

UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

PLANIFICATION DE CHEMIN ÉCOÉNERGÉTIQUE D'UN ROBOT
MOBILE AUTONOME DE MANUTENTION

THÈSE PRÉSENTÉE
COMME EXIGENCE PARTIELLE DU
DOCTORAT EN INGÉNIERIE

PAR
EMNA MEJRI

NOVEMBRE 2023

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire ou de cette thèse a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire ou de sa thèse.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire ou cette thèse. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire ou de cette thèse requiert son autorisation.

UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

DOCTORAT EN INGÉNIERIE (PH.D.)

Direction de recherche :

Prof. Sousso KELOUWANI

Directeur de recherche

Prof. Yves DUBÉ

Codirecteur de recherche

Jury d'évaluation :

Prof. Sousso KELOUWANI

Directeur de recherche

Prof. Yves DUBÉ

Codirecteur de recherche

Prof. Pascal FORGET

Président du jury

Prof. Ali AIBIDI

Évaluateur externe

Prof. Ismail BISKRI

Évaluateur externe

Résumé

La robotique mobile joue un rôle clé dans l'industrie 4.0, grâce à une nouvelle génération de véhicules/robots autonomes de manutention industriels appelée *AMRs* (ou *AGVs*, anciennement) ayant pour mission le transport intelligent des biens au sein d'une manufacture ou entrepôt. Ces *AMRs* doivent posséder une grande flexibilité et agilité et ils doivent cohabiter avec des opérateurs humains. Cette agilité apporte de nouveaux défis opérationnels qui nécessitent une meilleure gestion énergétique. La planification du chemin est une décision clé dans la navigation des robots mobiles. Cette décision est prise plusieurs fois par jour et par suite l'optimisation de son efficacité énergétique peut apporter un gain important en termes de coût de déploiement de l'*AMR* (meilleure autonomie, plus de temps productifs, moins de temps improductifs de recharges des batteries) et en termes de coût de maintenance et/ou changement de batterie en fin de vie (en évitant ses décharges profondes). On distingue deux types de planifications de chemins (i) planification de chemin globale (entre deux positions source/destination) qui fait référence à la recherche d'un chemin sans collision entre une position source et une position cible dans un environnement comportant des obstacles statiques (cette planification peut être utilisée d'une manière incrémentale pour planifier la visite d'une suite de positions si l'ordre de visite est prédéfini), et (ii) planification de chemin plus générale multideestination qui cherche une route d'une position source unique vers un ensemble bien déterminé de positions non ordonnées. On considère, dans cette étude, particulièrement le problème du routage de préparation de commandes dans un entrepôt qui fait référence à la collecte des articles d'une liste de commande client à partir de leurs emplacements de stockage. La plupart des recherches portant sur ces deux types de planifications se basent majoritairement sur le critère de plus courte distance, ignorant l'aspect énergétique. En effet,

dans le contexte des bâtiments intérieurs, les travaux existants dans la littérature visant l'amélioration de l'efficacité énergétique des *AMRs* à travers la planification de chemin abordent principalement la planification de chemin locale (le contrôle des vitesses des actionneurs) et non la planification de chemin globale (le processus qui lui génère le chemin à suivre d'un point à l'autre), en supposant par défaut que le chemin le plus court est le plus économe énergétiquement. Cette planification globale est généralement basée sur une carte métrique de l'environnement reflétant l'information des obstacles et des dimensions. Cependant, mis à part la distance, l'environnement comporte d'autres informations susceptibles d'influencer la demande énergétique d'un chemin versus l'autre, telles que les conditions du sol, la congestion des voies de passages, etc. En outre, dans la situation où le robot doit changer de masse durant son chemin, tel que le problème de routage de préparation de commandes (le type (ii) de planification), il est indispensable de considérer cette variation lors de la décision de routage. Par suite, l'objectif de cette thèse est de proposer deux nouvelles approches de planification de chemin d'un *AMR*, qui réduisent sa consommation d'énergie, tout en respectant les contraintes opérationnelles du temps (avec un compromis temps/énergie). À cet égard, premièrement, une approche de planification de chemin globale écoénergétique (*Energy Efficient Global Path Planning (EE-GPP)*) est proposée pour la recherche de chemin entre deux positions de l'espace de déploiement qui minimise, simultanément, le temps et l'énergie consommée. Cette approche tient compte de l'effet de la variation de la friction au sol et du trafic sur la demande énergétique des chemins alternatifs. Deuxièmement, une approche de planification de chemin écoénergétique pour le routage de préparation de commande (*Energy Efficient Order Picking Routing (EE-OPR)*) d'un « *Pick support-AMR* », avec un bon compromis temps / énergie, est présentée. Dans cette partie du travail, l'effet de la variation de la masse transportée par les robots d'un point de ramassage à l'autre est souligné. Cette variation est considérée dans la formulation du problème d'optimisation à travers le modèle dynamique du véhicule. De plus, étant donné le manque des travaux qui traitent l'efficacité énergétique de la planification de chemin global et l'absence des travaux sur la planification de chemin écoénergétique des *Ps-AMRs* (étant leurs nouveautés), des analyses comparatives ont été réalisées pour évaluer nos approches avec des approches utilisées dans l'état de l'art qui se basent seulement sur le critère de distance/temps pour chaque problème séparément. Les résultats montrent que les

chemins optimisés par *EE-GPP* et les tournées plus écologiques données par *EE-OPR* sont efficaces et robustes, avec des économies moyennes significatives en termes d'énergie et de coût.

Remerciements

Je tiens tout d'abord à remercier mon directeur de thèse, monsieur Sousso Kelouwani pour son encadrement enrichissant et son aide à la réalisation de ce travail. Je suis honoré d'avoir collaboré avec une personne ayant vos qualités scientifiques et humaines. Merci pour vos conseils, votre générosité, votre écoute et votre compréhension.

Je remercie également monsieur Yves Dubé, mon codirecteur de thèse, pour son aide scientifique, son soutien continu et sa disponibilité. J'apprécie beaucoup votre modestie, votre disponibilité et votre compréhension. Puisse ce travail vous satisfaire et vous refléter ma haute considération.

Je remercie ensuite l'ensemble des membres du jury, d'avoir accepté d'évaluer mon travail et de l'avoir étudié avec attention.

Je veux dédier ce travail de recherche à :

Mes chers parents,

Merci pour votre soutien continu, vos prières et pour votre amour inconditionnel

Mon mari et mon meilleur ami,

Ce travail n'aurait pu aboutir sans ta présence, ta patience et ton soutien inconditionnel tout au long de mon aventure, notre aventure : tous les deux au doctorat, avec un enfant en charge ! : on savait que le défi était grand. Mais on savait aussi que rien n'est impossible ; il suffit juste d'y croire et de travailler pour. Merci pour tout Hatem !

Mon trésor, mon fils champion Skander

Tu es ma source d'amour, d'espoir et de motivation. Nous sommes vraiment chanceux de t'avoir comme enfant, mon grand. Je m'excuse si je n'ai pas pu te donner plus de temps, et que je n'étais pas toujours présente. Merci mon Skandou tu es toujours raisonnable et

compréhensif et tu m'encourageais continuellement, comme un grand! Merci mon chéri

Ma sœur et mon frère et mon beau-frère,

Merci pour votre soutien et votre encouragement et votre présence continue malgré les distances.

Mes parents au Canada,

Merci Monique et Jacques pour votre support continu. Merci d'être toujours là pour nous.

Mrs Melancon et Larry,

Une pensée particulière à vous deux. Je n'oublierais jamais tout ce que vous avez fait pour nous. Ceci a contribué à l'aboutissement de ce rêve. Merci infiniment.

Table des matières

Résumé	i
Remerciements	iv
Table des matières	vi
Table des figures	xi
Chapitre 1 - Introduction	1
1.1 Contexte général	1
1.1.1 AGV versus AMR	2
1.1.2 Types des AMR et utilisations	3
1.1.3 Flexibilité accrue et nouveaux défis émergents	4
1.1.4 Aspect économique et écologique	6
1.1.5 Planification de chemin des AMR	10
1.2 Problématique de recherche.....	14
1.3 Objectifs de recherche	19
1.4 Méthodologie.....	19
1.5 Structure de la thèse.....	22
Chapitre 2 - État de l'art	23
2.1 Les AMR et la durabilité	23
2.2 Planification de chemin.....	24
2.2.1 Recherche dans un graphe	24
2.2.2 Formulation du problème de planification de chemin	25
2.2.3 Catégories de planification de chemin	26

2.3	Planification de chemin globale (GPP).....	29
2.3.1	Représentation de l'espace	29
2.3.1.1	Les méthodes probabilistes	31
2.3.1.2	Méthodes déterministes	33
2.3.2	Recherche de chemin dans un graphe	36
2.3.2.1	L'algorithme de Dijkstra	36
2.3.2.2	L'algorithme de A^* et ses successeurs	39
2.3.3	Méthode de champs de potentiel artificiel	40
2.3.4	Approche métaheuristique pour la résolution du problème de planification de chemin global	41
2.3.5	Planification de chemin globale dans le contexte industriel d'un AMR	42
2.4	Planification de chemin et critère énergétique.....	43
2.5	Synthèse sur les méthodes de planification globale.....	45
2.6	Routage de préparation de commandes d'un Ps -AMR (planification de chemin multi-destinations).....	50
2.6.1	Définition du problème <i>Steiner TSP</i>	51
2.6.2	Méthodes exactes pour la résolution du problème de routage de préparation de commandes	52
2.6.2.1	Méthodes de résolution du problème de routage de préparation de commandes basées sur la programmation dynamique	52
2.6.2.2	Formulation du programme linéaire mixte en nombres entiers	54
2.6.3	Méthodes heuristiques pour la résolution du problème de routage de préparation de commandes	56
2.7	Synthèse sur le routage économe en énergie pour la préparation des commandes.....	59
	Chapitre 3 - Approches pour la planification de chemin écoénergétique	64
3.1	Planification de chemin global écoénergétique.....	65
3.1.1	Introduction	65
3.1.2	Hypothèses	66

3.1.3	Représentation de l'espace par un graphe de rue et représentation des facteurs de l'environnement	67
3.1.3.1	Représentation de l'espace avec une carte de grille d'occupation et une carte de résistance au roulement	67
3.1.3.2	Création des matrices d'adjacences distance et résistance au roulement	69
3.1.3.3	Représentation de l'espace avec une carte topologique et une carte de trafic	71
3.1.3.4	Définition de la matrice d'adjacence topologique et la matrice d'adjacence indicatrice du trafic	72
3.1.4	Définition du problème	75
3.1.5	Formules de programmation en nombres entiers pour la planification de chemin globale écoénergétique	78
3.1.5.1	Calcul de la consommation énergétique entre deux nœuds adjacents	80
3.1.5.2	Gestion du compromis temps/ énergie	82
3.1.6	Algorithme pour la planification de chemin globale écoénergétique . .	82
3.2	Routage de préparation de commande écoénergétique.....	85
3.2.1	Introduction	85
3.2.2	Représentation de la configuration de l'entrepôt	87
3.2.3	Description du problème	89
3.2.4	Hypothèses	91
3.2.5	Création d'un graphe dynamique	92
3.2.5.1	Définition des nœuds «states»	93
3.2.5.2	Définition des arcs de transitions	95
3.2.5.3	Création des nœuds d'états et des arcs de transitions	96
3.2.5.4	Coût de transition en termes de temps et d'énergie	99
3.2.6	Le problème de recherche dans un graphe	103

Chapitre 4 - Évaluation des approches pour la planification de chemin écoénergétique	108
4.1 Évaluation de la méthode <i>EE-GPP</i> , avec un <i>grid-graphe</i> de résistance au roulement	109
4.1.1 Configuration de l'espace et représentation du facteur de la résistance au roulement	109
4.1.2 Résultats de simulation	111
4.1.3 Analyse et discussion	114
4.2 Évaluation de la méthode <i>EE-GPP</i> avec un graphe topologique avec considération de l'effet du trafic	118
4.2.1 Configuration de l'espace et représentation des facteurs de la résistance au roulement et du trafic	118
4.2.2 Résultats de simulation	119
4.2.3 Analyse et discussion	122
4.3 Évaluation de la deuxième méthode <i>EE-OPR</i> , avec variation de la masse	122
4.3.1 Configuration de l'entrepôt	123
4.3.2 Résultats de simulation	123
4.3.3 Discussion	130
4.3.3.1 Flexibilité du graphe d'entrepôt et la différence entre la méthode <i>EE-OPR</i> et les méthodes basées sur <i>RRs</i>	131
4.3.3.2 Économie d'énergie et d'argent	133
4.3.3.3 Importance de la considération du compromis temps/énergie	134
Chapitre 5 - Conclusions et recommandations	141
5.1 Synthèse des travaux.....	142
5.1.1 <i>EE-GPP</i>	142
5.1.2 <i>EE-OPR</i>	145
5.2 Recommandations.....	148
5.3 Travaux en cours.....	151
5.4 Conclusion	153

Bibliographie	155
Annexe A - Articles publiés	183

Table des figures

1-1	Exemple d'AMR utilisées dans un <i>FMS</i> [12].	4
1-2	Exemples de AMR utilisés pour les opérations de préparations de commandes[13]-[17].	5
1-3	Différents types, formes, masses et capacités d'AMR [18].	6
1-4	Cycle de vie expérimental de la batterie au lithium-ion en fonction du DOD [34]	8
1-5	Effet de développement des environnements industriels et de la croissance de leur flexibilité sur la demande énergétique pour les déplacements des AMR .	11
1-6	Diagramme de processus de navigation d'un robot mobile	13
1-7	Le rapport entre le poids d'un véhicule et le nombre de kilomètres parcourus [56].	16
1-8	Un exemple simplifié de planification de chemin globale démontrant l'importance d'aborder le compromis entre le temps et l'énergie	17
1-9	Un exemple simpliste de routage de ramassage de commandes démontrant l'importance d'aborder le compromis entre le temps et l'énergie	18
1-10	La méthodologie de recherche suivie	21
2-1	Espace de configuration [76].	26
2-2	Classification de planification de chemin selon le sujet[77].	27
2-3	Planification de chemin d'un robot mobile : Classification [72].	30
2-4	Un exemple d'une carte routière (<i>roadmap</i>) dans un environnement avec des obstacles O_1 , O_2 et O_3 [85].	31

2-5	Deux exemples d'utilisation de la méthode <i>PRM</i> (exemple à gauche : la méthode réussit à trouver un chemin faisable ; exemple à droite : la méthode échoue à construire un graphe connexe).	32
2-6	Exemples d'utilisation de la méthode <i>RRT</i>	32
2-7	Méthode de décomposition en cellules approximatives (partie gauche de la figure) et en cellules uniformes (partie droite de la figure)[50], [88].	34
2-8	Exemple de carte topologique [96].	35
2-9	Un exemple d'exécution de l'algorithme de <i>Dijkstra</i> [107].	38
2-10	Comparaison entre l'exécution de l'algorithme A^* et l'exécution de l'algorithme <i>Dijkstra</i> de point de vue spatial [110].	40
2-11	Méthode de champ de potentiel artificiel pour la navigation du robot mobile [50].	41
2-12	(a) Exemple de différents chemins <i>Manhattan</i> avec la même longueur dans une représentation de carte <i>2D</i> avec l'information distance ; (b) Représentation de la même carte avec une information supplémentaire concernant la friction de l'espace de déploiement (carte <i>3D</i>) et changement de perspectives	49
2-13	Différence entre la distance euclidienne, de <i>Manhattan</i> et la distance la plus courte du chemin réel	50
2-14	(a) Un entrepôt Uni-bloc (b) Un entrepôt Multi-blocs	52
2-15	Une illustration d'un entrepôt et son graphe correspondant (a) Un exemple typique d'un entrepôt multi-blocks dont les carrés noirs représentent les emplacements de stockage que le robot doit visiter (b) Le graphe de l'entrepôt correspondant (les emplacements de stockage sont remplacés par des nœuds noirs (requis) et les croisements sont représentés par des nœuds blancs (de passage). [171]	53
2-16	Les mouvements possibles dans les allées verticales et transversales définis par <i>Ratliff</i> et <i>Rosenthal</i> [172]	55
2-17	Quelques stratégies de routage de prélèvement de commandes [171].	58
2-18	La fréquence d'utilisation des différentes stratégies de routage dans les systèmes manuels de préparation de commandes <i>picker-to-parts</i> [185].	59

2-19	Potentiel de gain en énergie en ne soulevant pas l'article requis d'un sommet lors du premier passage (contrairement à la solution 1, la solution 2 retarde le ramassage d'un article requis)	61
2-20	Potentiel de gain en énergie en changeant la direction dans laquelle les articles sont ramassés (la solution 2 tient compte de cet aspect alors que la solution 1 ne le fait pas (l'article <i>B</i> est plus lourd que l'article <i>C</i>))	61
3-1	Un exemple illustratif d'un plan de collecte et de dépôt	65
3-2	Exemple d'une carte métrique d'une manufacture et de l'espace de configuration du robot	68
3-3	La définition des nœuds et des arêtes d'un " <i>grid graph</i> " à partir d'une carte binaire	68
3-4	Un exemple de représentation d'un graphe par une matrice d'adjacence (b) ou une liste d'adjacence (c).	69
3-5	Représentation d'un graphe "distance" (à droite de la figure) à partir d'une carte <i>OGM</i> (à gauche de la figure (les cellules noires représentent les obstacles et les cellules blanches reflètent les zones de passage permises)).	70
3-6	Représentation d'un graphe "énergie" (a) Carte <i>OGM</i> avec les coûts de résistance au roulement des zones (colorées en gris dégradé selon la rugosité des surfaces) (b) Graphe pondéré avec le coût de la consommation énergétique	71
3-7	Processus de la méthode de planification globale avec considération du facteur de la résistance au roulement.	72
3-8	Exemples de représentation d'un graphe topologique (graphe de rue) de deux photos de manufactures réelles ((a)-(c) dont la photo (c) affiche des robots de systèmes FetchRobotics)[13] (b)-(d) Représentation des cartes topologiques correspondantes aux manufactures (respectivement) en marquant les positions d'intérêts avec des cercles rouges).	73
3-9	Schématisation du processus de définition de coefficient de pénalité pour un arc.	74
3-10	Effet du trafic sur la consommation énergétique [47].	74

3-11	Processus de la méthode de planification globale écoénergétique avec la considération des facteurs de la résistance au roulement et du trafic.	75
3-12	Un exemple de représentation d'un graphe topologique d'une manufacture.	76
3-13	Vue d'ensemble de la configuration d'un entrepôt (vue de dessus de la zone du stockage)	86
3-14	Aménagement d'un entrepôt et représentation du graphe (a) Définition des nœuds "intersections" (les cercles bleus) et des nœuds lieux du prélèvement (<i>picking locations</i>) (les cercles jaunes) (b) Définition des arcs de connexions et du graphe G_0	87
3-15	Un graphe $G(V,A)$ (nœuds rouges représentent les positions à visiter et les nœuds bleus représentent les intersections de passage)	90
3-16	Un exemple d'un nœud « <i>State</i> » avec une <i>pick-list</i> de taille 4.	94
3-17	Un exemple explicatif de la création des nœuds « <i>États</i> »	95
3-18	L'application d'un masque pour activer un bit qui correspond à un nœud donné dans une <i>pick-list</i>	97
3-19	(a) La procédure de la création de $state_1$ (b) La procédure de la création de $state_2$	97
3-20	Un exemple d'arborescence de création des nœuds d'états (façon naïve).	100
3-21	Dynamic states graph G'	102
3-22	Exemple de matrice d'adjacence d'états avec toutes les connexions possibles	103
4-1	Exemple de génération aléatoire d'environnements de manufactures (10 configurations différentes présentées par une grille 10*10 avec un retrait de 3 nœuds aléatoires et une sélection de 3 sommets d'intérêts aléatoires (en rouge)).	110
4-2	Informations sur l'environnement avec la structure du graphe en grille (a) graphe en grille avec l'information sur la distance (b) graphe en grille avec l'information sur la friction (la dégradation en bleue reflète la grandeur du coefficient de résistance au roulement).	110

4-3	Représentation des configurations pour les simulations : <i>layout 1</i> (a) <i>grid-graph</i> "résistance au roulement" de <i>Layout 1</i> (b) <i>grid-graph</i> "résistance au roulement" de <i>Layout 2</i> (c) <i>grid-graph</i> "résistance au roulement" de <i>Layout 3</i> .	111
4-4	Résultats des simulations <i>EE-GPP</i> des trois <i>layouts</i>	113
4-5	Résultats de simulations de 100 instances de <i>GPP</i> pour le <i>layout 1</i>	113
4-6	Exemple d'un scénario qui appartient au groupe 1 (a) <i>SP-GPP path</i> (b) <i>EE-GPP path</i>	114
4-7	Exemple d'un scénario qui appartient au groupe 2 (a) <i>SP-GPP path</i> (b) <i>EE-GPP path</i>	115
4-8	Exemple d'un scénario qui appartient au groupe 3 (a) <i>SP-GPP path</i> (b) <i>EE-GPP path</i>	115
4-9	Un aperçu des résultats de compilation de 100 instances de <i>GPP</i> et des coûts des méthodes <i>ECM-GPP</i> et <i>EE-GPP</i> versus <i>SP-GPP</i>	116
4-10	Informations sur l'environnement avec la structure topologique (a) graphe topologique avec l'information sur la friction (la dégradation en bleue reflète la grandeur du coefficient de résistance au roulement) (b) graphe topologique avec l'information du trafic (la dégradation entre le jaune et le rouge reflète la grandeur de la pénalité attribuée à chaque arc du graphe à cause du nombre moyen d'obstacles rencontrés (jaune :faible trafic, rouge :fort trafic)	119
4-11	Résultats de simulation de <i>EE-GPP</i> (avec considération d'un trafic élevé)	121
4-12	Résultats de simulation de <i>EE-GPP</i> (avec considération d'un faible trafic)	121
4-13	La configuration de l'entrepôt considérée pour les études de cas	123
4-14	Pourcentage de gain qu' <i>EE-OPR</i> donne versus les <i>SP-OPRs</i>	126
4-15	Comparaison des tours résultants après application les approches <i>SP-OPR</i> (a) et <i>EE-OPR</i> (b) appliquées sur différentes configurations.	127
4-16	Exemple de comparaison des tournées de ramassage de commandes résultants de l'approche <i>SP-OPR</i> (à gauche) et de l'approche <i>EE-OPR</i> (à droite)	129
4-17	Les différents éléments qui affectent l'efficacité de la stratégie de routages [186]	131
4-18	Les différents aménagements (configurations) possibles d'un entrepôt ou d'un DC (figure adaptée de [207](a) [208],(b)[171],(c)[209]	132

4-19	Comparaison entre les tours de prélèvement des commandes fournis par les deux approches <i>SP-OPR</i> et <i>EE-OPR</i> (l'effet de l'aspect <i>Direction</i>) (a) Configuration d'entrepôt de <i>RR (unibloc)</i> (b) Configuration d'entrepôt à deux blocs	135
4-20	Un exemple concret de chemin plus économe en énergie, mais qui est plus long du chemin le plus court	136
4-21	Un exemple concret de chemin plus économe en énergie, mais qui est plus long que le chemin le plus court (analyse).	137
4-22	Les avantages et les inconvénients de la méthode <i>ECM-OPR</i>	139
5-1	Rôle du planificateur local pour adoucir l'effet des virages (a) Chemin global dessiné en vert avec des angles droits (b) chemin local dessiné en jaune avec lissage des virages (point de départ en mauve et point d'arrivée en rouge) . .	144
5-2	Un exemple de solution de calcul d'un meilleur chemin plus court à l'aide de l'élimination des coins (un chemin le plus court qui utilise une distance rectiligne (tirets courts) par rapport à un chemin qui coupe les coins (tirets longs)))[211].	144
5-3	Un exemple de stratégie de stockage combiné (la ligne continue indique la répartition des types des produits sur les emplacements)[214].	149
5-4	Un exemple de stratégie de stockage <i>Midpoint</i> (la ligne continue indique la répartition des types des produits sur les emplacements) [214].	149
5-5	Deux solutions de chemins de prélèvement de deux articles représentés par des points en gras qui commencent et retournent au dépôt (en bas au centre.) [211].	150
5-6	(a) Représentation d'un entrepôt avec la configuration « fishbone » utilisant la méthode définie par Gue et al. [209] (b) Graphe de visibilité d'un entrepôt avec la configuration « fishbone »[211].	150
5-7	Explication de l'effet du temps sur les décisions du routage à travers un exemple de planification de chemin <i>source-target</i> sur une route urbaine utilisant <i>GoogleMap</i>	152

5-8 Carte colorée d'indicateur énergétique et son effet sur la décision de routage . 153

Liste des acronymes

AGV : Automated Guided Vehicle

AMHR : Autonomous Material Handling Robot

AMR Autonomous Mobile Robot

APF : Artificial Potential Field

DA : Dijkstra Algorithm

DOD : Depth Of Discharge

DP : Dynamic programming

ECM : Energy Consumption Minimization

ECM-GPP : Planification de chemin globale qui minimise seulement la consommation énergétique

ECM-OPR : Energy Consumption Minimisation of Order Picking Routing

EE : Energy Efficient

EE-GPP : Planification de chemin globale écoénergétique qui minimise le temps/distance et l'énergie

EE-OPR : Energy Efficient Order Picking Routing

FMS : Flexible Manufacturing System

GPP : Global Path Planning

LPP : Local Path Planning

MILP : Mixed-Integer Linear Programming

MPG : miles per gallon

OGM : Occupancy Grid Map

OP : Order Picking

OPP : Order Picking Problem

OPR : Order Picking Routing

PDP : Pickup and Delivery Plan

PP : Path Planning

PP : Path Planning

PPP : Path Planning problem

PRM : Probabilistic Roadmaps

RMFS : Robotic Mobile fulfillment systems

RR : Ratliff et Rosenthal method

RRT : Rapidly Exploring Random Trees

SKU : Stock Keeping Unit

SLAM : Simultaneous Localization and Mapping

SP-GPP : Planification de chemin globale qui minimise seulement le temps/distance parcourue

SP-OPR : Shortest Path Order Picking Routing

SPP : Shortest Path Planning

ST-PP : Source-Target Path Planning

STSP : Steiner Travelling Salesman Problem

TCAC : Taux de Croissance Annuel Composé

TSP : Travelling Salesman Problem

UGV : Unmanned Ground Vehicle

Liste des symboles

(i, j) : Arc qui part d'un noeud i vers un noeud j

β : pente de la route

$\delta^+(i)$: l'ensemble des arcs sortants du noeud i

$\delta^-(i)$: l'ensemble des arcs entrants du noeud i

Γ_i^+ : l'ensemble des arcs sortants du noeud d'état i

Γ_i^- : l'ensemble des arcs entrants au noeud d'état i

μ : coefficient de la résistance au roulement sur un arc (i, j)

$\overline{w}_{ob}^{d,t}(i, j)$: la médiane du nombre d'obstacles contournés sur l'arête (i, j) selon des données antérieures fournies par la base de données historique reflétant le degré de la congestion sur le segment de rue (i, j)

$\overline{w}_{tr}^{d,t}(i, j)$: le coefficient de pénalité attribuée à un arc donné selon le nombre d'obstacles rencontrés sur l'arc (i, j)

ρ : densité de l'air en kg/m^3

A' : ensemble d'arcs de transition entre les noeuds d'états

a_i : l'arrangement (le tuple) qui reflète le nombre et l'identité des articles déjà soulevés

A_v : surface avant du véhicule en m^2

A : l'ensemble des arcs du graphe G

c_e : le coût d'une unité d'énergie (joule) en dollar

c_t : le coût d'une unité de temps en dollar

C_d : coefficient de traînée

c_{ij} : coût de l'arc (i, j) en dollar des deux critères temps et énergie

$d_{(i.vertex\ j.vertex)}$: la distance qui sépare la position de l'état i et la position de l'état j

d : jour de la semaine

$e_{ij}(m_i^L)$: valeur de référence de l'énergie demandée pour voyager d'un nœud i vers un nœud j avec une masse donnée m_i^L

e_{ij} : énergie requise pour traverser l'arc (i, j)

F_a : force aérodynamique

F_g : force de gravité sur une pente de route

F_r : force de résistance au roulement

F_T : force de traction longitudinale dans les roues

G' : un graphe d'états pondéré (spatial et de collecte)

G_{OGM} : graphe correspondant à la carte de grille d'occupation (*Occupancy Grid Map*)

g : constante de gravité m/s^2

G : un graphe dirigé pondéré (spatial)

I : l'ensemble de nœuds d'intérêts (positions de ramassage ou de dépôt)

M_d^{OGM} : matrice d'adjacence d'une carte *OGM* pondérée avec le coût de distance

M_d^{topo} : matrice d'adjacence d'une carte topologique pondérée avec le coût de distance

M_e^{OGM} : matrice d'adjacence d'une carte *OGM* pondérée avec le coût de l'énergie consommée

M_i^L : la masse totale de l'AMR lorsqu'il quitte (*L: Leave*) le nœud i

M_i^R : la masse totale de l'AMR lorsqu'il atteint (*R: Reach*) le nœud i

M_r^{OGM} : matrice d'adjacence d'une carte *OGM* pondérée avec le coût de coefficient de résistance au roulement

M_r^{topo} : matrice d'adjacence d'une carte topologique pondérée avec le coût de coefficient de résistance au roulement

M_r : ensemble des valeurs des masses des AMHRs correspondant à R

m_t : masse totale du véhicule

q_i : pose de l'AMR (représentation ponctuelle)

R : ensemble d'AMHRs

t_{ij} : temps requis pour traverser l'arc (i, j)

t : plage horaire

V' : ensemble des nœuds d'états

v_w : vitesse du vent

v : vitesse du véhicule

V : l'ensemble des nœuds du graphe G

w_d : distance weight (le pondération en distance d'un arc)

w_e : energy weight (le pondération en énergie d'un arc)

w_r : roulement resistance (le pondération en résistance au roulement d'un arc)

x_{ij} : variable binaire qui indique si l'arc (i, j) appartient ou non au chemin optimal

Y : l'ensemble des masses m_i de tous les articles dans la localisation de stockage v_i en kg

Chapitre 1 - Introduction

1.1 Contexte général

L'intelligence des industries est une composante principale de la quatrième révolution industrielle (*Industrie 4.0*). Durant les prochaines années, les futurs processus de fabrication, de production et d'entreposage devraient changer en adoptant de nouveaux concepts tels que l'interopérabilité, la virtualisation, la décentralisation, l'orientation vers les services et la modularité, offrant ainsi une meilleure souplesse de déploiement, mobilité et extensibilité [1]. Afin de répondre à ces « exigences 4.0 » et aux conditions du marché, le niveau d'automatisation dans le secteur industriel et d'entreposage ne cesse d'augmenter, notamment suite à la croissance accrue du commerce électronique, la personnalisation de masse et la pénurie de main d'œuvre [2]. En effet, le besoin de robots flexibles pouvant être rapidement reprogrammés pour effectuer de nouvelles tâches a encouragé les entreprises à investir dans des solutions robotisées pour l'automatisation des systèmes de transport intralogistique. D'où l'apparition des solutions robotisées de véhicules de transport intralogistiques appelées AMR (*Autonomous Mobile Robot*) qui présentent une nouvelle génération plus intelligente d'AGV (*Automated Guided Vehicle*). Comparativement aux AGV traditionnelles, principalement filoguidés ou attachés à des installations fixes rigides aux changements, les AMR offrent une meilleure efficacité, flexibilité et soutien aux opérateurs humains. Présentement, des centaines de fournisseurs dans le monde entier proposent des AMR [3] et le marché de ces systèmes monte en flèche à cause de la demande croissante d'automatisation dans le commerce électronique, l'industrie automobile, la santé et autres. Selon *MarketsandMarkets*, le marché de la robotique de service devrait passer de 36,2 à 103,3 milliards de dollars américains entre 2021 à 2026 avec un taux de croissance annuel composé (TCAC) de 23,3 % [4]. Cette évolution est accompagnée par de nouveaux défis en termes d'optimisation de l'utilisation de ces systèmes,

étant donnée leur complexité accrue. C'est la raison pour laquelle plusieurs chercheurs et entrepreneurs se focalisent sur l'amélioration de l'agilité des AMR (leur capacité à s'adapter de manière autonome à l'environnement de navigation) à travers l'optimisation de la gestion des tâches des AMR, de leurs routages, de leurs gestions énergétiques et de leurs interactions avec les opérateurs humains [5].

1.1.1 AGV versus AMR

La tendance de l'automatisation des véhicules de transport dans l'industrie n'est pas nouvelle. En effet, le premier AGV a été développé par Arthur M.(Mac) Barrett, en 1954, pour l'utiliser dans son industrie, Barrett Electronics [5]. Ce véhicule naviguait en suivant les fils montés au plafond de l'usine [6]. De même, les notions de la robotique mobile sont connues depuis des années, avec l'invention du premier robot mobile par Paul J. Matarboni en 1987 [7]. Cependant, l'exploitation de la maturation des technologies utilisées principalement dans le domaine de la robotique mobile pour développer des AGV plus intelligents est relativement récente, notamment les technologies de navigation, de localisation, de détection et de contrôle du mouvement [8]. De façon générale, un AGV est un véhicule sans conducteur utilisé pour le transport intérieur des marchandises ou des palettes entre les différents emplacements d'une manufacture et/ou entrepôt [9]. Les techniques de navigation des systèmes AGV ont progressivement évolué d'une navigation qui utilise un guidage électromagnétique, mécanique, optique, inductif ou inertiel, à une navigation basée sur le laser et la vision et dernièrement à une navigation dite « naturelle » utilisant de multiples technologies [10]. Cette évolution croissante permet une meilleure habileté, une plus grande flexibilité, augmente la capacité des robots à apprendre des tâches sans programmation formelle et à collaborer de manière autonome avec d'autres dispositifs autonomes et des opérateurs humains [11]. Dans le domaine de la manutention industrielle, le terme AMR désigne généralement la nouvelle génération des AGV avec une navigation naturelle. Cependant, certains documents utilisent, encore, le terme AGV pour les nommer, étant sa signification populaire dans l'industrie (un véhicule de manutention intralogistique veut dire AGV) [3]. Le tableau 1-1 présente quelques points de différences entre les AGV et les AMR, qui commence d'ailleurs par leurs noms *Automated* (gérer par un automate) versus *Autonomous* (autonome) et *Guided* (guidé) vs. *Mobile* (mobile).

	AGV	AMR
Nom	<i>Automated Guided Vehicle</i>	<i>Autonomous Mobile Robot</i>
Navigation	Navigation basée sur une infrastructure de guidage (électromagnétique, optique, inductif, inertiel, laser, etc.)	Utilise un ordinateur de bord
Flexibilité	Sensible aux changements : un changement d'aménagement ou d'emplacement de matériaux entraîne des périodes d'inactivité et risquerait des pertes économiques et des baisses de productivité. Complexité d'ajouter des nouvelles routes ou destinations	Flexible : s'adapte rapidement aux changements de l'environnement d'exploitation.
Gestion d'obstacles	S'arrête devant les obstacles	Contourne l'obstacle et retrouve son chemin selon sa carte locale
Précision	Collé à un chemin, l'AGV est plus fiable	Plus sensible aux variations de l'environnement (risque de perdre sa position)

TABLEAU 1-1 Les points de différence entre les AGV et les AMR [11].

1.1.2 Types des AMR et utilisations

L'utilisation des systèmes AMR a augmenté considérablement dans les milieux industriels et manufacturiers, mais aussi dans les milieux logistiques et d'entreposage. La Figure 1-1 montre un système de fabrication flexible FMS (*Flexible Manufacturing System*) qui utilise des AMR pour transporter des matériaux dans une ligne de production aidant à la livraison «*Just in time*» (des AMR standard et des «*Roller top- AMR* » dans une infrastructure des convoyeurs). En outre, la figure 1-2 illustre quelques variétés d'AMR, sur le marché, utilisées pour optimiser la préparation de commande (*Order Picking* (OP)). On distingue deux grandes catégories de ces deniers. La première regroupe les « *rack-moving* » AMR utilisés dans des systèmes « *parts-to-picker* » dits aussi « *goods-to-person* », tels que les « *puzzle-based storage systems* » et les « *Robotic Mobile fulfillment systems* » (RMFS) comme le populaire système *Kiva* de Amazon. La deuxième catégorie comporte les AMR utilisés dans des systèmes

Les AMR sont utilisés aussi dans d'autres environnements tels que les hôpitaux, les terminaux à conteneurs, les hôtels, les stationnements, etc. Nous sommes intéressés par les AMR standard pour les déplacements horizontaux et les *Ps-AMR* pour le ramassage

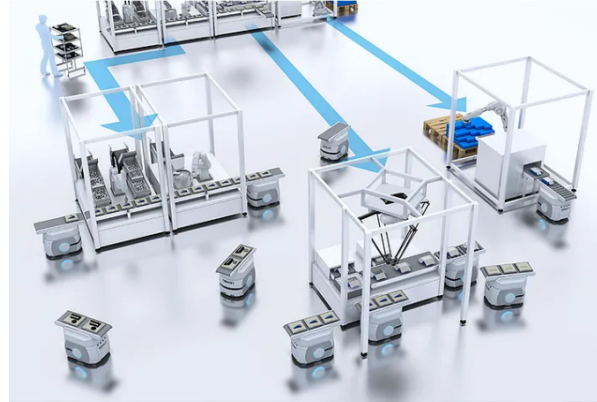


FIGURE 1-1 Exemple d'AMR utilisées dans un FMS [12].

des commandes, particulièrement, l'optimisation de leur navigation. Nous allons utiliser, dans ce document, le terme AMR pour désigner tous types de véhicules/robots autonomes de manutentions (*AMHRs (Autonomous Material Handling Robots)*), tel que les chariots élévateurs robotisés (*autonomous forklift*), AGV intelligent, base de robot mobile standard, etc). Par exemple, la figure 1-3 montre différents types d'AMR utilisés pour des applications manufacturières et d'entreposage.

