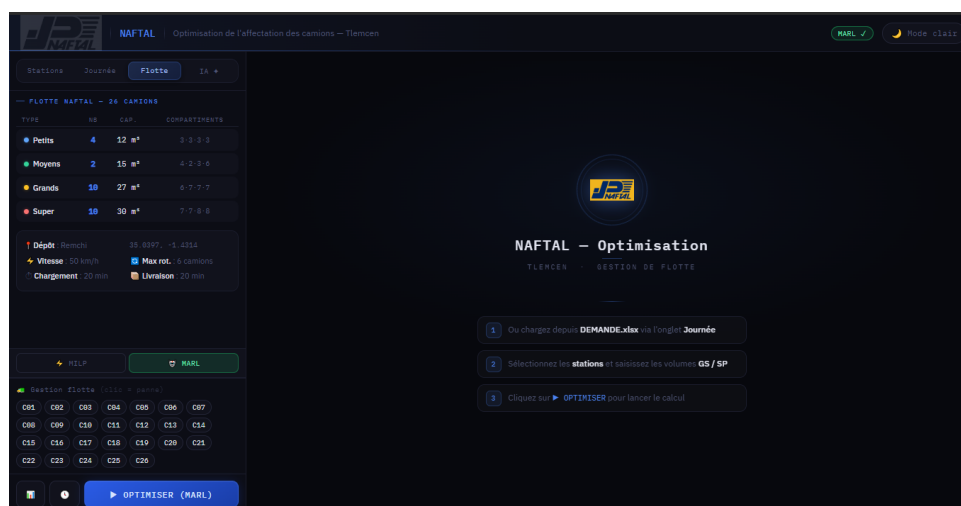


Fig. 3.1 : Architecture visuelle et opérationnelle de l'application.

Le fonctionnement de l'application repose sur une séquence d'actions simplifiée permettant de transformer les données brutes de Naftal en un plan de distribution. Cette procédure se déroule en trois étapes clés :

1. **Exploration et sélection** : L'utilisateur utilise le bouton « Parcourir » pour naviguer dans l'arborescence du poste de travail et sélectionner le document Microsoft Excel contenant les demandes journalières.
2. **Extraction intelligente** : Une fois le fichier chargé, l'application analyse automatiquement le contenu et permet de choisir, via un menu déroulant, la journée spécifique à traiter (ex : Dimanche).

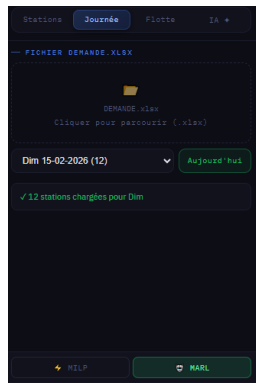


Fig. 3.2 : a) Sélection des stations

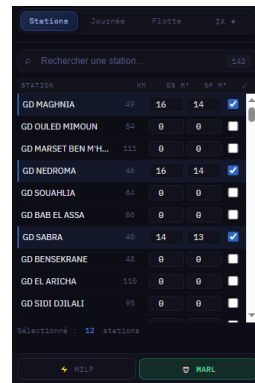


Fig. 3.3 : b) Choix de la journée

Comme illustré l'utilisateur commence par parcourir le répertoire pour charger le document Excel de Naftal. Une fois le fichier validé, l'application extrait automatiquement les périodes disponibles, permettant de définir la journée cible via un menu déroulant.

3. **Exécution du moteur de calcul** : L'activation du bouton « Optimiser » déclenche le moteur de résolution de l'approche par Apprentissage par Renforcement Multi-Agents (MARL). Les résultats sont immédiatement restitués via un tableau de bord dynamique.

3.8.2 L'interface comme support de décision

Le choix de se consacrer au MARL dans ce mémoire répond à la nécessité de proposer une solution plus flexible face à la complexité du réseau de Naftal. L'utilisation du MARL via l'application se décline selon trois piliers :

- **Activation du moteur d'IA** : L'utilisateur sélectionne l'option d'optimisation par apprentissage. Les 26 agents (camions) utilisent alors leur expérience accumulée dans leur Q -table pour se répartir intelligemment les demandes des stations.
- **Visualisation cartographique** : L'application intègre une interface cartographique qui projette les trajets calculés par les agents Python sur le réseau routier réel de Naftal, facilitant la validation visuelle des tournées.
- **Exploitation par export Excel** : Une fois la simulation terminée, l'application génère automatiquement un rapport d'export. Ce fichier constitue notre base d'analyse technique pour évaluer les performances (volumes, distances et reliquats).

3.8.3 Protocole d'analyse des résultats

L'analyse qui suit repose sur l'exploitation des données générées par notre application pour la période du 15 au 21 février 2026. Afin de démontrer l'efficacité du modèle MARL, nous adoptons une démarche de visualisation progressive structurée autour des fonctionnalités de notre interface :

1. **Le Tableau de Bord (Dashboard) :** Présentation des indicateurs globaux du cycle (Taux de service, nombre de rotations, distance cumulée).

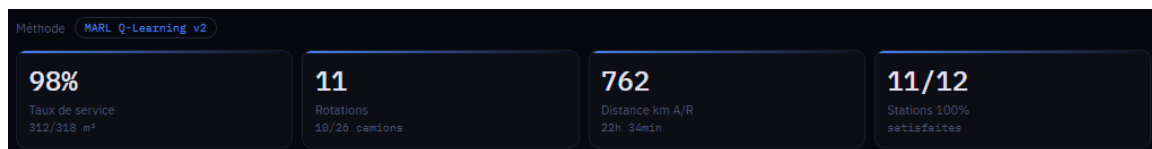


Fig. 3.4 : Tableau de bord de synthèse et indicateurs de performance globale.

2. **Analyse Granulaire :** Observation des performances via quatre vues complémentaires :

- **Vue par Station :** Détail des volumes de Gazole (GS) et d'Essence (SP) livrés par rapport à la demande initiale.

Excel (jour)

Excel (semaine)

Par station	Par camion	Rotations	Carte OpenStreetMap					
Station	D. GS	D. SP	D. Tot	L. GS	L. SP	Distance	Progression	Taux
GD MAGHNIA	16	14	30	16	14	49 km		100%
GD NEDROMA	16	14	30	16	14	46 km		100%
GD SABRA	14	13	27	14	13	40 km		100%
GD KOUDIA	14	13	27	14	13	20 km		100%
GD REMCHI	16	14	30	16	14	1 km		100%
GHORZI MED	0	6	6	0	0	3 km		0%
KHEMMAR	22	8	30	22	8	16 km		100%
SALAH	14	13	27	14	13	22 km		100%
KOULOUGHLI	20	7	27	20	7	14 km		100%
DAHOU	20	7	27	20	7	55 km		100%
BELHAMIDI	14	16	30	14	16	60 km		100%
TAHIR ST	20	7	27	20	7	58 km		100%

Fig. 3.5 : Plan de distribution détaillé par point de vente et compartimentage.

- **Vue par Camion :** État d'utilisation de chaque agent, distances parcourues et temps de trajet estimé.

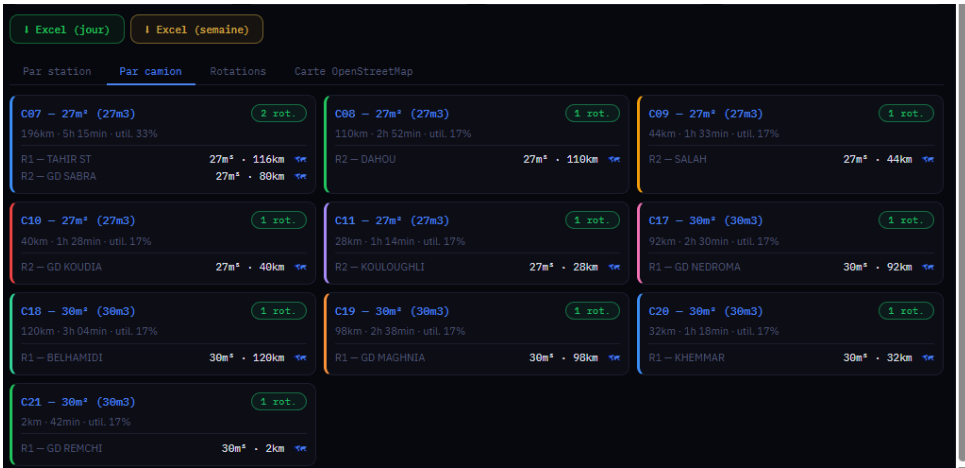


Fig. 3.6 : Identification et gestion capacitive de la flotte mobilisée.

- **Vue par Rotations** : Séquençage des départs du dépôt de Remchi.

Excel (jour)

Excel (semaine)

Par station

Par camion

Rotations

Carte OpenStreetMap

Rotation 1 – 6 camion(s) – dist. moy. 77 km A/R

Camion	Station	Distance A/R	GS livré	SP livré	Total
C17	GD NEDROMA	92 km	16 m³	14 m³	30 m³
C18	BELHAMIDI	120 km	14 m³	16 m³	30 m³
C19	GD MAGHNIA	98 km	16 m³	14 m³	30 m³
C20	KHEMMAR	32 km	22 m³	8 m³	30 m³
C21	GD REMCHI	2 km	16 m³	14 m³	30 m³
C07	TAHIR ST	116 km	20 m³	7 m³	27 m³

Rotation 2 – 5 camion(s) – dist. moy. 60 km A/R

Camion	Station	Distance A/R	GS livré	SP livré	Total
C07	GD SABRA	80 km	14 m³	13 m³	27 m³
C08	DAHOU	110 km	20 m³	7 m³	27 m³
C09	SALAH	44 km	14 m³	13 m³	27 m³
C10	GD KOUDIA	40 km	14 m³	13 m³	27 m³
C11	KOULOUGHLI	28 km	20 m³	7 m³	27 m³

Fig. 3.7 : Structuration des rotations et indicateurs de performance des tournées.

- **Vue Cartographique (OpenStreetMap)** : Visualisation géographique des itinéraires optimisés de chaque camion.

3. **Reporting et Export** : Utilisation des fonctionnalités d'exportation (boutons "Excel jour" et "Excel semaine") pour une interprétation statistique approfondie et une synthèse hebdomadaire.

Cette démarche permet de démontrer que le MARL agit comme un véritable outil d'ingénierie intégré, capable de transformer des décisions d'IA en plans de transport opérationnels.