1.1.3 Flexibilité accrue et nouveaux défis émergents

Au cours des dernières années, le commerce électronique a connu une croissance exponentielle amplifiée par l'effet de la pandémie de COVID-19. Selon *Salesforce*, le nombre d'acheteurs uniques en ligne a augmenté de 40% durant le premier trimestre de 2020 [19]. Ce bondissement augmente le besoin d'automatisation de la manutention au sein des entrepôts et DC et pousse les entreprises à suivre le rythme et à progressivement migrer vers des mécanismes robotisés pour l'exécution de commandes (*order fulfillment*) [20]. Le rapport de marché pour l'année 2022 réalisé par *LogisticsIQ* estime qu'à l'échelle mondiale, le marché de l'automatisation des entrepôts devrait atteindre 41 milliards de dollars d'ici 2027 [21]. De nombreuses solutions robotisées ont été adoptées. Certaines entreprises ont opté pour des solutions entièrement automatisées, avec des investissements élevés, un temps de mise en place important et des moyens peu flexibles. D'autres ont opté pour des pratiques partiellement automatisées qui offrent des alternatives relativement moins coûteuses, avec une flexibilité et une évolutivité élevée [22]. Les solutions partiellement autonomes sont de nos jours les

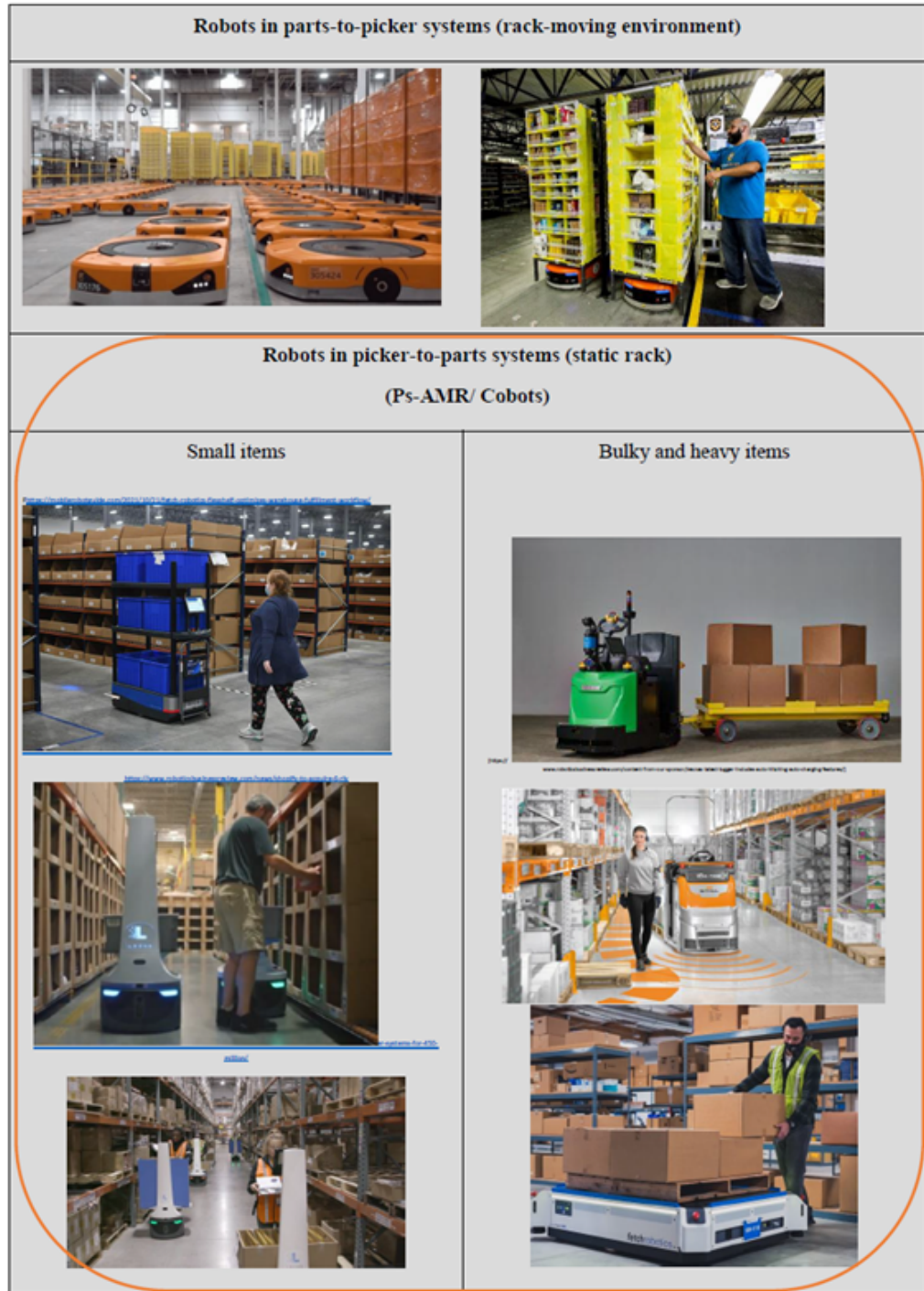


FIGURE 1-2 Exemples de AMR utilisés pour les opérations de préparations de commandes[13]-[17].

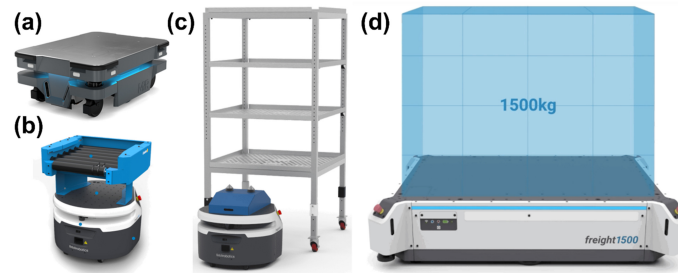


FIGURE 1-3 Différents types, formes, masses et capacités d'AMR [18].

plus tendance et de plus en plus demandés [23]. En effet, ces robots peuvent collaborer avec les humains et peuvent être utilisés dans des entrepôts déjà fonctionnels avec peu de temps d'intégration et sans besoin d'une infrastructure spéciale dans l'environnement de travail. Sauf que cette flexibilité augmente les exigences en termes de la sécurité dans l'environnement dynamique, la collaboration avec les humains et la consommation énergétiques [1]. Par exemple, étant donnée leur nouveauté, les Ps-AMR commencent à susciter l'intérêt de plusieurs chercheurs et plusieurs changements ou adaptations de quelques pratiques industrielles actuelles devront être considérés [24]. Cette constatation est confirmée par plusieurs chercheurs. Par exemple, Boysen et al. [25] proposent la réévaluation des affectations de stockage traditionnelles qui proposent généralement le stockage des items à rotation rapide à proximité du dépôt. Cette stratégie n'est pas nécessairement la meilleure dans les systèmes collaboratifs [24]. Azadeh et al. [24] à leur tour témoignent qu'il n'existe que quelques articles de recherche sur les Ps-AMR malgré la diversité des possibilités de recherches. Également, Meller et al. [26] mentionnent que la dynamique de la collaboration homme-robot est complexe et des recherches supplémentaires sont nécessaires. En outre, Vaidya et al. [27] affirment qu'avec la migration 4.0, de nouveaux axes de recherche devraient être envisagés au niveau de la collecte des données sur les lignes de production et l'optimisation de leur utilisation pour améliorer l'efficacité des machines, la gestion transparente de la chaîne d'approvisionnement et de l'industrie, l'optimisation de la maintenance et l'efficacité énergétique.

1.1.4 Aspect économique et écologique

L'intégration des AGV/AMR dans les systèmes de fabrication et d'entreposage existants offre un potentiel économique important à travers l'accroissement de la productivité ainsi que

la réduction des coûts d'accidents par l'amélioration de la sécurité de travailleurs, l'économie de coûts de main-d'œuvre, et l'abaissement de la consommation énergétique et des émissions de gaz à effet de serre [28]. D'ailleurs, certains travaux de recherche existants se concentrent sur les aspects économiques de l'application des AGV/AMR. Ces travaux visent à étudier et à fournir des solutions AGV/AMR qui contribuent à la croissance de l'entreprise, qui correspondent aux opérations de leurs solutions et qui offrent un retour sur investissement (ROI (*Return on investment*)) significatif. Cependant, un facteur crucial, peu exploré dans la littérature et qui devrait être pris en compte lors de l'utilisation des AMR est le coût énergétique de l'exploitation du véhicule ainsi que le coût de la maintenance de la batterie et de son échange, notamment lorsque la batterie atteint sa fin de vie. Ce facteur est essentiel pour l'adoption des AMR et, en particulier, pour l'amélioration du coût de déploiement de cette technologie. De plus, les exigences politiques concernant la réduction des émissions de gaz à effet de serre et le prix de l'énergie qui ne cesse d'augmenter rendent la réduction de la consommation d'énergie de ces robots industriels un objectif amplement souhaitable qui sollicite de plus en plus l'attention de non seulement les utilisateurs industriels, mais aussi des chercheurs [29]. Étant un véhicule électrique, l'AMR est confronté à de nombreux défis en termes d'autonomie de batterie et de coûts de manutention et/ ou de changement de cette dernière à la fin de sa vie [30]. De plus, un AMR utilise plusieurs batteries tout au long de sa vie, ce qui engendre un coût total important de ces batteries. Parmi les facteurs majeurs qui influencent la réalisation et la durée d'une mission d'un AMR est, assurément, son autonomie de batterie qui doit être changée ou chargée durant leur fonctionnement [8]. Le temps de changement et de recharge est considéré comme une perte de productivité. Ainsi, l'amélioration de l'efficacité énergétique (*Energy Efficiency (EE)*) des AMR a plusieurs bénéfices à court et à long terme tels que (i) la réduction de la consommation de l'électricité (ii) l'accroissement de leurs durées de fonctionnement productives et de leurs disponibilités (iii) la réduction des coûts de maintenance et (ou) de changement des batteries, du nombre de batteries utilisées le long d'une vie d'un AMR et des déchets néfastes causés par ces batteries [31]. Ces points sont expliqués davantage, Ci-dessous : (i) les véhicules de manutention industriels, en général, consomment une quantité notable d'énergie. Par exemple, les chariots élévateurs (*forklifts*) sont connus comme les équipements de manutention les plus énergivores utilisés

dans les entrepôts [32]. (ii) L'amélioration de l'efficacité énergétique des AMR prolonge leur temps de fonctionnement productif entre les arrêts de recharge (autonomie et disponibilité). Cette amélioration peut également augmenter le nombre de tâches de maintenance effectuées par jour, diminuer la congestion sur les stations de recharge (s'il y a lieu) et elle peut même diminuer le nombre de robots nécessaires dans l'industrie. (iii) Toute amélioration qui réduit la consommation d'énergie de l'AMR entraîne la réduction de sa profondeur de décharge DOD (*Depth Of Discharge*) ce qui aide à améliorer sa durée de vie [33], [34] et à retarder l'effet de la baisse de sa capacité maximale. En effet, il est admis que la dégradation des batteries des AMR provoque une perte de performance graduelle [35] et une autonomie réduite [36], [37]. Ce phénomène est dû, généralement, à leurs chimies, mais également à leur manière d'utilisation, aux facteurs environnementaux [38]-[40], sans oublier le phénomène de temps appelé usure calendaire. Par exemple, la courbe présentée dans la Figure 1-4 montre le cycle de vie expérimental des batteries au *lithium-ion* en fonction du pourcentage de décharge [33], [41]. On y remarque, clairement, l'influence majeure de la DOD sur le nombre de cycles de vie et par conséquent sur la durée de vie de la batterie. Ainsi, une mauvaise utilisation des batteries des AMR rend leur durée de vie bien inférieures à celle attendue par les consommateurs [42], [43] et accélère l'effet régulier de la diminution de sa capacité maximale et le contraire est également vrai.

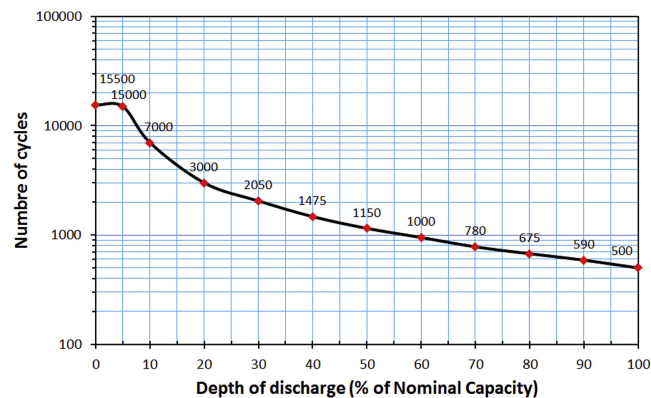


FIGURE 1-4 Cycle de vie expérimental de la batterie au lithium-ion en fonction du DOD [34]

Grâce à leur fiabilité et leur coût relativement faible, les batteries au plomb-acide sont les plus utilisées dans les AGV [44]. D'autres types de batteries, tels que les batteries au nickel-cadmium et au lithium-ion, présentent certains avantages par rapport aux batteries

au plomb (par exemple, un temps de charge plus court et un poids plus léger), mais avec des coûts quatre à cinq fois plus élevés qu'une batterie au plomb de capacité similaire [45]. L'amélioration de l'efficacité énergétique de l'AGV/AMR réduit la nécessité d'utilisation des batteries très chères ou aide à maximiser les bénéfices d'utiliser de telles batteries. Cette fin peut être atteinte à travers la gestion des différentes technologies du robot, dont la planification de chemin qui présente un point important de la recherche. Cependant, le processus de planification de chemins est de plus en plus challengeant à cause de la flexibilité croissante des manufactures et des entrepôts (voir figure 1-5). En effet, comme le montre la figure 1-5, le développement industriel et l'industrie 4.0 offrent plusieurs avantages et technologies pour une meilleure adaptabilité. Cependant, ces changements provoquent des variations importantes de l'environnement du robot et il est indispensable de les considérer lors des décisions de routage des AMR. En effet, mise à part la distance, il existe plusieurs facteurs qui affectent la consommation énergétique de ces véhicules tels que la vitesse, la masse transportées, la friction au sol, le trafic, etc.

Par exemple, la force de résistance au roulement présente la force dominante des ensembles des forces que le robot doit vaincre et qui doit être sérieusement considéré. En effet, étant dans un contexte intérieur, la force aérodynamique et la force de gradient sont négligeables. La résistance au roulement est susceptible de varier vu la largeur des surfaces de l'industrie moderne avec des surfaces différentes dans certains et des environnements salissants dans d'autres (ex. les manufactures et les entrepôts alimentaires). L'autre facteur dominant dans la force de résistance est la masse. D'autre part, les AMR doivent, quotidiennement, transporter des lourdes charges avec un large assortiment d'articles, marchandises et matières premières avec des demandes variables. En outre, les articles peuvent changer de position dans les manufactures et les entrepôts modernes ou encore être remplacés par d'autres articles. Cette variation de type d'articles et de leur masse aura un effet sur l'énergie nécessaire pour effectuer ses missions étant donné que le besoin en puissance augmente avec l'augmentation du poids des charges utiles transportées [46]. En outre, la croissance du commerce électronique et l'individualisation complexifient encore la situation avec des possibilités de fluctuations de la demande de quelques produits et avec la naissance des entrepôts à forte mixité.

De plus, le changement d'aménagement et l'extension des superficies des environnements manufacturiers engendrent un changement dans la carte routière et des distances à parcourir.

D'autre part, un AMR moderne doté d'un niveau haut d'autonomie, contrairement à un AGV traditionnel, n'a plus des voies réservées pour leur navigation. Il partage l'espace avec d'autres véhicules autonomes ou non autonomes, avec des opérateurs humains et il travaille en étroite collaboration avec les opérateurs humains. Ce qui rajoute un facteur important à prendre en considération qui est le trafic, car, pendant ses missions de transportations, le robot devrait éviter des obstacles dynamiques ou s'arrêter dans certaines situations, sur sa trajectoire prévue. Ces décélérations et accélérations augmentent ses pertes énergétiques [47], [48], pourrait fausser les estimations de son besoin de recharge et par suite décharger agressivement la batterie.

Toutes ces observations nous amènent à mettre en question les pratiques et les algorithmes de routage existants, spécialement lorsqu'on s'intéresse à l'économie énergétique. Cette constatation est aussi tenue par Koster et al. [49] qui expriment la nécessité de proposer de nouveaux modèles de conception et de gestion alors que de nouvelles questions surgissent avec l'essor des technologies modernes. En effet, les éventuels changements que nous avons cités ci-dessus donnent de nouvelles possibilités de chemins non explorés étant donné les nouvelles variations des conditions de routes. Ces dernières peuvent carrément fausser les prédictions pour la planification de recharge du robot qui peut attribuer au robot de longues missions et par suite décharger profondément sa batterie et réduire sa durée de vie.

1.1.5 Planification de chemin des AMR

Les décisions opérationnelles relatives aux AMR, particulièrement leurs routage et planification de chemin (*Path Planning (PP)*), peuvent considérablement influencer la quantité d'énergie requise pour leur fonctionnement et par la suite affecter leur autonomie énergétique. Les points de mérites et de démérites des résultats de ces techniques de planification de chemin affecteront directement l'efficacité énergétique de l'AMR, notamment à cause de la fréquence élevée de la prise de décision de routage par jour.

L'AMR, tel qu'un humain, peut faire ses propres décisions et effectuer des actions en conséquence. Il reçoit des tâches de transport à destination unique ou multideestination depuis

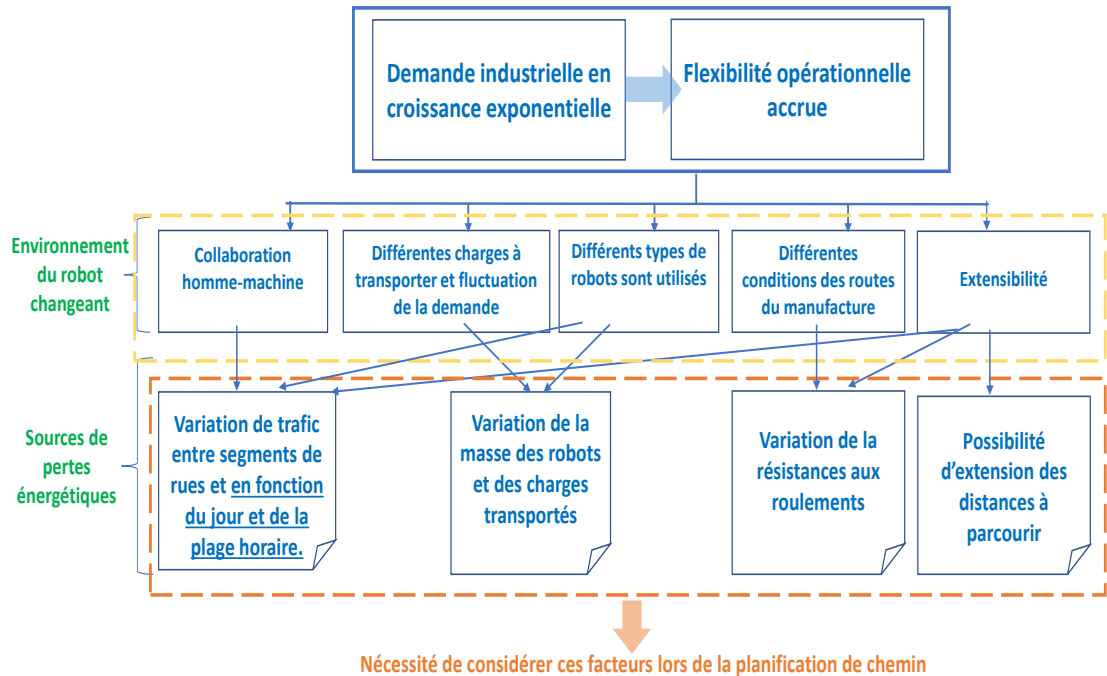


FIGURE 1-5 Effet de développement des environnements industriels et de la croissance de leur flexibilité sur la demande énergétique pour les déplacements des AMR

un « *Host* ». Ensuite, afin de pouvoir naviguer d'une manière autonome dans son environnement de travail, il utilise plusieurs technologies. L'AMR commence par percevoir son environnement à l'aide de différents capteurs et équipements (tel que le laser, lidar, caméra, etc.) pour construire sa carte d'environnement locale et sa carte d'environnement globale (ou la mettre à jour), tout en s'y localisant. Ce processus est connu sous le nom *SLAM* pour « *Simultaneous Localization and Mapping* », en anglais. Par la suite, ayant la carte, la position du robot, et la commande qui spécifie la prochaine position destination, une planification de chemin globale (*GPP* (*Global Path Planning*)) est requise. Ce processus est responsable de trouver un chemin continu, sans impasse ou collision entre une position de départ jusqu'à une position cible, dans une carte donnée [50]. Ce chemin, constitué de suite de points à visiter, est attribué au planificateur local (*LPP* (*Local Path Planning*)) qui cherche une séquence de mouvements valide pour que l'AMR suive le chemin tracé par le *GPP*, tout en adaptant ses mouvements à des environnements dynamiques. La séquence de mouvement est traduite par des commandes de vitesse attribuées aux actionneurs du robot. La Figure 1-6 résume le processus décrit. Chaque processus représenté dans cette figure révèle des défis de développements avec un degré de difficulté différent, dépendamment de type d'applications et

le motif d'utilisation de l'AMR. Par exemple, dans un environnement d'usine, le robot doit transporter des biens d'une station à l'autre. Cet environnement ne dispose pas d'un réseau routier comme les cartes urbaines qui sont définies par des boulevards, croisement, rond-point, etc., particulièrement lorsqu'il y a plusieurs espaces ouverts. Par suite, l'AMR doit suivre un chemin qui satisfait plusieurs propriétés liées à la fois aux contraintes du véhicule et à la morphologie de l'environnement dans lequel il navigue, tout en gardant le robot à une distance de sécurité des obstacles à tout moment. D'où l'importance de l'optimisation du planificateur global *GPP*. D'autre part, dans d'autres environnements avec une structure particulière, tels que les entrepôts et les *DC(Distribution Centers)*, les allées verticales parallèles et les allées transversales facilitent la représentation topologique de l'espace qui ne représente plus un enjeu majeur. Cependant, une planification multidestination est requise pour résoudre le problème de routage d'un *Ps-AMR* lors du processus de préparation de commandes. En effet, la tâche est plutôt de visiter plusieurs destinations pour collecter des commandes clients depuis leurs emplacements de stockage, à partir d'une position initiale vers une position prédéfinie avec un coût de déplacement minimal. Il s'agit du problème de préparation de commande appelé *OPP (Order Picking Problem)* ou *OPR (Order Picking Routing)*. Ce problème est défini comme un cas particulier du problème du voyageur de commerce (*Travelling Salesman Problem (TSP)*) qui est un problème d'optimisation combinatoire *NP-Hard*.

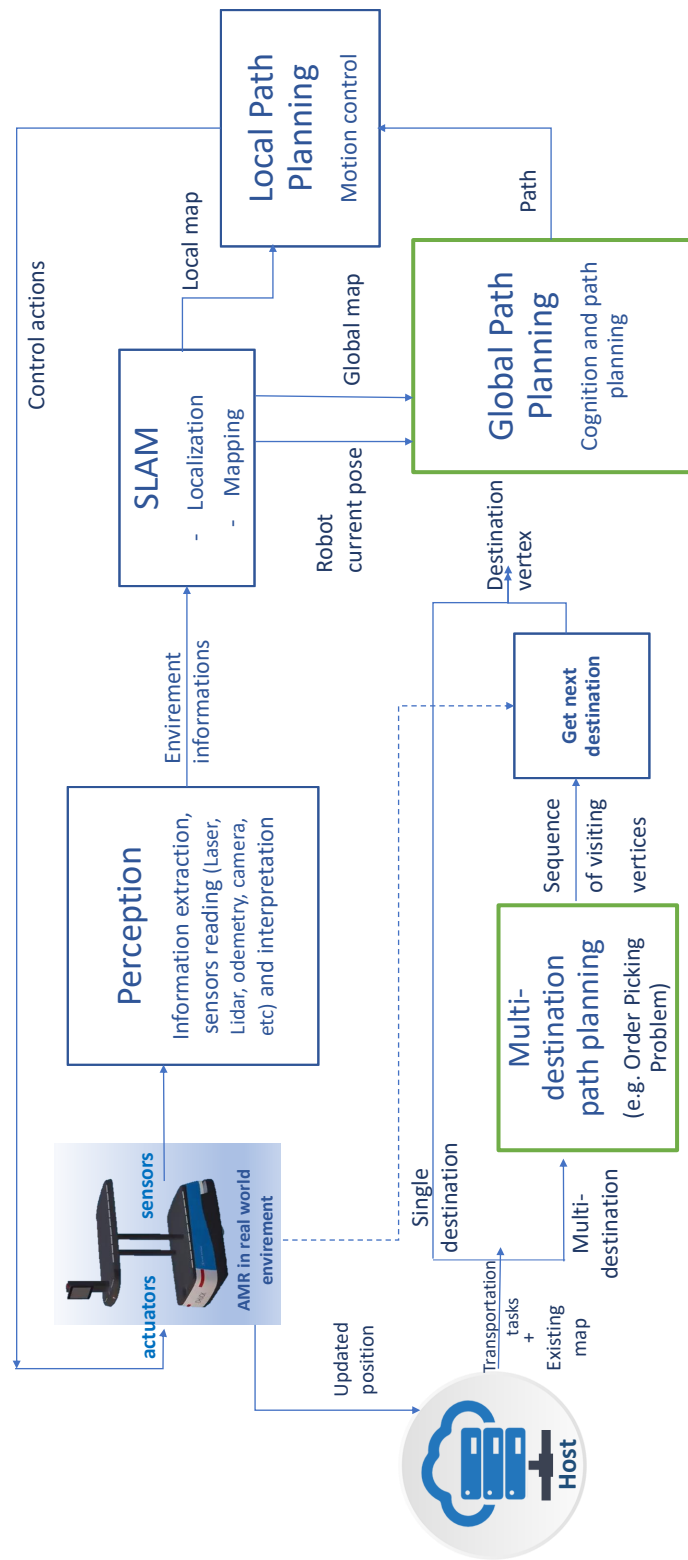


FIGURE 1-6 Diagramme de processus de navigation d'un robot mobile

1.2 Problématique de recherche

L'endurance opérationnelle des AMR est limitée par la quantité d'énergie stockée dans leurs batteries. Par conséquent, la minimisation de la consommation énergétique de ces robots est très importante pour effectuer des missions plus étendues sans recharge. Pour cette raison et pour des préoccupations environnementales, de nombreuses recherches ont été menées sur l'amélioration de l'efficacité énergétique des AMR [51] à travers différents moyens, tels que l'optimisation et le contrôle des sources énergétiques, l'évitement des accélérations et des décélérations soudaines [52], l'optimisation de recharges ainsi que l'optimisation de la planification de chemin.

La majorité des travaux qui considèrent l'*EE* via la planification de chemin sont des articles qui traitent particulièrement la planification de chemin locale (*LPP (Local Path Planning)*), c'est-à-dire l'amélioration des commandes des vitesses des AMR (contrôle de mouvements *Motion planning*). Rappelons que la planification du mouvement est le processus qui définit un ensemble d'actions à exécuter par les actionneurs du robot pour suivre un chemin planifié par le planificateur global. Cependant, l'aspect énergétique de la planification de chemin globale (*GPP*) est généralement omis [53]. Pourtant, le *GPP* représente un élément clé dans le mouvement du robot qui est largement étudié dans le domaine de la recherche sur la navigation des robots mobiles [54] et plusieurs algorithmes matures de *GPP* sont proposés. Ces derniers supposent par défaut que les besoins énergétiques de l'AMR sont approximativement proportionnels au coût de la distance parcourue et que le chemin optimal en termes de la distance est approximativement le chemin optimal en termes d'énergie [55]. Ils sont basés sur des indicateurs de performance de plus courte distance ou de plus courte durée, ignorant l'aspect énergétique. Ces pratiques ont tendance à écarter des chemins plus économes en énergie et ils peuvent, dans quelques situations, imposer des décharges trop profondes des batteries des AMR, réduisant leurs durées de vie. Rares sont les travaux qui considèrent la consommation énergétique dans la *GPP*. Ces dernières considèrent des robots utilisés à l'extérieur dans des environnements tout-terrains (*off-road* environnement) (appelés dans quelques articles *Autonomous Ground Vehicles*) et ils focalisent sur le facteur d'élévation du terrain qui varie et qui doit être considéré pour une meilleure planification économe en

énergie, dans ce contexte. Or, dans un contexte où les AMR sont utilisés à l'intérieur ou entre bâtiments de l'industrie, la surface est généralement plate. Cependant, il existe d'autres facteurs de l'environnement qui peuvent influencer l'énergie consommée par le robot pour parcourir son chemin et il serait intéressant d'investiguer ce champ et analyser s'il existe des chemins alternatifs plus économes énergétiquement que le chemin de plus court (*SP*).

Par la suite, afin de comprendre comment évolue la consommation d'énergie d'un véhicule sur les différentes possibilités de chemins, il est important d'analyser les paramètres qui l'influencent. Ces derniers peuvent être classés en deux catégories : (1) paramètres intrinsèques qui sont liés à la construction de la plate-forme, le type d'énergie de traction, l'architecture du train de propulsion, la structure mécanique et aérodynamique, etc. ; et (2) paramètres extrinsèques qui dépendent de l'interaction *AMR-environnement* de roulement, du contrôle des mouvements de l'AMR et des tâches de manutention à faire (par exemple, un déplacement à vide consomme 40 Ah, un déplacement à charge 60 Ah et une décélération 20 Ah [45]). Les paramètres de la première catégorie sont incontrôlables pour une meilleure efficacité énergétique, à moins de changer d'équipements. Cependant, la considération des facteurs de la deuxième catégorie lors de la planification de chemin de l'AMR peut apporter une économie énergétique importante. Par exemple, à basse vitesse de mouvement, la force de résistance au roulement domine l'ensemble des forces que l'AMR doit vaincre pour avancer. Les changements possibles dans les configurations de surface de l'environnement du robot (dans le cas des grandes surfaces d'usines, des industries alimentaires ou le transport entre bâtiments) sont des sources de variation de l'énergie (augmentation). En outre, la flexibilité croissante des systèmes AMR et leurs collaborations avec d'autres robots et avec des opérateurs humains, augmente les risques de congestions et la nécessité de contourner continuellement des obstacles (de différentes structures et formes), ce qui engendre des pertes énergétiques considérables inattendues (engendrées par les accélérations, décélérations, rotations continues et autres et amplifiées avec les facteurs masse et friction). Par ailleurs, lorsque le robot doit effectuer un tour de travail et visiter plusieurs destinations en ajoutant ou enlevant des charges de masses différentes (tel que le problème d'OPR). Cette variation a un impact majeur sur l'énergie nécessaire pour effectuer sa séquence de manutention. Le graphique représenté dans la figure 11-7 montre l'effet inévitable de l'augmentation de la masse d'un

véhicule conventionnel (en tonnes)) sur la distance (en *miles*) qu'il peut parcourir par gallon de carburant (*MPG (Miles Per Gallon)*). Les facteurs décrits ci-dessus sont souvent omis dans

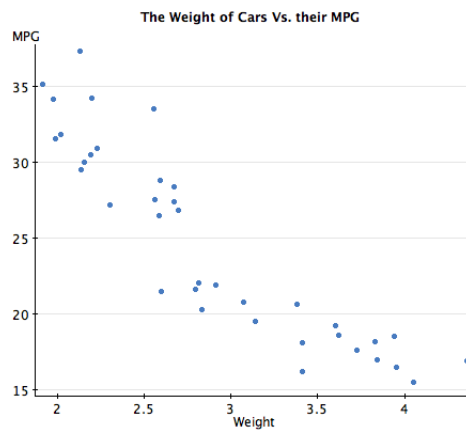


FIGURE 1-7 Le rapport entre le poids d'un véhicule et le nombre de kilomètres parcourus [56].

les travaux qui traitent le problème de planification de chemins des AMR dans la littérature et par conséquent leurs formulations de problème d'optimisation ne sont pas adaptées à les incorporer avec un modèle de consommation énergétique plus précis.

La planification du chemin requiert, premièrement, l'établissement d'un modèle environnemental et deuxièmement, une bonne stratégie de planification du chemin. Communément, la première phase fait référence à l'établissement d'une carte de l'environnement ou d'un modèle spatial incluant les obstacles statiques, la position du robot, les panneaux de signalisation, etc. Cette carte, construite à l'aide des données fournies par les capteurs locaux du robot et des informations de l'utilisateur, est souvent limitée à deux dimensions, ou bien rarement, à trois dimensions (dans le sens géométrique du mot) [57], [58]. Ces informations sont suffisantes pour que le robot planifie un chemin plus court entre le point de départ et le point cible sans collision, dans ce modèle. En revanche, ces informations ne sont pas nécessairement suffisantes pour un choix de chemin énergétiquement économe, mais des informations supplémentaires sont indispensables pour une bonne prédiction des coûts énergétiques relatifs aux différents chemins alternatifs. En effet, mis à part le facteur distance, plusieurs facteurs affectent la consommation énergétique de l'AMR lors de l'exécution de ces tâches (telles que) la rugosité et la pente de la chaussée, le contrôle des actionneurs, la masse des charges transportées et les conditions générales de l'environnement. Par ailleurs, les avancées récentes dans des technologies telles que l'internet des Objets (*IoT*), les données massives (*big data*), l'informatique

dans les nuages (*cloud computing*) et l'intelligence artificielle (*IA*) rendent disponible des informations relatives aux conditions générales de l'environnement, les masses des produits manutentionnés et peuvent modéliser le comportement dynamique du trafic sur les segments des rues (ce qui n'est pas traité dans cette thèse). Ainsi, l'exploitation de ces informations est indispensable pour avoir une estimation précise de la consommation d'énergie du véhicule et prendre des décisions de *PP* plus écologique.

Il est aussi important de mentionner que dans certains contextes, nous avons constaté que le chemin économe en énergie (*Energy Consumption Minimization (ECM)*) peut être plus long que le chemin le plus court (*Shortest Path (SP)*) que ce soit pour le problème de *GPP* ou de *OPR*. Par exemple, la Figure 1-8 représente deux solutions possibles de résolution d'un problème simplifié de *GPP* qui essaye de trouver un chemin sans collision à partir du point vert jusqu'au point rouge de l'environnement (la forme noire représente un obstacle statique). Le chemin de la solution 1, représenté dans la figure, utilise moins de temps de voyage, mais consomme plus d'énergie. Au contraire, la solution 2 consomme plus de temps, mais moins d'énergie. En effet, étant donnée qu'il existe deux surfaces différentes, la solution 2 contourne la surface rugueuse et sélectionne plutôt un chemin moins énergivore. Cependant, ce dernier est plus long que le chemin donné par la solution 1.

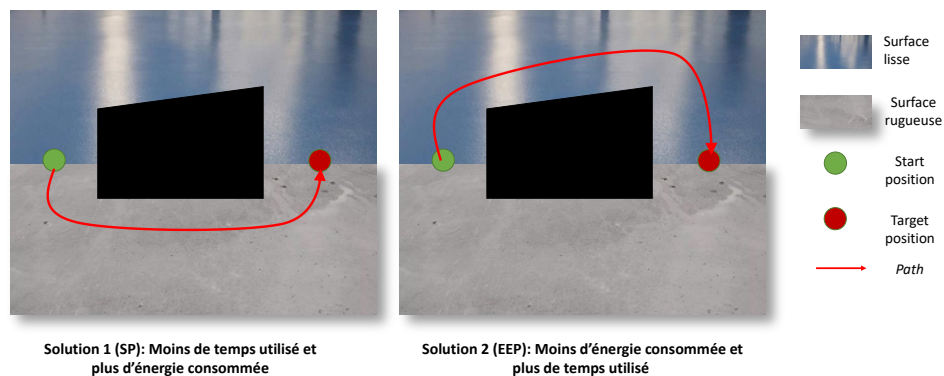


FIGURE 1-8 Un exemple simplifié de planification de chemin globale démontrant l'importance d'aborder le compromis entre le temps et l'énergie

De même, pour le problème du *OPR* (multidestinations), on présente un exemple dans la figure 1-9 dont le problème est de trouver la tournée optimale qui part du sommet *A* et arrive au sommet *E* tout en ramassant les articles dans les sommets *C* et *D*. La partie gauche de la

figure (solution 1) montre le chemin le plus court, mais qui consomme plus d'énergie que celui illustré à droite de la figure (solution 2) qui consomme moins d'énergie, mais qui est plus long. En effet, la *ECM* évite de traîner 80 kg le long de 8 mètres , contre un déplacement de 4 mètres de plus sans charges. Si un trajet un peu plus long permet d'économiser de l'énergie, il est peut-être intéressant pour des économies à long terme, surtout pour les grandes superficies. Il est donc intéressant de trouver un compromis entre le gain en termes du temps et le gain en

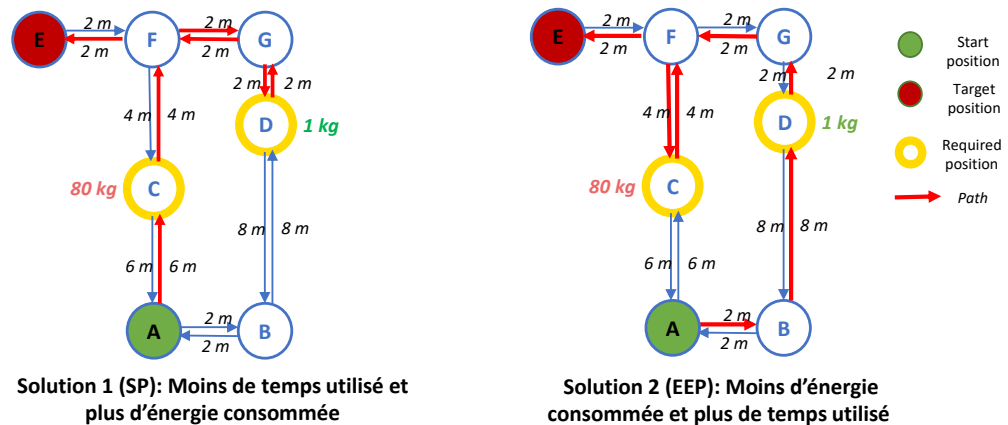


FIGURE 1-9 Un exemple simpliste de routage de ramassage de commandes démontrant l'importance d'aborder le compromis entre le temps et l'énergie

termes d'énergie. En outre, dans certaines situations, il est nécessaire de préciser les priorités des deux critères. Par exemple, le temps peut avoir une priorité élevée dans un contexte particulier, comme avoir trop de demandes dans des périodes spécifiques (vendredi fou, par exemple). Par conséquent, cette thèse s'intéresse à la planification de chemin écoénergétique¹ (économe à la fois en temps et en énergie), particulièrement, à la résolution du problème de *GPP* écoénergétique (*PP* entre deux points) pour les AMR industriels en général et à la résolution du problème d'*OPR* écoénergétique pour les robots *Ps-AMR* utilisés dans les entrepôts et les *DC* pour la préparation de commandes (multidestinations). Cette planification prend en considération les différents facteurs qui affectent la consommation énergétique.

1. Dans le reste du document, nous allons utiliser le terme : planification de chemin **écoénergétique** pour désigner la planification de chemin qui cherche à minimiser à la fois la consommation en temps et en énergie avec un certain compromis temps/ énergie consommés.

1.3 Objectifs de recherche

L'objectif de ce travail de recherche est de proposer de nouvelles approches écoénergétiques de planification de chemin d'un AMR, tenant compte des facteurs qui influencent sa demande énergétique, afin de réduire sa consommation et de limiter les décharges profondes des batteries, tout en respectant les contraintes opérationnelles du temps des missions. La réalisation de l'objectif principal est conditionnée par l'achèvement des objectifs secondaires, résumés comme suit :

1. Développer une approche de planification de chemin globale écoénergétique (*Energy Efficient Global path planning (EE-GPP)*) qui minimise simultanément la consommation de l'énergie et du temps avec un compromis entre les deux facteurs. Plus précisément :
 - Représenter l'espace de configuration du robot avec les informations métriques jumelées avec les informations de l'environnement, en considérant l'effet de la résistance au roulement et de la dynamique du trafic (création d'une matrice d'adjacence «*layered adjacency matrix*»).
 - Développer une stratégie de planification de chemin capable de gérer ces informations et d'intégrer le modèle dynamique du robot.

2. Développer une approche de routage écoénergétique (avec un compromis temps/énergie) pour résoudre le problème de préparation de commande (*EE-OPR (Energy Efficient-Order Picking Routing)*) d'un Ps-AMR dans un entrepôt (ou DC) avec la configuration *pickers-to-parts*. Particulièrement, développer un programme dynamique pour la résolution du problème de Steiner-TSP qui tient compte des charges transportées depuis les différents emplacements de collecte lors de la prise de décision de routage.

1.4 Méthodologie

Afin de résoudre la problématique de routage écoénergétique des AMR , une étude bibliographique générale est réalisée sur l'utilisation croissante des AMR et leur flexibilité progressive. Cette étude a mené à une remise en question sur l'efficacité énergétique des pratiques opérationnelles actuelles des AMR et la nécessité de les réévaluer ou les adapter

aux changements causés par cette flexibilité, y compris les changements de l'environnement qui affectent la consommation énergétique des différents chemins disponibles pour les robots et qui devront être prise en compte dans la planification de chemins. Pour ce faire, deux sous problèmes ont été traités (le *GPP* et l'*OPR*). Chaque problématique de ces deniers est abordée en trois phases (voir figure 1-10), à savoir l'étude bibliographique, la proposition des approches et les résultats et discussion :

- Une étude bibliographique est réalisée sur la problématique de la recherche de chemin globale (*GPP*) de l'AMR et de son efficacité énergétique. Cette étude a relevé le manque des travaux qui considèrent l'énergie consommée lors de la planification du chemin global. En effet, contrairement à la planification du chemin local et le contrôle de vitesse écologique de plus en plus abordés dans la littérature, il y a un manque de travaux sur le sujet de la *GPP* écoénergétique qui tient en compte de la résistance au roulement et le trafic dans leurs stratégies de *PP*. Par conséquent, une approche de planification de chemin globale écoénergétique (*EE-GPP*) qui considère ces facteurs est proposée. De plus, une discussion des résultats est réalisée ainsi que des comparaisons avec les approches proposées avec les méthodes classiques basées sur le critère distance ou temps seulement.

- En outre, une étude bibliographique était réalisée sur la problématique de routage de préparation de commandes (*OPR*) du *Ps-AMR*, particulièrement, et l'efficacité énergétique de leurs routages. Par la suite, dans ce contexte, la masse des items transportés est un facteur très important qui doit être considéré dans la stratégie de préparation de commande, mais qui est généralement omis dans la littérature. Conséquemment, une approche de planification de tour de préparation de commandes écoénergétique (*EE-OPR*) et qui considère la masse transportée durant le ramassage des commandes est proposée. De même, une discussion des résultats et des comparaisons avec les stratégies basées sur le critère distance ou temps seulement sont effectuées.

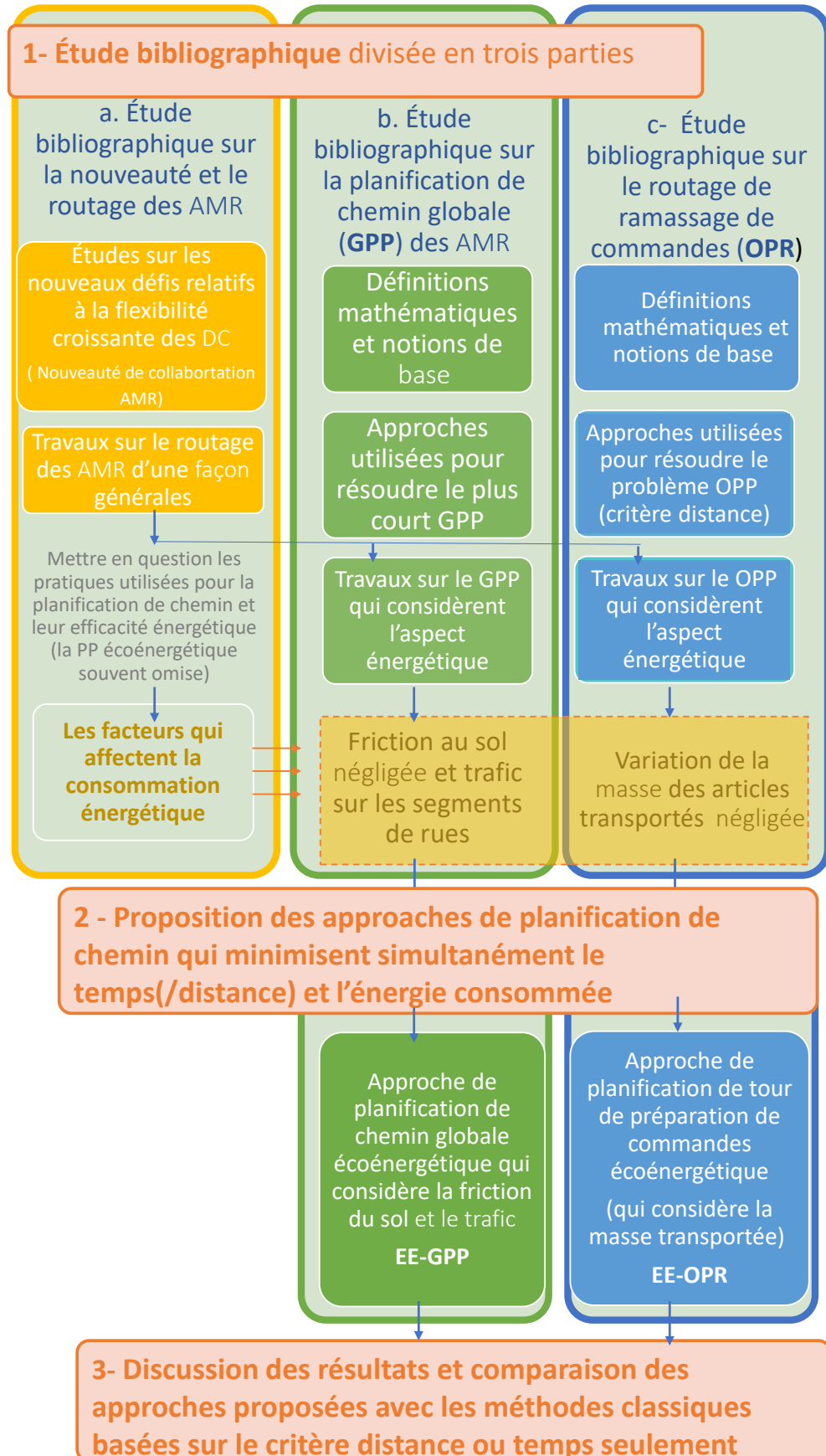


FIGURE 1-10 La méthodologie de recherche suivie

1.5 Structure de la thèse

La suite du document est organisée comme suit :

- **Chapitre 2** : présente une revue de littérature scientifique sur des thèmes pertinents relatifs à cette thèse.

[•] Les notions de base et les méthodes utilisées pour la planification de chemin globale (source et destination uniques) d'une façon générale et les travaux portant sur l'amélioration de l'efficacité énergétique à travers la GPP et les travaux connexes suivi par une synthèse.

[•] Les notions de base et les méthodes utilisées pour la planification de chemin multideestination non ordonnée, particulièrement le routage des préparateurs des commandes d'une manière générale et les travaux connexes qui visent l'optimisation de l'EE des AMR, suivie par une synthèse.

- **Chapitre 3** : expose une nouvelle approche de planification globale écoénergétique qui tient compte de l'effet de la résistance au roulement et la congestion de quelques segments de rue d'une manufacture intelligente, présente des simulations qui compare cette méthode avec les planificateurs de plus court chemin.
- **Chapitre 4** : présente une nouvelle approche de routage écoénergétique utilisée dans un contexte de collectes de commandes dans un entrepôt. Elle focalise sur l'effet de la masse sur la consommation énergétique des chemins des AMR étant donné que l'ordre de visite des points de ramassage est non défini, le problème de routage ne se limite pas à la définition de trajectoires de véhicule, mais aussi à l'ordre de visite des points d'intérêts et aux ramassages ou non des charges situées sur ces points. Le chapitre présente également des simulations qui comparent cette approche avec les algorithmes de préparations de commandes traditionnels.
- **Chapitre 5** : synthétise les travaux réalisés, énonce quelques limitations et finit avec des recommandations permettant des améliorations futures.

Chapitre 2 - État de l'art

2.1 Les AMR et la durabilité

L'utilisation des AGV/AMR offre de multiples avantages sur la gestion durable de la chaîne d'approvisionnement dans le contexte de la numérisation 4.0 [28]. Bechtsis et al. (2017) témoignent des avantages environnementaux offerts grâce à l'adoption des AGV, notamment dans le domaine de la distribution et les opérations logistiques. Ils définissent une hiérarchie du processus de prise de décision en relation avec la durabilité environnementale comme suit : (a) la prise de décision à l'échelon stratégique (b) la prise de décision à l'échelon tactique et (c) la prise de décision à l'échelon opérationnel. Afin d'accroître ces avantages, une meilleure utilisation efficiente et efficace de ces systèmes est souhaitable, offrant un plus long niveau de service et plus de productivité des opérations logistiques. Par conséquent, de nombreuses recherches ont été menées sur la performance énergétique des véhicules de manutention autonomes afin de mettre en place une démarche écologique au sein des usines et des entrepôts [51]. Certains chercheurs se sont concentrés sur le développement d'outils d'aide à la décision pour la sélection du type de véhicules de manutention (comme le gaz de pétrole liquéfié, le diesel ou l'électricité) afin de minimiser l'impact environnemental des activités d'entrepôt [59], [60]. D'autres axes de recherche visent l'évaluation des facteurs qui influencent les besoins énergétiques des AMR et à modéliser ces besoins [61], [62]. Certains chercheurs se sont concentrés sur l'optimisation de la charge [63]-[65], tandis que d'autres sur l'amélioration de l'efficacité énergétique des robots mobiles en agissant sur leurs planification de chemin locale et le contrôle de leurs mouvements [66]-[71]. Par ailleurs, la planification de chemin globale et la planification de chemin multidestinations (*OPR*, dans un *DC*) sont deux étapes très importantes qui précèdent la planification de chemin locales dans la navigation de l'*AMR* et l'amélioration de l'efficacité énergétique de ces derniers est indispensable.

Enfin, d'autres travaux de recherche pertinents visent à améliorer l'efficacité énergétique par les décisions de routage. Puisque l'objectif principal de la thèse est de résoudre le problème de planification de chemin de façon générale (uni et multide destinations) d'un AMR dans les manufactures, les entrepôts et les *DCs* de manière économe en énergie, l'analyse de la littérature existante a été principalement effectuée sur deux axes principaux : l'amélioration de l'*EE* de la planification globale d'un *AMR* et l'amélioration de l'*EE* du routage de préparation de commandes d'un *Ps-AMR* dans un entrepôt. Pour chaque axe, nous allons présenter l'état de l'art de la base du problème en question (définitions, notations mathématiques, formulation du problème et les stratégies utilisées pour le résoudre) et par la suite, les travaux qui considèrent l'efficacité énergétique en abordant ce problème et les travaux connexes. Par la suite, les lacunes de ces travaux qui sont comblées dans ce travail sont exposées.

2.2 Planification de chemin

Le planificateur de chemin est le système responsable de la conception des chemins à assigner au véhicule robot pour accomplir sa mission. Ces chemins doivent satisfaire plusieurs propriétés souhaitées liées à la fois aux contraintes du véhicule et à la morphologie de l'environnement dans lequel le véhicule navigue. En d'autres termes, ce module doit s'assurer que le chemin généré prend en compte les contraintes dynamiques du véhicule (telles que la courbure maximale et la vitesse), tout en gardant le véhicule à une distance de sécurité des obstacles à tout moment.

2.2.1 Recherche dans un graphe

Les différentes méthodes de planification de trajectoires reposent sur la recherche dans un graphe. Un graphe G est un tuple constitué d'un ensemble fini de sommets (ou nœuds) V et d'un ensemble fini d'arêtes E . Nous symbolisons leurs nombres par le cardinal $|V| = n$ et $|E| = m$. Une arête $e \in E$ est représentée par une paire de sommets $(v, w) \in V$. Les arêtes peuvent être unidirectionnelles (appelés arcs) dans un graphe orienté, comme ils peuvent être bidirectionnels dans un graphe non orienté. On considère une arête non dirigée comme deux arcs de sens contraires (v, w) et (w, v) . Un chemin P est une suite d'arêtes $\{e_1, e_2, \dots, e_k\}$ avec $e_i = (v_i, w_i) \in E$, $i = 1, \dots, k$ et $v_{i+1} = w_i$, $i = 1, \dots, k - 1$. Il peut aussi être représenté par la

suite des sommets correspondants. Un chemin est appelé unitaire si la séquence des sommets $\{v_1, v_2, \dots, v_k\}$, qui le construit ne contient aucun nœud w_k doublement. Un chemin est appelé cycle si $v_1 = w_k$. De plus, un graphe peut avoir un coût (poids) c ou bien une fonction de coût qui caractérise chaque arête. Un tel graphe $G = (V, E, c)$ est appelé un graphe pondéré. Le poids (ou le coût) d'un chemin est égal au coût total (la somme) de tous les coûts d'arêtes contenues dans ce dernier ($c(p) = \sum_{e \in p} c(e)$).

Dans un graphe de rues $G(V, E)$, les arêtes représentent les rues et les nœuds présentent leurs points d'intersection. Les graphes de rue présentent généralement les caractéristiques distinctives des autres graphes auxquels les algorithmes de routage sont appliqués pour les graphes de grille uniformes nécessaires à la navigation robotisée, aux réseaux de transport public ou à la communication. En effet, les graphes de rue sont presque planaires. Ils sont creux et ils révèlent une structure hiérarchique. Le degré de leurs nœuds maximal est généralement petit. Cependant, il doit exister quelques rares exemples de graphes de rue qui n'ont pas les caractéristiques citées. Néanmoins, la considération de ces hypothèses lors de la conception ou de l'ingénierie des algorithmes est nécessaire.

2.2.2 Formulation du problème de planification de chemin

Le problème de planification de chemin (*PPP (Path Planning Problem)*) d'un AMR consiste à chercher le chemin optimal sans obstacles entre un état initial et un état destination. Le chemin résultant est construit d'un ensemble d'états (positions et/ou orientations) ou de points de passage (*waypoints*). Le *PPP* est généralement composé d'une description géométrique du véhicule et d'une représentation de l'environnement de navigation. Le véhicule doit chercher un chemin optimal depuis l'état de départ jusqu'à l'état d'arrivée selon certains critères de performance, tels que l'évitement d'obstacles, la longueur et le lissage du chemin et la consommation d'énergie [72], [73]. La formulation conventionnelle du *PPP* est définie par Latombe [74] comme suit :

Pour un robot R dans l'espace de navigation W (W : un espace euclidien \mathbb{R}^2 ou \mathbb{R}^3 , la géométrie, la position et l'orientation de R sont connues a priori, les obstacles O sont des objets rigides stationnaires dans W , dont la localisation est connue avec précision. La planification du chemin $P \subset W$ se traduit par l'ensemble de points tel que $R(p) \cap O = \emptyset$, pour chaque point

$p \in P$. Le planificateur détermine et retourne un chemin P entre un point de départ et un point d'arrivée, ou bien il retourne l'ensemble vide \emptyset , le cas échéant. L'optimalité de chemin résultant est évaluée selon les critères d'optimisation préalablement définis. Lozano-Pérez a introduit la notion de l'espace de configuration ($C_{space} \subset W$) du robot, introduit dans [75], qui caractérise la position et l'orientation du robot par un seul point $q = (x, y) \in \mathbb{R}^2$ (position en x , position en y). La trajectoire correspond à une suite de points dans un plan 2D. Cette perspective facilite la recherche de chemin. L'espace de configuration est constitué de deux parties : espace de configuration libre, où le robot peut circuler (C_{free}) et espace de configuration interdit, où il y a des obstacles (C_{obs}). La taille des obstacles est agrandie selon la taille du robot pour compenser la représentation du robot par un point, comme représenté dans la figure 2-1 (l'espace gris représente le C_{free} , l'espace noir et l'espace jaune de l'agrandissement des obstacles représentent C_{obs} et le cercle blanc représente le robot). Par ailleurs, il existe des travaux de recherches qui incluent la contrainte de la géométrie du véhicule dans leurs formulations de planification de chemin. Ces méthodes ne sont pas traitées dans cette thèse qui considère l'espace C_{free} et les voies définies par les cartes topologiques suffisants pour la navigation du robot dans son espace de déploiement.

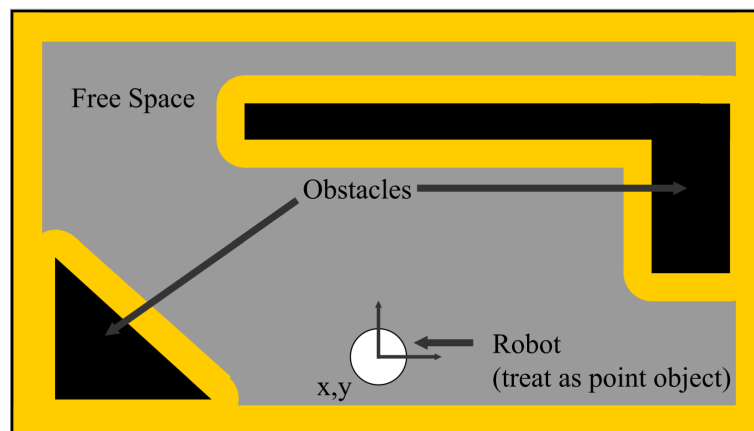


FIGURE 2-1 Espace de configuration [76].

2.2.3 Catégories de planification de chemin

La planification de chemin des robots mobiles est reliée à différents sujets qui peuvent être classifiés en trois catégories comme, schématisés dans la figure 2-2.

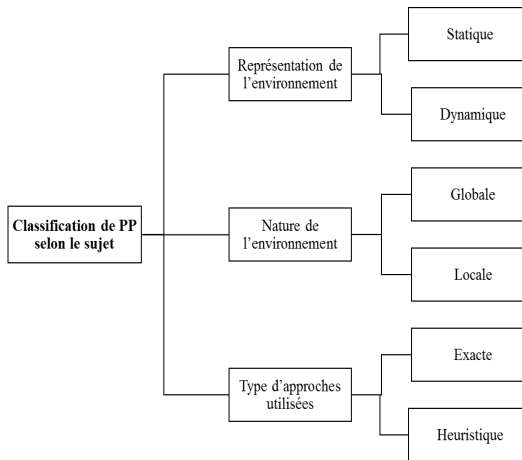


FIGURE 2-2 Classification de planification de chemin selon le sujet[77].

— Représentation de l'environnement

Le problème de planification de chemin peut être résolu dans des environnements statiques ou bien dans des environnements dynamiques. Dans le premier cas (statique, dont nous nous intéressons), l'environnement, le point départ, le point destination et les obstacles sont fixes. Ils ne changent pas de position au cours du temps. Cependant, pour le second cas (dynamique) l'emplacement des obstacles et la position de point destination peuvent changer au cours du processus de recherche.

— Nature de l'environnement

La planification de chemin d'un robot mobile repose principalement sur la carte de son environnement, pour son repérage et le repérage de sa destination. Dépendamment de la disponibilité ou non de l'information de la carte et de la qualité de celle-ci se définit le type de planification à concevoir. Sur cette base de connaissance, la planification de chemin peut être divisée en deux catégories : planification de chemin global et planification de chemin local (ou navigation réactive). Pour la planification de chemin globale, le véhicule dispose d'une connaissance a priori de l'environnement, représentée sous forme d'une carte métrique. Par contre, pour la planification de chemin locale tous ou la majorité de l'information sur l'environnement sont inconnu pour le robot avant son exécution [50], [72], [77], [78]. Il

TABLEAU 2-1 Planification de chemin globale vs locale [77]

Planification de chemin local (GPP)	Planification de chemin global (LPP)
Basée sur les capteurs	Basée sur une carte
Navigation réactive	Navigation calculée
Réponse rapide	Relativement plus lente
Suppose que l'espace de travail est incomplet ou partiellement incomplet	L'espace de travail est connu
Génère un chemin et se dirige vers la destination en évitant les obstacles et les objets	Génère un chemin faisable avant partir vers la destination
En ligne	Hors ligne

doit, donc, détecter la position des obstacles durant le processus de recherche (en temps réel) afin de les éviter et de construire un chemin approprié vers la destination. Le tableau 2-1 présente quelques caractéristiques de planification de chemin globale versus la planification locale. L'optimisation de l'efficacité énergétique de cette dernière est beaucoup plus abordée dans la littérature de la robotique mobile que la planification globale. Les quelques travaux qui traitent l'EE de GPP sont généralement adressés à des véhicules de surface sans pilote (unmanned surface vehicles) [79] et surtout à des robots *off-road* utilisés dans un contexte extérieur communément appelé, véhicule terrestre sans pilote (UGV (*Unmanned Ground Vehicle*)) [80]-[83]. Ces derniers sont généralement utilisés dans des situations où la présence humaine est dangereuse ou impossible et/ou sur des terrains difficiles. D'ailleurs c'est pour cette raison que l'amélioration de l'efficacité énergétique de la GPP de ces robots est souvent gérée à travers la considération de la friction au sol et des pentes lors de la planification. Ces applications sont différentes du contexte intérieur considéré dans ce document où la pente et les élévations sont négligeables.

— Type d'approches utilisées

Un algorithme de planification de chemin est appelé complet lorsqu'il considère ou élimine tous les états possibles dans l'espace de configuration [84]. En fonction de leur complétude, les approches utilisées pour résoudre un problème de planification de chemin peuvent être exacts ou bien basés sur des heuristiques. Les approches exactes garantissent l'optimalité des résultats s'ils en existent ou prouvent la non-existence d'une solution réalisable, le cas échéant. Cependant, les approches basées sur des heuristiques essaient de trouver des bonnes solutions

dans de plus brefs délais (temps de calcul plus court). Néanmoins, elles ne garantissent pas l'optimalité de la solution.

Une autre classe possible est la planification de chemins de plus haut niveau lorsqu'il s'agit d'un problème de routage à destinations multiples avec un ordre indéfini de positions à visiter. Cette classe de planification de chemin multidestinations peut être vue comme une planification de chemin entre deux points avec la contrainte de passage sur des terminaux bien déterminés. Ce cas de figure est abordé plus tard via le problème *OPR* dans le contexte de collecte des articles des commandes clients par un *Ps-AMR*, dans un entrepôt.

2.3 Planification de chemin globale (GPP)

Comme précédemment annoncé, la planification de trajectoire est composée de deux étapes distinctes : la planification de trajectoire globale et la planification de trajectoire locale. Généralement, trois étapes constituent le processus de la planification de chemin globale (voir figure 2-3) :

1. Représentation de l'espace : de navigation de robot est converti à des indications faciles pour stocker les caractéristiques cartographiques ;
2. Critères d'optimisation ;
3. Algorithme de recherche de chemin : identifier un chemin qui mène du point de départ au point de destination, en minimisant certains critères d'optimisation.

Nous allons expliquer d'avantage ces notions dans les prochaines sections.

2.3.1 Représentation de l'espace

La représentation de l'espace est la première étape que le planificateur de chemin doit faire. Il s'agit de saisir les connectivités dans l'espace de configuration avec une carte routière (*roadmap*) construite par des courbes unidimensionnelles (graphe ou réseau (exemple dans la figure 2-4)). Elle est ensuite utilisée pour connecter un état de départ q_s à un état cible q_t par un chemin dans C_{free} , en évitant les obstacles O .

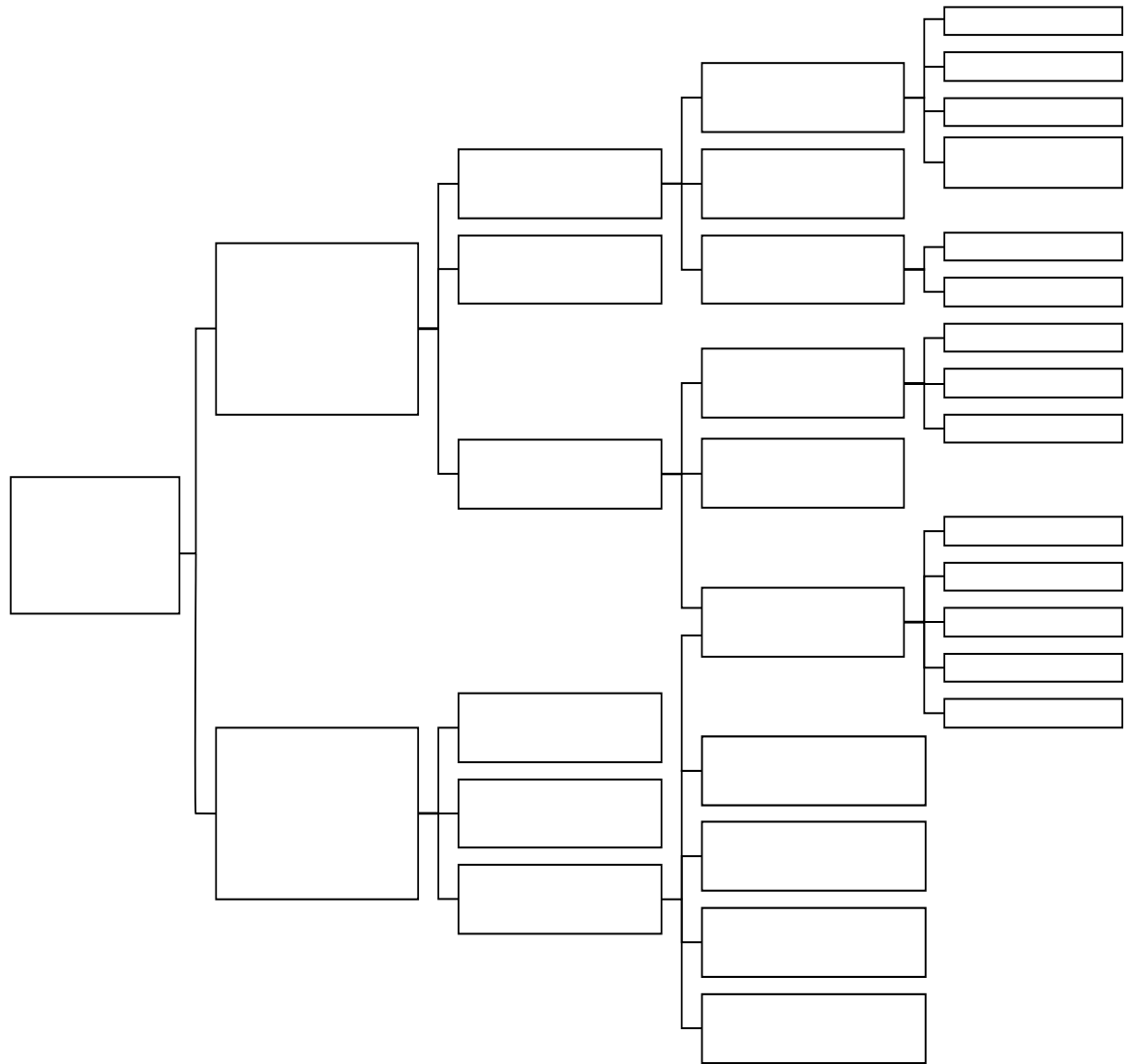


FIGURE 2-3 Planification de chemin d'un robot mobile : Classification [72].

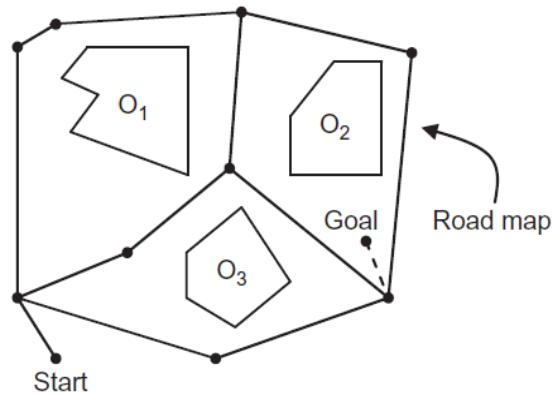


FIGURE 2-4 Un exemple d'une carte routière (*roadmap*) dans un environnement avec des obstacles O_1 , O_2 et O_3 [85].

Pour construire la *roadmap*, deux groupes de travaux se distinguent dans la littérature : (i) des travaux qui utilisent les méthodes probabilistes (ou planification basée sur l'échantillonnage) et (ii) des travaux qui utilisent les méthodes déterministes traditionnelles .

2.3.1.1 Les méthodes probabilistes

Les algorithmes de planification de chemin basés sur l'échantillonnage dépendent fortement de la sélection des points de l'espace de configuration du véhicule. Nous citons l'exemple de la méthode (*PRM*) [86] (*Probabilistic Roadmaps* et la méthode *RRT* [87] (*Rapidly Exploring Random Trees*). En effet, la méthode *PRM* utilise des algorithmes probabilistes pour construire un graphe de réseau de chemins possibles (*roadmap*) dans l'espace de configuration C_{free} , par un échantillonnage arbitraire. Elle commence par une génération arbitraire des nœuds. Ensuite, un planificateur local essaye de connecter ces nœuds à l'aide d'un chemin. Si un chemin est trouvé, un nouveau nœud est additionné au graphe. Par la suite, après avoir connecté la configuration initiale et la configuration finale au graphe, un chemin faisable est déterminé à l'aide d'un algorithme de recherche dans un graphe. La méthode *PRM* réussit dans certaines situations de planification de chemin complexes [88] et elle échoue dans d'autres telles que les encoches et autres [89] (voir figure 2-5). Quant à la méthode *RRT*, elle permet de rechercher les espaces concaves, de grande dimension, en construisant aléatoirement un arbre de remplissage. Ces arbres résultent en des trajectoires non lisses (des virages inutiles), surtout avec un faible nombre d'itérations. La figure 2-6 montre la structure de *RRT*, un

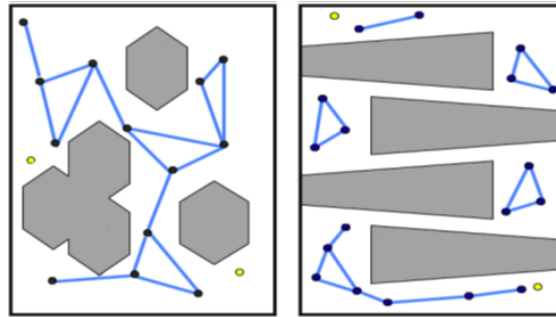


FIGURE 2-5 Deux exemples d'utilisation de la méthode *PRM* (exemple à gauche : la méthode réussit à trouver un chemin faisable ; exemple à droite : la méthode échoue à construire un graphe connexe).

exemple de chemin résultant de *RRT* dans un environnement vide et un exemple de chemin résultant de *RRT* dans un environnement avec obstacle. Souvent, un post-traitement est essentiel pour améliorer la qualité du chemin, vu la nature probabiliste de l'algorithme.

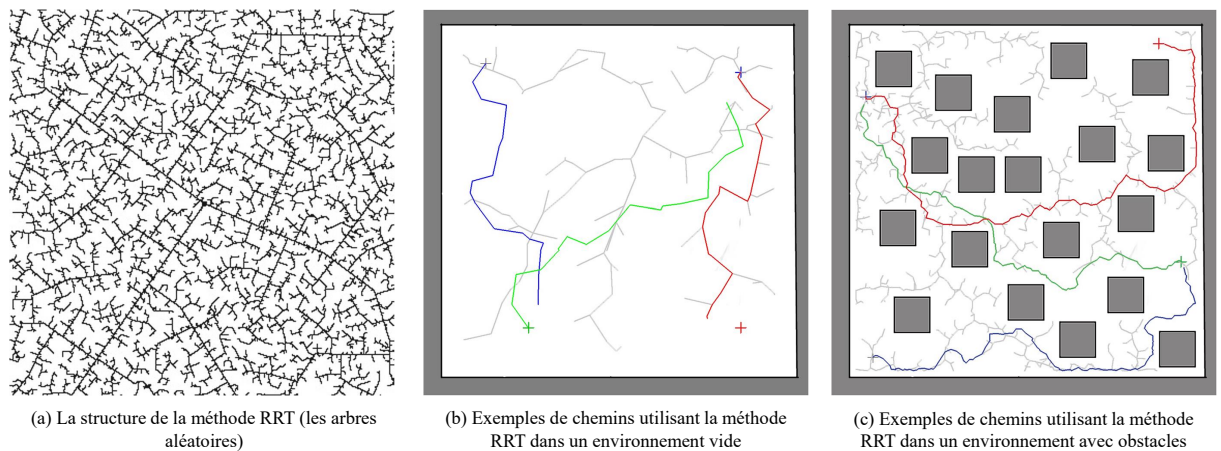


FIGURE 2-6 Exemples d'utilisation de la méthode *RRT*.

Les deux méthodes décrites ci-dessus peuvent obtenir un taux de réussite élevé, avec un faible temps d'exécution. Cependant, elles ne sont pas complètes en résolution (non exactes). Même avec des conditions initiales similaires, ces méthodes ne trouveront pas nécessairement le même chemin à chaque exécution. Dans un contexte d'usine ou d'entrepôt, les méthodes probabilistes sont écartées pour résoudre le *GPP*. En effet, l'exécution de tels algorithmes génère à chaque fois des *roadmaps* différentes et par suite des solutions distinctes. Ces méthodes ne seront pas alors une bonne alternative de discrétisation de l'espace permettant de trouver un chemin écoénergétique optimal pour l'*AMR*. Cependant, ces méthodes

probabilistes et leurs extensions peuvent être une solution efficace utilisée en temps réel pour la planification locale des robots mobiles [90].

2.3.1.2 Méthodes déterministes

Dans un espace de configuration continu, les approches déterministes sélectionnent les chemins sans faire appel à des algorithmes d'approximations (contrairement aux algorithmes basés sur l'échantillonnage). Pour cette raison, ils sont aussi appelés des algorithmes exacts [87]. Parmi les méthodes utilisées pour développer le réseau *roadmap* dans la catégorie des méthodes déterministes. On trouve :

- la planification combinatoire ;
- la méthode topologique.

Parmi les méthodes les plus connues dans la planification combinatoire. On distingue les méthodes de graphes de visibilité, la méthode de graphes de *Voronoi* et la décomposition en cellules qui donnent un graphe de réseau [50], [91]. Ensuite, la méthode de recherche dans un graphe relie la position initiale et la position cible à travers les nœuds de la carte [50], [91]. La décomposition de cellules consiste à décomposer le C_{free} , délimité par des polygones, en cellules trapézoïdales, triangulaires, ou quadratiques en projetant des rayons de haut en bas à partir de chaque sommet de polygone. Chaque cellule résultante est numérotée et représentée par un nœud et les nœuds adjacents sont reliés avec une arête formant ainsi le graphe de connectivité. Les points milieu des droites communes des cellules adjacentes (intersections) et (/ou) les centroïdes de cellules sont utilisés pour connecter le q_s à q_r . Il existe des variantes de la méthode de décomposition en cellules, telles que la décomposition en cellules exactes et la décomposition en cellules approximatives (voir la figure 2-7 [50], [88]). La décomposition en cellules uniformes est un sous-ensemble de la décomposition en cellules approximatives (la partie droite de la figure 2-7 [92], appelée aussi carte de grille d'occupation *OGM* (une matrice $NbLineOccGrid * NbColOccGrid$)).

Cette méthode est la plus classique pour la représentation de l'espace d'un robot mobile [92]. D'ailleurs Patle et al. [50] affirment que les algorithmes de planification de chemin dans des environnements connus sont basés sur des approches classiques (déterministes). La principale difficulté de la grille d'occupation consiste à trouver la taille de la grille (le

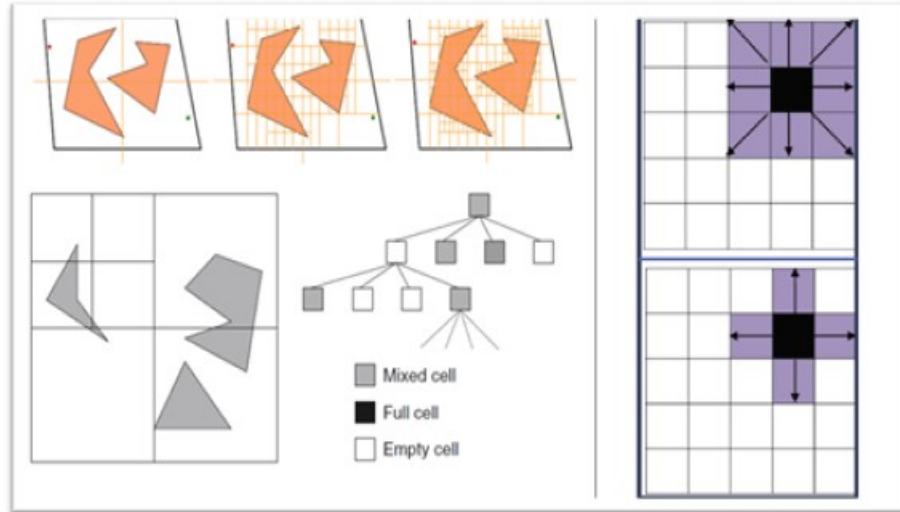


FIGURE 2-7 Méthode de décomposition en cellules approximatives (partie gauche de la figure) et en cellules uniformes (partie droite de la figure)[50], [88].

nombre de cellules adéquat). En effet, plus les cellules sont petites, plus la représentation de l'environnement serait précise. Cependant, l'utilisation des cellules de moindre taille entraînera une augmentation exponentielle de l'espace mémoire demandé et de la plage de recherche [93]. Néanmoins, cette méthode garantit la précision de recherche de chemin vu sa nature déterministe pour discrétiser l'espace C_{free} et garantir l'optimalité de résultat si elle est utilisée avec un algorithme de recherche exact [94]. Pour cette raison et sur la base de la démarche adoptée dans la majorité des travaux issus du contexte industriel, nous avons opté pour ladite méthode dans la première partie de la thèse, ensuite nous avons opté pour la solution topologique. La méthode topologique réduit la dimension de PPP dans un espace géométrique à haute dimension en éliminant les connectivités à faible dimension. Elle représente un environnement sous forme d'un graphe constitué d'un ensemble de nœuds et d'arêtes reliant les nœuds discriminant des connectivités à faible dimension [72], [95]. Un bon exemple de carte topologique serait la carte de bus ou de métro, où on trouve différentes stations (qui correspondent à des nœuds de la carte topologique) et différentes lignes de bus ou de métro (qui correspondent à des arêtes de la carte topologique) qui relient les stations connectées. La Figure 2-8 représente un exemple d'une carte topologique pour stocker l'environnement des robots de manière qualitative [97]. Chaque nœud correspond à un emplacement distinctif dans l'environnement ou une balise (point de repère) et les arêtes correspondent aux connectivités

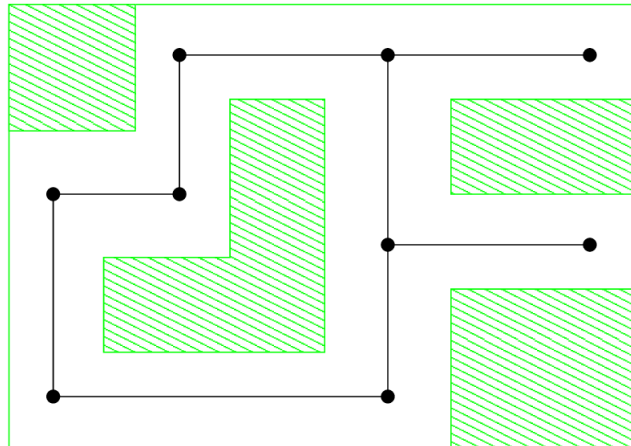


FIGURE 2-8 Exemple de carte topologique [96].

entre les nœuds et aux routes possibles reliant les différents emplacements. Ils se basent sur des caractéristiques telles que les intersections et les angles pour leurs nœuds. D'autres particularités peuvent être utilisées, telles que les murs et les couloirs [98]. Cette approche topologique assure une planification rapide, comparée à d'autres approches de planification. En effet, elle économise le temps pour la construction du modèle et elle requiert moins d'espace de stockage. Sa complexité est seulement liée aux nombres d'obstacles. En outre, les cartes métriques, résultant des techniques de la cartographie dans la robotique mobile, sont très vulnérables aux imprécisions dans les capteurs, les actionneurs et l'odométrie [96]. De ce fait, d'autres moyens de discrétisation peuvent être affectés par cette imperfection. À l'opposé, une cohérence topologique globale est robuste contre ces imprécisions [99]. Le point faible de cette méthode qualitative est sa faible résolution. Afin de bénéficier de l'avantage de la robustesse de la méthode topologique et de la précision de la méthode métrique, des cartes hybrides étaient proposées [100]. Dans cette optique, nous avons choisi pour la méthode topologique où l'utilisateur définit un *roadmap* : un ensemble de chemins prédéterminés virtuels le long desquels les *AMR* sont contraints de se déplacer tout en permettant au planificateur local de faire une planification plus raffiner (ajouter un maillage plus détaillé dans les espaces ouverts) et de contourner les obstacles imprévus [101]. Nous nous sommes inspirés de quelques travaux récents effectués dans un contexte industriel [100], [101].