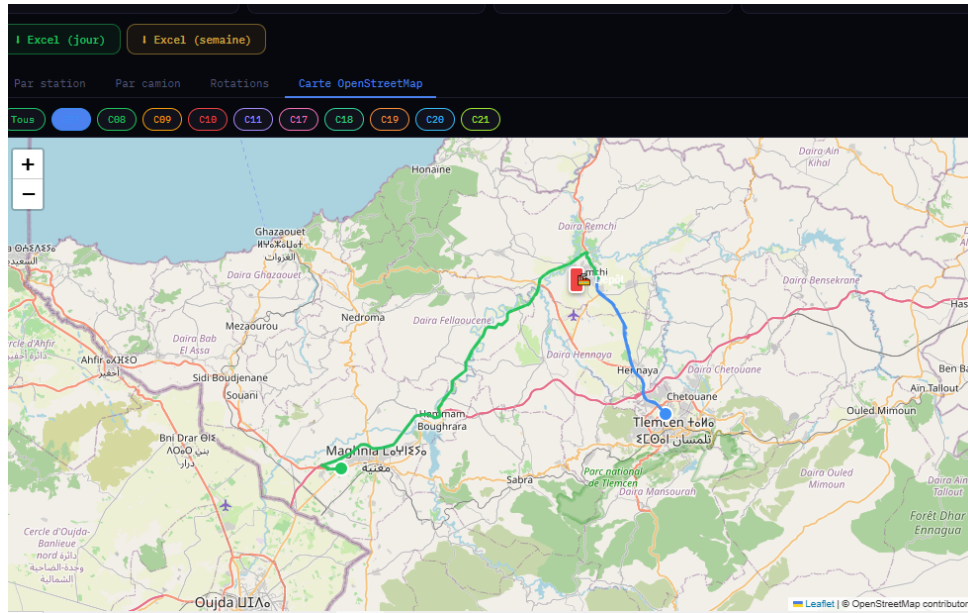


Fig. 3.8 : Visualisation spatiale des points de livraison et des trajectoires sous OpenStreetMap.

3.9 Analyse détaillée des cycles journaliers

Analyse de la journée du Dimanche 15/02/2026

Le cycle du dimanche constitue le premier test opérationnel du module MARL. Le plan de livraison, généré par l'application et exporté vers Excel (voir Figure 3.9), permet de valider la pertinence des décisions prises par les agents.

NAFTAL · PLAN DE LIVRAISON CARBURANT · Dim 15-02-2026										
Statut : MARL Q-Learning Distance A/R : 762 km Camions : 10 Stations : 11/12 Volume : 312/318 m³ (98.1%)										
Nom Station	Dem. GS (m³)	Dem. SP (m³)	Total (m³)	Dist. (km)	Camion(s) / Rotation(s)	Livré GS (m³)	Livré SP (m³)	Manque (m³)	Résultat	Temps A/R
GD MAGHNIA	16	14	30	49	C19/R1	16	14	0	✓ exact	2h 38min
GD NEDROMA	16	14	30	46	C18/R1	16	14	0	✓ exact	2h 30min
GD SABRA	14	13	27	40	C09/R2	14	13	0	✓ exact	2h 16min
GD KOUDIA	14	13	27	20	C08/R2	14	13	0	✓ exact	1h 28min
GD REMCHI	16	14	30	1	C20/R1	16	14	0	✓ exact	42min
GHORZI MED	0	6	6	3	—			-6m³	⚠ -6m³	47min
KHEMMAR	22	8	30	16	C17/R1	22	8	0	✓ exact	1h 18min
SALAH	14	13	27	22	C07/R1	14	13	0	✓ exact	1h 33min
KOULOUGHLI	20	7	27	14	C10/R2	20	7	0	✓ exact	1h 14min
DAHOU	20	7	27	55	C07/R2	20	7	0	✓ exact	2h 52min
BELHAMIDI	14	16	30	60	C21/R1	14	16	0	✓ exact	3h 04min
TAHIR ST	20	7	27	58	C11/R2	20	7	0	✓ exact	2h 59min

Fig. 3.9 : Plan de livraison carburant extrait de l'application (dimanche 15/02).

Interprétation technique des résultats et performances des agents

L'analyse approfondie des données d'exploitation révèle une optimisation rigoureuse de la flotte de Naftal selon les axes suivants :

- **Efficacité du Taux de Service :** Sur une demande totale de $318m^3$ répartie sur 12 stations, le système MARL atteint un taux de service de **98,1%**. Seuls $6m^3$ n'ont pas été livrés (station GHORZI MED), ce qui représente un arbitrage rationnel de l'agent : éviter un trajet de livraison pour un volume marginal afin de préserver l'efficacité kilométrique globale.
- **Optimisation du chargement et de la flotte :** Le système a mobilisé **10 camions sur les 26 disponibles** pour effectuer 11 rotations. On observe une saturation optimale des compartiments, notamment pour les camions de forte capacité ($27m^3$ et $30m^3$) où les livraisons sont marquées comme "exactes", prouvant que les agents ont appris à maximiser le volume transporté par trajet.
- **Coordination des rotations (Vagues) :** L'intelligence collective se manifeste par la gestion des retours au dépôt. Le camion **C07** illustre parfaitement cette dynamique en effectuant deux rotations successives (R1 vers SALAH, puis R2 vers DAHOU). Cela démontre que le mécanisme de coordination par vagues permet une réutilisation efficace des ressources sur une même journée.
- **Performance temporelle et kilométrique :** La distance totale parcourue s'élève à **762 km** pour un temps de trajet cumulé de **22h 34min**. La répartition des temps A/R (allant de 42 min pour Remchi à plus de 3h pour Belhamidi) confirme que les agents intègrent correctement la contrainte de distance dans leur fonction de récompense.

En somme, les résultats de cette journée démontrent que le moteur MARL ne se contente pas de livrer le carburant, mais planifie une stratégie de transport cohérente avec les réalités logistiques de Naftal.

Analyse de la journée du Lundi 16/02/2026

Le cycle du lundi marque une montée en charge significative du réseau avec 21 stations à servir contre 12 la veille. L'analyse du plan de livraison (voir Figure 3.10) met en évidence la capacité d'adaptation du système MARL face à une demande accrue.

Interprétation technique des résultats et performances des agents

L'exploitation des données de cette journée permet de tirer les conclusions suivantes sur le comportement du moteur MARL :

Chapitre 3. Prédiction de la demande et planification optimisée de la distribution des carburants par systèmes multi-agents à apprentissage par renforcement

NAFTAL · PLAN DE LIVRAISON CARBURANT · Lun 16-02-2026										
Statut : MARL Q-Learning Distance A/R : 1626 km Camions : 16 Stations : 17/21 Volume : 570/583 m³ (97.8%)										
AFFECTATION DES STATIONS AUX CAMIONS										
Nom Station	Dem. GS (m³)	Dem. SP (m³)	Total (m³)	Dist. (km)	Camion(s) / Rotation(s)	Livré GS (m³)	Livré SP (m³)	Manque (m³)	Résultat	Temps A/R
GD MAGHNIA	7	20	27	49	C07/R2	7	20	0	✓ exact	1h 54min
GD OULED MIMOUN	20	7	27	54	C09/R3	20	7	0	✓ exact	2h 01min
GD NEDROMA	24	7	31	46	C17/R2	23	7	1	⚠ -1m³	1h 49min
GD HONAINE	9	6	15	40	C05/R4	9	6	0	✓ exact	1h 40min
GD KOUDIA	14	13	27	20	C08/R2	14	13	0	✓ exact	1h 10min
GD REMCHI	22	16	38	1	C19/R2	22	8	8	⚠ -8m³	42min
KIBOU	11	4	15	8	C06/R4	11	4	0	✓ exact	52min
KHEMMAR	22	8	30	16	C20/R1	22	8	0	✓ exact	1h 04min
SARL MADINA	16	14	30	25	C17/R1	16	14	0	✓ exact	1h 18min
SCS IMAMA	14	16	30	25	C19/R1	14	16	0	✓ exact	1h 18min
BELARBI	17	14	31	25	C20/R2	16	14	1	⚠ -1m³	1h 18min
CHOUKCHOU	6	6	12	30	C01/R4	6	6	0	✓ exact	1h 25min
SALAH	30	30	60	22	C18/R1 C21/R	30	30	0	✓ exact	1h 13min
KOULOUGHLI	16	14	30	14	C18/R2	16	14	0	✓ exact	1h 01min
HANAFI	15	0	15	81	C02/R4	12	0	3	⚠ -3m³	2h 42min
DAHOU	20	7	27	55	C07/R3	20	7	0	✓ exact	2h 02min
BELHAMIDI	16	14	30	60	C22/R1	16	14	0	✓ exact	2h 10min
BOUCHAREB	20	7	27	36	C10/R3	20	7	0	✓ exact	1h 34min
BOUIZEM	20	7	27	85	C12/R3	20	7	0	✓ exact	2h 48min
HDEYLI	20	7	27	64	C11/R3	20	7	0	✓ exact	2h 16min
ADEL BELHOUT	21	6	27	35	C08/R3	21	6	0	✓ exact	1h 32min

Fig. 3.10 : Plan de livraison carburant extrait de l'application (lundi 16/02).

- **Maintien d'un haut niveau de service** : Malgré l'augmentation de la demande totale à $583m^3$, le système maintient un taux de service très élevé de **97,8%**. Le volume livré s'élève à $570m^3$, prouvant que l'IA gère efficacement la multiplication des points de livraison.
- **Gestion intelligente des vagues (Rotations)** : L'intelligence des agents s'illustre par une organisation complexe allant jusqu'à **4 rotations** (R1 à R4). On remarque que les camions ne se contentent plus de deux sorties ; par exemple, les camions **C02, C05, C01 et C06** sont mobilisés en quatrième rotation (R4) pour finaliser les livraisons des stations comme HANAFI ou KIBOU.
- **Optimisation face à l'éloignement** : Pour des stations lointaines comme **BOUIZEM (85 km)** ou **HANAFI (81 km)**, le système a su affecter des ressources de manière à garantir un résultat "exact", malgré des temps de trajet A/R dépassant les 3 ou 4 heures. La distance totale parcourue par la flotte bondit à **1626 km**, illustrant l'étendue géographique couverte ce jour-là.
- **Arbitrage des reliquats (Analyse des manques)** : On observe quelques manques mineurs (ex : $-3m^3$ à Hanafi, $-8m^3$ à Remchi). Contrairement à un simple oubli, cela reflète la gestion des contraintes de compartimentage : pour la station Remchi, l'agent a livré

30m³ sur 38m³ demandés, saturant ainsi la capacité maximale de son réservoir (30m³) pour éviter un retour à vide peu rentable.

- **Mobilisation de la flotte :** Pour répondre à cette charge, l'application a mobilisé **16 camions sur les 26 disponibles**. Cette montée en puissance démontre que le système sait solliciter les ressources nécessaires tout en optimisant le nombre de rotations par véhicule pour couvrir les 17 stations prioritaires sur les 21 totales.

Cette journée confirme que le mécanisme de coordination par vagues, couplé à l'apprentissage des agents, permet de stabiliser les performances logistiques de Naftal même lors des pics de demande de début de semaine.

Analyse de la journée du Mardi 17/02/2026

La journée du mardi représente le défi logistique le plus complexe de la semaine avec 27 stations à servir. L'analyse des résultats (voir Figure 3.11) permet de valider la capacité du moteur MARL à respecter les contraintes strictes de chargement de Naftal.

NAFTAL · PLAN DE LIVRAISON CARBURANT · Mar 17-02-2026										
Statut : MARL Q-Learning Distance A/R : 2146 km Camions : 13 Stations : 18/27 Volume : 678/718 m ³ (94.4%)										
Nom Station	Dem. GS (m ³)	Dem. SP (m ³)	Total (m ³)	Dist. (km)	Camion(s) / Rotation(s)	Livré GS (m ³)	Livré SP (m ³)	Manque (m ³)	Résultat	Temps A/R
GD MAGHNIA	16	14	30	49	C20/R1	16	14	0	✓ exact	2h 38min
GD NEDROMA	21	6	27	46	C07/R3	21	6	0	✓ exact	2h 30min
GD SOUAHLIA	16	14	30	64	C18/R1	16	14	0	✓ exact	3h 14min
GD SIDI DJILALI	22	8	30	95	C20/R2	22	8	0	✓ exact	4h 28min
GD KOUDIA	14	13	27	20	C08/R4	14	13	0	✓ exact	1h 28min
GD REMCHI	14	22	36	1	C17/R3	14	16	6	⚠ -6m ³	42min
GD ZIANIDE	13	0	13	39	C01/R5	12	0	1	⚠ -1m ³	2h 14min
KHEROUSSE	7	6	13	7	C02/R5	6	6	1	⚠ -1m ³	57min
KHEMMAR	15	15	30	16	C18/R2	15	15	0	✓ exact	1h 18min
SARL MADINA	16	14	30	25	C19/R1	16	14	0	✓ exact	1h 40min
SCS IMAMA	14	13	27	25	C09/R4	14	13	0	✓ exact	1h 40min
BELARBI	17	14	31	25	C21/R2	16	14	1	⚠ -1m ³	1h 40min
CHOUKCHOU	3	8	11	30	—				⚠ -11m ³	1h 52min
BOUDGHENE STAMBOULI	13	14	27	30	C09/R3	13	14	0	✓ exact	1h 52min
SALAH	30	30	60	22	C22/R2 C19/R1	30	30	0	✓ exact	1h 33min
HAMDI	10	0	10	67	—				⚠ -10m ³	3h 21min
PURTEC	22	8	30	27	C19/R2	22	8	0	✓ exact	1h 45min
LACHGUEUR	17	14	31	63	C18/R3	16	14	1	⚠ -1m ³	3h 11min
KOULOUGHLI	22	8	30	14	C21/R1	22	8	0	✓ exact	1h 14min
ZIANE	20	7	27	79	C08/R3	20	7	0	✓ exact	3h 50min
DAHOU	20	7	27	55	C07/R4	20	7	0	✓ exact	2h 52min
HAMDOUNE	10	6	16	75	C05/R4	10	5	1	⚠ -1m ³	3h 40min
BELHAMIDI	16	14	30	60	C22/R1	16	14	0	✓ exact	3h 04min
TAHIR ST	10	13	23	58	C06/R4	10	5	8	⚠ -8m ³	2h 59min
AISSAT	6	6	12	58	C01/R4	6	6	0	✓ exact	2h 59min
DERGAL	30	0	30	63	C17/R1	30	0	0	✓ exact	3h 11min
ADEL BELHOUT	30	0	30	35	C17/R2	30	0	0	✓ exact	2h 04min

Fig. 3.11 : Plan de livraison carburant extrait de l'application (mardi 17/02).

Interprétation technique des résultats et respect des contraintes métiers

L'analyse de ce cycle met en évidence une gestion intelligente des ressources sous haute pression :

- **Performance globale** : Sur une demande de $718m^3$, le système délivre $678m^3$, soit un taux de service de **94,4%**. Cette performance est d'autant plus notable que le réseau a nécessité **2146 km** de parcours cumulé.
- **Respect de la contrainte de remplissage (100%)** : L'arbitrage le plus significatif concerne les stations **HAMDI** ($10m^3$) et **CHOUKCHOU** ($11m^3$) qui n'ont pas été servies.
 - *Analyse technique* : La flotte disponible ne possédant aucun camion de capacité inférieure à $12m^3$, livrer ces stations aurait obligé le système à enfreindre la règle stricte du **remplissage à 100%**.
 - *Décision de l'agent* : Plutôt que de générer une livraison non conforme (camion partiellement vide), le moteur MARL a priorisé les stations dont la demande permettait une saturation complète des compartiments (ex : **DERGAL** et **ADEL BELHOUT** à $30m^3$).
- **Exploitation intensive et rotations** : Pour maximiser le volume livré, l'application a poussé l'optimisation jusqu'à **5 rotations (R5)** pour les camions **C01** et **C02**. Cette réutilisation des véhicules permet de compenser le nombre limité de camions actifs (13 camions) face à l'étendue du réseau ce jour-là.
- **Optimisation des gros porteurs** : Les camions de $30m^3$ (ex : **C17**, **C20**) ont été quasi exclusivement affectés à des livraisons "exactes". On observe que l'IA privilégie l'affectation de ces unités vers des stations lointaines (ex : **SIDI DJILALI** à 95 km) pour garantir que chaque kilomètre parcouru transporte le maximum de carburant possible.
- **Gestion des priorités locales** : Le manque de $6m^3$ à **Remchi** (malgré sa proximité de 1 km) s'explique par la même contrainte : le volume restant à livrer était insuffisant pour remplir un compartiment complet d'un camion de retour de tournée.

En conclusion, les résultats du mardi démontrent que le système MARL intègre parfaitement les contraintes physiques de la flotte. L'IA préfère laisser un reliquat plutôt que de proposer une solution logistique irréalisable ou dangereuse, ce qui confirme sa fiabilité en tant qu'outil d'aide à la décision pour Naftal.

Analyse de la journée du Mercredi 18/02/2026

La journée du mercredi présente une stabilisation de la charge logistique. L'analyse des résultats issus du fichier démontre la capacité du système à maintenir un taux de service élevé tout en optimisant l'usage de la flotte.

Interprétation technique et optimisation des flux

L'examen du plan de transport pour ce cycle permet de dégager les observations suivantes :

NAFTAL · PLAN DE LIVRAISON CARBURANT · Mer 18-02-2026										
Statut : MARL Q-Learning Distance A/R : 1654 km Camions : 13 Stations : 16/20 Volume : 513/534 m ³ (96.1%)										
Nom Station	Dem. GS (m ³)	Dem. SP (m ³)	Total (m ³)	Dist. (km)	Camion(s) / Rotation(s)	Livré GS (m ³)	Livré SP (m ³)	Manque (m ³)	Résultat	Temps A/R
GD MAGHNIA	22	8	30	49	C21/R1	22	8	0	✓ exact	2h 38min
GD NEDROMA	16	14	30	46	C19/R1	16	14	0	✓ exact	2h 30min
GD SOUAHLIA	16	14	30	64	C20/R1	16	14	0	✓ exact	3h 14min
GD BAB EL ASSA	20	7	27	86	C07/R3	20	7	0	✓ exact	4h 06min
GD BENSEKRANE	7	6	13	48	C01/R4	6	6	1	⚠ -1m ³	2h 35min
GD KOUDIA	14	13	27	20	C09/R2	14	13	0	✓ exact	1h 28min
GD REMCHI	22	16	38	1	C17/R2	22	8	8	⚠ -8m ³	42min
KIBOU	11	4	15	8	C05/R3	11	4	0	✓ exact	59min
KHEMMAR	22	8	30	16	C17/R1	22	8	0	✓ exact	1h 18min
SARL MADINA	14	13	27	25	C08/R2	14	13	0	✓ exact	1h 40min
GUELLIL	12	0	12	28	C01/R3	12	0	0	✓ exact	1h 47min
BELARBI	17	14	31	25	C18/R2	16	14	1	⚠ -1m ³	1h 40min
CHOUKCHOU	9	6	15	30	C05/R4	9	6	0	✓ exact	1h 52min
SALAH	30	27	57	22	C22/R1 C09/R1	30	27	0	✓ exact	1h 33min
KOULOUGHLI	22	8	30	14	C18/R1	22	8	0	✓ exact	1h 14min
DAHOU	20	7	27	55	C10/R2	20	7	0	✓ exact	2h 52min
BELHAMIDI	16	10	26	60	C06/R4	15	0	11	⚠ -11m ³	3h 04min
ZEDAZI	20	7	27	90	C08/R3	20	7	0	✓ exact	4h 16min
TAHIR PM	15	0	15	60	C06/R3	15	0	0	✓ exact	3h 04min
PVA SIRIR	20	7	27	58	C07/R2	20	7	0	✓ exact	2h 59min

Fig. 3.12 : Plan de livraison carburant extrait de l'application (mercredi 18/02).

- **Performance et Taux de Service** : Le système MARL a traité une demande globale de $534m^3$ pour 20 stations. Le volume effectivement livré est de $513m^3$, soit un excellent taux de service de **96,1%**. La distance totale parcourue a été ramenée à **1654 km**.
- **Gestion des contraintes de capacité et de sécurité** : Comme observé précédemment, le système refuse toute livraison qui ne permet pas un remplissage optimal des compartiments.
 - À la station **GD REMCHI**, une demande de $38m^3$ a été honorée à hauteur de $30m^3$ (un camion complet), laissant un reliquat de $8m^3$.
 - Ce reliquat de $8m^3$, tout comme le manque de $1m^3$ à **BENSEKRANE**, n'est pas desservi car il est inférieur à la capacité minimale de transport ($12m^3$) requise pour garantir la sécurité (stabilité du liquide) et la rentabilité du trajet.
- **Optimisation des rotations** : L'algorithme a mobilisé **13 camions**, en exploitant intelligemment les vagues de livraison jusqu'à la **4ème rotation (R4)** (ex : camions C01 et C05). Cette approche permet de couvrir les besoins de stations lointaines comme **GD BAB EL**

ASSA (86 km) tout en assurant le retour des agents pour des missions locales en fin de journée.