2.3.2 Recherche de chemin dans un graphe

Une fois que l'environnement avec des obstacles est convenablement représenté par un graphe, des algorithmes de recherche de chemin (*Graph Search algorithm*) sont utilisés pour trouver le chemin optimal entre la configuration du début jusqu'à la configuration de destination. Ci-dessous, quelques méthodes sont présentés.

2.3.2.1 L'algorithme de Dijkstra

L'algorithme de Dijkstra (*DA (Dijkstra Algorithm)*) est l'algorithme de planification de chemin le plus classique et le plus connu [102]. C'est un algorithme efficace pour effectuer une recherche de graphe optimal à source unique. Il détermine le chemin de moindre coût dans un graphe pondéré ou non d'un nœud initial (q_s) (*start*) à un nœud de destination (q_t) (*target*). La fonction de coût peut être, simplement, une distance euclidienne ou bien une fonction de coût beaucoup plus complexe avec différents critères. L'algorithme fournit un résultat complet, ce qui signifie qu'il garantit l'obtention de la solution optimale, ou signale la non-existence d'une solution. Toutefois, il est encore possible d'améliorer et d'étendre cet algorithme pour des scénarios spécifiques [103]-[105].

L'algorithme de *Dijkstra* est considéré comme un algorithme gourmand (*greedy*), car il fait le choix localement optimal à chaque étape, sans tenir compte des implications globales de ce choix. En effet, si on utilise les appellations les plus courantes de "plus court chemin" pour exprimer le chemin le plus optimal (avec un critère donné) et "*distance* ou *dist*" d'un nœud pour exprimer le coût optimal du nœud depuis le sommet de départ, *DA* peut être décrit comme suit : À chaque étape, l'algorithme sélectionne le sommet ayant la plus petite valeur de distance au sommet source parmi l'ensemble des sommets dont les distances de chemin les plus courtes ne sont pas encore connues. Ce choix est localement optimal, car il minimise la distance entre le sommet sélectionné et le sommet source. Cependant, ce choix peut ne pas être globalement optimal, car il ne prend pas en compte la possibilité que la sélection d'un sommet différent puisse entraîner un chemin global plus court vers un autre sommet. Malgré sa nature gourmande, l'algorithme de *Dijkstra* garantit la résolution optimale de problème du chemin le plus court à source unique dans un graphe pondéré. En effet, l'*AD* est fortement

inspiré du principe d'optimalité de Bellman [106] techniquement et conceptuellement ainsi il peut être également considéré comme une procédure d'approximations successives de *DP* pour résoudre le problème du plus court chemin à source unique dans un graphe pondéré.

Par ailleurs, étant donné sa popularité, sa flexibilité et sa capacité d'intégrer différents types de modèles et de contraintes, tout en donnant des résultats optimaux, la programmation dynamique est une méthode très utilisée pour résoudre les problèmes de contrôle optimal. De ce fait, nous nous sommes inspirés dans les approches proposées dans cette thèse de l'*AD*, tout en exploitant sa perspective de *DP*. En effet, l'algorithme de Dijkstra présente à la fois une programmation dynamique et des caractéristiques gourmandes, et ces propriétés ne sont pas mutuellement exclusives [106]-[108]. Par ailleurs, étant donné sa popularité, sa flexibilité et sa capacité d'intégrer différents types de modèles et de contraintes, tout en donnant des résultats optimaux, la programmation dynamique est une méthode très utilisée pour résoudre les problèmes de contrôle optimal. De ce fait, nous nous sommes inspirés dans les approches proposées dans cette thèse de l'*AD*, tout en exploitant sa perspective de *DP*. En effet, l'algorithme de Dijkstra présente à la fois une programmation dynamique et des caractéristiques gourmandes, et ces propriétés ne sont pas mutuellement exclusives [106]-[108].

À chaque étape de l'algorithme *AD*, un ensemble de sommets avec les distances de plus court chemin depuis le sommet source sont connues, est défini. De plus, un tableau de distance qui stocke la distance connue la plus courte entre le sommet source et chaque sommet du graphe est mis à jour. L'algorithme sélectionne itérativement le sommet avec la plus petite valeur de distance par rapport au sommet source parmi l'ensemble des sommets dont les distances de chemin les plus courtes ne sont pas encore connues. Il relâche ensuite toutes les arêtes sortantes du sommet sélectionné pour mettre à jour les distances des sommets adjacents. L'étape de relaxation met à jour la valeur de distance d'un sommet v au sommet source en considérant la valeur de distance de son sommet prédécesseur u et le poids de l'arête (u, v) les reliant. Si la valeur de distance nouvellement calculée est inférieure à la valeur de distance actuelle, nous mettons à jour le tableau de distance avec la nouvelle valeur. Ce processus est répété jusqu'à ce que tous les sommets du graphe soient inclus dans l'ensemble de sommets dont les distances de chemin les plus courtes depuis le sommet source sont connues. À ce

stade, le tableau de distance contient les distances de chemin les plus courtes entre le sommet source et tous les autres sommets du graphe. La perspective de programmation dynamique vient du fait que l'algorithme résout les sous-problèmes de recherche des distances de chemin les plus courtes entre le sommet source et chaque sommet du graphe, et combine les solutions à ces sous-problèmes pour obtenir la solution globale. L'étape de relaxation correspond à la propriété de sous-structure optimale de la programmation dynamique, qui stipule qu'une solution optimale à un problème peut être obtenue en combinant des solutions optimales à ses sous-problèmes. Un exemple d'exécution de l'algorithme de *Dijkstra* est illustré dans la figure 2-9.

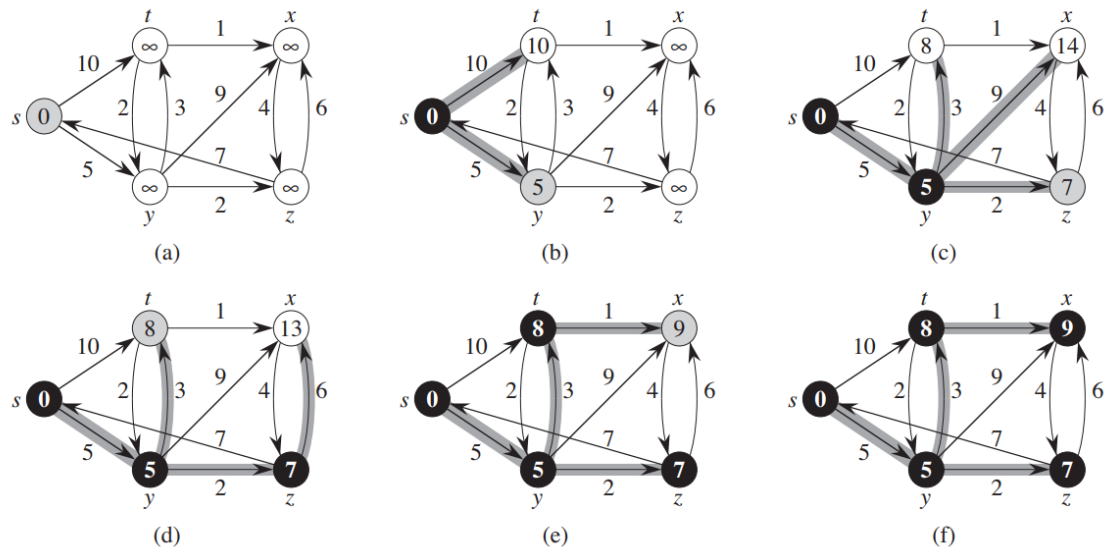


FIGURE 2-9 Un exemple d'exécution de l'algorithme de *Dijkstra* [107].

Si on considère le nœud *s* comme étant le nœud de départ, le déroulement des étapes présentées est le suivant :

- (a) : Initialiser la distance du nœud *s* à zéro (distance entre *s* et lui-même) et du reste des nœuds à infini. Ces valeurs sont écrites dans les cercles des nœuds dans la figure, représentant le coût optimal actuel entre chaque nœud et le nœud source *s*.

- (b) : Sélectionner le nœud avec la distance minimale et explorer les nœuds voisins et mettre à jour leurs distances, si possible. Ainsi, le nœud avec la distance minimale est *s* avec un coût de 0. Les voisins de *s* sont *t* et *y* avec des coûts de $0 + 10 < \infty$ et $0 + 5 < \infty$, respectivement.

- (c) : Dans cette étape, le nœud y est sélectionné avec le coût de 5. Ce dernier est marqué comme visité et son coût ne peut plus être amélioré (le cercle du nœud est coloré en noir). Les voisins de y sont t , x et z avec des coûts de $5 + 3 < 10$ (mettre à jour le coût) et $5 + 9 < \infty$ (mettre à jour le coût) et $5 + 2 < \infty$ (mettre à jour le coût), respectivement.

- (d) : Maintenant, le nœud z est choisi avec le coût de 7 (il est marqué comme visité). Les nœuds sortants de z sont s (on ne retourne pas vers la source) et x avec le coût de $7 + 6 < 14$ (mettre à jour le coût).

- (e) : Ensuite, le nœud t est sélectionné avec le coût de 8 (il est marqué comme visité en noir). Le voisin de t est x avec des coûts de $8 + 1 < 13$ (mettre encore à jour le coût optimal de x).

- (f) : Enfin, le nœud x est sélectionné avec le coût de 9 (il est marqué comme visité en noir).

2.3.2.2 L'algorithme de A^* et ses successeurs

L'algorithme A^* est très utilisé dans la planification de trajectoires des *AMR* dans une carte de grille d'occupation [109]. Cette méthode est considérée comme un algorithme gourmand de meilleur en premier (*Greedy Best-First-Search*). Il était proposé par Hart et al. en 1968, puis en version améliorée en 1972 [104]. L'algorithme est basé sur une heuristique de recherche de chemin qui combine le coût réel depuis le point de départ et le coût estimée entre le nœud exploré et le nœud destination. En effet, A^* a une fonction de coûts (totale) $f(x)$ composée de deux fonctions de coûts $g(x)$ (le coût estimé entre le nœud de départ et le nœud x) et $h(x)$ (la distance estimée entre le nœud x et le nœud destination (généralement la distance euclidienne ou de *Manhattan*)) pour chaque nœud exploré. Cette fonction est définie comme suit :

$$f(x) = g(x) + h(x)$$

Par suite, l'algorithme A^* est l'équivalent à l'algorithme de *Dijkstra* si on ne considère pas $h(x)$. L'algorithme A^* est plus efficace que l'algorithme de *Dijkstra* dans la recherche de chemin en terme de temps d'exécution, car il réduit la taille du graphe exploré en utilisant ces heuristiques [104]. Cette manière accélère la résolution du problème de planification de chemin, mais elle empêche l'exploration de quelques zones de l'environnement qui peuvent être plus favorable de point de vue énergétique (voir la figure 2-10). En fait, l'heuristique $h(x)$

joue le rôle d'un amant vers le nœud destination. D'autre part, *Dijkstra* explore les différentes possibilités et garantit la solution optimale, en conséquence.

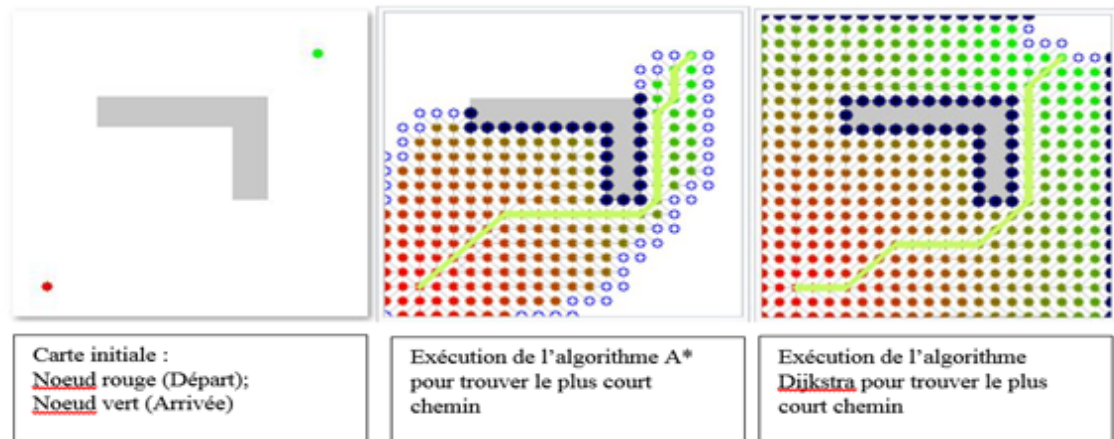


FIGURE 2-10 Comparaison entre l'exécution de l'algorithme A* et l'exécution de l'algorithme Dijkstra de point de vue spatial [110].

Nous mentionnons également que l'algorithme A^* est étendu pour donner d'autres méthodes adaptées à des besoins particulier. Par exemple, l'algorithme D^* était proposé par Stentz en 1994 pour répondre au besoin de l'environnement dynamique du robot mobile et de la nécessité d'y prendre conscience [111], contrairement à l'algorithme A^* utilisé principalement pour la recherche globale dans un environnement statique (d'où l'appellation D^* qui vient du terme *Dynamic A**). Il suit le même principe que A^* , excepté que le coût des arcs change au fur et à mesure que l'algorithme s'exécute. D'autres algorithmes successeurs de D^* ont été proposés, à savoir, l'algorithme *Field D^** [112] et l'algorithme *Theta** [113]. D^* et ses variantes (*D^* Lite*, *Focused D^**) ont été largement utilisés pour la navigation dynamique des robots mobiles, particulièrement *D^* Lite*.

2.3.3 Méthode de champs de potentiel artificiel

L'approche du champ de potentiel artificiel (APF) est conçue par Khatib et al. [114] et elle est développée ultérieurement dans [115]. L'idée générale de la méthode est de créer un champ de potentiel en fonction de l'ensemble des obstacles et du point destination et ensuite modéliser le robot comme une particule qui se déplace sous l'influence de ce champ. Pour que le robot puisse éviter les obstacles, des forces répulsives sont attribués aux obstacles. D'autre

part, des force attractives sont affectées au point cible afin qu'il attire le robot (voir figure 2-11). Toutefois, dans certaines situations, le robot peut se retrouver piégé. C'est un problème majeur des méthodes de potentiel artificiel, à savoir, la présence du minimum local. Afin de surmonter ce problème, plusieurs chercheurs ont utilisé des fonctions potentielles sans minimum local, appelée des fonctions de navigation [116]-[118]. L'approche du champ potentiel artificiel était utilisée dans [119] et [120] pour la planification de chemin de robot mobile. Le travail dans [119] contenait aussi une comparaison de *APF*, un *APF* modifié et un algorithme génétique. Sfeir et al dans [120], ont proposé une amélioration de *APF* pour la planification dans un environnement inconnu. Cette amélioration se base sur une nouvelle formule de potentiel répulsif qui réduit les oscillations et évite les confusions dans le cas où il y a des obstacles à proximité d'un point destination. Kim el al. [121] ont proposé une conception basée sur la *APF* pour la planification local pour s'échapper d'une situation d'emplacement minimum.

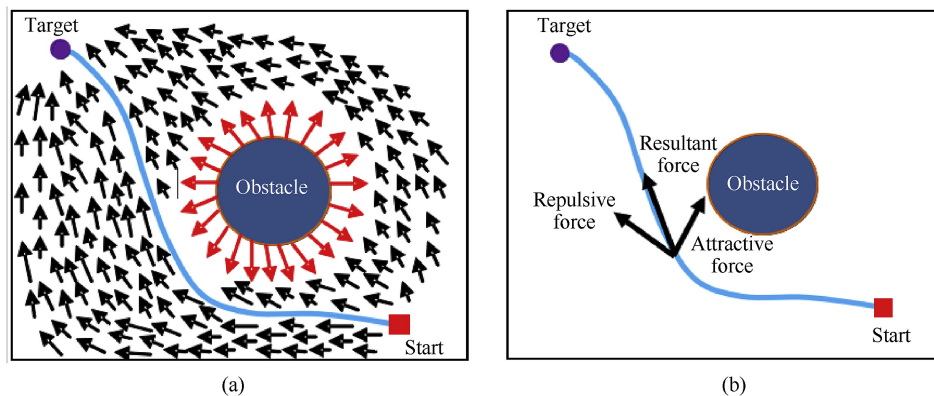


FIGURE 2-11 Méthode de champ de potentiel artificiel pour la navigation du robot mobile [50].

2.3.4 Approche métaheuristique pour la résolution du problème de planification de chemin global

Pour trouver le chemin optimal, les approches classiques deviennent moins efficaces en termes de temps de calcul lorsque l'environnement est complexe [72], [122]. D'où l'apparence et l'évolution des approches métaheuristiques. Parmi ces approches les plus utilisées pour la planification de chemin d'un véhicule autonome, il y a l'algorithme génétique (GA (*Genetic Algorithm*)), l'optimisation par essaims particulaires (PSO(*Particle Swarm Optimization*)) et l'algorithme de colonies de fourmis (ACO (*Ant Colony Optimisation*)) [72]. Malgré une

vitesse de calcul plus rapide des méthodes métaheuristiques, les approches classiques telles que *Dijkstra* et la programmation dynamique, par exemple, restent plus précises [72]. Dans cette optique, on a opté pour ces dernières qui garantissent l'optimalité de résultat s'il en existe ou prouvent la non-existence d'une solution réalisable. En effet, les méthodes métaheuristiques se basent sur des approximations pour être plus rapides. Ceci remet en question l'optimalité de leurs résultats. Dans notre cas, la précision et la garantie de trouver une solution optimale est indispensable. En effet, vu l'exigence très sévère de la sécurité et l'efficacité dans le cadre industriel, les méthodes basées sur des approximations ne sont pas désirables. De plus, nous parlons d'une planification de chemin globale, qui contrairement au planificateur local qui demande un temps de réaction très rapide (requiert des algorithmes avec un temps de calcul très court), dispose d'exigences peu strictes en termes de temps de calcul et se caractérise par un espace de recherche raisonnable. Pour cela, la densité de graphe joue un rôle indispensable dans le choix de type de l'algorithme à adopter. En effet, la méthode topologique, choisie précédemment, fournit un graphe appelé « graphe de rue ». Ce type de graphe a des caractéristiques particulières : un graphe presque polaire (il peut être dessiné de manière à ce qu'aucune arête ne s'entrecroise), un graphe creux (*sparse graph*). Le degré maximal des nœuds est petit, révèle une structure hiérarchique et des chemins optimaux pratiquement uniques. Ces caractéristiques appuient aussi notre choix des algorithmes exacts, car l'espace de recherche est relativement petit.

2.3.5 Planification de chemin globale dans le contexte industriel d'un AMR

Les travaux de recherche antérieurs pour les systèmes AMR sont principalement axés sur l'optimisation des chemins et la gestion du trafic des AMR [123] (il existe encore des travaux qui utilisent toujours la notation AGV). On trouve des travaux qui s'orientent vers l'optimisation de trajectoire de flux des AMR dans son ensemble, tels que [124]-[127] et d'autres qui s'intéressent principalement à l'optimisation de chemin (GPP) d'un seul AMR comme dans notre cas, mais avec seulement le critère distance. Par exemple, Yang et al. [128] et Wang et al. [129] présentent des algorithmes de planification de chemin pour les AMR, basé sur l'algorithme A^* . L'algorithme dans [128] utilise la méthode de la décomposition en grille uniforme pour discrétiser l'espace et une autre de lissage de trajectoire qui est ajoutée à

l'algorithme A^* , en considérant le rayon de braquage de l'AMR. Praserttaweelap et al. [130] se focalisent sur la contrainte d'évitement d'obstacles et le compromis entre la sécurité et le temps d'exécution de la plateforme. Ils utilisent PSO pour la recherche de chemin en intégrant la fonction de fitness à la fonction d'évitement d'obstacles. Le travail décrit dans [131] traite le problème de routage de plusieurs AGV et il se base sur la classification des collisions. Les auteurs planifient le chemin de chaque tâche en utilisant l'algorithme de Dijkstra. Ils présentent des études de cas pour montrer l'efficacité de leur approche. Kelen et al. [132], présente une vue d'ensemble sur des robots chariots élévateurs pour les entrepôts intelligents. Pour ce qui est GPP de l'AMR, ils utilisent une carte topologique de l'entrepôt et choisissent l'algorithme de Dijkstra pour déterminer le chemin le plus court. Les résultats obtenus montrent que l'outil proposé était capable d'éviter l'afflux et les collisions avant que le robot exécute sa tâche selon le chemin sélectionné.

2.4 Planification de chemin et critère énergétique

Les recherches sur l'efficacité énergétique des systèmes AMR (ou AGV) et leurs batteries ont principalement porté sur la gestion de la batterie de ces véhicules [133], sur l'évaluation de l'impact de la gestion de recharge sur le système au complet [64], [134], ou encore sur le contrôle optimal (vitesse/accélération) des véhicules pour effectuer leurs tâches [135]-[137]. Cependant, l'optimisation de l'efficacité énergétique d'AMR à travers l'optimisation de sa trajectoire globale n'a pas été suffisamment abordée. Toutefois, d'autres travaux s'intéressent à la planification écoénergétique (EEPP : *Energy Efficient Path Planning*) en dehors du contexte industriel et nous distinguons deux catégories de travaux de recherche qui s'intéressent à la planification de chemin avec un critère énergétique (i) La première catégorie regroupe les travaux sur les véhicules électriques (non autonomes) sur les routes de façon générale. (ii) La deuxième catégorie regroupe les travaux qui se concentrent sur la planification de chemin avec un critère énergétique pour les robots mobiles de façons générales ou les UGV (*Unmanned Ground Vehicle*). Les travaux de la première catégorie essayent d'améliorer les logiciels d'aide à la navigation routière conçus pour les véhicules conventionnels afin de combler les insuffisances de ces logiciels avec les véhicules électriques [138]-[148]. Par exemple, l'atout de freinage régénératif donné par ces véhicules conduit à des coûts négatifs sur certains arcs

du graphe du réseau routier qui ne sont pas acceptés par l'algorithme de *Dijkstra* généralement utilisé pour la recherche de chemin des véhicules conventionnels. Plusieurs chercheurs ont travaillé sur la planification de chemin et la navigation autonome. La majorité des travaux considèrent seulement la distance comme indicateur de performance, pour se déplacer d'un point *A* vers un point *B* [45], [149]-[152]. Par contre, dans le monde réel, il y a des cas où la surface rugueuse de la route, par exemple, rend le chemin le plus court en distance loin d'être optimal. Cependant, rares sont les travaux qui exploitent l'information de coût énergétique associé aux sous-trajectoires pour planifier le chemin parcouru [153]. Nous allons en citer quelques-uns ci-dessous. Mei et al. [66] s'intéressent au problème de *PP* pour les multirobots. Ils abordent le déploiement de plusieurs robots qui réalisent collectivement certaines tâches d'exploration de la manière la plus économe en énergie. D'autres recherches se focalisent sur l'exploration totale de l'environnement, dont le travail présenté dans [154]. Les auteurs proposent un algorithme génétique pour optimiser la solution au problème de planification du chemin choisi en tenant compte de la consommation énergétique, des obstacles et des élévations. Ce travail ne considère pas le facteur résistance aux roulements qui a une influence majeure sur la consommation énergétique. D'autres chercheurs, comme Khuller et al. [155] traitent le problème du voyageur de commerce qui utilise une matrice de coût pour chaque paire de nœuds. Cependant, on distingue une étape floue dans ce travail qui est la planification de chemin à source unique pour connaître le coût réel de voyage entre les nœuds. Il existe également des travaux tels que [155]-[157] qui s'intéressent à réduire la consommation énergétique d'un robot mobile, mais en s'accentuant principalement sur l'aspect contrôle. Par exemple, Liu et al. [156] proposent un modèle de commande pour ajuster le mouvement du robot et de déterminer les sources des erreurs systématiques affectant sa cinématique. Cette approche permettra de mieux suivre les trajectoires préalablement planifiées. Les travaux de Plonski et al. [158] présentent une méthode pour construire une carte de disponibilité de l'énergie solaire. Ils utilisent une stratégie de programmation dynamique pour planifier un chemin à travers la carte précédemment construite. Zhao et al. proposent une amélioration de l'algorithme de *A** pour le problème de planification de chemin avec une consommation d'énergie minimale [159]. Ils traitent l'impact des différentes conditions du sol et de l'angle de virage sur la consommation de l'énergie du robot, ainsi que, les contraintes multipoints

du robot mobile sur l'angle de virage. Cependant, le document ne contient pas de détail sur le calcul de l'énergie consommée. De plus, ils attribuent des coefficients grossiers pour les conditions de sol sans expliquer la source de ces chiffres. D'autre part, la méthode utilisée ne garantit pas le retour d'une solution optimale, car elle peut éliminer des sous-trajectoires optimales. Wang et al. [160] proposent un algorithme de planification de trajectoire, *SLEEP* (*Staying-aLive and Energy-Efficient Path planning*), qui assure la recharge avant une décharge totale et qui se préoccupe de l'efficacité en énergie. La méthode de recherche Tabou était employée pour résoudre ce problème. D'autre part, pour pouvoir faire une recherche de chemin le plus court/optimal par rapport un critère donné dans un graphe (espace de recherche), il faut, en premier lieu, attribuer des poids (coûts) pour chaque segment (arêtes). Pour le critère énergétique, il faut entre autres estimer, en premier lieu, l'énergie consommée pour traverser un segment donné. Plusieurs chercheurs ont travaillé sur l'estimation de l'énergie consommée par une plateforme que ce soit un robot mobile ou un véhicule électrique de façon générale [161]-[164]. Par contre, certains ont omis quelques paramètres qui sont indispensables pour la bonne prédiction, tels que la masse, la friction du sol, l'élévation. Or, la négligence de ces dernières affecte la planification de recharge et la réussite de la mission, d'où l'importance de développer un algorithme d'apprentissage automatique pour apprendre de façon naturelle la consommation d'énergie des différents chemins alternatifs.

2.5 Synthèse sur les méthodes de planification globale

Le fonctionnement d'un AMR dépend de sa batterie rechargeable. Il est indispensable de s'assurer que les batteries ne sont pas épuisées de manière inattendue avant la fin de mission. De nombreux travaux ont été réalisés sur la prédiction de l'énergie consommée pour les véhicules électriques (autonomes ou pas). Cependant, la précision de ces derniers reste irrégulière et dépend de facteurs extrinsèques (environnementaux), souvent négligés dans la littérature. Or, la prédiction de l'énergie consommée utilise généralement un modèle de véhicule qui dépend de différents paramètres complexes à déterminer. Une prédiction inexacte de la consommation d'énergie par segment de rue impacte directement la planification de chemin écoénergétique. C'est pourquoi nous nous sommes concentrés, dans la première partie de notre travail sur la compréhension et la quantification adéquate des pertes associées aux paramètres extrinsèques

du véhicule, mais aussi sur le développement d'un outil de planification écoénergétique sur cette base. Cette partie vise particulièrement les robots électriques autonomes à faible vitesse dans les entrepôts (AMR) pour qui le frottement roue-chaussé peut conduire à une résistance au roulement variable qui impacte la consommation d'énergie.

Peu de travaux en robotique mobiles s'intéressent à la planification de chemin écoénergétique, contrairement à ceux qui considèrent le chemin le plus court comme indicateur de performance [10]. Parmi ces travaux, on a noté également l'imprécision de leur modèle pour bien prédire les besoins énergétiques. En effet, quelques chercheurs ont choisi d'utiliser des méthodes métaheuristiques pour accélérer le processus de recherche de chemin. Ces méthodes sont intéressantes quand il s'agit d'un large réseau ou quand le temps de calcul est primordial (temps réel). Cependant, la solution obtenue est une solution approchée et non optimale. Rares sont les travaux qui fusionnent les axes de recherches sur la navigation autonome évitant les obstacles statiques, la navigation qui consomme moins d'énergie, la navigation qui tient compte du temps opérationnel. Hors, la minimisation de la consommation d'énergie de la même mission diminue la profondeur de décharge de la batterie et prolonge le temps d'opération de véhicule et prolonger la durée de vie de la batterie de l'AMR. C'est pourquoi cette thèse vise l'optimisation énergétique des AMR utilisés dans les usines et les entrepôts, à travers la gestion de trajectoire de ces derniers. Il faut mentionner que l'effet de la température de l'industrie sur la performance de la batterie et son autonomie n'est pas considéré dans cette étude étant donné qu'il est relativement faible par rapport aux autres variables considérées.

Pour la réalisation de cet objectif et vu le manque de travaux qui considèrent la consommation énergétique dans la *GPP*, une revue sur les méthodes classiques utilisées pour la planification de chemin global en général a été effectuée, afin de concevoir une méthode de planification plus économe énergétiquement [165]. Comme précédemment annoncée, la procédure de planification de chemin d'un AMR se réalise en deux étapes clés. La première étape est la représentation de l'environnement de déploiement du robot qui produit un graphe utilisé dans la deuxième étape qui est la stratégie de recherche de chemin.

Les approches de représentation de l'espace peuvent être regroupées en trois catégories, à savoir la représentation basée sur une grille d'occupation, la représentation basée sur une carte de trajectoire et la représentation topologique [50], [53], [166]. La représentation en

grille est référée à la décomposition en cellule uniforme et elle est effectuée en décomposant la carte de l'environnement de l'AMR en grille d'une certaine résolution avec des cellules occupées (qui reflètent les murs, les stations, les tables, et tous les obstacles statiques) et d'autres libres. Ces derniers représentent les positions de passage permises. L'enjeu majeur de cette méthode est la définition de la taille de grille qui a un effet direct sur la précision de la recherche de chemin. Par ailleurs, la représentation basée sur une carte de trajectoire, quant à elle, reflète la représentation de la *roadmap* en utilisant des courbes de trajectoire de mouvement du robot [50]. Ces dernières sont couramment générées par le graphe de *Voronoi* ou bien le graphe de visibilité. En outre, la représentation topologique contient des informations de l'environnement réel de l'AMR et elle spécifie des informations nécessaires telles que les intersections, les points d'intérêts (positions de ramassage ou dépôt), les points de recharge ainsi que les connections entre ces points par des arcs ou arêtes pondérées. Cette flexibilité et extensibilité fait de cette représentation une bonne alternative de représentation de graphe d'industrie. De plus, pour le cas où, l'énergie consommée est considérée lors de la planification de chemin, des informations supplémentaires devront être disponibles telles que la friction, le trafic, la masse des objets transportés et autres. La représentation topologique est compatible pour supporter ces informations. Cependant, la capacité de décrire les informations locales est relativement faible. Cette limitation est négligeable, étant donné que le planificateur local s'occupe de la génération des chemins locaux en créant à son tour une représentation de l'environnement en temps réel avec une résolution plus petite.

Une revue sur des méthodes de planification de chemin appliquées sur des cartes *OGMs* à deux dimensions était réalisée par Tsardoulias et al. [45]. Selon sept indicateurs de performances, ils évaluent quatre familles algorithmiques (*PRM*, les graphes de visibilité, le *RRT* et les méthodes d'échantillonnage spatial) en les testant sur cinq cartes *OGMs* de nature différente. Les auteurs démontrent que les méthodes d'échantillonnage spatial présentent le meilleur choix d'approches utilisées pour la planification de chemin dans les *OGMs*. Ces méthodes avaient un temps d'exécution bas, des chemins résultants courts sans virages excessifs et un taux de réussite élevé. Parmi les méthodes d'échantillonnage spatiales, on trouve la décomposition en cellules uniformes (*grid-graph*) que nous adoptons, dans un premier lieu, dans les méthodes proposées dans le prochain chapitre (*EE-GPP*). Dans un second

lieu, nous utilisons la représentation topologique pour la méthode *EE-GPP*, pour les raisons susmentionnées.

Concernant la deuxième étape de stratégie de recherche de chemin, les méthodes existantes dans la littérature peuvent être classées en (i) méthodes conventionnelles telles que la programmation entière mixte, *AD*, *A**, *D**, et *APF* et (ii) méthodes métaheuristiques ou méthodes basées sur l'intelligence artificielle, comme l'algorithme génétique, l'algorithme de colonies de fourmis, l'optimisation par essais particuliers, le réseau de neurones artificiels.

Une étude comparative entre quelques méthodes utilisées dans la planification de chemin des *AMR* est effectué par Wahab et al.[167] (les méthodes sont : *RRT*, *PRM*, *APF*, l'algorithme de Dijkstra, l'algorithme génétique (*GA*), l'optimisation par essaims de particules (*CPSO* (*Constricted Particle Swarm Optimization*)), l'évolution différentielle (*DE*) et *Cuckoo Search* (*CSA*)). Ils démontrent que l'*AD* est la méthode la plus performante, étant donné qu'elle utilise des connaissances préalables. La performance de cette dernière est utilisée comme valeur de référence puisqu'elle offre le meilleur résultat possible dans les configurations expérimentales proposées par les auteurs. De plus, sans considérer l'*AD*, l'étude démontre que l'algorithme *CPSO* surpasse, majoritairement, toutes les autres approches de planification conventionnelles et basées sur des métaheuristiques précédemment indiquées en considérant les facteurs de performance suivant : le temps d'exécution, la cohérence du tracé de la trajectoire, la consommation de batterie et la cohérence de la trace de la trajectoire et la distance parcourue. Par contre, elle a été surpassée par d'autres méthodes méta-heuristiques pour le critère de performance qui concerne le nombre d'itérations de convergence [167].

— Constatations importantes :

L'utilisation des algorithmes de planification de chemin exactes susmentionnés considérant seulement le critère de distance tel qu'ils sont présentés peut écarter de meilleures solutions écoénergétiques. En outre, les méthodes heuristiques à leurs tours peuvent donner des approximations pour un meilleur temps de calcul, mais il risque de fausser des éléments relatifs à la prise de décision. En effet, on souligne deux constatations à ce sujet. La première concerne, la possibilité d'avoir plusieurs chemins optimaux avec le même coût optimal en distance. Par exemple, la distance *Manhattan* peut être présentée de différente manière avec le même nombre de pas (voir figure 2-12-a). D'un point de vue longueur de chemin à traverser, les

quatre chemins présentés dans la figure ont le même nombre d'unités, à savoir 12 (pas) et le choix entre ces derniers est souvent au hasard (n'a pas d'importance). Cependant, si on considère le critère énergétique et particulièrement d'autres facteurs de l'environnement, tel que la condition du chaussez (voir 2-12-b), le chemin *path 3* qui évite la surface rugueuse pourrait être une meilleure alternative. Par conséquent, le choix de la stratégie de recherche de chemin n'est pas suffisant pour une bonne *GPP* d'un *AMR*. La deuxième remarque, quant

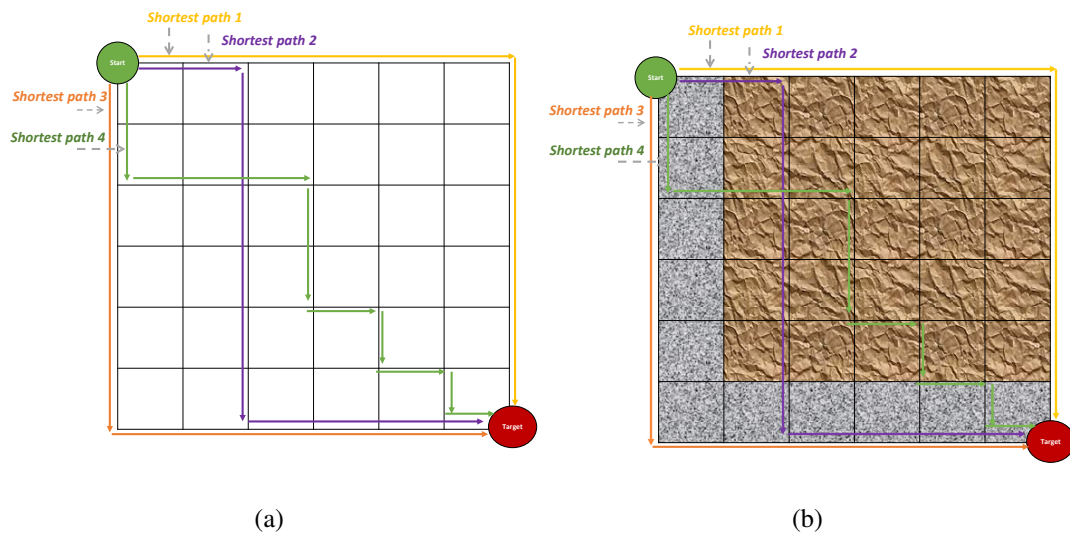


FIGURE 2-12 (a) Exemple de différents chemins *Manhattan* avec la même longueur dans une représentation de carte *2D* avec l'information distance ; (b) Représentation de la même carte avec une information supplémentaire concernant la friction de l'espace de déploiement (carte *3D*) et changement de perspectives

à elle, s'adresse à l'inconvénient des méthodes heuristiques dans certaines conditions où ils peuvent fausser la valeur de la distance qui sépare deux points. Par exemple, plusieurs approches parmi celles décrites ci-dessus utilisent la distance euclidienne ou de *Manhattan* pour approximer la distance la plus courte entre deux points (par exemple, l'heuristique utilisée par *A** pour s'approcher rapidement du nœud destination). Toutefois, cette valeur peut réussir dans certains cas et elle peut échouer dans plusieurs autres. En effet, comme on peut le constater dans la figure 2-13, la distance du plus court chemin réel est loin des distances euclidiennes et de *Manhattan*. En fait, partant du point *Start* en vert et pour se rendre au point *Target* en rouge, le robot doit faire le tour pour éviter l'obstacle en noir). Par ailleurs, il est important de considérer cette notion, particulièrement, lorsqu'on parle d'une amélioration de l'efficacité

énergétique et du temps et que l'exactitude des méthodes est parmi les objectifs estompés, particulièrement dans un contexte industriel, où chaque petit gain économique compte.