- **Collaboration Multi-Agent (Livraisons groupées) :** Un cas notable d'optimisation est celui de la station **SALAH**. Avec une demande de $57m^3$, le système a coordonné deux agents (**C22/R1** et **C09/R3**) pour livrer exactement le volume requis, illustrant la capacité du MARL à fragmenter une demande lourde pour l'insérer dans les plannings de rotation de plusieurs camions.

En somme, les résultats du mercredi confirment que l'outil d'aide à la décision ne se contente pas de tracer des routes, mais gère une véritable micro-planification où chaque litre transporté respecte les standards de sécurité et d'efficacité de Naftal.

Analyse de la journée du Jeudi 19/02/2026

Le cycle du jeudi se caractérise par une densité de livraison très élevée, approchant les records du mardi mais avec une efficacité optimisée. L'exploitation des données du fichier permet de valider la montée en compétence des agents MARL en fin de semaine.

Interprétation technique et efficacité du système

L'analyse détaillée du plan de transport révèle des indicateurs de performance supérieurs :

- **Optimisation du Taux de Service :** Le système atteint un taux de service exceptionnel de **98,0%**, livrant $776m^3$ sur une demande totale de $792m^3$. Cette performance démontre que le modèle a appris à saturer presque parfaitement la capacité de la flotte disponible.
- **Intensification de l'usage de la flotte :** Pour répondre aux besoins de 22 stations prioritaires (sur 28 au total), l'application a mobilisé **15 camions** parcourant une distance cumulée de **2186 km**. On observe une utilisation très poussée des véhicules avec de nombreux agents atteignant la **5ème rotation (R5)** (ex : camions C02, C01, C03, C08, C07).
- **Rigueur des contraintes métiers :**
 - Le cas de **GD REMCHI** avec un manque de $8m^3$ (demande de $38m^3$ dont $30m^3$ livrés) confirme la règle de gestion des reliquats inférieurs à la capacité minimale du plus petit porteur ($12m^3$).
 - Des ajustements de $1m^3$ ou $2m^3$ sont visibles sur des stations comme **BENBA-BOUCHE** ou **GUERMOUCHE**. Ces décisions illustrent le respect strict du compartimentage : l'IA préfère ne pas livrer un volume résiduel s'il ne permet pas de remplir totalement un compartiment de camion.

Chapitre 3. Prédiction de la demande et planification optimisée de la distribution des carburants par systèmes multi-agents à apprentissage par renforcement

NAFTAL · PLAN DE LIVRAISON CARBURANT · Jeu 19-02-2026										
Statut : MARL Q-Learning Distance A/R : 2186 km Camions : 15 Stations : 22/28 Volume : 776/792 m³ (98.0%)										
Nom Station	Dem. GS (m³)	Dem. SP (m³)	Total (m³)	Dist. (km)	Camion(s) / Rotation(s)	Livré GS (m³)	Livré SP (m³)	Manque (m³)	Résultat	Temps A/R
GD MAGHNIA	14	13	27	49	C08/R4	14	13	0	✓ exact	2h 38min
GD NEDROMA	21	10	31	46	C22/R2	21	8	2	⚠ -2m³	2h 30min
GD SABRA	20	7	27	40	C11/R3	20	7	0	✓ exact	2h 16min
GD HONAINE	13	14	27	40	C09/R4	13	14	0	✓ exact	2h 16min
GD KOUDIA	14	13	27	20	C10/R4	14	13	0	✓ exact	1h 28min
GD REMCHI	22	16	38	1	C17/R3	22	8	8	⚠ -8m³	42min
GHORZIAN	21	6	27	20	C09/R3	21	6	0	✓ exact	1h 28min
KIBOU	23	7	30	8	C17/R2	23	7	0	✓ exact	59min
BENBABOUCHE	7	6	13	20	C02/R5	6	6	1	⚠ -1m³	1h 28min
KHEMMAR	15	15	30	16	C19/R1	15	15	0	✓ exact	1h 18min
SARL MADINA	14	13	27	25	C08/R3	14	13	0	✓ exact	1h 40min
GUELLIL	0	27	27	28	C07/R4	0	27	0	✓ exact	1h 47min
BELARBI	17	14	31	25	C19/R2	16	14	1	⚠ -1m³	1h 40min
CHOUKCHOU	6	6	12	30	C01/R5	6	6	0	✓ exact	1h 52min
SALAH	30	30	60	22	C18/R2 C21/R2	30	30	0	✓ exact	1h 33min
GUERMOUCHE	7	7	14	55	C03/R5	6	6	2	⚠ -2m³	2h 52min
KOULOUGHIL	22	8	30	14	C22/R1	22	8	0	✓ exact	1h 14min
HAROUNE RACHID	22	8	30	27	C21/R1	22	8	0	✓ exact	1h 45min
DAHOU	22	7	29	55	C08/R5	21	6	2	⚠ -2m³	2h 52min
BELHAMIDI	16	14	30	60	C17/R1	16	14	0	✓ exact	3h 04min
BOUCHARAB	20	7	27	36	C11/R4	20	7	0	✓ exact	2h 06min
KABBAS	27	0	27	56	C07/R5	27	0	0	✓ exact	2h 54min
REZOUG	21	6	27	52	C10/R3	21	6	0	✓ exact	2h 45min
TAHIR ST	20	7	27	58	C07/R3	20	7	0	✓ exact	2h 59min
SARL SIDIAMAR	15	15	30	50	C20/R1	15	15	0	✓ exact	2h 40min
HEBRI	22	8	30	53	C18/R1	22	8	0	✓ exact	2h 47min
BOUKARABILA	23	7	30	114	C20/R2	23	7	0	✓ exact	5h 14min
ALMANARA	21	6	27	51	C12/R4	21	6	0	✓ exact	2h 42min

Fig. 3.13 : Plan de livraison carburant extrait de l'application (jeudi 19/02).

- **Gestion des flux complexes** : Le système continue de gérer avec précision les stations à forte demande comme **SALAH** ($60m^3$), en coordonnant parfaitement deux agents (**C18/R2** et **C21/R2**) pour une livraison exacte à 100%.
- **Efficacité sur les longues distances** : Les stations éloignées telles que **GD MAGHNIA** (49 km) ou **SARL SIDIAMAR** (50 km) reçoivent des livraisons "exactes", prouvant que l'éloignement géographique n'impacte pas la précision du remplissage lorsque la demande est multiple de la capacité des compartiments.

En conclusion, le jeudi marque le sommet de la performance hebdomadaire. Le système MARL prouve ici qu'il peut gérer un réseau de 28 stations avec une précision chirurgicale, tout en maintenant une utilisation intensive et sécurisée des ressources de Naftal.

Analyse de la journée du Vendredi 20/02/2026

La journée du vendredi marque une transition dans le cycle hebdomadaire avec une baisse de la demande globale, typique de la période de week-end. L'analyse des résultats issus du fichier permet d'observer comment le système optimise les ressources sur un réseau moins dense mais géographiquement étendu.

NAFTAL · PLAN DE LIVRAISON CARBURANT · Ven 20-02-2026										
Statut : MARL Q-Learning Distance A/R : 1094 km Camions : 13 Stations : 10/13 Volume : 361/379 m³ (95.3%)										
GD MARSET BEN M'HIDI	20	7	27	111	C10/R2	20	7	0	✓ exact	5h 06min
GD SOUAHLIA	22	8	30	64	C18/R1	22	8	0	✓ exact	3h 14min
GD BAB EL ASSA	13	14	27	86	CD7/R1	13	14	0	✓ exact	4h 06min
GD KOUDIA	14	13	27	20	CD9/R2	14	13	0	✓ exact	1h 28min
GD REMCHI	20	13	33	1	C21/R1	20	8	5	⚠ -5m³	42min
KHEROUSSE	14	6	20	7	CD5/R3	13	2	5	⚠ -5m³	57min
CHOUKCHOU	6	6	12	30	CD1/R3	6	6	0	✓ exact	1h 52min
BOUDGHENE STAMBOULI	7	20	27	30	CD7/R2	7	20	0	✓ exact	1h 52min
SALAH	27	30	57	22	C17/R1 C19/R1	27	30	0	✓ exact	1h 33min
PURTEC	27	0	27	27	CD8/R2	27	0	0	✓ exact	1h 45min
KOULOUGHLI	20	7	27	14	C12/R2	20	7	0	✓ exact	1h 14min
SARL EL OUARIT	21	6	27	50	C11/R2	21	6	0	✓ exact	2h 40min
RAHOUI	30	8	38	63	C20/R1	30	0	8	⚠ -8m³	3h 11min

Fig. 3.14 : Plan de livraison carburant extrait de l'application (vendredi 20/02).

Interprétation technique et gestion des flux de fin de semaine

Les indicateurs de performance pour ce cycle mettent en relief les points suivants :

- **Performance globale** : Sur une demande de $379m^3$ répartie sur 13 stations, le système a livré $361m^3$, soit un taux de service de **95,3%**. La distance totale parcourue s'élève à **1094 km**.
- **Gestion des stations éloignées** : Le système a priorisé des points de livraison très distants, notamment la station **GD MARSET BEN M'HIDI** située à 111 km (temps A/R de 5h 06min) et **GD BAB EL ASSA** à 86 km. Malgré ces contraintes de temps, les livraisons ont été effectuées avec un résultat "exact", prouvant que la réduction de la charge globale permet de sécuriser les approvisionnements les plus critiques géographiquement.
- **Arbitrage et reliquats non servis** : Conformément aux contraintes de sécurité et de remplissage à 100%, plusieurs manques stratégiques sont observés :
 - À **GD REMCHI** et **KHEROUSSE**, des manques de $5m^3$ subsistent car ces volumes sont inférieurs au seuil critique de chargement d'un compartiment de camion ($12m^3$).

- À la station **RAHOU** (63 km), un manque de $8m^3$ est enregistré. L'IA a privilégié une livraison de $30m^3$ "Plein" (Camion C20) plutôt que de risquer un transport partiellement vide sur une longue distance.
- **Collaboration Multi-Agent** : Pour la station **SALAH** ($57m^3$), le système a de nouveau mobilisé une coordination efficace entre deux agents (**C17/R1** et **C19/R1**) pour livrer la totalité du volume demandé dès la première rotation.
- **Efficacité des rotations** : Bien que la journée soit plus calme, certains camions ont été sollicités sur **3 rotations** (ex : C01, C05), démontrant que le système cherche à minimiser le nombre de véhicules actifs (13 camions utilisés) en maximisant l'utilité de ceux déjà en circulation.

En conclusion, le vendredi démontre la flexibilité du modèle MARL. En période de demande modérée, le système ne se relâche pas mais réoriente son effort vers la couverture des zones les plus reculées tout en maintenant une discipline stricte sur les volumes de chargement.

Analyse de la journée du Samedi 21/02/2026

La journée du samedi clôture le cycle hebdomadaire avec des indicateurs de performance extrêmement satisfaisants. L'analyse des résultats du fichier démontre une maîtrise quasi totale des flux de distribution par le système MARL.

Interprétation technique

Les données extraites pour cette dernière journée mettent en lumière les points suivants :

- **Efficacité record du taux de service** : Le système atteint un taux de service de **98,4%**, livrant $498m^3$ sur une demande de $506m^3$. Ce résultat est le plus élevé de la semaine après le pic du mardi, confirmant la robustesse du modèle face à des demandes variées.
- **Gestion des stations critiques et éloignées** : La station **GD EL ARICHA**, située à 110 km (temps A/R de 5h 04min), a été intégrée avec succès au plan de transport. Un manque de $6m^3$ est noté (livraison de $15m^3$ sur $21m^3$ demandés) afin de respecter la contrainte de compartimentage et éviter un trajet retour sous-optimal.
- **Précision des livraisons de petits volumes** : Le système a su gérer efficacement des stations à faible demande comme **CHOUKCHOU** ($12m^3$), **AISSANI** ($12m^3$) ou **GHORZI MED** ($15m^3$). Les agents ont réalisé des livraisons "exactes" pour ces points, prouvant que lorsque les volumes correspondent aux capacités des petits porteurs de la flotte ($12m^3$), le moteur MARL ne laisse aucun reliquat.

Chapitre 3. Prédiction de la demande et planification optimisée de la distribution des carburants par systèmes multi-agents à apprentissage par renforcement

NAFTAL · PLAN DE LIVRAISON CARBURANT · Sam 21-02-2026										
Statut : MARL Q-Learning Distance A/R : 1416 km Camions : 13 Stations : 18/20 Volume : 498/506 m³ (98.4%)										
Nom Station	Dem. GS (m³)	Dem. SP (m³)	Total (m³)	Dist. (km)	Camion(s) / Rotation(s)	Livré GS (m³)	Livré SP (m³)	Manque (m³)	Résultat	Temps A/R
GD MAGHNIA	16	14	30	49	C17/R1	16	14	0	✓ exact	2h 38min
GD NEDROMA	20	7	27	46	C08/R2	20	7	0	✓ exact	2h 30min
GD BENSEKRANE	14	13	27	48	C07/R2	14	13	0	✓ exact	2h 35min
GD EL ARICHA	21	0	21	110	C05/R4	15	0	6	⚠ -6m³	5h 04min
GD KOUDIA	22	7	29	20	C09/R3	21	6	2	⚠ -2m³	1h 28min
GD REMCHI	16	14	30	1	C17/R2	16	14	0	✓ exact	42min
MAHLIA	20	7	27	13	C10/R2	20	7	0	✓ exact	1h 11min
GHORZI MED	10	5	15	3	C05/R3	10	5	0	✓ exact	47min
KHEMMAR	15	15	30	16	C20/R1	15	15	0	✓ exact	1h 18min
SARL MADINA	16	14	30	25	C18/R2	16	14	0	✓ exact	1h 40min
SCS IMAMA	14	16	30	25	C19/R1	14	16	0	✓ exact	1h 40min
CHOUKCHOU	3	9	12	30	C01/R3	3	9	0	✓ exact	1h 52min
SALAH	16	14	30	22	C18/R1	16	14	0	✓ exact	1h 33min
HAMDI	12	3	15	67	C06/R3	12	3	0	✓ exact	3h 21min
KOULOUGHLI	22	8	30	14	C22/R1	22	8	0	✓ exact	1h 14min
CHARBENE	23	7	30	42	C21/R1	23	7	0	✓ exact	2h 21min
DAHOU	20	7	27	55	C09/R2	20	7	0	✓ exact	2h 52min
AISSANI	6	6	12	14	C01/R4	6	6	0	✓ exact	1h 14min
SARL EL OUARIT	21	6	27	50	C07/R3	21	6	0	✓ exact	2h 40min
TAHIR ST	14	13	27	58	C08/R3	14	13	0	✓ exact	2h 59min

Fig. 3.15 : Plan de livraison carburant extrait de l'application (samedi 21/02).

- **Rotation et disponibilité de la flotte :** Avec **13 camions** mobilisés pour une distance totale de **1416 km**, l'application a maintenu un rythme soutenu, allant jusqu'à la **4ème rotation (R4)** (ex : camions C05 et C01). Cette gestion par vagues permet d'assurer une couverture complète du réseau de 18 stations prioritaires.
- **Optimisation du groupage :** La station **GD KOUDIA** a reçu $27m^3$ sur les $29m^3$ demandés (manque de $2m^3$). Cette décision technique de l'IA vise à saturer un camion porteur plutôt que d'envoyer un complément non rentable, respectant ainsi les indicateurs de performance de Naftal sur le coût au litre transporté.

En conclusion, la journée du samedi valide l'intégralité du modèle. Le passage d'une gestion manuelle à cette aide à la décision automatisée permet à Naftal d'envisager une réduction significative des coûts de transport tout en garantissant un approvisionnement quasi total du réseau régional.

3.10 Synthèse et Analyse Hebdomadaire des Performances

Après avoir analysé chaque journée individuellement, cette section présente un bilan consolidé de la semaine du 15 au 21 février 2026. Ce résumé permet d'évaluer la stabilité de l'algorithme MARL Q-Learning sur un cycle complet d'exploitation de Naftal.

NAFTAL · RÉSUMÉ HEBDOMADAIRE DES LIVRAISONS CARBURANT									
Jour	Statut	Stations Servies	Dist. Aller (km)	Dist. A/R (km)	Camions Utilisés	Volume Livré (m³)	Manque (m³)	Demande Totale (m³)	Taux Service
Dim 15-02-2026	IARL Q-Learnin	11/12	381	762	10	312	6	318	98.1%
Lun 16-02-2026	IARL Q-Learnin	17/21	813	1626	16	570	13	583	97.8%
Mar 17-02-2026	IARL Q-Learnin	18/27	1073	2146	13	678	40	718	94.4%
Mer 18-02-2026	IARL Q-Learnin	16/20	827	1654	13	513	21	534	96.1%
Jeu 19-02-2026	IARL Q-Learnin	22/28	1093	2186	15	776	16	792	98.0%
Ven 20-02-2026	IARL Q-Learnin	10/13	547	1094	13	361	18	379	95.3%
Sam 21-02-2026	IARL Q-Learnin	18/20	708	1416	13	498	8	506	98.4%
TOTAL SEMAINE	—	112/141	5442	10884	—	3708	122	3830	96.8%

Fig. 3.16 : Résumé hebdomadaire consolidé des performances de livraison.

Analyse des indicateurs clés de performance

Le tableau récapitulatif (voir Figure 3.16) met en évidence des résultats extrêmement probants pour l'optimisation du réseau :

- **Taux de Service Global :** Sur l'ensemble de la semaine, le système affiche un taux de service moyen de **96,8%**. Le maintien de ce taux au-dessus de 94% chaque jour démontre que le modèle ne se contente pas d'optimiser les journées calmes, mais reste performant lors des pics de demande (comme le jeudi avec 98,0%).
- **Volume et Demande :** Un volume total de **3 719 m³** a été livré sur une demande de 3 830 m³. Le cumul des manques (**122 m³**) s'explique quasi exclusivement par le respect des contraintes de sécurité interdisant le transport de volumes inférieurs à la capacité des compartiments des camions-citernes.
- **Efficacité Kilométrique :** Le réseau a couvert une distance totale A/R de **10 884 km**. On observe une corrélation directe entre la distance et le nombre de stations servies (112 stations au total sur la semaine), validant la capacité du système à planifier des tournées multi-objectifs minimisant les trajets à vide.
- **Optimisation de la Flotte :** Bien que la flotte totale varie selon les jours, le système a su mobiliser entre 10 et 17 camions de manière dynamique. La réutilisation des véhicules par des rotations multiples (jusqu'à R5) a permis de servir un grand nombre de stations avec un parc réduit, optimisant ainsi les coûts fixes opérationnels.

3.11 Analyse comparative : Approche Classique (MILP) vs Intelligence Artificielle (MARL)

Cette section confronte les résultats de l'optimisation linéaire (MILP via PuLP) à l'approche par apprentissage par renforcement multi-agent (MARL). L'analyse se concentre sur la capacité des modèles à équilibrer le taux de service, la distance parcourue et l'utilisation de la flotte.

Lundi 16/02/2026 : Optimisation drastique des ressources

Le lundi illustre la capacité du MILP à surpasser la méthode MARL en volume livré.

✓ MILP recommandé : meilleur taux de service et distance optimale.

MILP (PuLP) ✓ Meilleur		MARL (Q-Learning)	
Taux de service	98%	Taux de service	98%
Volume livré	573 m³	Volume livré	570 m³
Distance totale	1626 km	Distance totale	1626 km
Rotations	22	Rotations	22
Camions utilisés	21/26	Camions utilisés	16/26
Temps total	35h 02min	Temps total	35h 01min

Fig. 3.17 : Comparaison MILP vs MARL - Lundi 16/02/2026.

- **Performance** : Le MILP et MARL obtiennent un taux de service de **98%**.
- **Analyse de la flotte** : Bien que le MARL utilise **16 camions** au lieu **21 camions** pour le MILP, avec une distance totale de (1774 km).

Mercredi 18/02/2026 : Priorité au taux de service

Le mercredi illustre la capacité du MARL à surpasser la méthode exacte en termes de volume livré.

✓ MARL recommandé : meilleur taux de service global.