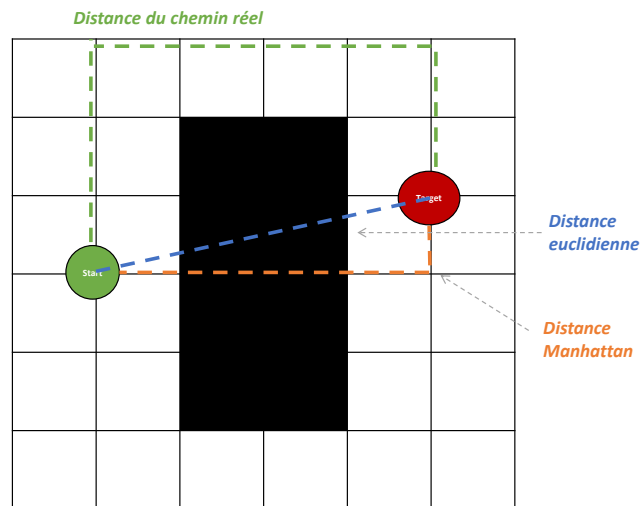


FIGURE 2-13 Différence entre la distance euclidienne, de *Manhattan* et la distance la plus courte du chemin réel

2.6 Routage de préparation de commandes d'un *Ps-AMR* (planification de chemin multi-destinations)

Dans cette section, nous allons aborder la planification de chemin multideestination à travers la résolution de la problématique particulière de routage de prélèvements de commandes (*Order Picking Problem*) écoénergétique d'un *Ps-AMR* dans un entrepôt, en considérant les masses des articles à transporter.

Comme précédemment présenté, le problème de routage unique d'un *PS-AMR* porte sur la détermination du chemin que le robot doit parcourir pour collecter un ensemble d'articles demandés par des clients internes ou externes, dans un entrepôt de distribution. Ce problème, bien connu, est appelé problème de préparation de commandes (*OPP (Order Picking Problem)*) dans les entrepôts rectangulaires. Des méthodes exactes et heuristiques ont été largement utilisées pour traiter l'*OPP* dans la littérature. L'*OPP* est représenté comme un cas particulier

du problème classique du voyageur de commerce (*TSP*), où le vendeur est l'*AMR*, et les villes sont les éléments à collecter [168].

2.6.1 Définition du problème Steiner TSP

La problématique de *OPP* est considérée comme un problème *Steiner TSP (STSP)*. Il s'agit d'une extension du problème classique du voyageur de commerce où on dispose d'une liste de villes, ainsi que des longueurs de routes qui les relient. Certaines villes sont à visiter (obligatoires) et d'autres de passages (peuvent être traversées ou non). L'objectif est de trouver le chemin optimal qui visite chacune des villes obligatoires et revient ensuite à la ville d'origine [169]. Le *STSP* est décrit par un graphe orienté $G = (V, E)$ où V est un ensemble de sommets et E est un ensemble d'arêtes. Dans ce graphe, $P \subseteq V$ représente les sommets requis et $V \setminus P$ représente les points de *Steiner*. Une tournée *Steiner* de G est une marche fermée qui visite chaque sommet de P au moins une fois. Il y a donc deux différences entre une tournée *Steiner* et une tournée de voyageur de commerce ordinaire (*TSP (Travel Salesman Problem)*). La première différence est que dans une tournée *Steiner*, les points *Steiner* peuvent ne pas être visités. La deuxième différence est qu'un tour *Steiner* peut contenir plusieurs sommets plus qu'une fois [170]. Dans le *STSP* classique, le but est de minimiser la longueur d'un tour de *Steiner* dans le digraphe G . Au lieu de cela, dans notre étude, nous visons à trouver la tournée *Steiner* minimisant la consommation de l'énergie et du temps, simultanément.

La configuration classique des entrepôts rectangulaires à allées parallèles contient des croisements seulement dans les extrémités des allées, appelées entrepôts uni-bloc (voir figure 2-14a). Cependant, il existe, également, des entrepôts muti-blocs tels que l'exemple illustré dans la figure 2-14b. En outre, une illustration d'un entrepôt et son graphe correspondant est représentée dans la figure 2-15. Le même principe de représentation de graphe d'entrepôt est utilisé dans la définition de tous les travaux de la littérature que nous décrivons ci-dessous. En effet, la problématique de *OPR* est très abordée dans la littérature, particulièrement pour les opérateurs "*picker*" humain et plusieurs méthodes ont été proposées. Ces méthodes sont classées en deux grandes catégories : les méthodes exactes et les méthodes basées sur des heuristiques (décrites ci-dessous).

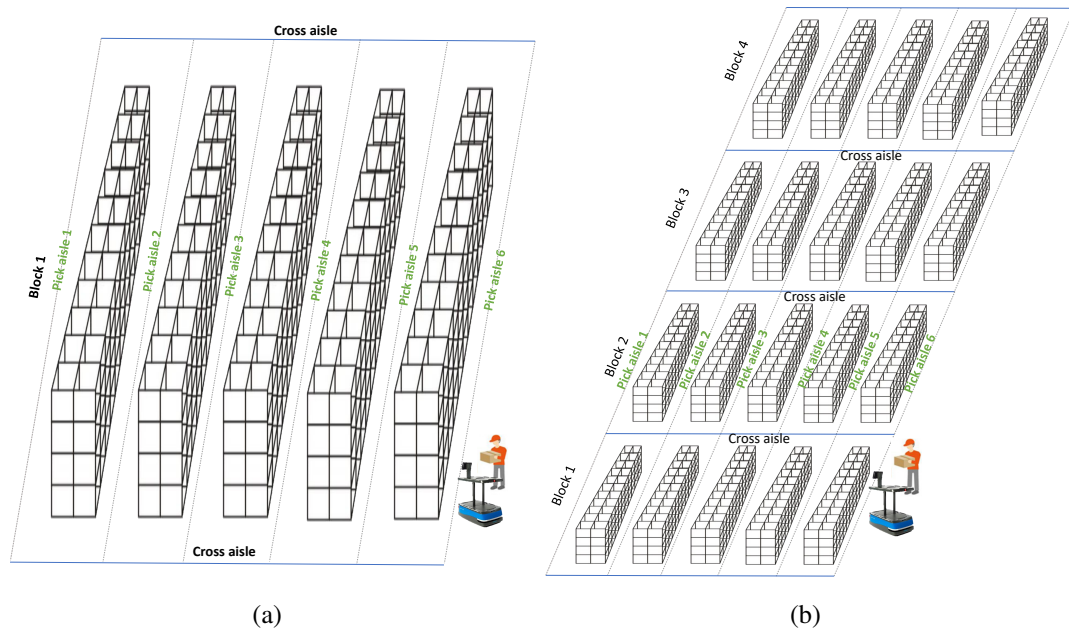


FIGURE 2-14 (a) Un entrepôt Uni-bloc (b) Un entrepôt Multi-blocs

2.6.2 Méthodes exactes pour la résolution du problème de routage de préparation de commandes

2.6.2.1 Méthodes de résolution du problème de routage de préparation de commandes basées sur la programmation dynamique

En 1983, Ratliff et Rosenthal [172] ont présenté une solution en temps polynomial comme une stratégie exacte de préparation de commandes dans un entrepôt uni-bloc. Cette méthode très connue, communément appelée méthode *RR* (pour *Ratliff* et *Rosenthal*), a été étendue par Roodbergen et De Koster pour convenir à des entrepôts avec deux blocs [173]. Par la suite, Löffler et al. et Masae et al. ont traité des problèmes de préparation de commandes avec des points de départ et d'arrivée arbitraires en appliquant les concepts de *RR* [174], [175]. Plusieurs autres travaux de recherches adoptent cette méthode conventionnelle pour la résolution du problème *OPR* et ils l'améliorent ou l'adaptent à un besoin particulier [173], [176], [177]. Par exemple, Pansart et al. ont proposé une approche de programmation dynamique étendant les algorithmes *RR* [177] pour s'adapter à des applications réelles. Ils réussissent à créer un algorithme exact pour n'importe quel nombre d'allées transversales, cependant, ce nombre doit être limité, car l'approche a une complexité exponentielle en nombre de couloirs transversaux

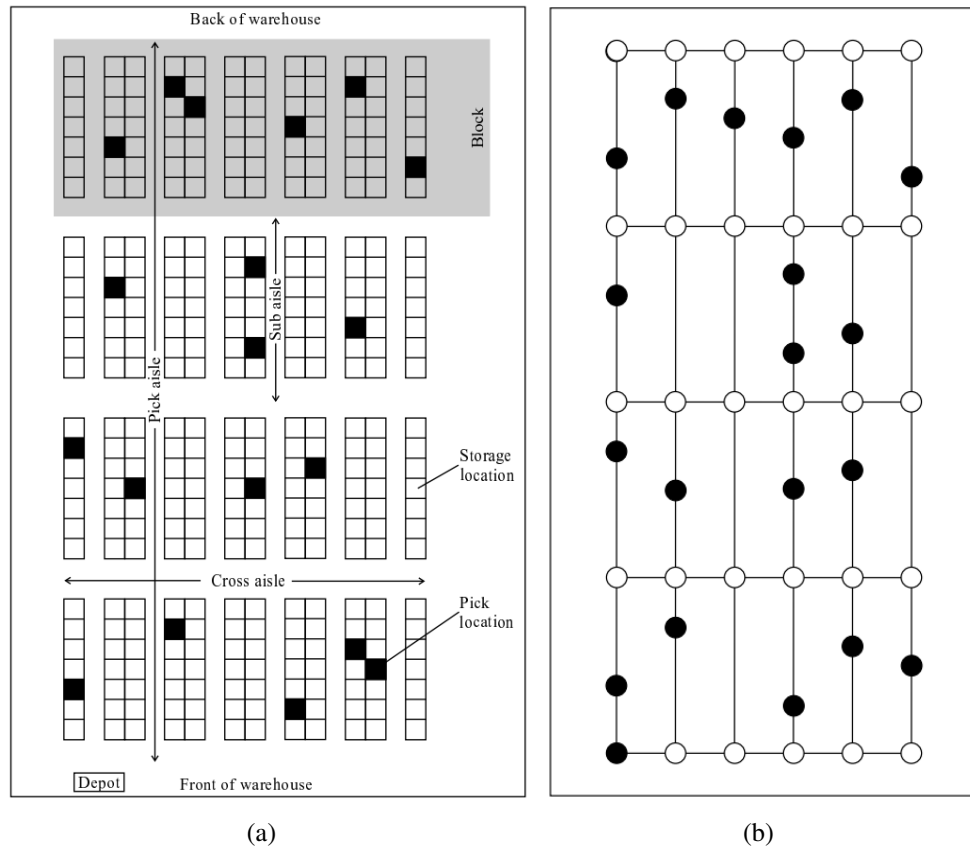
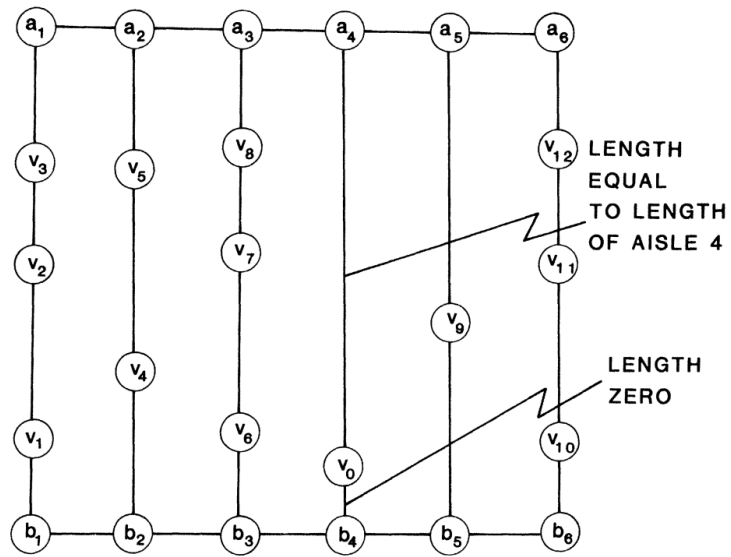


FIGURE 2-15 Une illustration d'un entrepôt et son graphe correspondant (a) Un exemple typique d'un entrepôt multi-blocks dont les carrés noirs représentent les emplacements de stockage que le robot doit visiter (b) Le graphe de l'entrepôt correspondant (les emplacements de stockage sont remplacés par des nœuds noirs (requis) et les croisements sont représentés par des nœuds blancs (de passage). [171]

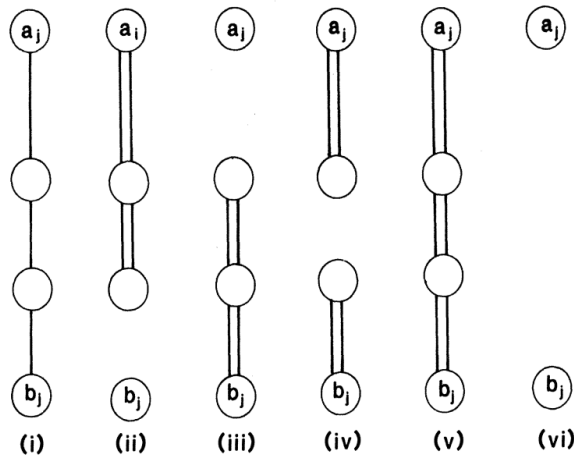
(*cross-aisles*). L'avantage des approches *RR* est leur capacité de résoudre d'une manière optimale l'*OPR* avec un algorithme de complexité temporelle linéaire par rapport au nombre d'allées. Néanmoins, le principe de ces méthodes est basé sur la configuration rectangulaire particulière à allées parallèles des entrepôts et elle n'est pas applicable à d'autres configurations ou graphes non traditionnels. Par exemple, la figure 2-16 montre les mouvements possibles dans les allées verticales et transversales définies par *RR*, pour le graphe d'entrepôt 2-16.a (a_j représentent les nœuds des allées en arrière de l'entrepôt (*back cross aisles*) et b_j représentent les nœuds des allées en avant de l'entrepôt (*front cross aisles*)). On remarque que l'algorithme limite les possibles configurations des arcs pour toute les allées j dans un sous-graphe de tournée optimale à six possibilités (voir la figure 2-16.b : (i) traverser l'intégralité de l'allée ; (ii) faire un demi-tour pour retourner sur a_j ; (iii) faire un demi-tour pour retourner sur b_j ; (iv) faire un demi-tour sur les deux bords a_j et b_j ; ou (v) faire un aller-retour sur l'intégralité de l'allée). De même, ils définissent cinq possibilités pour les allées horizontales *cross aisles* (voir la figure 2-16.c). Ceci montre que l'algorithme se base, principalement, sur ces configurations pour définir les états de leur programme dynamique, sauf que ces derniers peuvent changer dans des entrepôts non conventionnels ou avec un changement d'aménagement. Cette limitation empêche les méthodes *RR* de supporter d'autres contraintes telles que le critère énergétique [177], [178].

2.6.2.2 Formulation du programme linéaire mixte en nombres entiers

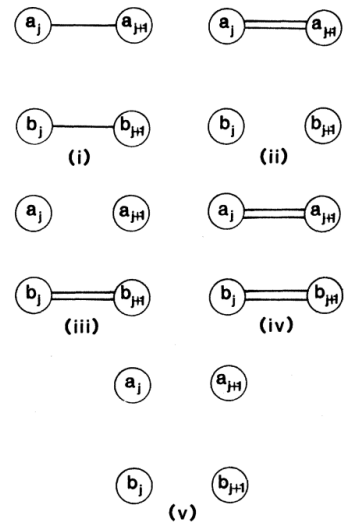
Plusieurs travaux considèrent la formulation mathématique en programmation linéaire mixte en nombres entiers pour résoudre le problème de routage de préparation de commande (*Mixed-Integer Linear Programming (MILP)*). Par exemple, Pansart et al. [177] présentent un deuxième algorithme exact pour résoudre l'*OPP*. Ils démontrent que le problème peut être résolu de manière optimale avec la *MILP* en utilisant une formulation creuse renforcée par un pré-traitement et des inégalités valides. Le problème est considéré comme un *Steiner TSP* et les auteurs ont utilisé une formulation compacte de flux de produit unique (*compact single-commodity flow formulation*) qui a été proposée dans un autre travail de Letchford et al. [179]. Ainsi, le préparateur doit déposer une unité d'articles à chaque fois qu'il en prélève un (principe du flux). Il est important de mentionner que la direction de parcours



(a) Graphe de prélèvements de commandes conventionnel (de RR)



(b) Les possibles configurations des arcs pour toute les allées j dans un sous-graphe de tournée optimale



(c) Les possibles configurations des arcs pour le croisement entre l'allée j et l'allée $j + 1$ dans un sous-graphe de tournée optimale.

FIGURE 2-16 Les mouvements possibles dans les allées verticales et transversales définis par *Ratlift et Rosenthal* [172]

est cruciale dans ce contexte, mais cette étape de post-traitement pour trouver la séquence de tournée de sélection à partir du sous-graphe de tournée résultant n'est pas détaillée, dans la majorité des travaux qui utilisent *MILP* pour la résolution du problème. Ils focalisent sur la détermination des arêtes qui construisent le sous-graphe optimal. C'est aussi le cas de Scholz et al. qui suggèrent une solution différente au problème de routage [180] basée sur de nouvelles formulations mathématiques afin de prendre en compte la spécificité de l'agencement de l'entrepôt. Leurs formulations et celle de Pansart et al. ont une contrainte principale qui consiste à imposer le ramassage d'un article au premier passage sur un sommet requis. Cependant, si la consommation d'énergie est prise en compte, cette contrainte peut aboutir à une solution non-optimale.

2.6.3 Méthodes heuristiques pour la résolution du problème de routage de préparation de commandes

En plus des méthodes exactes, des méthodes heuristiques ont été également proposées dans la littérature pour résoudre l'*OPR*. Par exemple, la méthode traversée (ou *S-shape*), plus grand écart (*largest Gap*), retour (*return*), point médian (*midpoint*) et composée (*combined*) [181], [182]. La figure 2-17 montre le déroulement de ces stratégies de routage. De plus, nous expliquons brièvement le principe de chaque méthode ci-dessous :

- La méthode *S-shape* (appelée aussi traversée (*traversal*) : le préparateur de commandes commence par traverser entièrement les allées qui contiennent minimum un article demandé dans les positions de prélèvements à droite ou à gauche. Il continue cette procédure jusqu'à ce que tous les articles de la commande soient récupérés.
- La méthode *largest Gap* : le préparateur de commandes traverse en premier lieu la première allée jusqu'à la *cross-aisle* arrière. Ensuite, il rentre dans chaque allée comportant des articles demandés en laissant le plus grand écart. Sachant que l'écart le plus grand est l'espace laissé entre deux éléments demandés dans une allée ou entre les sorties d'allée. Ce dernier est l'élément clé pour définir la partie avant et la partie arrière de chaque allée de la tournée de prélèvement.
- La méthode *return* : dans cette heuristique, le préparateur de commandes entre dans chaque allée contenant au moins un article demandé depuis la *cross-aisle* de l'avant

de l'entrepôt, il collecte tous les articles demandés dans cette allée, puis il retourne à l'extrémité avant de l'allée.

- La méthode *midpoint* : l'entrepôt est divisé en deux moitiés égales, appelées parties avant et arrière. Le préparateur de commandes ramasse les articles de la partie avant de l'entrepôt, en entrant dans les allées contenant des articles requis et il quitte chaque allée du côté où il y est entré. Une fois terminé, l'opérateur se déplace vers la partie arrière de l'entrepôt pour compléter sa récolte d'article en visitant les allées requises de la même façon.
- La méthode *combined* : comme son nom l'indique, cette méthode combine l'heuristique *return* et l'heuristique *S-shape*. Le préparateur de commandes décide pour chaque allée s'il traverse entièrement l'allée ou bien retourne sur le *cross-aisle* de l'avant selon l'option qui donne le chemin le plus court.

Les heuristiques sont les politiques les plus utilisées pour résoudre l'*OPP*, car la route optimale peut sembler illogique pour un opérateur humain [183]. Cependant, dans le cas d'un système semi-autonome ou complètement autonome où le robot est le leader, ce n'est plus un problème. Koster et Poort [184] présentent une comparaison pratique entre les algorithmes exacts et les heuristiques en comparant la méthode *S-shape* avec la programmation dynamique. Dans la stratégie en forme de *S*, les allées sont entièrement parcourues s'il y a des produits à prélever, sinon, les allées sont sautées. Par exemple, lorsque le préparateur part du coin inférieur gauche de l'entrepôt (dépôt) et entre et quitte les allées de différents côtés (avant et arrière), il retourne au dépôt après avoir terminé le prélèvement, ce qui donne un itinéraire en forme de *S*. Koster et Poort concluent que malgré la facilité d'utilisation de la stratégie *S-shape*, les algorithmes optimaux apportent de meilleures économies en temps de trajet. Ce résultat motive l'utilisation d'algorithmes exacts. Pour plus de détails, Masae et al. [185] présentent une revue systématique de la littérature sur le routage des préparateurs de commandes dans les entrepôts. Ils résument la fréquence d'utilisation des différentes politiques de routage dans les systèmes manuels de préparation de commandes *picker-to-parts* à travers un graphique intéressant représenté dans la figure 2-18.

Dans les systèmes *PS-ARM*, le routage *S-shape* présente la méthode la plus utilisée [24]. Cette méthode est quasi-optimale lorsqu'il y a une densité de prélèvement excessive [186].

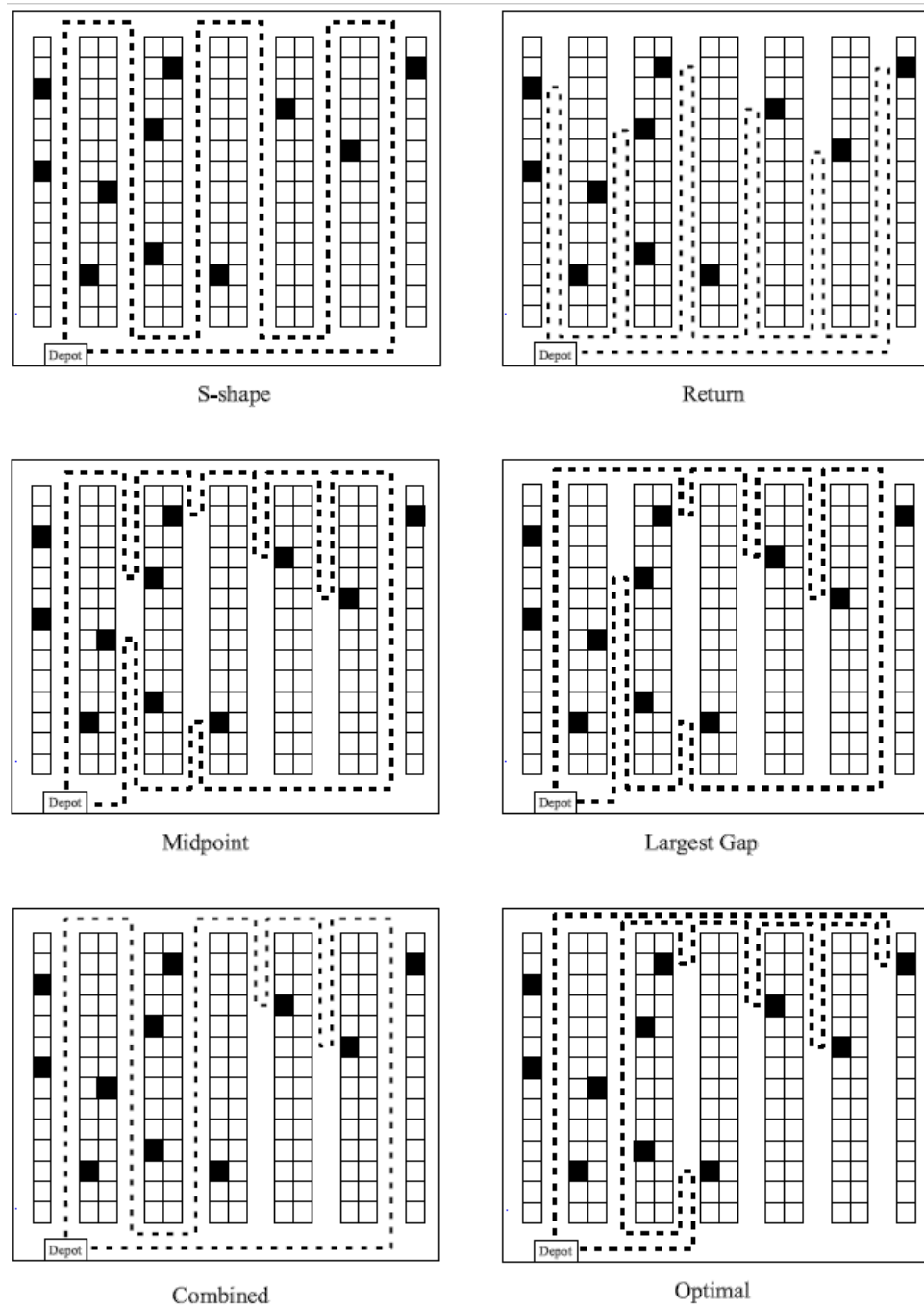


FIGURE 2-17 Quelques stratégies de routage de prélèvement de commandes [171].

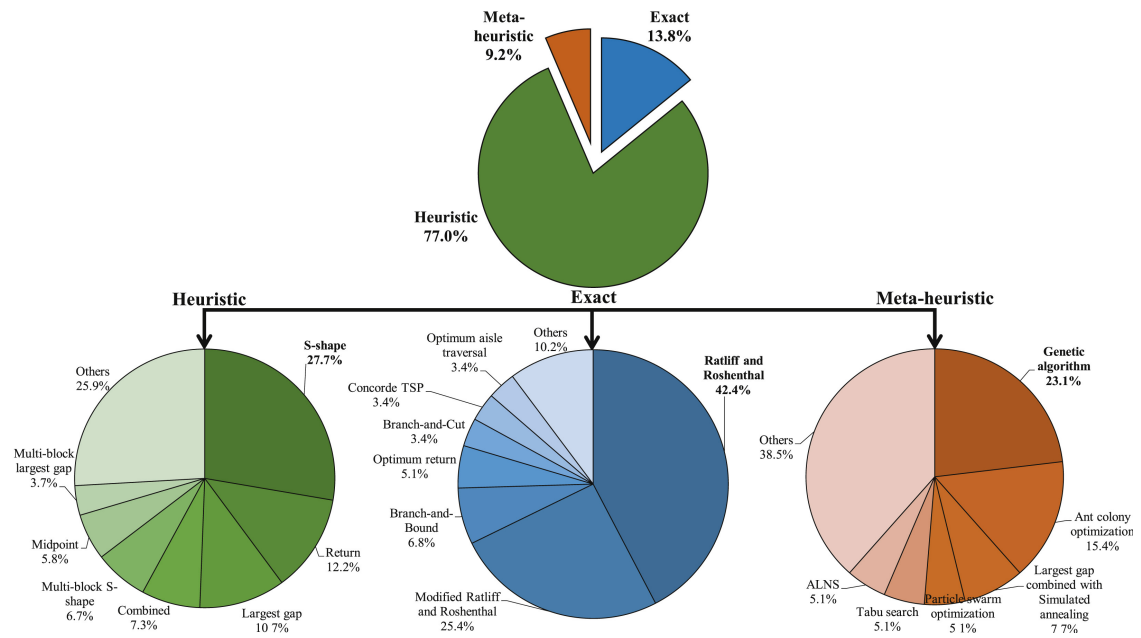


FIGURE 2-18 La fréquence d'utilisation des différentes stratégies de routage dans les systèmes manuels de préparation de commandes *picker-to-parts* [185].

Löffler et al. [187] étendent l'algorithme *RR* pour le problème de la sélection d'une seule commande avec des emplacements de début et de fin donnés. Ils présentent également une adaptation des stratégies *S-shape* et *largest Gap* pour s'adapter à la préparation de commandes assistée par *AGV* où les points de départ et d'arrivée peuvent être différents et ne se limitent pas à l'emplacement du dépôt. Cependant, cette méthode, ainsi que d'autres heuristiques, sont moins adaptées pour tenir compte des contraintes annexes, telles que la consommation d'énergie et la masse de la charge transportée.

Il est important de mentionner que l'objectif principal et commun des études susmentionnées est le temps et/ou la distance de déplacement. Cependant, ni la consommation d'énergie ni l'effet des masses sur la consommation d'énergie n'ont été pris en compte et étudiés.

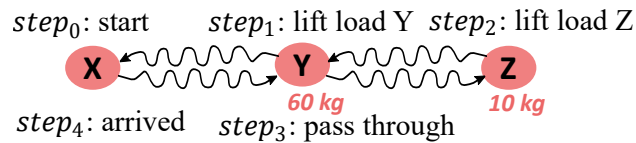
2.7 Synthèse sur le routage économe en énergie pour la préparation des commandes

Comme mentionné dans la section 2.6.1, l'objectif de la plupart des approches liées à l'*OPR* dans la littérature est de réduire le temps et/ou la distance de déplacement tout en négligeant la performance environnementale de l'entrepôt [59], [188]. Cependant, certaines études tentent de trouver un compromis entre le temps de trajet et la minimisation de la consommation

d'énergie afin d'optimiser l'acheminement de la préparation de commandes. Par exemple, Ene et al. [189] ont développé un algorithme génétique pour le problème de préparation de commandes dans les entrepôts. Cet algorithme vise à minimiser le temps de service ainsi que la consommation d'énergie via l'optimisation de la mise en lots des commandes et de l'acheminement du picking. Les auteurs ont prouvé par des exemples l'économie d'énergie significative lors de l'application de leur approche. Cependant, leur travail se base uniquement sur la vitesse verticale et horizontale et la distance parcourue du chariot élévateur pour estimer la consommation d'énergie. En outre, ils supposent une consommation d'énergie constante par unité de temps, négligeant les forces de charge et de résistance dans leur calcul. Rojanapitoon et Teeravaraprug [190] introduisent un nouveau modèle mathématique pour le routage des préparateurs qui minimise le temps de déplacement et la consommation d'énergie compte tenu d'une variation du niveau de trafic dans un entrepôt rectangulaire. Leur modèle mathématique est ensuite utilisé sur un logiciel de simulation informatique qui a été présenté dans un travail précédent [191]. Ils comparent leurs résultats avec le modèle échelonné dans le temps et les valident par la stratégie de recherche par force brute. Par rapport à la méthode qui optimise le temps, les auteurs ont rapporté que leur modèle proposé optimise à la fois le temps et l'énergie et permet d'économiser jusqu'à 17 % d'énergie. Cependant, ils ont exclu la possibilité de récupérer l'article demandé au deuxième passage (s'il y en a un) et ont négligé le sens de déplacement dans la construction d'une tournée de prélèvement complète, ce qui peut conduire à des solutions sous-optimales. Lee et al. [192] ont développé un algorithme dynamique intégré comme solution au problème de routage des chariots élévateurs électriques en fonction des contraintes de charge de la batterie. Autrement dit, l'algorithme prend en compte les itinéraires de prélèvement/rangement du chariot élévateur électrique et les horaires de charge de la batterie, ainsi que le nombre de chariots élévateurs électriques. En plus de cela, leur algorithme tient compte de la consommation d'électricité dans l'entrepôt. Aussi, Makris et al. [193] ont abordé l'OPP du point de vue de l'efficacité énergétique. Ils présentent un algorithme de routage basé sur le TSP afin d'obtenir un compromis entre le temps de trajet et la consommation d'énergie de la préparation des commandes dans l'entrepôt. Les auteurs excluent cependant le poids de leur évaluation de la consommation d'énergie. De même, la plupart des travaux de recherche ne considèrent pas la masse comme un facteur critique pour

les décisions d'acheminement des véhicules, en particulier en ce qui concerne les économies d'énergie. Elbert et Müller [194] étudient l'impact du poids de l'article transporté sur la vitesse du préparateur de commandes et le temps de déplacement lors d'une préparation de commandes manuelle du préparateur aux pièces. Dans leurs travaux, ils se concentrent sur le problème de l'affectation du stockage et proposent de nouvelles politiques d'affectation du stockage basées sur la classe de poids pour réduire le temps de déplacement.

Solution 1



Solution 2

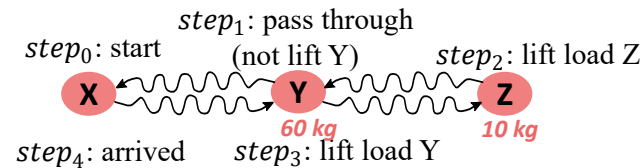
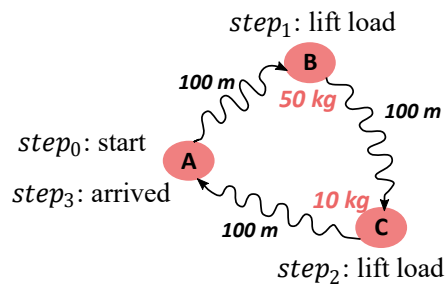


FIGURE 2-19 Potentiel de gain en énergie en ne soulevant pas l'article requis d'un sommet lors du premier passage (contrairement à la solution 1, la solution 2 retarde le ramassage d'un article requis)

Solution 1



Solution 2

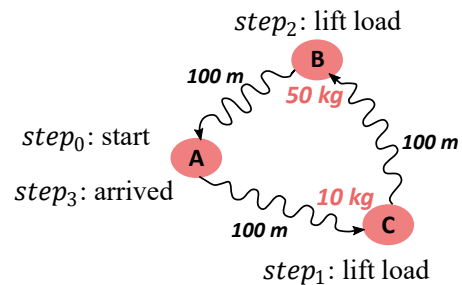


FIGURE 2-20 Potentiel de gain en énergie en changeant la direction dans laquelle les articles sont ramassés (la solution 2 tient compte de cet aspect alors que la solution 1 ne le fait pas (l'article B est plus lourd que l'article C))

Par conséquent, deux points importants sont négligés dans les études de recherche susmentionnées, qui sont examinées dans cette étude. Nous abordons ces deux points dans ce qui suit. Premièrement, dans la majorité des articles de recherche liés à la résolution de l'OPP, les contraintes appliquées dans la formulation mathématique forcent la solution à ne passer

qu'une seule fois sur chaque emplacement de prélèvement requis ou à prélever les produits de la commande lors du premier passage. Ces formulations conduisent à un résultat optimal ou quasi-optimal qui minimise le temps de trajet de la tournée de préparation des commandes. Cependant, la prise en compte du gain d'énergie dans la formulation d'optimisation n'a pas été prise en compte. Ce gain peut être obtenu en passant par un point voulu sans prendre la charge (pour ne pas la traîner) et en la reprenant au retour pour économiser de l'énergie. La figure 2-19-a montre un exemple de prise en compte du gain d'énergie en ne soulevant pas l'élément requis lors de la première passe. Dans cette illustration, nous supposons un graphe bidirectionnel avec un ensemble de sommets X, Y, Z et un ensemble d'arêtes $(X, Y), (Y, Z)$. Les flèches ondulées représentent les chemins les plus courts entre deux nœuds. La Police provinciale de l'Ontario tente ici de partir du point X , pour visiter des points Y et Z , puis pour revenir au point X avec un coût minime. Dans cet exemple, le plus court chemin résultant nécessite le passage deux fois sur le même nœud Y . La solution 1 ne tient pas compte du gain d'énergie et porte donc la charge dans Y tout le chemin à partir de Y à Z , puis de Z à Y . Cependant, la solution 2, qui prend en compte le gain d'énergie, sélectionne d'abord la charge en Z , et sur le chemin du retour, il prend la charge dans Y . Deuxièmement, la plupart des travaux discutés se concentrent sur la recherche d'un sous-tour optimal et considèrent la construction d'un tour complet comme allant de soi. Néanmoins, un changement de direction d'un même sous-tour peut apporter un gain d'énergie si la perte d'énergie liée au déplacement d'objets lourds est retardée. La Figure 2-20 donne un exemple d'économie d'énergie en changeant de direction. Nous supposons dans cette figure un graphe bidirectionnel différent avec trois sommets A, B, C , et trois arêtes $(A, B), (B, C)$ et (C, A) . Ici, le voyageur de commerce (l'AMR dans notre cas) tente de partir du point A , capte les charges des points B et C , et revient au point A avec un coût minime. Selon cet exemple, la charge dans B est plus lourde que C . La solution 1 ne tient pas compte du changement de direction et, par conséquent, elle prend d'abord la charge la plus lourde et la transporte jusqu'à C et revient à A . Cependant, la solution 2 sélectionne d'abord la charge la plus légère dans C . Ensuite, il prend le plus lourd dans B et revient à A . Ce changement de direction peut entraîner des économies d'énergie.

Ainsi, l'apport principal de la deuxième partie du travail est d'aborder ces deux points peu étudiés dans la littérature. Ainsi, dans notre approche appelée *Energy Efficient Order Picking*

Routing (EE-OPR), nous prenons en compte le temps, l'énergie et la masse dans la décision de planification afin de réaliser une tournée de préparation de commandes efficace pour un véhicule de manutention autonome. De plus, contrairement aux approches classiques basées sur la *RR* qui reposent principalement sur la structure particulière des entrepôts rectangulaires à allées parallèles pour créer les sous-problèmes d'un programme dynamique, l'approche proposée peut être utilisée pour d'autres entrepôts avec des dispositions différentes. En fait, afin de résoudre l'*OPP* tout en minimisant simultanément le temps et l'énergie, un programme dynamique est développé. Ce programme considère le problème comme un *STSP* écoénergétique et le transforme en un problème de chemin le plus court (*SPP (Shortest Path Planning)*) en créant un graphe d'état dynamique acyclique et en effectuant un processus de recherche de graphe.