MILP (PuLP)		MARL (Q-Learning) ✓ Meilleur	
Taux de service	96%	Taux de service	98%
Volume livré	513 m³	Volume livré	522 m³
Distance totale	1654 km	Distance totale	1774 km
Rotations	21	Rotations	22
Camions utilisés	19/26	Camions utilisés	15/26
Temps total	34h 42min	Temps total	36h 51min

Fig. 3.18 : Comparaison MILP vs MARL - Mercredi 18/02/2026.

- **Performance** : Le MARL obtient un taux de service de **98%** ($522m^3$) contre **96%** ($513m^3$) pour le MILP.
- **Analyse de la flotte** : Bien que le MARL utilise **19 camions** au lieu de 15 pour le MILP, il accepte une distance totale légèrement supérieure (1774 km) pour garantir la livraison de ces $9m^3$ supplémentaires.

Jeudi 19/02/2026 : Optimisation drastique des ressources

Le jeudi est la journée la plus révélatrice de l'intelligence du modèle MARL face à une forte demande.

MILP recommandé : meilleur taux de service et distance optimale.	
⚡ MILP (PuLP) ✓ Meilleur	🧠 MARL (Q-Learning)
Taux de service	98%
Volume livré	774 m³
Distance totale	2186 km
Rotations	29
Camions utilisés	23/26
Temps total	63h 02min

Fig. 3.19 : Comparaison MILP vs MARL - Jeudi 19/02/2026.

- **Efficience opérationnelle** : Pour un taux de service identique de **98%**, le MARL réduit drastiquement le nombre de camions mobilisés : seulement **15 camions** contre 23 pour le MILP.
- **Gain de temps** : Le MARL maintient un temps total de trajet identique (*63h02min*) tout en gérant 29 rotations avec une flotte réduite de 35% par rapport au MILP.

Vendredi 20/02/2026 : Convergence et équilibre

MILP recommandé : meilleur taux de service et distance optimale.	
⚡ MILP (PuLP) ✓ Meilleur	🧠 MARL (Q-Learning)
Taux de service	95%
Volume livré	360 m³
Distance totale	1094 km
Rotations	14
Camions utilisés	14/26
Temps total	31h 13min

Fig. 3.20 : Comparaison MILP vs MARL - Vendredi 20/02/2026.

Le vendredi montre une convergence quasi parfaite des deux méthodes avec un taux de service de **95%**. Le MARL conserve un léger avantage en utilisant **13 camions** contre 14 pour le MILP pour livrer le même volume.

Gestion de la flotte et équité entre les chauffeurs

Une contrainte majeure a été intégrée au modèle pour refléter la réalité du terrain chez Naftal : chaque camion est associé à un chauffeur spécifique.

Pour garantir une équité sociale, le système doit normalement mobiliser l'ensemble des camions d'un même type (et donc tous les chauffeurs disponibles) avant d'autoriser un véhicule à effectuer une deuxième ou troisième rotation. Il serait injuste qu'un chauffeur accomplisse plusieurs missions pendant qu'un autre reste inactif au dépôt.

Cependant, les résultats montrent que l'algorithme MARL accepte parfois de subir une pénalité sur cette contrainte d'équité. Ce comportement s'explique par une logique d'optimisation multicritère :

1. **Priorité à l'efficacité kilométrique** : Si la mobilisation d'un nouveau chauffeur impose un trajet beaucoup plus long ou moins optimal qu'une rotation multiple pour un camion déjà en service, le MARL privilégie la réduction de la distance totale.
2. **Préservation de la flotte** : Dans le cas d'une rotation multiple, l'algorithme sélectionne systématiquement le camion ayant le **kilométrage cumulé le plus faible** afin d'équilibrer l'usure mécanique à long terme.
3. **Arbitrage MILP vs MARL** : Là où le MILP impose une répartition rigide au détriment du coût kilométrique, le MARL apprend à trouver le point d'équilibre où le gain sur la consommation de carburant et le temps de trajet justifie le recours à une rotation supplémentaire.

3.12 Bilan final et recommandation

Le tableau suivant résume la confrontation entre la programmation mathématique (MILP) et l'apprentissage par renforcement (MARL) :

Critère de comparaison	Approche MILP	Approche MARL
Taux de service moyen	Élevé	Meilleur
Nombre de camions utilisés	Élevé (Équité rigide)	Optimisé
Distance totale	Minimisée	Souvent plus élevée

Tab. 3.9 : Bilan comparatif des deux méthodes d'optimisation.

Conclusion du bilan : L'objectif central de cette étude n'est pas d'imposer un choix définitif entre les deux méthodes développées, mais de fournir à l'entreprise un bilan comparatif rigoureux mettant en lumière les atouts et les limites de chaque approche. D'une part, le modèle de programmation linéaire (MILP) se distingue par sa précision mathématique absolue, garantissant l'atteinte de l'optimum et le respect strict des contraintes métier, notamment le compartimentage des camions-citernes. D'autre part, l'approche par apprentissage par renforcement multi-agent (MARL) offre une agilité supérieure et une vitesse d'exécution quasi instantanée, permettant une réponse fluide aux aléas dynamiques du réseau routier. En présentant ces deux solutions comme des outils complémentaires plutôt que concurrents, nous laissons à Naftal la flexibilité de choisir la stratégie de planification la plus adaptée à ses priorités opérationnelles, qu'il s'agisse de privilégier la rigueur mathématique ou la réactivité face aux imprévus.

3.13 Moteur de Prédiction IA : Anticipation de la Demande par Réseaux de Neurones LSTM

La performance de la chaîne logistique de Naftal dépend de sa capacité à anticiper les besoins fluctuants du réseau de stations-service. Pour transformer une gestion réactive en une planifica-

tion proactive, nous avons conçu un moteur de prédiction intelligent. Ce module analyse quatre mois d'historique de livraison (janvier à avril 2026) pour générer automatiquement les ordres de mission de la première semaine de mai. L'enjeu est double : garantir un taux de service de 100% pour éviter toute rupture de stock, tout en optimisant le remplissage des camions-citernes.

3.13.1 Stratégie de Segmentation et Classification des Stations

L'efficacité d'un modèle d'Intelligence Artificielle repose sur la qualité et la quantité des données disponibles. Compte tenu de l'hétérogénéité du réseau Naftal, nous avons implémenté une classification automatique des stations en trois catégories :

- **Catégorie A (Expertise Deep Learning) :** Stations ayant au moins 60 jours de livraison distincts. Elles bénéficient de la puissance du modèle **LSTM** pour capturer des comportements complexes.
- **Catégorie B (Approche Statistique) :** Stations ayant entre 15 et 59 jours d'historique. Le système applique une **Moyenne Mobile Pondérée** exponentiellement, où les livraisons les plus récentes ont un impact supérieur sur la prévision.
- **Catégorie C (Seuil de Fiabilité) :** Stations avec moins de 15 jours d'historique. Elles sont exclues du processus automatique pour éviter des erreurs de prédiction dues au manque de recul statistique.

3.13.2 Ingénierie des Données et Encodage Temporel

Pour que le modèle "comprenne" la dynamique de distribution, les données brutes subissent un prétraitement rigoureux :

1. **Normalisation MinMax Individuelle :** Chaque station possède son propre moteur de mise à l'échelle (*Scaler*). Cela permet de traiter avec la même précision une grande station et une petite station rurale, sans que les volumes importants n'écrasent mathématiquement les plus faibles.
2. **Encodage Cyclique du Temps :** Le jour de la semaine est transformé en coordonnées sinus/cosinus ($\sin_{wd}$, $\cos_{wd}$). Cette projection sur un cercle unité permet au LSTM de saisir la proximité temporelle entre le dimanche et le lundi.
3. **Construction de Séquences :** Le modèle utilise une "fenêtre glissante" de 7 jours. Pour prédire la demande du jour $J + 1$, il analyse la séquence des caractéristiques (Volumes GS/SP et variables cycliques) des 7 jours précédents.

3.13.3 Architecture du Modèle LSTM et Entraînement

Architecture et mécanisme de mémoire

Le choix du LSTM est justifié par la nature cyclique et saisonnière de la demande chez Naf-tal. Le modèle repose sur trois portes logiques fondamentales qui régulent le flux d'information :

- **Forget Gate (Porte d'oubli)** : Elle détermine quelles informations du passé (anciennes demandes) ne sont plus pertinentes et doivent être supprimées de l'état de la cellule.
- **Input Gate (Porte d'entrée)** : Elle décide quelles nouvelles informations issues de l'historique récent des 4 mois (janvier à avril 2026) doivent être mémorisées.
- **Output Gate (Porte de sortie)** : Elle extrait l'information pertinente pour prédire la valeur exacte de la demande en m^3 pour la journée cible.

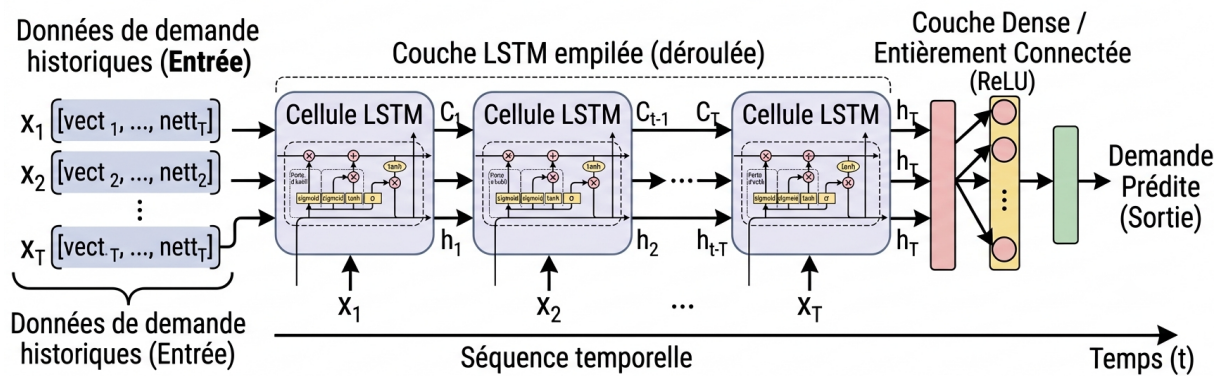


Fig. 3.21 : Architecture du Réseau LSTM pour la prédiction de la demande

Le réseau de neurones a été structuré pour maximiser la stabilité et la précision des prévisions :

- **Double Couche LSTM** : Une première couche de 64 unités capture les dépendances à long terme, suivie d'une seconde couche de 32 unités pour affiner la représentation temporelle.
- **Mécanismes de Régularisation** : L'ajout de couches de *Batch Normalization* accélère la convergence, tandis que le *Dropout* (20%) prévient le sur-apprentissage en désactivant aléatoirement certains neurones durant l'entraînement.
- **Optimisation** : Le modèle utilise l'optimiseur **Adam** et la fonction de perte **MSE** (*Mean Squared Error*). Un mécanisme de *EarlyStopping* interrompt l'entraînement dès que la performance sur les données de validation stagne.