Chapitre 3 - Approches pour la planification de chemin écoénergétique

Dans ce chapitre, nous allons expliquer nos deux méthodes proposées pour résoudre le problème de la planification de chemin écoénergétique d'un *AMR* qui minimise à la fois le temps et l'énergie consommée par ce dernier pour la réalisation d'une mission de transport donnée. La première méthode, appelée (*EE-GPP*), résout le problème de la planification de chemin globale d'un *AMR* (une planification source-destination). Cette approche considère le critère énergétique et le critère de temps pour planifier un chemin sans collision entre un point *A* et un point *B* tout en considérant les facteurs qui influencent la demande énergétique du mouvement du robot, particulièrement la résistance au roulement et le trafic sur les voies de passage. La deuxième méthode vise, cependant, la résolution d'un problème de planification de chemin multidestinations (avec une liste de destinations non ordonnées), précisément le problème de la planification de chemin écoénergétique d'un *Ps-AMR* pour un tour de ramassage de commande (*OP*) dans un entrepôt ou *DC*. Cette situation peut être, également, vue comme un problème de planification de chemin d'un point de départ *A* vers un point d'arrivée *B* avec la contrainte de passer sur des nœuds obligatoires (*picking locations*) d'une liste non ordonnée de commande à ramasser. De plus, le critère énergétique est considéré dans la formulation du problème en incluant la variable de la masse des articles de la commande client. Afin de rejoindre la littérature de cette problématique particulière d'un contexte de prélèvement dans un entrepôt, nous allons utiliser les termes les plus connus *OPR*, *OPP* ou *Steiner TSP*, pour désigner le problème. La méthode proposée, appelée *EE-OPR*, considère l'effet de la masse lors de la planification de chemin, étant donné que la masse change entre les points d'intérêts (le robot doit collecter des articles appartenant à une commande donnée depuis plusieurs emplacements de l'entrepôt).

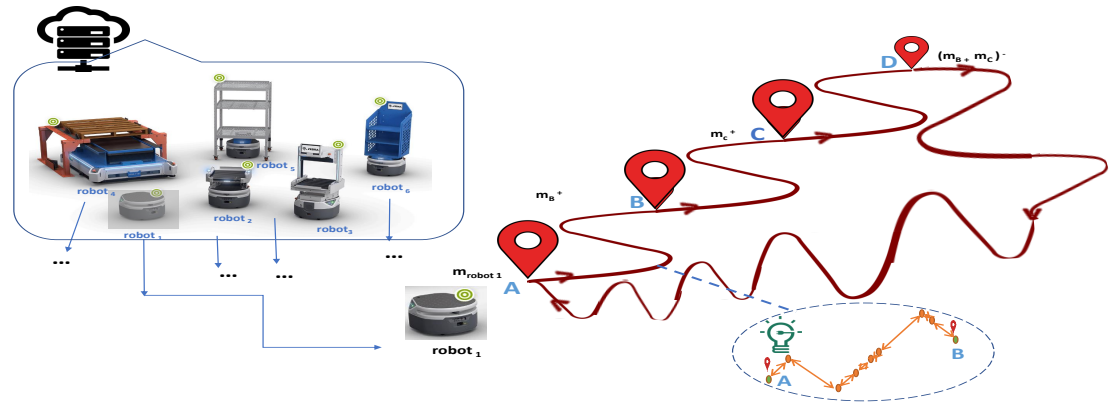


FIGURE 3-1 Un exemple illustratif d'un plan de collecte et de dépôt

3.1 Planification de chemin global écoénergétique

3.1.1 Introduction

Dans un contexte d'une manufacture intelligente, des robots de manutention autonomes de différents types et formes sont utilisés pour déplacer différentes charges à partir de plusieurs sources jusqu'à plusieurs destinations. Par exemple, la figure 3-1 schématise un plan de ramassage et de livraison (*Pickup and delivery plan*), montrant la variété de robots proposée par la marque Fetch robotique (différentes formes, capacités et masses) [18]. D'où l'importance de considérer la masse, la friction et le trafic dans la planification de chemin des AMRs, que nous expliquons en détail dans les sections qui suivent.

Comme précédemment expliquée, la planification de chemin se repose sur deux processus principaux qui sont la représentation de l'espace de configuration et la stratégie de recherche de chemins qui se base à son tour sur le critère d'optimisation. Dans notre cas, nous considérons les deux critères temps/énergie et nous étudions le compromis entre ces deux derniers. Dans cette optique, nous allons commencer par la représentation de l'espace de l'industrie visant à avoir un graphe de rue (*roadmap*) dans lequel nous allons effectuer la recherche de chemin (*graph search*). Deux représentations de graphes sont étudiées dans ce document : le graphe OGM (*Occupancy Grid Map*) et le graphe topologique, vu qu'ils représentent les graphes les plus communs en robotique mobile et étant donné le contexte de leur utilisation. Au début, le graphe OGM est utilisé pour étudier l'effet du facteur de résistance aux roulements sur la

décision de routage. Cette représentation donne plus de résolution permettant d'explorer plus de zones de l'espace de déploiement du robot. Par la suite, pour étudier le facteur trafic, un graphe de rue topologique est considéré afin d'avoir une structure plus réaliste pour modéliser le trafic par segment de rue. Nous présentons, en second lieu, la formulation du problème de programmation en nombre entier dans laquelle nous définissons les points à visiter et les masses à transporter. La définition des coups entre les nœuds successifs nécessite la définition de l'équation dynamique du véhicule, présenté subséquentement. Une stratégie générale de recherche de chemin écoénergétique appelée *EE-GPP* est enfin présentée. Cette dernière est convenable pour chacun des deux graphes précédemment définis (*Grid-graph* et graphe topologique) et tout autres graphes représentant l'industrie. La stratégie *EE-GPP* est basée sur la programmation dynamique avec une approximation successive inspirée de l'algorithme de *Dijkstra*. Elle considère tous les facteurs de l'environnement précédemment définis et elle cherche un bon compromis entre le temps et l'énergie.

3.1.2 Hypothèses

La liste suivante énumère les hypothèses que nous avons posées dans *EE-GPP* :

- La vitesse de l'*AMR* pour se déplacer sur un arc (i, j) est supposée constante parce que le robot se déplace d'une manière uniforme sur ce dernier.
- L'historique du trafic/congestion est exprimé par l'historique de nombre d'obstacles rencontrés sur chaque arête du graphe de la manufacture pour chaque jour de la semaine et chaque plage horaire de la journée. Il est supposé donné.
- La valeur médiane du nombre d'obstacles contournés sur une arête (i, j) est supposé le même pour les deux directions (ij) et (ji) .
- La pénalité attribuée à une arête donnée à cause du nombre d'obstacles contournés sur cet arête est supposée causer par l'extension de la distance parcourue sur ce segment de rue en effectuant des contournements d'obstacles. Elle est définie par le modèle de régression déduit à partir de la courbe de la figure 3-9 (description détaillée ci-dessous).
- La consommation de l'énergie causée par l'effet de l'accélération et de décélération n'est pas pris en compte de façon directe dans ce document, car le déplacement du robot

est supposé uniforme sur une ligne droite. Cependant, si le facteur trafic est considéré, le robot doit contourner les obstacles en effectuant des trajectoires en vagues et retourner sur son chemin global. Dans ce cas, nous supposons que le modèle de régression qui modélise l'augmentation de l'énergie consommée sur un segment de rue en fonction du nombre d'obstacles contournés sur ce dernier englobe tous les facteurs qui causent cette augmentation de consommation.

- La force de gravité est négligée dans ce travail, vu que la surface de l'industrie est supposée plane (absence de pentes).
- Seulement la consommation d'énergie relative aux mouvements horizontaux de l'AMR est considéré. Les pertes d'énergies causées par d'autres sources de consommation énergétique ne sont pas considérées dans cette étude, car elles n'affectent pas la décision du routage (ex. la manutention des charges, les capteurs utilisés par le robot, le froid).
- Le coût de la dégradation de la batterie peut affecter la décision du routage, mais ce critère n'est pas considéré dans cette thèse.

3.1.3 Représentation de l'espace par un graphe de rue et représentation des facteurs de l'environnement

3.1.3.1 Représentation de l'espace avec une carte de grille d'occupation et une carte de résistance au roulement

Dans un premier temps, une carte de grille d'occupation *OMG* est considérée. Un exemple d'une telle carte est illustré dans la figure 3-2 où toutes les cellules ont la même dimension (décomposition en cellules uniformes). Les cellules noires représentent les zones occupées par des obstacles (interdites C_{obs}), les cellules grises présentent l'agrandissement des obstacles (selon la forme du robot pour sa sécurité et sa dynamique) (des zones interdites C_{obs}) et les cellules blanches représentent les zones libres (permises C_{free}). Nous considérons une représentation ponctuelle d'un AMR à l'aide de sa position $q_i=(x_i,y_i)$ en nous basant sur la configuration spatiale discrète d'un robot introduite par Latombe et al. [75] constituée par des triplets $q_i=(x_i,y_i,\theta_i)$ où x_i,y_i et θ_i représentent, respectivement, l'abscisse, l'ordonnée et l'orientation du véhicule. Par conséquent, l'espace de configuration est l'ensemble de tous

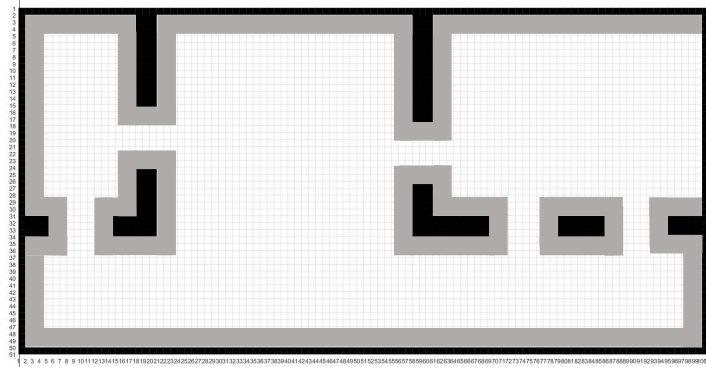


FIGURE 3-2 Exemple d'une carte métrique d'une manufacture et de l'espace de configuration du robot

les q_i atteignables par le robot. Étant le type de représentation *OGM*, l'*AMR* peut exercer huit déplacements possibles vers les cellules adjacentes (vers la cellule à droite, à gauche, en haut, en bas et vers les quatre directions diagonales). En connectant chaque cellule de l'espace de configuration à ses voisines, un treillis ou un graphe (*Grid graph*) est construit [195]. Un exemple de ce dernier est illustré dans la figure 3-3 où les cellules inoccupées sont représentées par des nœuds et les liens entre les nœuds adjacents sont des arêtes (*Grid graph* est représenté par un traçage rouge). Le robot est contraint à se déplacer entre les nœuds connectés avec des arêtes du graphe définissant ainsi les zones qui lui sont accessibles (les nœuds gris sont éliminés, car ils font partie de C_{obs}).

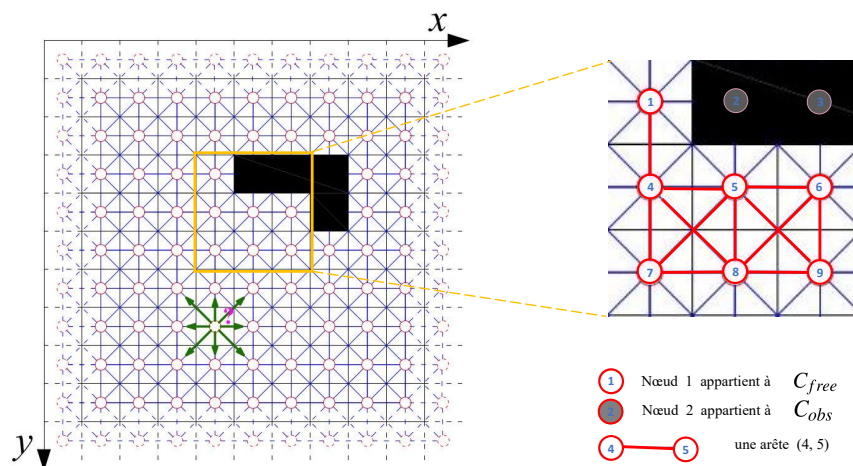


FIGURE 3-3 La définition des nœuds et des arêtes d'un "grid graph" à partir d'une carte binaire

De cette façon, nous définissons le "grid graph" $G_{OGM}(V_{OGM}, A_{OGM})$, tels que V_{OGM}

l'ensemble des nœuds et A_{OGM} l'ensemble d'arcs pondérés avec les coûts de transition entre chaque paire de nœuds adjacents. Ces pondérations des arêtes sont définies dans la section suivante.

3.1.3.2 Création des matrices d'adjacences distance et résistance au roulement

La représentation mathématique d'un graphe est réalisée par la définition d'une matrice d'adjacence nn (où n est le nombre de nœuds du graphe) ou bien d'une liste d'adjacence de longueur n . Ces représentations indiquent si deux points ou régions de la carte sont connectés et la pondération de leurs connexions (le poids de chaque arête (i, j) du graphe). Un exemple de représentation d'un graphe par une matrice d'adjacence et une liste d'adjacence est illustrée dans la figure 3-4. Les pondérations dans cet exemple sont binaires, cependant, le coût de transition entre les nœuds peut refléter d'autres critères. Étant donné que, l'approche

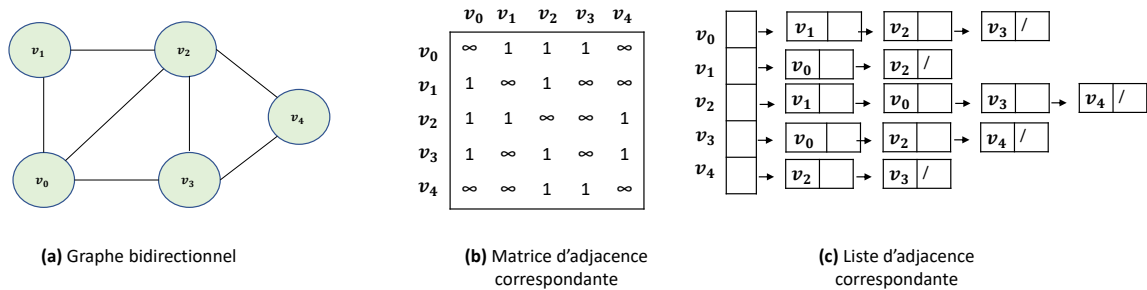


FIGURE 3-4 Un exemple de représentation d'un graphe par une matrice d'adjacence (b) ou une liste d'adjacence (c).

proposée vise la minimisation de deux différents critères : le critère de temps de déplacement, proportionnel au critère de distance et le critère de la consommation énergétique en tenant compte de l'impact de la résistance au roulement, trois matrices d'adjacences présentatrices de "grid graph" G_{OGM} sont définies comme suit :

— M_d^{OGM} "matrice distance" :

$M_d^{OGM}(i, j)$ est la distance euclidienne entre i et j s'il existe une connexion entre les nœuds i et j , sinon $M_d^{OGM}(i, j)$ est égal ∞ .

— M_r^{OGM} "matrice résistance au roulement" :

$M_r^{OGM}(i, j)$ est égal à la résistance au roulement moyenne sur une arête (i, j) s'il existe une connexion entre les nœuds i et j , sinon $M_r^{OGM}(i, j)$ est égal à ∞ .

— M_e^{OGM} "matrice énergie" :

$M_e^{OGM}(i, j)$ est la quantité d'énergie utilisée par le véhicule pour se déplacer de i à j s'il existe une connexion entre les nœuds i et j (calculer à l'aide du modèle dynamique de l'AMR). $M_e^{OGM}(i, j)$ est égal à ∞ , le cas échéant.

Chacune de ces trois matrices donne une couche de poids supplémentaires des arêtes qui appartiennent à G_{OGM} . Ce graphe peut alors être redéfini comme $G_{OGM}(V_{OGM}, A_{OGM}, [w_d, w_r, w_e])$, avec w_d, w_r , et w_e représentent les pondérations (*weights*) en distance, résistance au roulement et énergie, respectivement.

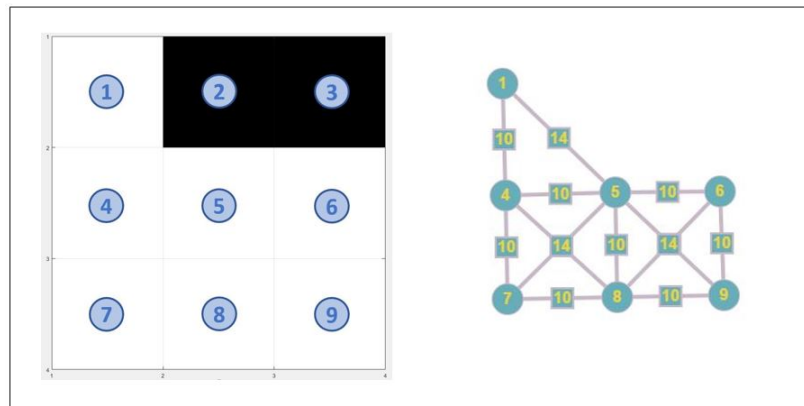


FIGURE 3-5 Représentation d'un graphe "distance" (à droite de la figure) à partir d'une carte *OGM* (à gauche de la figure (les cellules noires représentent les obstacles et les cellules blanches reflètent les zones de passage permises)).

En tenant compte de la particularité du "*grid graph*", le poids des arêtes est égal à la valeur unitaire de la résolution de la carte en grille pour les déplacements orthogonaux et il est égal à $\sqrt{2}$ multiplié par cette dernière quant aux déplacements diagonaux (théorème de Pythagore). La figure 3-5 illustre une matrice d'adjacence basée sur la distance dans laquelle un mouvement vertical ou horizontal a une distance de $10m$, tandis que la distance d'un mouvement diagonal est de $10\sqrt{2} \approx 14m$. Par ailleurs, la matrice "résistance" M_r^{OGM} est supposée connue par l'intermédiaire des travaux dans la littérature qui proposent des méthodes d'identification du coefficient de résistance au roulement lors du déplacement du véhicule [162], [196]. En outre, pour construire la matrice d'adjacence basée sur l'énergie consommée

M_e^{OGM} , le modèle dynamique de l'AMR (défini dans la section qui suit) est employé. Celui-ci prend en compte la résistance au roulement pneu/route exprimée par M_r^{OGM} . Par exemple, la figure 3-6 illustre à sa gauche une carte de résistance au roulement correspondante à la carte *OMG* représentée dans la figure 3-5 et à sa droite le graphe énergie correspondant à ces deux informations. On constate bien que ce graphe est différent du graphe distance avec des poids différents même avec des arêtes de même longueur. La détermination de ces matrices est une étape clé pour la planification de chemin écoénergétique. Par suite, le planificateur global prend en entrée ces dernières, comme expliqué dans la figure 3-7.

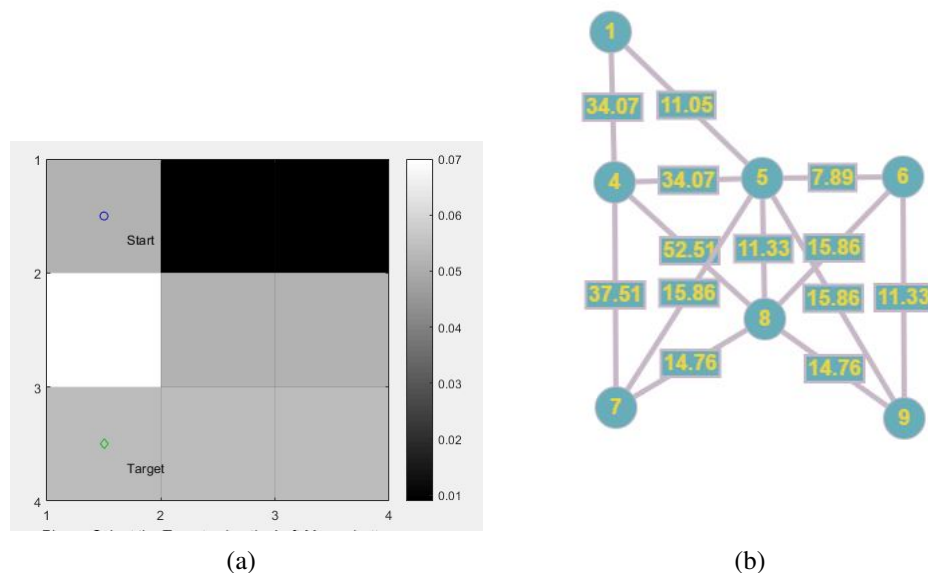


FIGURE 3-6 Représentation d'un graphe "énergie" (a) Carte *OMG* avec les coûts de résistance au roulement des zones (colorées en gris dégradé selon la rugosité des surfaces) (b) Graphe pondéré avec le coût de la consommation énergétique

3.1.3.3 Représentation de l'espace avec une carte topologique et une carte de trafic

Nous considérons également la représentation de l'espace de déploiement du robot par des cartes/ graphes topologiques, tel que les graphes représentés dans la figure 3-8. Il s'agit de définir les voies de passages par laquelle les AMRs peuvent naviguer, sans pour autant leur empêcher de contourner les obstacles qui leur rencontrent sur leurs chemins planifiés.

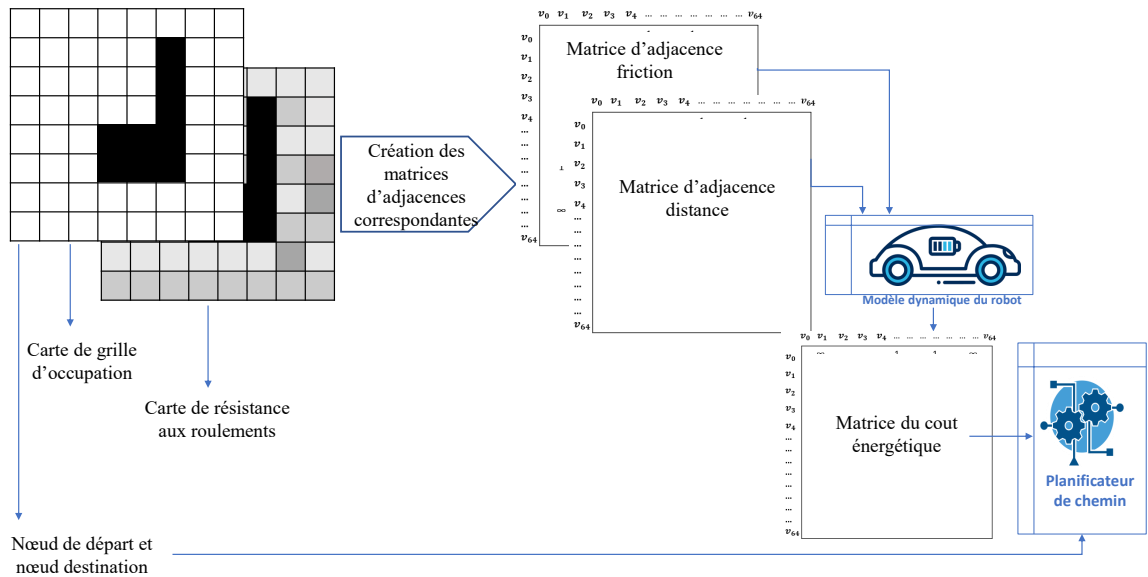


FIGURE 3-7 Processus de la méthode de planification globale avec considération du facteur de la résistance au roulement.

3.1.3.4 Définition de la matrice d'adjacence topologique et la matrice d'adjacence indicatrice du trafic

Comme dans la section précédente 3.1.3.1, nous définissons les matrices des caractères de l'environnement comme suit :

- M_d^{topo} représente la "matrice distance" topologique.
- M_r^{topo} désigne la "matrice de résistance au roulement" topologique.

Par contre, les déplacements dans ce type de représentation suivent la structure de la carte topologique et ils ne sont plus restreints aux mouvements dans les 8 directions. De plus, nous définissons une autre matrice appelée M_{ob}^{topo} "matrice d'historique des obstacles contournés" qui donne l'information du trafic sur chaque arête de la carte topologique chaque jour d et plage horaire t , avec $d \in \{0, \dots, 6\}$ et $t \in \{0, \dots, 23\}$. Par exemple, $d = 0$ correspond au jour dimanche et $t = 8$ correspond à la plage horaire entre 8 heure et 9 heure ($[8, 9]$). Cette matrice est composée de valeurs médianes du nombre d'obstacles rencontrés et contournés sur chaque arête du graphe de la manufacture, pour chaque plage horaire et chaque jour donné. Elle est définie comme suit :

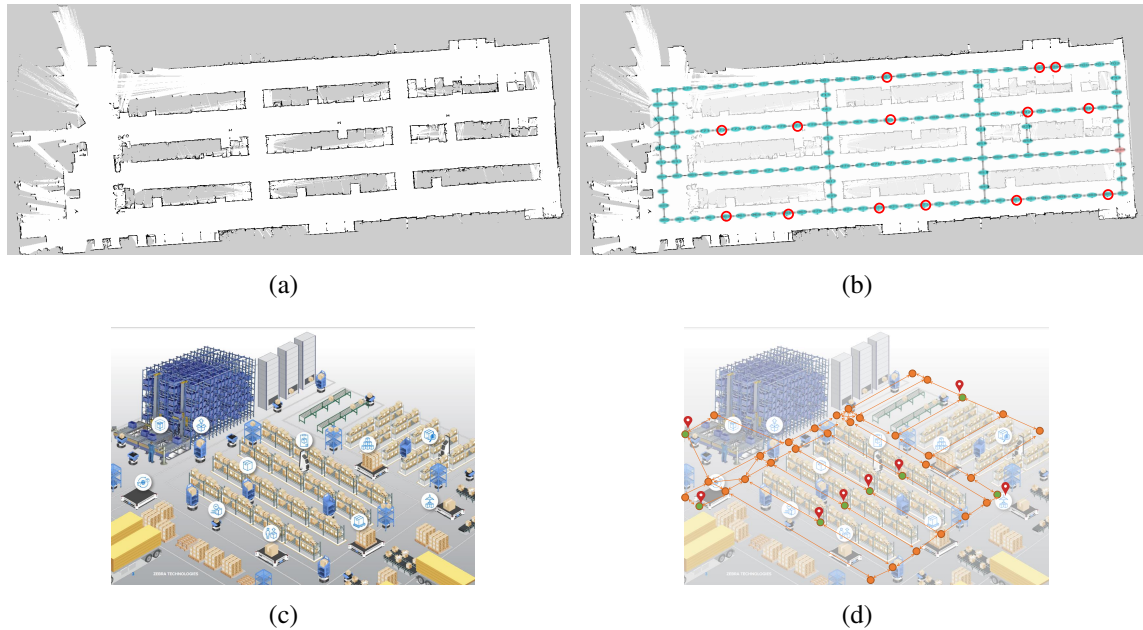


FIGURE 3-8 Exemples de représentation d'un graphe topologique (graphe de rue) de deux photos de manufactures réelles ((a)-(c) dont la photo (c) affiche des robots de systèmes FetchRobotics)[13] (b)-(d) Représentation des cartes topologiques correspondantes aux manufactures (respectivement) en marquant les positions d'intérêts avec des cercles rouges).

$M_{ob}^{topo}(d,t) = [\bar{w}_{ob}^{d,t}(i,j), \forall (i,j) \in A_{topo}]$ tel que $\bar{w}_{ob}^{d,t}(i,j)$, représente la médiane du nombre d'obstacles contournés sur l'arête $(i,j) \in A_{OGM}$, selon des données antérieures fournies par la base de données historique, reflétant le degré de la congestion sur le segment de rue (i,j) , sachant que nous supposons que $\bar{w}_{ob}^{d,t}(i,j) = \bar{w}_{ob}^{d,t}(j,i)$ (le coefficient du trafic est le même dans les deux sens du même segment de rue (arête)).

Il est important de mentionner que ces informations de trafic sont supposées connues à l'avance (grâce à la disponibilité des données avec *IOT*). Par ailleurs, l'approche proposée ne les gère pas en temps réel. La base de données schématisée par une boîte avec un contour bleu (nommée historique de données) sur la figure 3-9 est supposée connue. Par la suite, au besoin, le planificateur global énergétique pointe l'information requise à partir de la base de données de l'historique de trafic dans une plage horaire et une journée donnée. Par exemple, en explorant l'arc (u,v) , l'information de la médiane du nombre d'obstacles contournés sur cet arc est sélectionnée à partir des données antérieures. Cette information permet de définir un coefficient de pénalité que nous allons appliquer dans la fonction du coût de l'optimisation de chemin écoénergétique. Cette dernière augmente en fonction du nombre d'obstacles rencontrés en se basant

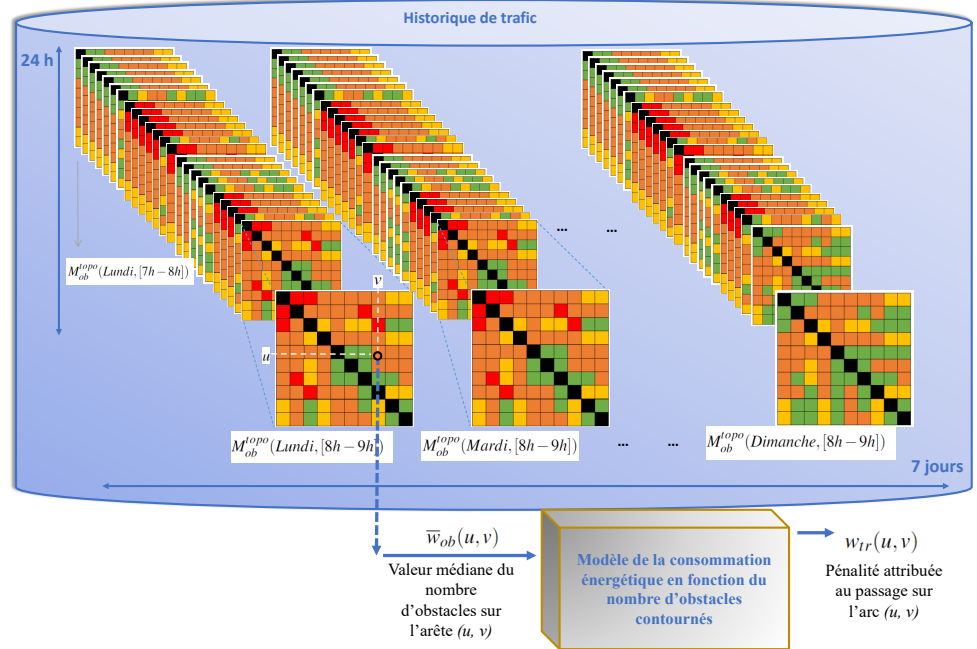


FIGURE 3-9 Schématisation du processus de définition de coefficient de pénalité pour un arc.

sur le tableau 3-1 tels que $\bar{w}_{ir}^{d,t}(i, j) \in \{1, 1.14, 1.31, 1.51, 1.68, 1.91, 2.15, 2.49, 2.74, 3.12\}$ indique le coefficient de pénalité attribuée à un arc donné, selon le nombre d'obstacles rencontrés sur cet arc. $\bar{w}_{ir}^{d,t}(i, j) = 1$ s'il y a zéro obstacle rencontré ou bien que l'information du trafic n'est pas disponible (ce segment de rue n'était jamais traversé). Ce tableau est défini grâce à la courbe représentée dans la figure 3-10 réalisée par Alajlan et al. dans [47], qui démontre l'effet du nombre d'obstacles rencontrés sur un segment de rue sur la consommation énergétique du robot (voir figure 3-10).

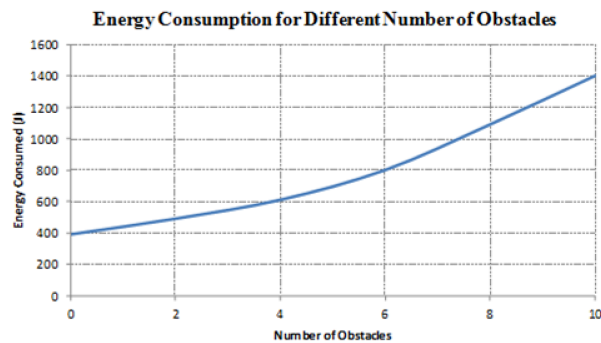


FIGURE 3-10 Effet du trafic sur la consommation énergétique [47].

Ainsi la définition des matrices d'adjacences topologiques M_d^{topo} , M_r^{topo} et $M_{ob}^{topo}(d, t)$ est une étape initiale pour une recherche de chemin plus économe énergiquement. En effet, ces

TABEAU 3-1 Les coefficients de pénalité associés au volume du trafic en fonction du nombre d'obstacles rencontrés.

Nombre d'obstacles	Coefficient de pénalité
0	1 (pas de pénalité)
1	1,14
2	1,31
3	1,51
4	1,68
5	1,91
6	2,15
7	2,49
8	2,74
9	3,12

matrices forment les paramètres d'entrées du planificateur global écoénergétique *EE-GPP*. Elles présentent des paramètres cruciaux pour le calcul de la fonction du coût (détail dans la section 3.1.5.1). Ce processus est illustré dans la figure 3-11 et expliqué dans la section suivante.

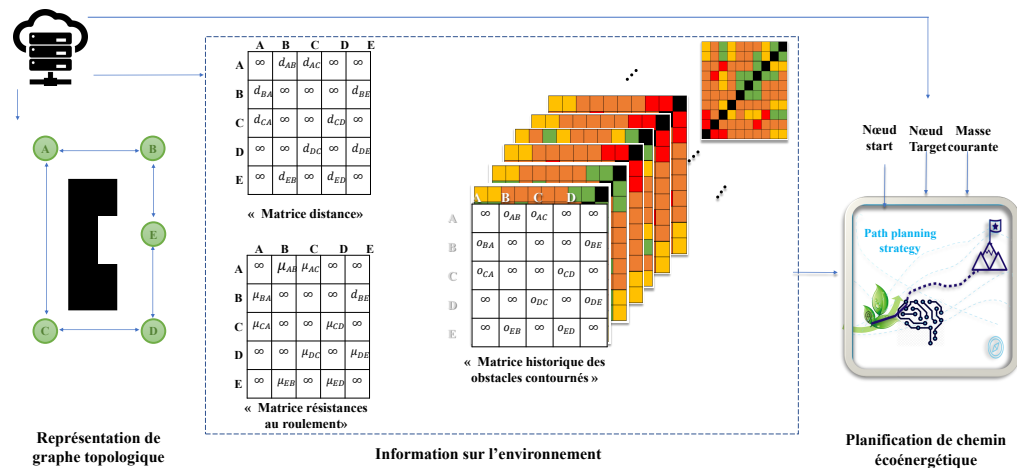


FIGURE 3-11 Processus de la méthode de planification globale écoénergétique avec la considération des facteurs de la résistance au roulement et du trafic.

3.1.4 Définition du problème

Considérons une carte topologique d'une manufacture représentée par un graphe de rue $G(V,A)$, avec $A = \{a_0, a_1, \dots, a_m\}$ ($m = |A|$) l'ensemble des arcs représentant les voies de passages (ou les segments de rue) de l'environnement des robots et $V = \{v_0, v_1, \dots, v_n\}$

($n = |V|$), l'ensemble des nœuds indiquant les intersections à travers lesquelles les *AMRs* peuvent naviguer. Les voies de passage sont virtuelles et peuvent être modifiées au besoin, de plus, les robots peuvent contourner les obstacles rencontrés sur un segment de rue et reprendre leurs voie (au besoin). Une représentation d'exemple d'un graphe de manufacture est illustrée dans la figure 3-12. Parmi l'ensemble des nœuds (V) du graphe G , une sélection de *nœuds d'intérêts* est réalisée. Ces nœuds sont regroupés dans l'ensemble ($I \in V, p = |I|$) qui représente les positions éventuelles de charge ou de décharge des biens (*Pick-up or Drop-Off positions*) et qui comprend la position de dépôt. Les nœuds qui appartiennent à I sont encerclé en rouge dans la figure 3-12.

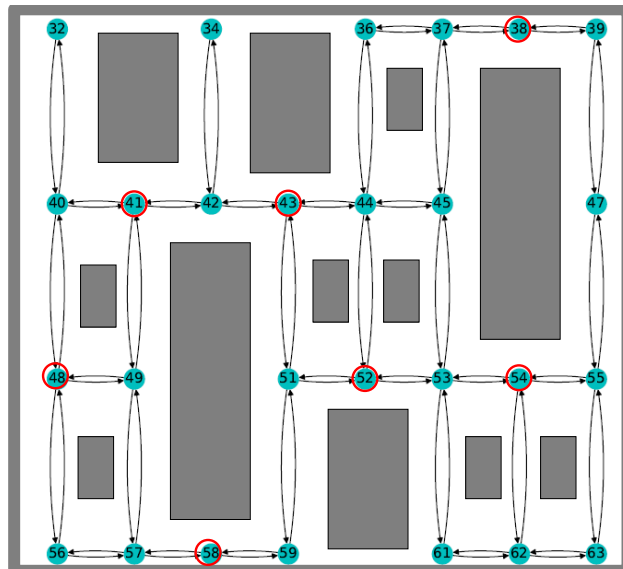


FIGURE 3-12 Un exemple de représentation d'un graphe topologique d'une manufacture.

Dans un contexte réel, le planificateur de chemin *GPP*, étant un planificateur *Source-Target* (*ST-PP*), est exécuté successivement pour qu'un robot voyage d'un point d'intérêt à l'autre en réalisant des manutentions spécifiques. Par conséquent, pour étudier le problème de *ST-PP* écoénergétique (avec plusieurs données et pour avoir l'information de la masse de départ), nous considérons un plan de collecte et de ramassage (*Pickup and Delivery Plan (PDP)*) préalablement élaboré. Il s'agit de transporter des charges à partir des positions d'origines vers des positions destinations par un ensemble de véhicules/robots. Chaque véhicule, avec une capacité donnée, a une mission de transportation à accomplir [197]. Soit un ensemble R d'*AMHRs* qui partagent le même environnement manufacturier, $R = \{r_0, r_1, \dots, r_l\}$, avec l est

le nombre de robots qu'y opèrent. Chaque robot de type particulier a sa propre masse définie par l'ensemble $M_r = \{m_{r_0}, m_{r_1}, \dots, m_{r_l}\}$ correspondant à R (ayant le même ordre, ex. m_{r_1} est la masse du robot r_1). Il est important de noter que cette thèse ne traite pas la gestion des robots multiples ni la définition de la liste des points à visiter à travers la mission. Cependant, nous considérons une mission donnée (des points d'intérêts à visiter) et l'information de la masse du robot qui exécute la tâche et la masse de la charge transportée à chaque étape, pour les intégrer dans la définition du problème de chemin écoénergétique d'un point d'intérêt v_k jusqu'à la position suivante v_{k+1} .

Soit un plan de ramassage et de livraison (PDP) R définissant des missions particulières pour chaque robot r_i ; $R := \{S_r \mid r \in R\}$ Une mission S_r est traduite par une liste de stations (positions d'intérêts ordonnées) à visiter par le robot r et par des actions de manutentions à effectuer sur chaque position visitée (collecte ou dépôt de charges), avec $S_r \subseteq I$.