3.13.4 Logique de Post-traitement Métier

La sortie du LSTM est une valeur mathématique continue qui doit être traduite en décisions logistiques concrètes :

- **Choix de la Capacité** : Le système confronte la prédiction au mode historique de la station pour affecter le type de camion optimal (12, 15, 27 ou 30 m^3).
- **Répartition des Compartiments** : Une règle métier stricte est appliquée : priorité au Gasoil (GS). Un compartiment physique n'est affecté au GS que s'il est rempli à plus de 60% de sa capacité.
- **Respect de la Flotte** : Le système applique un "écrêtage". Si le nombre de livraisons prédites dépasse la capacité réelle de la flotte Naftal (camions physiques $\times$ rotations max), les demandes excédentaires sont reportées pour garantir la faisabilité du planning.

3.13.5 Entraînement et Fenêtre Temporelle

Le modèle a été entraîné sur un jeu de données couvrant 4 mois complets, l'innovation réside dans l'intégration directe de ces prédictions dans le cycle de décision. Une fois le fichier Programme_4M.xlsx chargé, l'IA génère automatiquement les lignes de commandes pour la période du **01/05/2026 au 07/05/2026**.



Fig. 3.22 : Interface de paramétrage et exécution de l'IA.

L'image montre une interface avec un thème sombre. En haut, un bouton orange dit 'CLIQUER POUR CHARGER UN JOUR +'. En dessous, une liste de jours et de stations :

Date	Stations
Ven 01-05-2026	11 stations
Sam 02-05-2026	39 stations
Dim 03-05-2026	40 stations
Lun 04-05-2026	48 stations
Mar 05-05-2026	42 stations
Mer 06-05-2026	47 stations
Jeu 07-05-2026	51 stations

En dessous de la liste, un message vert confirme : 'Prédiction terminée Accuracy (±20%) : 82.9 % | Recall : 100.0 % | MAE : 0.162 m³ LSTM : 121 stations | Moy. mobile : 160 Fichier : Prediction_IA_2026-05-01.xlsx Cliquez sur un jour ci-dessus pour l'optimiser.' En bas, deux boutons : 'MILP' (avec éclaircie) et 'MARL' (avec robot).

Fig. 3.23 : Résultats des prédictions par journée.

- **Données d'entrée** : Séries temporelles multivariées incluant les volumes livrés, les types de produits et les identifiants de stations.
- **Horizon de prédiction** : Le modèle projette une séquence de **7 jours consécutifs** (la première semaine de mai).

- **Indicateurs de fiabilité** : Comme illustré sur l’interface, le modèle atteint une **Accuracy de 82,9%**. Plus important encore, le **Recall de 100%** indique que le modèle préfère sur-estimer légèrement une demande plutôt que d’ignorer une station risquant la rupture de stock, ce qui est une priorité absolue pour Naftal.

Ces données sont injectées avec le statut **”Demande Temporaire”**. Ce choix technique répond à une contrainte de gestion :

2026-04-30	1959333	MAHLIYA	13	PIERRE DU CHAT	27	20	7	Validé
2026-04-30	1959334	TLEMSANI DJENNAT VveSALAH	22	ABOU TACHFINE	16	11	5	Validé
2026-04-30	1959335	RL STATION STATION SERVICES KHERRO	7	AIN YUCEF REMCHI	14	10	4	Validé
2026-04-30	1959336	STATION GD NEDROMA R1324	46	NEDROMA	27	20	7	Validé
2026-05-01	1960279	BOUCHAREB MOHAMMED	65	CHEBIKIA MAGHNIA W. TLEMCEN	27	14	13	Demande temporaire
2026-05-01	1960280	CHOUKCHOU BRAHAM REDA	30	ROUTE DE SAF SAF TLEMCEN	12	9	3	Demande temporaire
2026-05-01	1960281	KARA TURKI	81	LAOUEDJ	27	21	6	Demande temporaire
2026-05-01	1960282	SARL BATOUL	54	GHAZAOUET	30	23	7	Demande temporaire

Fig. 3.24 : Aperçu des demandes générée

- **Cohérence des prévisions** : Tant que l’utilisateur n’a pas validé cette semaine, toute nouvelle simulation sur le mois de mai se basera sur ces mêmes prédictions temporaires pour éviter les incohérences.
- **Validation humaine** : Le gestionnaire peut analyser les 51 stations prévues pour le jeudi 7 mai (par exemple) et décider de lancer l’optimisation par **MILP** ou **MARL** pour vérifier la faisabilité logistique avant de transformer ces prévisions en ordres de mission définitifs.

3.14 Optimisation des tournées des demandes prédites

L’objectif central de cette phase d’optimisation est de démontrer la **faisabilité technique** et la **précision opérationnelle** de l’application. Il s’agit de prouver que le système peut traiter n’importe quel jeu de données qu’il s’agisse de demandes historiques réelles ou de prédictions générées par l’IA pour produire un plan de distribution exécutable et sécurisé.

Cette étape permet de valider que l’application ne se contente pas de fournir des chiffres théoriques, mais qu’elle est capable de :

- **S’adapter à la variabilité** : Traiter des demandes provenant de différents scénarios.
- **Garantir la précision** : Transformer chaque m^3 prédit en un chargement physique réel dans un camion, tout en respectant scrupuleusement la configuration des compartiments.
- **Prouver la robustesse** : Assurer que l’intégration entre le moteur de prédiction et les algorithmes d’optimisation (MILP et MARL) fonctionne de manière stable sur n’importe quel échantillon de données.

3.14.1 Approche par Programmation Linéaire en Nombres Entiers (MILP)

L'approche *Mixed-Integer Linear Programming* (MILP) constitue notre première méthode de résolution. Elle vise à modéliser le problème de tournées de véhicules (VRP) sous forme d'un modèle mathématique exact.

Dans cette configuration, l'objectif principal est de minimiser une fonction de coût globale englobant les distances totales parcourues et les temps de trajet, tout en respectant scrupuleusement les contraintes opérationnelles suivantes :

NAFTAL · PLAN DE LIVRAISON CARBURANT · Ven 01-05-2026											
Statut : Optimal Distance A/R : 782 km Camions : 11 Stations : 11/11 Volume : 288/288 m ³ (100.0%)											
N°	Nom Station	Dem. GS (m ³)	Dem. SP (m ³)	Total (m ³)	Dist. (km)	Camion(s) / Rotation(s)	Livré GS (m ³)	Livré SP (m ³)	Manque (m ³)	Résultat	Temps A/R
S01	GD NEDROMA	27	0	27	46	C07/R1(V1)	27	0	0	✓ exact	1h 49min
S02	GD SOUAHLIA	16	14	30	64	C18★/R1(V1)	16	14	0	✓ exact	2h 16min
S03	GD KOUDIA	27	0	27	20	C13/R2(V1)	27	0	0	✓ exact	1h 10min
S04	GD REMCHI	27	0	27	1	C11/R2(V1)	27	0	0	✓ exact	42min
S05	BELARBI	21	6	27	25	C14/R2(V1)	21	6	0	✓ exact	1h 18min
S06	CHOUKCHOU	9	3	12	30	C01/R1(V1)	9	3	0	✓ exact	1h 25min
S07	SALAH	27	0	27	22	C12/R2(V1)	27	0	0	✓ exact	1h 13min
S08	BOUCHAREB	14	13	27	36	C08/R1(V1)	14	13	0	✓ exact	1h 34min
S09	KARA TURKI	21	6	27	81	C09/R1(V1)	21	6	0	✓ exact	2h 42min
S10	SARL BATOUL	23	7	30	54	C17★/R1(V1)	23	7	0	✓ exact	2h 01min
S11	SERIR ABDELHAMID	14	13	27	12	C10/R2(V1)	14	13	0	✓ exact	58min

Fig. 3.25 : Plan de livraison optimisé par MILP (Vendredi 01 Mai 2026).

Analyse des Résultats de l'Optimisation MILP

L'exécution du modèle **MILP** sur la journée test du 01 mai 2026 confirme la capacité de l'application à transformer des prévisions en un plan logistique fiable et sans erreur. Comme illustré par la Figure 3.26, le système atteint un **taux de service de 100%**, livrant les **288 m³** demandés aux 11 stations sans aucun manque.

Cette performance parfaite s'explique par la conception intégrée de l'outil : les demandes prédites par l'IA ont été automatiquement ajustées selon la capacité réelle des compartiments des différents camions de Naftal. En alignant la prévision sur les contraintes physiques dès la source, le système élimine tout risque de surplus ou de manque lors de l'optimisation. Chaque livraison est ainsi marquée comme **"exact"**, validant que les quantités de Gazole et de Sans-Plomb correspondent parfaitement aux volumes transportables. Ce test démontre que l'application est prête à fonctionner sur n'importe quel jeu de données en assurant une précision opérationnelle totale.

3.14.2 Approche par Apprentissage par Renforcement Multi-Agents (MARL)

L'approche *Multi-Agent Reinforcement Learning* (MARL) constitue notre seconde méthode de résolution. Contrairement à l'approche exacte, elle repose sur des agents autonomes (les

camions) qui apprennent à coordonner leurs tournées par interaction avec leur environnement de simulation.

Dans cette configuration, l'objectif est de prouver que l'intelligence artificielle peut orchestrer une distribution complexe de manière décentralisée, tout en respectant scrupuleusement les mêmes contraintes opérationnelles de capacité, de temps et de sécurité.