Soit $S_r = \{(s_0, m_0), (s_1, m_1), \dots, (s_s, m_{nb_s})\}$, tel que $nb_s \leq p$ est le nombre des nœuds d'intérêts à visiter par le robot r , s_i correspond à un nœud $v \in I$, $m_i \in \mathbb{R}$ est la masse manutentionnée (collecte ou dépôt) en arrivant sur le nœud s_i ($m_i > 0$ s'il s'agit d'une collecte de charge, $m_i < 0$ s'il s'agit d'un dépôt de charge, $m_0 = m_r$, $m_{nb_s} = 0$). L'exemple explicatif du contexte de *PDP* présenté dans la figure 3-1 schématise un *Host* qui attribue des missions de transports aux différents robots disponibles (la ligne en vague schématise le chemin optimal entre deux positions d'intérêts que nous cherchons à trouver (un chemin non direct qui contient des points de passages de réseaux de rue (présenté dans le cercle de *zoom-in* en bleue)). Chaque robot est responsable d'une suite de positions d'intérêts à visiter et à manutentionner des charges spécifiques. Par exemple, dans la figure 3-1, le *robot*₁ doit se rendre à la position B et ramasser une charge de masse m_B , par la suite il doit se diriger vers C et collecter une marchandise de masse m_C , ensuite il a à voyager vers D et déposer les deux masses m_B et m_C et finalement retourner à sa position initiale A . Étant donné qu'il peut exister une multitude de chemins plus courts ou quasi plus court entre chaque paire d'origines/destinations, nous ne se contentant pas seulement d'un chemin plus court au hasard, de la distance euclidienne ni d'un chemin avec une distance Manhattan plus courte. Par contre, l'intégralité des chemins intermédiaires du réseau routier de la manufacture est considérée pour chercher un chemin plus écoénergétique. Par conséquent, le problème peut être décrit comme suit : étant donnée

une mission attribuée à un robot donné, quelle est la route écoénergétique pour que le robot effectue sa mission de transport d'un point d'intérêt à un autre? Ceci revient à résoudre successivement le problème EE-GPP (d'un point d'intérêt à un autre), tous en considérant les facteurs environnementaux dont la masse du robot à la position d'intérêt précédente, le trafic variable et les conditions des surfaces de rues.

3.1.5 Formules de programmation en nombres entiers pour la planification de chemin globale écoénergétique

Le problème de planification de chemin écoénergétique est considéré comme un problème de recherche de chemin optimal élémentaire. Afin d'être général et pour convenir à tout type de graphe, le problème est défini dans un graphe adaptatif $G(V, A)$, avec V présente l'ensemble des nœuds de cardinalité n et A présente l'ensemble des arcs avec une cardinalité m . Ce graphe peut être remplacé par les deux représentations de l'espace de robot précédemment définies en employant :

- le *grid-graph* $G_{OGM}(V_{OGM}, A_{OGM}, w_d, w_r)$ avec les attributs distance et résistance au roulement;
- le graphe topologique $G_{topo}(V_{topo}, A_{topo}, w_d, w_r, w_{tr})$ avec les attributs distance, résistance au roulement et trafic.

Le premier cas de figure (*grid-graph*) ne considère pas le facteur trafic. Par suite, si l'information de trafic n'est pas disponible, on suppose que $\overline{w}_{tr}^{d,t}(i, j) = 1 \forall (ij \in A_{OGM})$. En outre, dans le deuxième cas de figure, la matrice d'adjacence "énergie" ne peut pas être calculé au préalable ni admis comme entrée pour l'algorithme de *PP* tel que pour le cas constant. Cependant, cette valeur doit être calculée lors de la planification de chemin parce qu'elle dépend du coût optimal du nœud précédent.

Pour déterminer un chemin élémentaire écoénergétique d'un nœud de départ $s \in V$ à un nœud destination $t \in V$ une formulation standard de programmation en nombres entiers est proposée. Un chemin $(v_{start} - P - v_{target})$ est dit élémentaire s'il ne passe qu'une seule fois sur les nœuds du chemin optimal $P \subseteq V - \{v_{start}, v_{target}\}$. On note par $\delta^+(i)$ l'ensemble des arcs

sortants du nœud i et par $\delta^-(i)$ l'ensemble des arcs entrants dans ce nœud. Par conséquent, $|\delta^-(s)| = |\delta^+(t)| = 0$. La formulation est la suivante :

$$J = \min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (3.1)$$

$$\sum_{(i,j) \in \delta^+(i)} x_{ij} - \sum_{(j,i) \in \delta^-(i)} x_{ji} = \begin{cases} 1 & \text{si } i = v_{start} \\ -1 & \text{si } i = v_{target} \\ 0 & \text{sinon} \end{cases} \quad \forall i \in V \quad (3.2)$$

$$\sum_{(i,j) \in \delta^+(i)} x_{ij} \leq 1 \quad \forall i \in V \quad (3.3)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (3.4)$$

Tel que $c_{ij} \in \mathbb{R}_+$ représente le coût d'arc (i, j) en argent (coût de l'énergie et du temps consommés) et x_{ij} est une variable binaire qui indique si l'arc (i, j) appartient ou non au chemin optimal (elle prend la valeur 1 si oui et 0 sinon). Les contraintes 3.2 définissent l'existence d'un chemin $v_{start} - v_{target}$ dans G . Elles imposent les contraintes classiques de conservation de flux qui s'assurent que le nombre de flux entrants dans un nœud est égal au nombre de flux sortants de ce nœud. En d'autres termes, si un nœud reçoit un certain nombre de flux, il doit également émettre le même nombre de flux. De plus, les contraintes 3.3 assurent que le degré de sortie de chaque nœud est au plus égal à un. Tandis que 3.4 est la contrainte binaire de x_{ij} . Par suite, pour résoudre le problème de planification de chemin globale (élémentaire) écoénergétique, nous avons développé un algorithme appelé *EE-GPP* que nous présentons dans la prochaine section. Toutefois, nous allons, d'abord, expliquer la manière dont la fonction de coût c_{ij} est calculée. Cette dernière dépend de la demande en énergie e_{ij} et en temps t_{ij} pour se déplacer de i à j ainsi qu'à la manière de gérer le compromis entre ces deux critères. Ces paramètres sont exprimés dans les deux sous-sections qui suivent.

3.1.5.1 Calcul de la consommation énergétique entre deux nœuds adjacents

Soit $e_{ij}(m_i^L)$ la valeur de référence de l'énergie demandée pour voyager d'un nœud i vers un nœud j avec une masse donnée m_i^L (masse totale lorsque le robot quitte le nœud (i)). On désigne, par valeur de référence, la consommation énergétique calculée à partir du modèle de véhicule. Dans cette partie de travail, la masse est constante (entre les points d'intérêts), étant donné que le robot ne change pas de masse lors de déplacement d'un point à l'autre. Cependant, la considération de cette valeur est cruciale pour le calcul de la demande énergétique (la masse influence la valeur (l'ampleur du coût e_{ij}), en revanche elle n'affecte pas la décision de PP). La construction de la matrice d'adjacence M_e est constituée de toutes les valeurs e_{ij} correspondantes à toutes les arêtes (i, j) . Ces valeurs sont calculées à l'aide de l'équation longitudinale du véhicule [198], donnée par 3.5 :

$$F_T = m_t \dot{v} + F_a + F_g + F_r \quad (3.5)$$

Tel que F_T , F_a , F_g et F_r représentent la force de traction longitudinale dans les roues, la force aérodynamique, la force de gravité sur une pente de route et la force de résistance au roulement, respectivement, et m_t and v représentent la masse totale du véhicule et sa vitesse, respectivement. En outre, la force aérodynamique définie dans l'équation 3.6 est la force qui s'oppose à l'avancement du véhicule dans l'air. Cela dépend de la densité de l'air ρ en kg/m^3 , à la surface avant du véhicule A_v en m^2 , coefficient de traînée C_d , la vitesse du véhicule et la vitesse du vent v_w . Ensuite, la force de gravité est donnée par l'équation (3.7), où β indique la pente de la route et g la constante de gravité. Finalement, la force de résistance au roulement est définie dans l'équation 3.8 avec μ est le coefficient de la résistance au roulement sur un segment de rue (i, j) .

$$F_a = \frac{1}{2} \rho C_d A_v (v - v_w)^2 \quad (3.6)$$

$$F_g = m_t g \sin(\beta) \quad (3.7)$$

$$F_r = m_t g \mu \cos(\beta) \quad (3.8)$$

Cette dernière force présente le focus de ce document, car nous nous concentrons sur l'effet de la résistance au roulement et la masse sur la consommation d'énergie et sur la

décision de routage. Par conséquent, nous nous sommes concentrés sur la force de résistance au roulement. En effet, la plupart des manufactures et des entrepôts ont un sol plat, $\beta = 0$ (force de gradient peut être négligée). En outre, la vitesse du vent dans un entrepôt peut être négligée. Par conséquent, la force de traction devient :

$$F_T = m_t \dot{v} + \frac{1}{2} \rho C_d A_v (v - v_w)^2 + m_t g \mu \quad (3.9)$$

Or, comme le véhicule est similaire lorsqu'il se déplace le long des arêtes, la force aérodynamique $\frac{1}{2} \rho C_d A_v (v)^2$ est similaire sur chaque arête du graphe. De plus, l'accélération est limitée pour éviter les secousses, les mouvements brusques et le basculement. Par conséquent, la composante la plus importante qui affectera la force est :

$$F_T = m_t g \mu \quad (3.10)$$

Ainsi, le travail e (qui est équivalent à l'énergie mécanique en translation) est donné par $F_T d$ (équation 3.11), où d est la longueur d'une arête :

$$e(m_t) = m_t g \mu d \quad (3.11)$$

Précisément, la demande énergétique estimée pour le déplacement d'un AMR depuis i vers j sur un arc (i,j) ayant une distance d_{ij} qui sépare deux nœuds i et j , transportant une masse totale m_i^L , désignée par $e_{ij}(m_i^L)^{ref}$ est donnée par l'équation 3.12. Cette fonction peut être développée davantage. Cependant, les paramètres ajoutés ne contribueront pas à la décision du routage :

$$e_{ij}(m_i^L)^{ref} = m_i^L g \mu d_{ij} \quad (3.12)$$

où $e_{ij}(m_i^L)^{ref}$ indique la valeur de référence de calcul du coût énergétique à l'aide du modèle du robot.

Cependant, si le facteur trafic est considéré, nous supposons que cette valeur de référence est multipliée par un coefficient de pénalité définie à partir du tableau 3-1 précédemment introduit, $w_{tr}^{d,t}(i,j)$.

Par conséquent, nous définissons l'énergie consommée avec la considération de l'effet

du trafic sur un arc (i, j) , e_{ij} égale à l'énergie de référence e_{ij}^{ref} calculée à partir du modèle longitudinal du robot multiplié par le coefficient de pénalité de trafic $\bar{w}_{tr}^{d,t}(i, j)$, définie comme suit :

$$e_{ij}(m_i^L) = \bar{w}_{tr}^{d,t}(i, j)[m_i^L g \mu d_{ij}] \quad (3.13)$$

3.1.5.2 Gestion du compromis temps/ énergie

Maintenant que nous avons défini le modèle du coût énergétique $e_{ij}(m_i^L)$ (équation 3.13), définissons le modèle (simplifié) du temps nécessaire pour passer du nœud i au nœud j (t_{ij}) comme suit :

$$t_{ij} = d_{ij}/V \quad (3.14)$$

avec d_{ij} représente la distance qui sépare i et j . Dans ce modèle, on suppose que la vitesse est constante et on ne considère pas le retard qui peut être causé par le trafic (temps proportionnel à la distance).

Maintenant, afin de pouvoir assembler les deux critères temps et énergie dans la même fonction de coût et puisque t_{ij} et e_{ij} ont des unités différentes, nous avons unifié l'unité de ces derniers en définissant le coût total en dollars pour que l'AMR se déplace de i à j , c_{ij} comme suit :

$$c_{ij} = c_t(t_{ij}) + c_e(e_{ij}) \quad (3.15)$$

tel que c_t et c_e sont deux coefficients représentant, respectivement, le coût d'une unité de temps en dollar et le coût d'une unité d'énergie (joule) en dollar. Ces deux coefficients définissent l'importance de chaque critère. Ainsi, la définition de c_{ij} , permet de calculer le coût total de déplacement sur l'arc (i, j) en fonction du temps lors de la planification de chemin écoénergétique, expliquée dans la section qui suit.

3.1.6 Algorithme pour la planification de chemin globale écoénergétique

Comme expliqué précédemment, la nouveauté de notre approche *EE-GPP* réside dans la considération des facteurs de l'environnement qui influencent la consommation énergétique et l'intégration de tous ces deniers dans la prise de décision du routage. Particulièrement,

l'approche proposée basée sur l'équation fonctionnelle de la programmation dynamique (avec une approximation successive) est capable de gérer les coûts des arêtes dépendant du temps. En effet, comme l'énergie consommée par segment de rue peut être non proportionnelle à la distance pour le parcourir et que le coût d'un arc peut varier en fonction de temps, les coûts des arcs sont modélisés à l'aide d'une fonction de coût qui représente le coût (en énergie et en temps) de déplacement en fonction du temps. Ces coûts sont calculés au fur et à mesure de la relaxation des arêtes. L'approche *EE-GPP* est décrite à travers l'algorithme 1 (ci-dessous), tel que la fonction *makeQueue* initialise un queue de priorité appelé *H*, *Extract-Min* sert à extraire le nœud avec le coût minimal du queue *H* et la fonction *decreaseKey* met à jour le coût d'un nœud s'il existe une meilleure option. L'algorithme 1 (ci-dessous) essaye d'examiner des sous-trajectoires et d'exclure celles qui ne sont pas favorables à l'obtention du chemin optimal jusqu'à l'identification de ce dernier [84], [102]. Plus précisément, il s'agit de construire un sous-graphe, progressivement, dans lequel sont classés les différents nœuds dans l'ordre croissant par rapport à leurs coûts minimum depuis le nœud de départ. À chaque itération, un nœud à coût minimum est sélectionné et ajouté au sous-graphe, initié par le nœud de départ avec un coût égal à 0. Ensuite, le coût des nœuds adjacents au nœud choisi est mis à jour. Cette opération se poursuit jusqu'à ce que le nœud cible soit sélectionné. L'opération de relaxation dans l'algorithme de *EE-GPP* (ligne 12 -19) consiste à mettre à jour le coût le plus faible connu pour atteindre un nœud voisin en le comparant au coût actuel pour atteindre ce nœud, et en prenant le coût le plus faible des deux. Si le coût le plus faible est mis à jour, le nœud voisin est ajouté à la liste des nœuds non visités pour être explorés ultérieurement. La conservation des coûts optimaux des nœuds préalablement calculés fait référence au principe d'optimalité qui est un concept clé en programmation dynamique. Il énonce que la solution optimale à un problème peut être obtenue en combinant les solutions optimales de ses sous-problèmes. Une fois les coûts optimaux de chaque nœud sont calculés et le prédécesseur de chaque étape est enregistré, un algorithme de suivie de chemin est requis. Nous définissons l'algorithme 2 (ci-dessous) qui prend en entrée la liste des prédécesseurs de chaque nœud, pour construire le chemin optimal entre v_{start} et v_{target} . Pour ce faire, une liste *path* est défini et initiée avec le nœud v_{target} . Ensuite, tous les prédécesseurs de chaque nœud sont extraits un par un, jusqu'à vider la liste *pred* et construire le chemin complet.

Algorithm 1: *EE-GPP* ($G(V, E, w_d, w_r, w_{tr}^{d,t}), v_{start}, v_{target}$)

```

1 initialization :
2 for each  $v \in G$  do
3    $cost[v] \leftarrow \text{infinite}$ ;
4    $pred[v] \leftarrow \text{undefined}$ 
5 end
6  $cost[v_{start}] \leftarrow 0$ 
7  $H \leftarrow \text{makeQueue}(V)$ 
8  $H \leftarrow [(0, v_{start})]$ 
9
10 while  $H \neq \emptyset$  do
11    $u \leftarrow \text{Extract-Min}(H)$ 
12   for all edges  $(u, v) \in G$  do
13     if  $cost[v] < cost[u] + \text{energy-time-cost}_1(w_r(u, v), w_d(u, v), w_{tr}(u, v))$ 
14       then
15          $cost[v] = cost[u] + \text{energy-time-cost}_1(w_r(u, v), w_d(u, v), w_{tr}(u, v))$ 
16          $pred[v] \leftarrow u$ 
17         decreaseKey ( $H, v$ )
18       end
19     end
20 end
21 return  $cost, pred$ 

```

Algorithm 2: *TruckPath* ($pred, v_{target}$)

```

1 initialization :
2  $path \leftarrow []$ ;
3  $u \leftarrow v_{target}$ 
4
5 while  $pred[v]$  is defined do
6   path.insertFirst ( $u$ )
7    $u \leftarrow pred[v]$ 
8 end
9 return path

```

3.2 Routage de préparation de commande écoénergétique

3.2.1 Introduction

Dans cette partie du travail, nous nous intéressons spécialement à l'optimisation énergétique du routage des robots Ps-AMRs étant donné leur expansion et leur utilisation accrue afin de réduire leurs coûts de déploiement affectant le coût global des opérations logistiques et de la chaîne d'approvisionnement. Comme précédemment énoncés, ces véhicules sont utilisés dans les milieux d'entrepôts et les DCs pour ramasser les commandes ou assister les opérateurs humains lors du processus de préparation de commandes (OP). Ils sont généralement déployés dans des entrepôts de préparation de commandes manuelles avec des configurations *picker-to-parts* appelé "entrepôts rectangulaires réguliers" ou "entrepôts à allées parallèles" (*traditional picker-to-parts order picking systems*) qui sont utilisés par 80 % des entrepôts d'Europe de l'Ouest et par la majorité des entrepôts de détails actuels [186]. L'inconvénient majeur de ce genre d'entrepôts est la marche improductive de l'opérateur humain lors des déplacements entre les étagères et le retour au dépôt central. Cet inconvénient est partiellement comblé par les Ps-AMRs. Cependant, comme maintenant ce sont les robots qui doivent parcourir de longues distances et l'énergie dépensée par ces derniers doit être considérée. En effet, l'AMR doit naviguer entre les couloirs, s'arrêter devant les différents lieux de prélèvements (*picking locations (PLs)*) des items demandés, les ramasser et retourner au dépôt central ou s'acheminer vers la station d'emballage une fois que la liste de la commande est bien collectée [199]. Par conséquent, l'amélioration de l'efficacité énergétique de ces déplacements est indispensable, surtout que cette opération est répétée plusieurs fois par jour avec différents déplacements à effectuer, différentes positions à visiter et différents poids à manutentionner. Dans ce cas de figure, il ne suffit pas de planifier le chemin entre les couples de positions, mais plutôt de faire une planification multidestinations qui considère l'ordre de visite des points à visiter ainsi que le chemin qui les relie. D'où vient l'objectif de notre approche que nous allons appeler *EE-OPR (Energy Efficient- Order Picking Routing)* visant à trouver un chemin écoénergétique d'un tour de ramassage de commandes d'un Ps-AMR, sans trop affecter le temps opérationnel d'OP. Par ailleurs, les sections qui suivent contiennent une description de la méthodologie suivie pour le développement de cette approche.

— Analogie avec la première partie du travail

Contrairement à la problématique traitée dans la section précédente (*EE-GPP*), nous traitons, maintenant, la problématique de routage d'un AMR pour la collecte des articles d'une commande donnée à partir de leurs emplacements de stockage dans un entrepôt. Par conséquent, la liste des points d'intérêts à visiter est exprimée dans cette partie de travail par une *pick-list*, soit la liste de points de ramassage que le robot doit visiter pour collecter une commande donnée (définie mathématiquement ultérieurement). Cette liste est non ordonnée et la recherche de l'ordre de visite des points d'intérêts fait partie de la problématique à résoudre. Par ailleurs, étant donné que la masse des articles à ramasser varie depuis différents emplacements, le facteur de la masse est considéré dans la formulation du problème de planification de chemin et il présente le focus de cette partie de travail. En outre, dans cette partie du travail l'entrepôt est représenté par un graphe topologique G_0 . À partir duquel, nous définissons un sous-graphe G . Ce dernier représente le graphe Steiner avec lequel la problématique *OPP* est définie (détails ci-dessous).

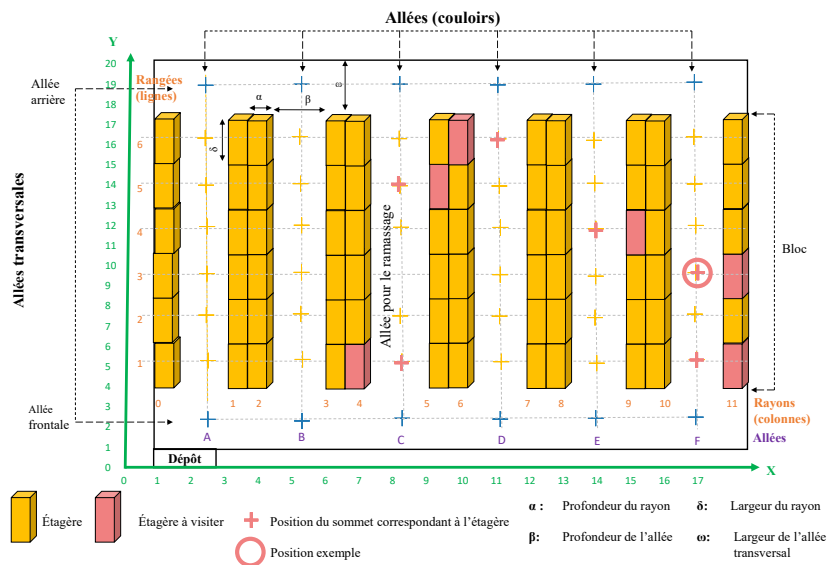


FIGURE 3-13 Vue d'ensemble de la configuration d'un entrepôt (vue de dessus de la zone du stockage)

3.2.2 Représentation de la configuration de l'entrepôt

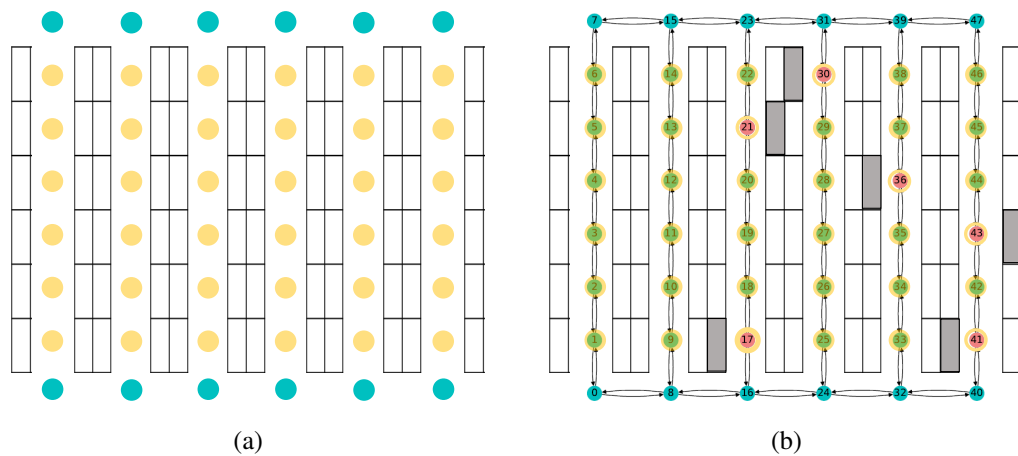


FIGURE 3-14 Aménagement d'un entrepôt et représentation du graphe (a) Définition des nœuds "intersections" (les cercles bleus) et des nœuds lieux du prélèvement (*picking locations*) (les cercles jaunes) (b) Définition des arcs de connections et du graphe G_0

Un entrepôt conventionnel uni-bloc à allées parallèles avec un dépôt unique est considéré (comme la configuration classique de la majorité des travaux qui traite l'*OPR*). Cependant l'*EE-OPR* peut être utilisée dans le cas d'entrepôts multi-blocs voire même avec des entrepôts non conventionnels (nous allons discuter cet aspect dans le prochain chapitre). Cet entrepôt se compose de g allées verticales 2 allées transversales horizontales. La Figure 3-13 représente une vue d'ensemble de la configuration de l'entrepôt avec six allées verticales et deux allées transversales. Les allées (*aisles*) contiennent des produits des deux côtés, tandis que les allées transversales (*cross-aisles*) forment des intersections à travers lesquelles le Ps-AMR peut naviguer. À partir de cette représentation traduite par une carte métrique ou image de carte, une carte topologique est créée selon des points de repère spécifiques considérant la spécificité de la configuration de l'entrepôt. Ce sont des points clés qui vont permettre la définition d'un réseau (graphe) qui va être utilisé pour la détermination des chemins de ramassage de commandes optimaux. Ces points sont symbolisés dans la figure 3-13 par des croix et ils sont remplacés par des nœuds dans la partie gauche de la figure 3-14a. En outre, les connections possibles entre les nœuds adjacents sont représentés par deux flèches de sens contraire dans la partie droite de la figure 3-14b, formant ainsi un graphe d'entrepôt $G_0(V_0, A_0)$, tel que $V_0 = \{v_0, \dots, v_k\}$ est l'ensemble de k nœuds (les cercles bleus et jaunes dans la Figure 3-14)

et A_0 l'ensemble des arcs du graphe bidirectionnel. On distingue deux catégories de nœuds : des nœuds d'intersections qui se situent sur les allées transversales (les cercles bleus dans la figure 3-14) et des nœuds *pick-locations*, qui représentent les emplacements de prélèvement éventuels situés dans les allées (les cercles jaunes dans la figure 3-14)). Les deux catégories sont représentées par deux sous-ensembles de V_0 ; V_I et V_L , respectivement. Chaque nœud $v_i \in V_0$ dans l'entrepôt possède une position et un identifiant : (a) une position (x, y) dans un système de coordonnées cartésiennes $2D$ (comme représenté dans la figure 3-13), avec x représente l'abscisse et y l'ordonnée par rapport à l'origine (le point d'intersection des axes X et Y représentant la direction est-ouest nord-sud, respectivement, et (b) un identifiant qui décrit dans quelle allée il se trouve et dans quel rang. Par conséquent, une étiquette est attribuée pour chaque allée (voir les lettres représentées en mauve dans la figure 3-13). Par conséquent, un autre repère relatif à l'identification d'un article et sa position dans l'allée est représentée par deux axes orange (l'axe horizontal définit le numéro de rayon où il se trouve et l'axe vertical identifie la rangée de l'article dans l'allée). On suppose que dans la même position d'une allée, le robot peut ramasser des articles du côté droit ou du côté gauche. Par exemple, les coordonnées de la "position exemple" représentée par un cercle rose dans la figure 3-13) sont $(17, 12)$; son identifiant est $(F, 3)$. Cette position permet de collecter les articles $(10, 3)$ et/ou $(11, 3)$. Elle est représentée par le nœud v_{43} dans la figure 3-14, représentant le graphe de l'entrepôt. Pour des raisons de simplification, dans le reste du document, nous allons désigner chaque position par le nœud qui lui correspond (les positions sont utilisées pour le calcul des distances).

Suite à une/des commande(s) faite(s) par des clients internes ou externes, un ensemble de prélèvement appelé « *pick-list* » est construit selon le type de prélèvement (lignes de commande d'un seul client (*pick-by-order* ou de plusieurs commandes clients *pick-by-batch* ou lots)). Il s'agit des points d'intérêts à visiter, si on veut suivre la même nomination que la première partie, cependant le non « *pick-list* » est plus universelle dans un contexte de *OP* dans un entrepôt, de plus, pour faire la distinction des listes ordonnées (partie 1) et listes non ordonnées (partie 2). Par ailleurs, dans la suite de ce document, nous utilisons le terme « article » pour désigner l'(les) article(s) ou le lot qui se situent dans le même emplacement de prélèvement représenté par un nœud bien déterminé. De plus, nous allons définir la *pick-list*

par les nœuds qui correspondent aux articles listés dans la commande. Cette liste est définie par un sous-ensemble $V_p \subseteq V_L$ avec une cardinalité $p \geq 2$ (si $p \leq 1$, la problématique revient à chercher *EE-GPP*), contenant des nœuds associés à p articles, qui sont décrits par leurs *SKUs* (*Stock Keeping Units*) et leur emplacement dans l'entrepôt (les nœuds corails dans la figure 3-14). Par exemple, le lieu de prélèvement de l'article i s'exprime par v_i . De plus, les masses de tous les articles sont données par l'ensemble $Y = \{m_1, \dots, m_p\}$ dans lequel chaque élément m_i représente la masse liée à l'article i , en kilogramme (kg). Aussi, m_0 désigne la masse de l'*AMR* sans charge.

3.2.3 Description du problème

Maintenant que le graphe d'entrepôt G_0 est défini, nous allons définir le problème *OPR* écoénergétique qui est considéré comme un problème *Steiner TSP*. Ce problème est basé sur le graphe qui contient des nœuds requis (terminaux) qui appartiennent à la *pick-list* et des nœuds *Steiner* non requis (de passage). Le graphe G_0 contient les nœuds nécessaires, cependant, pour réduire le temps de calcul, la taille du problème peut être considérablement réduite si on considère seulement les nœuds et les arcs nécessaires pour résoudre le *OPR* écoénergétique. Par conséquent, un graphe dirigé $G(V, A)$, sous graphe de $G_0(V_0, A_0)$ est défini. Par exemple, la figure 3-15 représente un exemple d'un graphe G associé au problème *OPR* sous graphe de G_0 étant donné une *pick-list* $V_p = \{v_{21}, v_{30}, v_{41}, v_{43}, v_{36}, v_{17}\}$. Ainsi, G contient 16 nœuds V , dont 6 sont requis construisant l'ensemble (V_p) et le reste sont des nœuds *Steiner* formant l'ensemble (V_I) . D'une façon générale, le graphe $G(V, A)$ est constitué de n nœuds $V = \{v_0, \dots, v_n\}$ et e arcs $A = \{a_0, \dots, a_e\}$. La pondération de chaque arc correspond au coût de la distance euclidienne, d_{ij} , entre les i^{th} et les j^{th} nœuds, reliés par un arc (sachant que $d_{ij} = d_{ji}$). De cette manière, la matrice distance est construite et nous supposons que la matrice résistance est connue et que le facteur trafic n'est pas considéré (pour des raisons de simplification). Cependant, ce dernier peut facilement être intégré dans l'approche de manière linéaire.

Par conséquent, dans le travail présenté, le problème de routage de la préparation des commandes traite l'optimisation de la tournée de l'*AMR* pour collecter tous les produits de la *pick-list* en minimisant le temps et l'énergie consommés lors du déplacement de la position

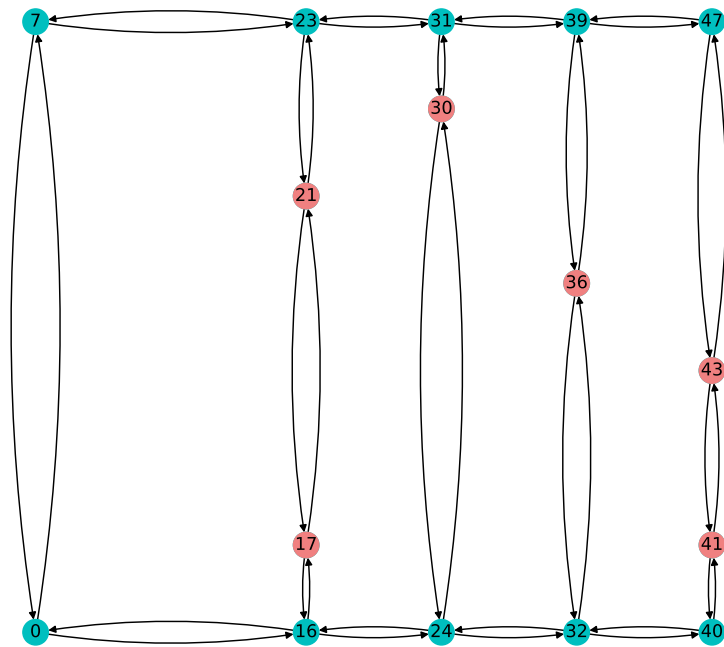


FIGURE 3-15 Un graphe $G(V,A)$ (nœuds rouges représentent les positions à visiter et les nœuds bleus représentent les intersections de passage)

initiale à la position cible (dépôt ou autres positions prédéfinies), en passant sur toutes les positions de prélèvement requises et en collectant les articles de la commande (décrite à travers la *pick-list*). Ce problème d'optimisation tient également compte de la masse des éléments de la *pick-list*. L'objectif étant de trouver un tour écoénergétique, pas nécessairement hamiltonien, puisqu'il s'agit d'un problème de *Steiner TSP* et non le *TSP* conventionnel (cela permet aux nœuds et aux arêtes d'être traversés plus qu'une fois, si nécessaire). Afin d'atteindre cet objectif, une nouvelle approche, applicable à n'importe quel aménagement d'entrepôt, est proposée. Cette approche, appelée *EE-OPR*, est basée sur la création des états dynamiques par la méthode *bitMasking* [200]. De plus, elle est inspirée de l'algorithme de "dynamized Dijkstra" pour la recherche de graphe de Sunita et al. [201]. Le *EE-OPR* prend en compte la consommation d'énergie de l'*AMR* et l'effet du poids de la cargaison transportée.

3.2.4 Hypothèses

La liste suivante énumère les hypothèses que nous avons posées dans *EE-OPR* :

- Seule la consommation d'énergie du déplacement horizontal de l'*AMR* est prise en compte. Les pertes du temps et d'énergie relatives aux arrêts de prélèvements peuvent être supposées constantes ou négligeables. De plus, selon le type de robot, s'il doit soulever lui-même des articles, la perte d'énergie due à ce travail n'est pas prise en compte dans la modélisation mathématique. En effet, la demande d'énergie pour vaincre la gravité de ces articles est presque la même et n'affecte pas la décision de routage.
- La minimisation du temps total de déplacement est équivalente à la minimisation de la longueur totale du tour de ramassage, étant donné que le véhicule peut se déplacer uniformément (vitesse constante (V) dans les deux axes de coordonnées). Par conséquent, la consommation d'énergie causée par l'effet de l'accélération et de décélération n'est pas prise en compte.
- La définition de la liste de prélèvement se fait au préalable (est donnée) de telle sorte que l'énergie à bord soit suffisante pour terminer le tour et que le volume et la masse des articles ne dépassent pas la capacité Q de l'*AMR*.

- La masse de toutes les *SKUs* est accessible et en connaissant la quantité requise dans chaque emplacement de prélèvement, la masse totale de cette dernière peut être calculée.
- Le préparateur de commandes continue à récupérer tous les articles requis dans le même lieu de stockage (à droite et à gauche de l'allée) lorsque l'*AMR* s'arrête sur la position en question (ou l'*AMR* lui-même).
- Les allées sont suffisamment larges pour permettre aux *AMRs* de se déplacer dans les deux sens. Cette hypothèse peut être modifiée pour des allées de prélèvement parallèles et fermées en modifiant simplement les arêtes du graphe avec des arcs unidirectionnels représentant les allées et des arcs bidirectionnels pour les arêtes des allées transversales.
- Par souci de simplicité, l'emplacement du dépôt est représenté dans le coin inférieur gauche de l'entrepôt (comme la majorité des travaux dans la littérature). Toutefois, cette hypothèse peut facilement être modifiée pour n'importe quelle disposition.
- Un seul *AMR* est considéré dans cette étude de cas. Elle peut être facilement étendue pour prendre en compte plusieurs *AMRs*.
- Le robot peut passer plus qu'une fois sur le même nœud au cours d'un tour de prélèvement.
- Le facteur trafic n'est pas considéré dans cette partie de travail (avec une pénalité égale à un), par souci de simplicité et pour mettre l'accent sur l'effet de la masse précisément. Cependant, le coefficient de pénalité peut être intégré, facilement, dans la fonction de coût, avec une complexité linéaire.

3.2.5 Création d'un graphe dynamique

Le *STSP* est un processus d'optimisation combinatoire qui peut être formulé comme un problème de plus court chemin [200]. La solution à ce problème est un chemin avec le coût minimum qui part de la position initiale avec une charge nulle et se termine à la position finale avec toutes les charges demandées (tous les articles de la *pick-list*). Nous avons abordé le problème à partir de cette idée. Par conséquent, les nœuds ici ne présentent pas seulement la position spatiale, mais plutôt un état avec l'information de la position et de la masse transportée. De plus, les arcs ici ne présentent plus une pondération constante, mais plutôt une transition

calculée en fonction du temps et en fonction de plusieurs facteurs qui influence l'énergie. Ainsi, un nouveau graphe $G'(V', A')$, appelé graphe d'états, est créé en plus du graphe spatial $G(V, A)$. L'ensemble V' représente les nœuds "états" que nous appelons «*state*» et l'ensemble A' définit les arcs de transition entre deux états successifs. $G'(V', A')$ est pondéré par une fonction de coût considérant l'état précédent et les coûts directs de transition. Cette fonction détermine le coût de temps et d'énergie nécessaire pour parcourir la distance d_{ij} avec une charge m . Par ailleurs, la résolution du problème du plus court chemin est traitée à l'aide du principe de *Bellman*. Ces notions sont détaillées dans les sections qui suivent.

3.2.5.1 Définition des nœuds «*states*»

Étant un cas particulier de *TSP*, le *STSP* peut utiliser la philosophie de l'arrangement *bitmask* soulignant que chaque donnée est représentée par un chiffre binaire (0 ou 1) avec une permutation particulière (*bitmask*) [200]. Soit p nœuds sont les éléments d'une *pick-list*. Pour cette liste, un tuple $\mathbf{a}_i = (a_0 \ a_1 \ \dots \ a_p)$ avec p chiffres binaires est créé avec le même ordre. Ce tuple définit les articles qui ont été sélectionnés à un moment donné, par le biais d'une valeur binaire 1 dans leur position correspondante dans \mathbf{a}_i . Par exemple, $\mathbf{a}_i = (0 \ 1 \ \dots \ 0)$ montre que le deuxième article de la *pick-list* a été prélevé par le robot. Par conséquent, $\mathbf{a}_i = (1 \ 1 \ \dots \ 1)$ signifie que tous les articles de la liste ont été collectés. Le nombre total d'arrangements possibles de a_i est 2^p , cependant, l'approche *EE-OPR* n'utilise que certains de ces derniers d'une manière intelligente. Puisque le problème est présenté dans un graphe de réseau incomplet et que seul un sous-ensemble de '*visiting vertices*' [170] est couvert par le tour de prélèvements, l'information spatiale des nœuds est également nécessaire pour formuler le problème d'*OPR*. En effet, le robot peut être positionné sur n'importe quel nœud $v_i \in V$ (requis ou non) et il peut transporter 0 à p articles. Par la suite, les informations spatiales et temporelles sont jumelées constituant un «*state*» qui est représenté comme un tuple $\mathbf{s}_k = (v_i, a_j)$ dans lequel v_i représente le nœud spatial (qui indique sa position) et a_j exprime l'arrangement (qui reflète le nombre et l'identité des articles déjà soulevés). Cet état est défini comme un nœud «*state*» 2D dans le nouveau graphe G' (incorporant les deux informations : spatial et d'arrangement). Un exemple d'un état possible avec une *pick-list* de taille 4 est illustré dans la figure 3-16.