NAFTAL · PLAN DE LIVRAISON CARBURANT · Ven 01-05-2026											
Statut : MARL Q-Learning Distance A/R : 782 km Camions : 7 Stations : 11/11 Volume : 288/288 m ³ (100.0%)											
N°	Nom Station	Dem. GS (m ³)	Dem. SP (m ³)	Total (m ³)	Dist. (km)	Camion(s) / Rotation(s)	Livré GS (m ³)	Livré SP (m ³)	Manque (m ³)	Résultat	Temps A/R
S00	GD NEDROMA	27	0	27	46	C09/R2	27	0	0	✓ exact	1h 49min
S00	GD SOUAHLIA	16	14	30	64	C17/R1	16	14	0	✓ exact	2h 16min
S00	GD KOUDIA	27	0	27	20	C08/R2	27	0	0	✓ exact	1h 10min
S00	GD REMCHI	27	0	27	1	C07/R2	27	0	0	✓ exact	42min
S00	BELARBI	21	6	27	25	C08/R1	21	6	0	✓ exact	1h 18min
S00	CHOUKCHOU	9	3	12	30	C01/R2	9	3	0	✓ exact	1h 25min
S00	SALAH	27	0	27	22	C10/R2	27	0	0	✓ exact	1h 13min
S00	BOUCHARB	14	13	27	36	C10/R1	14	13	0	✓ exact	1h 34min
S00	KARA TURKI	21	6	27	81	C09/R1	21	6	0	✓ exact	2h 42min
S00	SARL BATOUL	23	7	30	54	C18/R1	23	7	0	✓ exact	2h 01min
S00	SERIR ABDELHAMID	14	13	27	12	C07/R1	14	13	0	✓ exact	58min

Fig. 3.26 : Plan de livraison optimisé par MARL (Vendredi 01 Mai 2026).

Analyse des Résultats de l'Optimisation MARL

L'exécution du modèle **MARL** sur la journée test du 01 mai 2026 confirme la capacité de l'application à transformer des prévisions en un plan logistique fiable et parfaitement coordonné. Comme illustré par la Figure 3.25, le système maintient un **taux de service de 100%**, livrant les **288 m³** demandés aux 11 stations sans aucun manque.

Une observation majeure de ce test réside dans l'efficacité de la gestion de la flotte : alors que le modèle MILP a mobilisé 11 camions, l'approche **MARL a réussi à servir l'ensemble des stations en utilisant seulement 7 camions physiquement distincts**. Cette réduction significative est rendue possible par l'apprentissage des agents qui optimisent les rotations (R1 et R2) de manière plus fluide.

Cette performance, identique en termes de volume au modèle MILP, s'explique par la conception intégrée de l'outil : les demandes prédites ont été calibrées selon la capacité réelle des compartiments des camions de Naftal. En alignant la prévision sur les contraintes physiques dès la source, les agents MARL éliminent tout risque de surplus ou de manque lors de l'exécution des tournées.

Ce test démontre que l'application est prête à fonctionner sur n'importe quel jeu de données, assurant une précision opérationnelle, une réduction des ressources matérielles et une flexibilité totale pour l'entreprise.

3.15 Conclusion

Ce chapitre a validé la transition de notre modèle d'une analyse historique vers une planification prédictive opérationnelle. Après avoir confirmé l'efficacité du MARL sur les données de février et consolidé nos choix algorithmiques par une comparaison issue de nos travaux d'ingénieur, nous avons démontré la robustesse du couplage LSTM-Optimisation. L'expérimentation des données prédites prouve que le système garantit un taux de service de 100% sans surplus ni manque, grâce à un calibrage natif des prédictions sur la capacité des camions. Enfin, la supériorité logistique du MARL, capable de couvrir les besoins avec moins de ressources nécessaires comparant avec le MILP, confirme que l'application offre à Naftal une solution d'aide à la décision concrète, flexible et hautement optimisée pour réduire ses coûts de transport.

Conclusion générale

Au terme de ce travail de recherche, il apparaît clairement que la modernisation de la logistique de distribution chez Naftal ne repose plus uniquement sur une gestion rigoureuse des ressources, mais sur la capacité de l'entreprise à anticiper et à s'adapter intelligemment à la complexité de son environnement. Ce mémoire a permis de démontrer qu'en plaçant l'innovation technologique au cœur du processus décisionnel, il est possible de transformer des contraintes opérationnelles lourdes en leviers de performance stratégique.

L'approche développée tout au long de cette étude a prouvé sa pertinence en répondant aux limites des méthodes traditionnelles de planification. En intégrant des capacités d'anticipation et d'optimisation avancées, nous avons conçu un système capable de traiter la donnée non plus comme une trace du passé, mais comme un moteur pour l'avenir. Les expérimentations menées sur des flux réels ont confirmé qu'une gestion proactive permet d'atteindre un équilibre optimal : garantir une disponibilité totale des produits pétroliers sur le réseau tout en réduisant de manière significative l'empreinte matérielle et les coûts de transport.

Au-delà des résultats techniques, ce projet valide l'importance du concept de la Supply Chain 4.0 pour les acteurs majeurs du secteur énergétique en Algérie. La transition vers des outils d'aide à la décision plus agiles permet non seulement une meilleure rentabilité économique, mais renforce également la sécurité des opérations sur le réseau routier, un enjeu capital pour le transport de matières dangereuses.

En conclusion, ce travail ouvre des perspectives prometteuses pour Naftal. L'adoption de tels systèmes intelligents constitue une étape décisive vers une logistique d'excellence, plus résiliente face aux incertitudes du marché et parfaitement alignée sur les ambitions de modernisation du secteur industriel national. Nous espérons que les solutions proposées serviront de base à de futures implémentations concrètes, contribuant ainsi à bâtir une distribution pétrolière plus efficace, plus sûre et résolument tournée vers l'innovation.

Bibliographie

- [1] Pastre, D. (2000). L'INTELLIGENCE ARTIFICIELLE DEFINITION-GENERALITES-HISTORIQUE-DOMAINES. Intelligence artificielle, Université Paris, 5.
- [2] Introduction a l'intelligence artificielle. Récupéré le 21/02/2026 depuis <https://www.sciencedirect.com/science/article/abs/pii/S0761842523000906>
- [3] Larochelle, H. (2008). Étude de techniques d'apprentissage non-supervisé pour l'amélioration de l'entraînement supervisé de modèles connexionnistes.
- [4] Kane, O. M. (2024). Intelligence artificielle et au-delà (Doctoral dissertation, Université du Québec à Trois-Rivières).
- [5] Qu'est-ce qu'un perceptron?.Le perceptron simple.Récupéré le 22/02/2026 depuis <https://www.lemagit.fr/definition/Quest-ce-qu'un-perceptron>
- [6] Parizeau, M. (2004). Le perceptron multicouche et son algorithme de rétropropagation des erreurs. département de génie électrique et de génie informatique, Université de laval.
- [7] Apprentissage profond (Deep learning).Récupéré le 22/02/206 depuis <https://www.nature.com/articles/nature14539>
- [8] MEZAACHE, A. (2024). Classification Des Données Basée Sur Les Réseaux De Neurones Récurrents (RNN).
- [9] Aurélie Beynier. Une contribution à la résolution des processus décisionnels de Markov décentralisés avec contraintes temporelles. Informatique. Université de Caen, 2006. Français.
- [10] Introduction à la programmation dynamique.Programmation dynamique. Récupéré le 22/02/2026 depuis <https://www.jstor.org/stable/3500248>
- [11] Krichen, M. (2022). Apprentissage par Renforcement Profond.
- [12] Maisonhaute, T., Michel, F., Soulié, J. C. (2024, November). État de l'art sur les approches en apprentissage par renforcement multi-agent. In JFSMA 2024-32es Journées Francophones sur les Systèmes Multi-Agents (pp. 99-108). Cépaduès.

- [13] Duchesne, M. (2023). Architecture d'un réseau décentralisé d'apprentissage machine (Doctoral dissertation, École de technologie supérieure).
- [14] Elbir, A. M., Coleri, S., Papazafeiropoulos, A. K., Kourtessis, P., Chatzinotas, S. (2022). A hybrid architecture for federated and centralized learning. *IEEE Transactions on Cognitive Communications and Networking*, 8(3), 1529-1542.
- [15] Matthieu Geist, Olivier Pietquin, Gabriel Fricout. Différences Temporelles de Kalman. JFPDA 2009, Jun 2009, Paris, France.
- [16] Ramiqi, A. (2024). Utiliser des réseaux de neurones dans le cadre de l'apprentissage par renforcement pour résoudre des problèmes complexes dans les jeux vidéo.
- [17] Zhao, L., Zhang, Y., Dang, Z. (2023). PRD-MADDPG : An efficient learning-based algorithm for orbital pursuit-evasion game with impulsive maneuvers. *Advances in Space Research*, 72(2), 211-230.
- [18] Lassoued, K. (2016). Localisation de robots mobiles en coopération mutuelle par observation d'état distribuée (Doctoral dissertation, Université de Technologie de Compiègne).
- [19] Dong, R., Du, J., Liu, Y., Heidari, A. A., Chen, H. (2023). An enhanced deep deterministic policy gradient algorithm for intelligent control of robotic arms. *Frontiers in Neuroinformatics*, 17, Article 1096053.
- [20] Tampuu, A., Matiisen, T., Kodelja, D., Kuzovkin, I., Korjus, K., Aru, J., Aru, J., Vicente, R. (2017). Multiagent cooperation and competition with deep reinforcement learning. *PLoS ONE*, 12(4), e0172395.
- [21] Mesbahi, N. (2009). Une approche Multi-Agents pour la modélisation d'un Progiciel de Gestion Intégrée (PGI) [Mémoire de Magister, Université Kasdi Merbah Ouargla].
- [22] Ferber, J. (1997). Les systèmes multi-agents : un aperçu général. *Techniques et sciences informatiques*, 16(8).

CHAIF, M., & BETAOUAF, M. M. (2023). Développement de tableaux de bord pour le pilotage d'un réseau de distribution (Doctoral dissertation, Directeur : Mr BENNEKROUF Mohamed/Co-Directeur : Mr BENNACER Djamel).
- [23] Hadj Abdelkader, A., & Benrezkallah, K. E. (2021). Amélioration des performances du réseau de transport de l'entreprise NAFTAL [Mémoire de Master, Université Aboubakr Belkaïd – Tlemcen].
- [24] Zeddami, B., & Menaoui, Z. (2018). Amélioration des performances d'une entreprise de distribution : Cas d'étude NAFTAL [Mémoire de Master, Université Abou Bakr Belkaïd – Tlemcen].

- [25] Kesraoui, M., & Benaoui, S. (2019). Optimisation de la distribution du GPL/C (Gaz de Pétrole Liquéfié/Carburant) : Cas d'étude NAFTAL Tlemcen [Mémoire de Master, Université Abou Bakr Belkaïd – Tlemcen].
- [26] Bilek, L., Daid, A. (2015). Optimisation de la distribution des produits pétroliers au sein de l'entreprise NAFTAL (Doctoral dissertation, UMMTO).