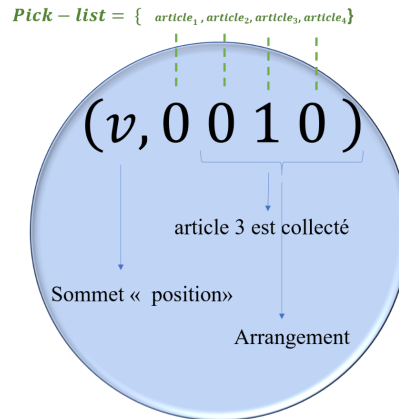


FIGURE 3-16 Un exemple d'un nœud «State» avec une *pick-list* de taille 4.

La complexité spatiale de la représentation des états est de $n2^p$, où n est la cardinalité de l'ensemble V . En général, le temps d'exécution global d'un *TSP* basé sur la programmation dynamique (*DP*) et le *bitmasking* avec n villes à visiter est $O(n^2 * 2^n)$ [200]. Par conséquent, la complexité de la résolution du *Steiner TSP*, utilisant le *DP bitmask* est $O(n^2 * 2^p)$. Néanmoins, cette complexité peut être diminuée si on considère la sparsité du graphe (creux) de l'entrepôt et la particularité des points *Steiner*. Cet avantage est réalisé par *EE-OPR* qui est capable de créer seulement les états et les arcs nécessaires pour la résolution du problème *OPR* d'une façon optimale .

Ainsi, pour résoudre le problème de routage de *Ps-AMR* écoénergétique, nous avons considéré l'*OPR* en tant que problème de plus court chemin entre l'état $s_{start} = (v_{start}, a_{start})$ et l'état $s_{target} = (v_{target}, a_{target})$ (les états de départ et de destination, respectivement). Pour ces états, v_{start} évoque la position de départ et v_{target} représente la position d'arrivée. En outre, a_{start} et a_{target} définissent, respectivement, l'arrangement de départ et l'arrangement cible ($a_{start} = (a_1, a_2, \dots, a_p)$ avec $a_i = 0 \forall i \in \{1, \dots, p\}$ et $a_{target} = (a_1, a_2, \dots, a_p)$ avec $a_i = 1 \forall i \in \{1, \dots, p\}$).

Dans la suite de la thèse, nous désignons l'arrangement de départ par '*empty arrangement*' et l'arrangement cible par '*full arrangement*', à savoir, le robot part de sa position de départ sans aucun article et arrive à destination avec tous les articles demandés. La figure 3-17 illustre de manière simplifiée le processus de la création des nœuds *états*. Dans cette représentation, le graphe principal comprend 4 nœuds : *A*, *B*, *C* et *D*, dont *B* et *D* sont les nœuds requis. Étant

donné, A à la fois point de départ et d'arrivée et $P = \{B, D\}$ une *pick-list*, les états de départ et d'arrivée peuvent être exprimés comme $(A, 00)$ et $(A, 11)$, respectivement. Afin d'éviter de créer tous les états possibles, un nouveau graphe acyclique dirigé, G' , est créé dynamiquement. Ce graphe est initié par le nœud de l'état de départ et il est par la suite étendu par un ensemble de nœuds d'états et des arcs de transitions en se basant sur l'exploration spatiale du graphe de l'entrepôt G , et sur l'arrangement actuel et les alternatives des arrangements successeurs.

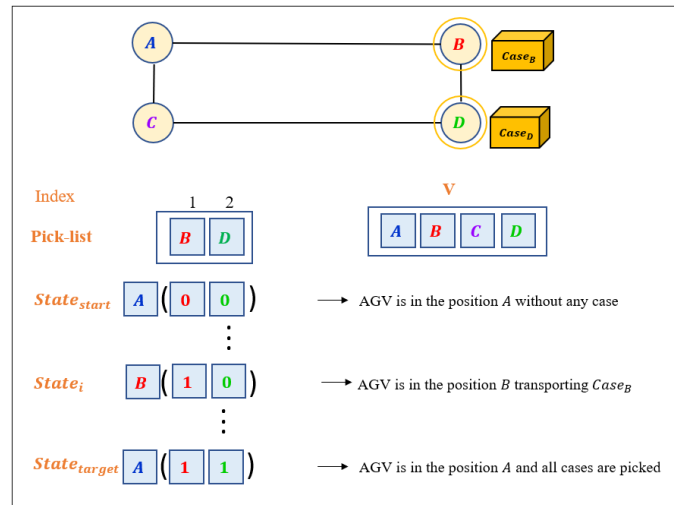


FIGURE 3-17 Un exemple explicatif de la création des nœuds «États»

3.2.5.2 Définition des arcs de transitions

Durant un tour de préparation des commandes, le robot peut se déplacer entre les nœuds en transportant de 0 à p articles. Ce mouvement correspond à une transition entre deux états qui est représentée par un arc $a \in A'$. La matrice d'adjacence $n2^p n2^p$ peut être utilisée pour définir les transitions possibles sur l'ensemble des états. Cependant, cette matrice inclut des transitions impossibles et des états inatteignables, étant donné des actions non permises, telque :

- ramasser plus qu'un seul article ou lot d'articles au même temps (plus qu'un item de la *pick-list*),
- enlever des charges pendant le tour de préparation des commandes,
- activer un bit dans l'arrangement (zéro \rightarrow un) lorsque l'article associé au nœud correspondant dans la *pick-list* n'est pas encore ramassé,

- activer un bit dans l'arrangement (zéro \rightarrow un) lorsque sa position n'est pas liée à celle du nœud dans la *pick-list*.

Par conséquent, *EE-OPR* est employé pour générer les états et les arcs du graphe G' de manière dynamique, pour réduire la complexité en espace et en temps et pour éviter les transitions interdites listées ci-dessus. À cet effet, des conditions de création de ramification sont appliqués. Ce processus est effectué en avançant dans l'exploration du graphe spatial G . De plus, la technique de relaxation des arêtes est utilisée pour mettre à jour les coûts actuels pour atteindre les états existants depuis la position de départ par des coûts inférieurs s'il en existe.

3.2.5.3 Création des nœuds d'états et des arcs de transitions

Selon l'explication ci-dessus, un état peut être représenté par plusieurs attributs comprenant le nom (v_i, a) , le nœud v_i , l'arrangement a , la masse m , le prédécesseur $state_{i-1}$ et le coût c relatif au déplacement depuis le nœud de départ. La phase d'exploration, pour la recherche de chemin, implique la création des états et des arcs de transition en parcourant le graphe G . Le nœud de départ est défini par le l'état dont la position est le nœud de départ et l'arrangement est '*empty arrangement*'. Les ramifications du graphe G' commencent à partir de ce nœud. Par la suite, chaque arc sortant implique une transition entre deux états, traduite par un changement de position (de nœud) et une mise à jour de l'arrangement (si possible). Le déplacement direct entre les nœuds peut se produire uniquement entre deux nœuds adjacents, $v_i, v_j \in V : (v_i v_j) \in A$. En outre, le changement de l'arrangement (état des charges) suit une certaine logique et il existe deux alternatives, décrites ci-dessous.

- a_1 : l'arrangement reste le même (ne pas transporter v_j).
- a_2 : l'arrangement est mis à jour en activant le bit correspondant à v_j (transportant v_j) (voir figure 3-18).

En jumelant les arrangements a_1 et a_2 avec un nœud voisin, deux états possibles peuvent être créés sur la base suivante :

- (v_j, a_1) : Cet état exprime le mouvement de l'AMR vers le nœud suivant v_j avec la même charge.

— (v_j, a_2) : Cet état désigne le mouvement de l'AMR vers le nœud suivant v_j , avec sa charge actuelle et la charge à ramasser dans v_j .

Si v_j appartient à la *pick-list*, les deux possibilités sont autorisées. Sinon, seule la première éventualité est autorisée. Notez, également, qu'un seul bit peut changer à chaque mise à jour de l'arrangement. En outre, si un état parmi ces deux existe déjà, l'étape de relaxation est effectuée, pour tester la possibilité de diminuer son coût.

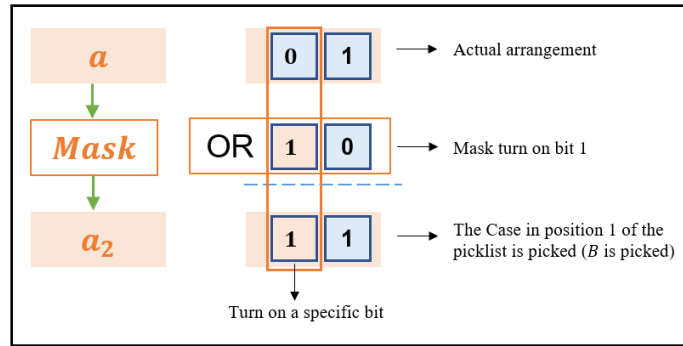


FIGURE 3-18 L'application d'un masque pour activer un bit qui correspond à un nœud donné dans une *pick-list*

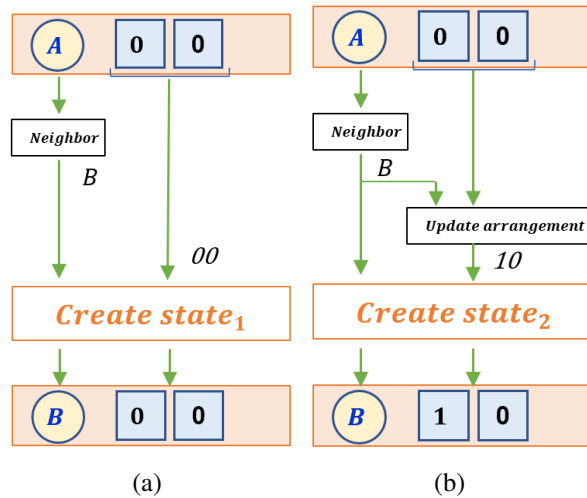


FIGURE 3-19 (a) La procédure de la création de $state_1$ (b) La procédure de la création de $state_2$

Afin de mieux comprendre le principe de ce processus, un exemple de création des états et des arcs de transition est présenté dans la figure 3-19. Cette illustration prend l'exemple du graphe représenté dans la figure 3-17, avec le point de départ et d'arrivée A et la *pick-list* $P = \{B, D\}$. Par suite, l'étape de l'initialisation consiste à définir l'état $(A, 00)$ comme état

de départ. Ensuite, la procédure d'exploration du graphe d'entrepôt est déclenchée (toujours le graphe représenté dans 3-17). Les nœuds voisins directs de A sont B et C . Le voisin B étant un nœud requis dans la première position de l'ensemble P et le voisin C est un nœud non requis. Pour le premier cas (nœud B), le $state_1$ et le $state_2$, successeurs de l'état $(A, 00)$ sont créés (voir figure 3-19). Cependant, pour le second (voisin C), seul le $state_1$ qui lui est associé est créé. Ces états sont expliqués par un tuple dans lequel le chiffre relatif au nœud du voisin peut être égal à 0 ($state_1$) ou à 1 ($state_2$). Dans cette figure, l'étape de "*update arrangement*" fait référence au processus de *bitmasking*, décrit dans la figure 3-18. En outre, la procédure de la création des états 1 et 2 est décrite dans l'algorithme 3 et l'algorithme 4, respectivement. Le premier algorithme décrit la fonction *CreateState1*. Elle prend en entrée l'état courant (son nœud u , son arrangement a , sa masse m , et son coût courant J^*) ainsi que le nœud spatial voisin $v \in V$. En sortie, il génère un tuple $state_1 = (v, a)$ qui associe le nœud successeur v avec le même arrangement actuel et la même masse ainsi que le coût cumulé pour passer de l'état actuel à l'état $state_1$ en utilisant la fonction *transitionCost* définie dans la sous-section suivante. En outre, l'état actuel est défini comme prédécesseur du $state_1$. Le deuxième algorithme définit la fonction *CreateState2* qui prend les mêmes informations

Algorithm 3: CreateState1 (s^*, v)

```

1  $u \leftarrow s^*.vertex$ ;  $m \leftarrow s^*.mass$ ;  $a \leftarrow s^*.arr$ ;  $J^* \leftarrow s^*.cost$ ;
2  $d \leftarrow dist(u, v)$ 
3  $c \leftarrow J^* + Transition\_cost(d, m)$ 
4  $s_1.vertex \leftarrow v$ 
5  $s_1.arrangement \leftarrow a$ 
6  $s_1.name \leftarrow (v, a)$ 
7  $s_1.cost \leftarrow c$ 
8  $s_1.mass \leftarrow m$ 
9  $s_1.predecessor \leftarrow s^*$ 
10 Return  $s_1$ 

```

d'entrées que l'algorithme 3, mais aussi l'information de la *pick-list*. L'algorithme commence par la détermination de y , la position/index de v dans la *pick-list* qui présente une information indispensable pour identifier les charges ramassées par le *Ps-AMR*. Ensuite, un masque est appliqué sur l'arrangement actuel pour donner un nouvel arrangement a_2 . Ce masque est un nombre binaire composé de p bits définis avec des zéros partout sauf le $y - th$ bit qui est fixé à

un. a_2 est par la suite jumelée au nœud successeur v pour créer $state_2 = (v, a_2)$. En outre, une nouvelle masse m_2 est calculée. Cette dernière est égale à la somme de la masse actuelle et de la masse associée au nœud v . De plus, le coût de $state_2$ correspond à la sommation du coût actuel J^* et le coût donné par la fonction *transitionCost*. Cette dernière utilise le nœud actuel u , le nœud successeur v , et m_2 pour calculer le coût de transition entre l'état actuel et $state_2$ à travers l'équation 3.15 préalablement définie. Ces deux algorithmes représentent les piliers de l'algorithme principal *EE-OPR*, défini dans la section 3.2.6 ci-dessous. Le principe de création des états peut être observé à travers la figure 3-20 qui illustre la totalité des ramifications de l'arborescence des états (la manière naïve) si on considère le graphe de la figure 3-17, avec les nœuds de ramassage requis B et D . Nous allons voir dans les sections qui suivent que *EE-OPR* utilise cette procédure d'une manière plus intelligente.

Algorithm 4: CreateState2 (s^*, v, V_p)

```

1  $u \leftarrow s^*.vertex$ ;  $m \leftarrow s^*.mass$ ;  $a \leftarrow s^*.arr$ ;  $J^* \leftarrow s^*.cost$ ;
2  $d \leftarrow dist(u, v)$ 
3  $y \leftarrow V_p.index(v)$ 
4  $a_2 \leftarrow a$  OR  $Mask(y)$ 
5  $m_2 \leftarrow m + M(y)$ 
6  $c_2 \leftarrow J^* + Transition\_cost(d, m_2)$ 
7  $s_2.vertex \leftarrow v$ 
8  $s_2.arrangement \leftarrow a_2$ 
9  $s_2.name \leftarrow (v, a_2)$ 
10  $s_2.cost \leftarrow c_2$ 
11  $s_2.mass \leftarrow m_2$ 
12  $s_2.predecessor \leftarrow s^*$ 
13 Return  $s_2$ 

```

3.2.5.4 Coût de transition en termes de temps et d'énergie

En définissant, progressivement, les nœuds d'états et les arcs de transition, un nouveau graphe d'états $G'(V', A')$ est créé. Ce graphe (qui est un *DAG (Direct Acyclic Graph)*) part du nœud de départ $s_{start} = (v_{start}, a_{start})$ et ramifie les nœuds d'états reliés par des arcs de transition. Le graphe est pondéré en fonction du critère à minimiser, soit les critères temps et énergie simultanément. Nous considérons, les mêmes modèles de temps et d'énergie consommée présentés dans la première partie du travail (t_{ij} 3.14 et e_{ij} 3.16). Cependant, les

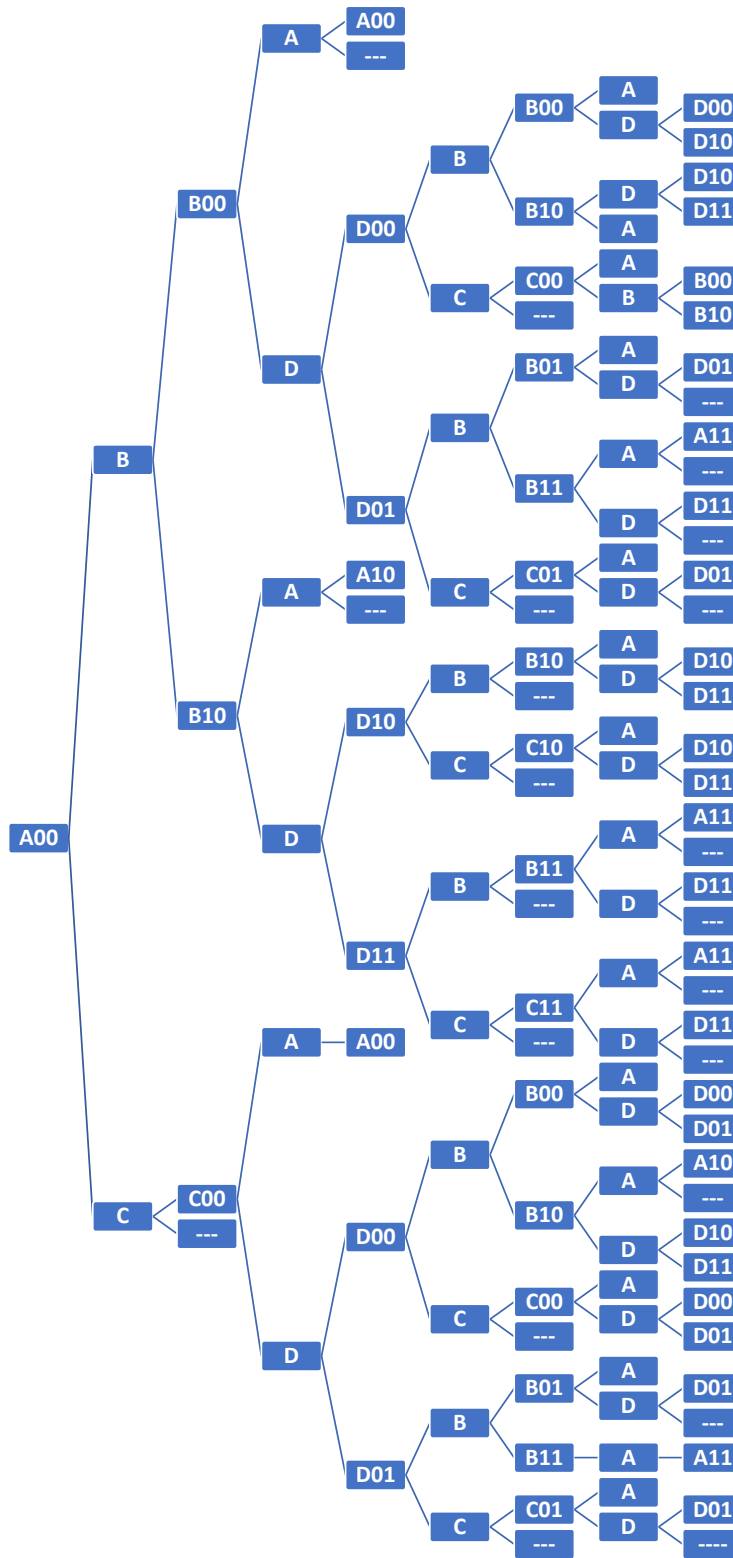


FIGURE 3-20 Un exemple d'arborescence de création des nœuds d'états (façon naïve).

nœuds i et j , dans cette partie, sont des nœuds *state* (non seulement des nœuds spatiaux). De plus, l'information de la masse actuelle en avançant dans l'algorithme est déterminée à partir de l'arrangement actuel. Par exemple, considérons le graphe dans la figure 3-17, si à une étape donnée, l'état actuel est $(B, 10)$ (c.-à-d la position actuelle est B et la charge transportée est m_B^L). La masse considérée dans le modèle pour passer de $(B, 10)$ à un état suivant est m_B^L . Par suite, si, on considère le voisin D comme successeur de B et que la charge D (m_D) est transportée, alors la nouvelle masse totale courante est $m_B + m_D$. Cependant, cette valeur est considérée dans le prochain calcul de coûts de transition, car celui-ci requiert la masse actuelle en partant du nœud en question.

Par suite, pour être générale, soit i est un nœud d'état et j est son successeur, M_i^R et M_i^L sont les masses totales de l'AMR lorsqu'il atteint le nœud $i.vertex$ et lorsqu'il le quitte, respectivement. La masse totale, exprimée en kg, comprend sa propre masse m_0 , les SKUs (les articles) qu'il transporte et la masse du préparateur de commandes (s'il est conduit pour se déplacer). Par exemple, si l'article (la charge) dans $i.vertex$ est prélevée lorsque l'AMR atteint ce nœud, $M_i^L = M_i^R + M[y_i]$ (masse actuelle plus la masse de l'article en position y_i de la *pick-list*). Sinon, la masse reste la même ; $M_i^L = M_i^R$. Par suite, la masse initiale en quittant l'état de départ M_{start}^L est m_0 . Par conséquent, l'énergie mécanique en translation de l'état i à l'état j (e_{ij}) est donnée par \hat{F}_T multiplié par la distance entre la position de i ($i.vertex$) et la position de j ($j.vertex$). De plus, comme nous ne considérons pas l'effet du trafic dans cette partie ($\bar{w}_{tr}^{d,t}(i, j) = 1, \forall (i, j) \in A_{topo}$), pas de pénalité attribuée à cause du trafic alors $e_{ij}(m_i^L) = e_{ij}(m_i^L)^{ref}$. Ainsi, le modèle de transition du coût de la consommation énergétique entre deux états i et j est exprimé comme suit :

$$e_{ij} = (M_i^L g \mu) d_{(i.vertex \ j.vertex)} \quad (3.16)$$

Tel que $d_{(i.vertex \ j.vertex)}$ présente la distance qui sépare la position de l'état i et la position de l'état j . Il est important de mentionner que la minimisation de seulement e_{ij} peut résulter des tours plus longs que les tours résultant des méthodes exactes qui minimise seulement le temps (les plus courts chemins). Nous allons appeler ces méthodes qui considèrent uniquement le temps/distance de parcours : *TMs* (*Travel Time Minimisation*)/ *SPs* (*Shortest Path*). En outre, nous appellerons les méthodes qui résolvent l'*OPR* considérant uniquement l'énergie : *ECM*

(*Energy Consumption Minimisation*).

Par exemple, supposons un graphe bidirectionnel C_n avec 4 nœuds A , C , D , et E et 5 arêtes pondérées $(AC, 6)$, $(AD, 10)$, $(CD, 8)$, $(CE, 6)$, et $(ED, 6)$. L'*OPR* consiste à partir de A , chercher C et D (l'ordre n'est pas prédéfini) et les amener au point E . On suppose que les masses m_0 de l'AMR, m_c (en C) et m_D (en D) sont : 100 kg, 80 kg et 1 kg, respectivement, et supposons que $\varepsilon = \mu g$:

- Le chemin qui consomme le moins de temps est : $P_1 = \{(AC), (CD), (DE)\}$ avec une longueur de 20 m et nécessitant 312,6 J pour exécuter la tournée de préparation des commandes ($E_1 = \varepsilon[d_{AC}m_0 + d_{CD}(m_0 + m_C) + d_{DE}(m_0 + m_C + m_D)]$).
- Le chemin qui consomme le moins d'énergie est : $P_2 = \{(AD), (DC), (CE)\}$ avec une longueur de 24 m et nécessitant 289,4J pour exécuter la tournée de préparation de commandes ($E_2 = \varepsilon[d_{AD}m_0 + d_{DC}(m_0 + m_D) + d_{CE}(m_0 + m_D + m_C)]$).

Ainsi, il est clair que P_2 surpasse P_1 en termes d'énergie. Cependant, P_2 consomme plus de temps que P_1 . D'où, l'importance de considérer les deux critères à la fois avec un bon compromis temps et énergie. Cet aspect est considéré dans l'approche proposée *EE-OPR* qui minimise les dépenses en termes de temps et d'énergie, simultanément. Nous considérons, par conséquent, la même fonction de coûts 3.15 définie dans la première partie de ce travail, où les deux coefficients représentant le coût en dollar de chaque unité de temps et d'énergie et peut prioriser un critère par rapport à l'autre. Par exemple, dans le cas d'un grand nombre de commandes dans un entrepôt, le facteur temps est crucial et il est recommandé d'augmenter le coût en dollars du temps dépensé. Si on applique ce principe de création progressive

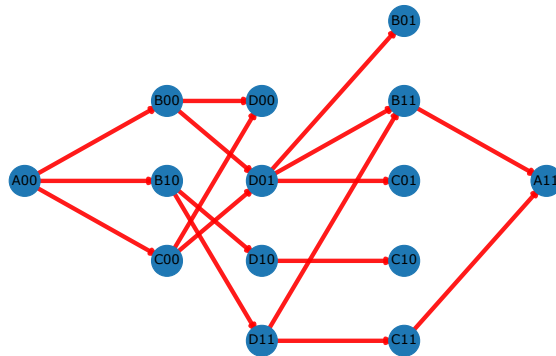


FIGURE 3-21 Dynamic states graph G' .

des états et d'arcs de transitions sur un graphe spatial simple comme le graphe représenté

précédemment dans la figure 3-17, nous allons avoir le graphe G' représenté dans la figure 3-21. De cette manière $EE-OPR$ élimine plusieurs états et transitions inutiles, comparativement à une approche naïve. Par exemple, si on considère un exemple de graphe composé de cinq nœuds dont 2 nœuds sont requis, la matrice d'adjacence exhaustive relative au problème OPR , a une taille de $2^2 \cdot 5^2 \cdot 5$ (voir figure 3-22). Ce chiffre peut augmenter rapidement en fonction de la taille de la pick list p (par exemple si la taille de la *pick-list* est 20, on aura $2^{20} = 1048576$ arrangements possibles, $2^{64} = 18446744073709551616$ arrangements possibles, etc). Par contre, $EE-OPR$ évite cette explosion en nombre d'états en créant d'une manière dynamique intelligente seulement les états et les transitions possibles.

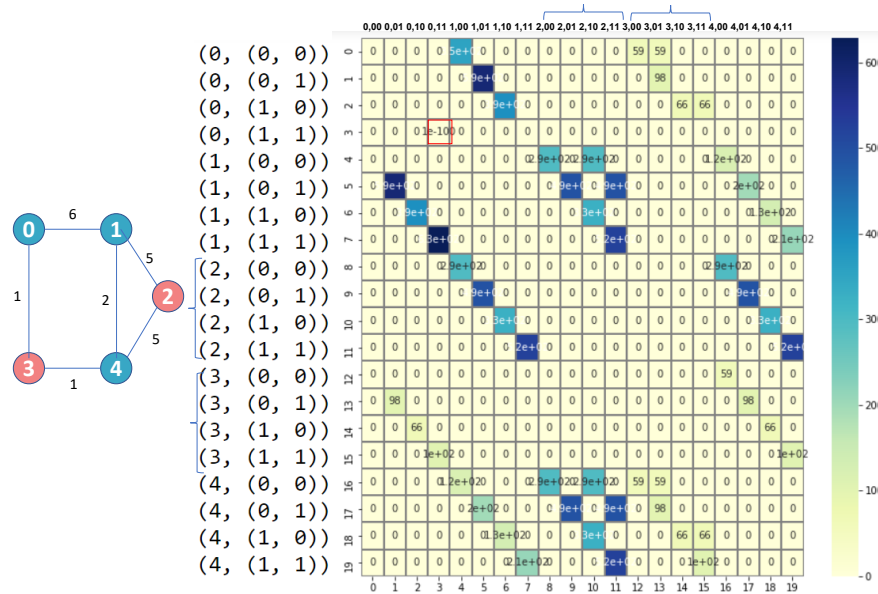


FIGURE 3-22 Exemple de matrice d'adjacence d'états avec toutes les connexions possibles

3.2.6 Le problème de recherche dans un graphe

Parallèlement à la construction du graphe d'états, le coût associé à chaque état est mis à jour en suivant le principe d'optimalité de Bellman [84]. Cette technique est inspirée de l'algorithme "dynamisé" de Dijkstra [202] (tel qu'il est nommé dans l'article) qui est considéré comme une procédure d'approximation successive de la programmation dynamique. En considérant le graphe $G'(V', A')$, chaque arc $(i, j) \in A'$ est pondéré par $c_{ij} \in \mathbb{R}_{\geq 0}$. Supposons que $s_{start}, s_{target} \in V'$ sont les nœuds "state" source et destination, respectivement. On définit

par X un chemin dirigé d'une séquence de nœuds v_i tel que : $X = \langle v'_1, \dots, v'_{nb_p} \rangle$, où $v'_1 = s_{start}$, $v'_m = s_{target}$, $(v_k, v_{k+1}) \in A'$, $\forall k \in \{1, \dots, nb_p - 1\}$. $c(X)$ est le coût du chemin, c'est-à-dire la somme des coûts des arcs dans X ; $c(X) = \sum_{i=1}^{nb_p-1} c_{v_i, v_{i+1}}$. Évidemment, les nœuds d'état qui constituent le chemin résultant sont visités au plus une fois. On précise qu'on parle des nœuds d'état et non des nœuds spatiaux appartenant à V . Cela signifie que nous pouvons passer deux fois sur un nœud, mais avec un arrangement différent.

Étant donné que le STSP écoénergétique est vu comme un problème du plus court chemin entre l'état s_{start} et l'état s_{target} , on présente la formulation standard suivante de programmation en nombres entiers : [203] (le plus court chemin désigne ici le chemin du coût optimal) :

$$\text{Minimiser } \sum_{(i,j) \in A'} (c_t(t_{ij}) + c_e(e_{ij}))x_{ij} \quad (3.17)$$

$$\text{tels que : } \sum_{j \in \Gamma_i^+} x_{ij} - \sum_{j \in \Gamma_i^-} x_{ji} = \quad (3.18)$$

$$\begin{cases} 1 & \text{if } i = s_{start} \\ 0 & \text{if } i \neq s_{start}, s_{target} \\ -1 & \text{if } i = s_{target} \end{cases} \quad (3.19)$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A' \quad (3.20)$$

avec i et j désignent des nœuds d'états, Γ_i^+ et Γ_i^- désignent les ensembles de nœuds successeurs et prédécesseurs, respectivement, x_{ij} représente la variable de décision définissant si l'arc (i, j) fait partie du plus court chemin ou non. Elle prend la valeur 1 dans le premier cas et 0, sinon. La contrainte 3.19 spécifie que pour chaque nœud appartenant au plus court chemin, différent de l'état de départ et de l'état cible, doit avoir le même nombre d'arcs entrants et sortants. Tandis que la contrainte 3.20 impose les valeurs binaires de x_{ij} .

Afin de résoudre le STSP écoénergétique présenté, *EE-OPR* est proposé (l'algorithme 5). L'approche suit les étapes suivantes : Créer une file d'attente Q et l'initialiser avec l'état de départ $state_{start}$ avec un coût de 0. Tant que cette file n'est pas vide, sélectionner l'état ayant

le coût minimal, le considérer comme un état courant et le supprimer de la file de priorité. Ensuite, définir ses composantes (identifier le nœud (u) et l'arrangement (a)). Explorez le nœud (u) en déterminant les nœuds voisins. Ces nœuds seront utilisés pour créer $state_1$ et $state_2$, et pour les ajouter à la file d'attente s'ils n'existent pas déjà. Sinon, vérifier si le coût calculé actuel peut être diminué (optimisé) en passant par l'état actuel pour atteindre l'état suivant. Ceci est décrit par la fonction RELAX (l'algorithme 6). En parallèle, la liste des coûts et la liste des prédécesseurs sont créées pour enregistrer les états ajoutés avec leurs coûts et leurs prédécesseurs, respectivement. Une mise à jour de ces coûts et prédécesseurs peut être effectuée en utilisant la fonction RELAX qui tente de minimiser le coût des chemins. Le coût final de l'état cible représente le coût optimal de la tournée de prélèvement. En outre, la liste des prédécesseurs (pred list) permet de tracer le chemin optimal depuis les états cibles jusqu'à l'état de départ. Notez que *EE-OPR* empêche le déplacement vers le nœud cible sans avoir récupéré tous les éléments de la *pick-list*. Ceci est réalisé pendant la phase d'exploration par l'interdiction de la création d'états et de transitions lorsque le nœud voisin est le nœud cible, mais que l'arrangement est différent du '*full arrangement*' (voir la première instruction *if* de l'algorithme). Par conséquent, si le nœud de destination est différent du nœud de départ, le seul état qui utilise ce nœud est l'état destination (le nombre total d'états possibles est donc inférieur à $(n - 1)2^p + 1$. Sinon, si la position du dépôt représente à la fois le point de départ et le point d'arrivée, les deux états possibles dans cette position sont les états ayant le "*empty arrangement*" (au départ) ou le "*full arrangement*" (à l'arrivée). Le nombre total d'états possibles est $(n - 1)2^p + 2$. Par conséquent, seuls les arcs quittant l'état de départ et les arcs entrant dans l'état de destination sont conservés. De plus, dans le cas où un nœud voisin v d'un nœud courant u appartient à la *pick-list* et que l'arrangement courant est déjà a_2 (le bit correspondant à v dans l'arrangement est égal à un). Dans ce cas, $state_2$ est égal à $state_1$. Par conséquent, seul $state_1$ est ajouté.

Nous allons maintenant discuter la complexité temporelle de *EE-OPR*. Puisque la file d'attente prioritaire Q est représentée comme un tas binaire, où les opérations sont effectuées en $O(\log(q))$ temps (avec q est la taille de Q), la complexité temporelle de *EE-OPR* est mesurée comme suit. Le temps pris pour chaque opération d'extraction est de $O(\log|V'|)$. De plus, l'itération sur tous les voisins des nœuds et la mise à jour de leurs valeurs *dist* sont

Algorithm 5: EE-OPR

```

1 initialization : set  $s_{start}$ 
2  $s_{start}.mass \leftarrow M_0$ ;
3  $cost[s_{start}] \leftarrow 0$ ;
4  $pred[s_{start}] \leftarrow \text{undefined}$ 
5  $Q \leftarrow [(0, s_{start})]$ 
6 while  $Q$  is not empty do
7    $s^* \leftarrow \text{Extract-Min}(Q)$ 
8    $u \leftarrow s^*.vertex$ ;
9    $a \leftarrow s^*.arrangement$ ;
10  for all  $v \in \Gamma_u^+$  do
11    if  $v == s_{target}.vertex$  and  $a \neq s_{target}.arrangement$  then
12      | continue
13    end
14     $S_1 \leftarrow \text{CreateState1}(s^*, v)$ 
15    if  $S_1 \in Q$  then
16      | Relax( $s^*, S_1$ )
17    else
18      |  $Q.insert(S_1)$ 
19    end
20     $cost[S_1] = S_1.cost$ 
21     $pred[S_1] = S_1.pred$ 
22    if  $v \in V_P$  then
23       $S_2 \leftarrow \text{CreateState2}(s^*, v)$ 
24      if  $S_2 \in Q$  then
25        | Relax( $s^*, S_2$ )
26      else
27        |  $Q.insert(S_2)$ 
28      end
29       $cost[S_2] = S_2.cost$ 
30       $pred[S_2] = S_2.pred$ 
31    end
32  end
33 end

```

Algorithm 6: RELAX (s^*, st)

```

1  $m \leftarrow s^*.mass;$ 
2  $d \leftarrow dist(s^*.vertex, st.vertex)$ 
3  $w(s^*, st) \leftarrow s^*.cost + Transition\_cost(d, m)$ 
4 if  $st.cost > s^*.cost + w(s^*, st)$  then
5   |  $st.cost \leftarrow s^*.cost + w(s^*, st)$ 
6 end

```

exécutées $O(|V'|)$ fois et chaque mise à jour de la priorité des nœuds prend $O(\log|V'|)$ de temps. Par conséquent, le coût total des deux calculs prend $O(|V'| \log|V'|)$ de temps [204]. Par conséquent, la complexité temporelle globale de *EE-OPR* est de $O(|V'| \log|V'|)$. Étant donné que le nombre maximal d'états pouvant être créés est inférieur à $n2^p$, la complexité algorithmique de *EE-OPR* est de $O(n2^p \log(n2^p))$.

Chapitre 4 - Évaluation des approches pour la planification de chemin écoénergétique

Afin d'évaluer et discuter la performance des méthodes proposées décrites précédemment, ce chapitre présente une étude comparative entre les deux approches proposées écoénergétiques et les méthodes de *PP* utilisées dans la littérature optimisant seulement la distance parcourue (*SP Shortest Path*) que ce soit pour le problème *GPP* (*SP-GPP*) ou *OPR* (*SP-OPR*). En effet, comme précédemment indiqué, les articles qui traitent l'optimisation de chemin en considérant l'énergie consommée agissent plutôt sur la planification locale. Les méthodes de planification de chemin globale présument que la minimisation des distances parcourues est le seul moyen de minimiser l'énergie. De plus, pour mettre en évidence l'importance de la considération du compromis temps/énergie et analyser cet aspect, deux approches de résolutions de *GPP* et *OPR* qui considèrent seulement le critère énergétique sont considérées séparément pour chaque problème. Ces méthodes sont appelées *ECM-GPP* (*Energy Consumption Minimization*) et *ECM-OPR*. Par conséquent, dans la suite du document, nous allons utiliser les préfixes *SP*, *EMC*, et *EE*, pour désigner, respectivement, les méthodes qui considèrent seulement le critère distance, seulement le critère énergétique et les deux critères à la fois (le compromis géré par les approches proposées). Par suite, dans l'ordre de présentation des deux approches dans la section précédente, nous allons présenter les simulations réalisées pour évaluer les deux approches *EE-GPP* et *EE-ORP*, les résultats obtenus et la discussion de ces derniers.

À cet effet, différents scénarios de problèmes de planification de chemin *Source-destination* et multidestinations sont étudiés afin de démontrer l'efficacité des différentes méthodes implémentées. Ainsi, pour évaluer la méthode *EE-GPP*, diverses simulations ont été réalisées pour le problème de planification *GPP* (*Source-Target*) dans les sous-sections 4.1 et 4.2 pour évaluer respectivement l'effet de la résistance au roulement et l'effet du trafic sur la

consommation énergétique en relation avec la planification de chemin écoénergétique. De même, une évaluation de l'approche *EE-ORP* est présentée dans la section 4.3 qui traite le problème du routage multidestinations d'un *Ps-AMRs* dans un entrepôt, examinant l'effet de la variation de la masse entre les différentes positions de prélèvements et le séquençement de leurs visites.

4.1 Évaluation de la méthode *EE-GPP*, avec un *grid-graphe* de résistance au roulement

4.1.1 Configuration de l'espace et représentation du facteur de la résistance au roulement

Une structure de grille d'occupation pour la navigation de l'*AMR* est considérée. Par suite, trois configurations de *grid-graph* ($n \times n$) de différentes tailles sont étudiées (*Layout 1* : $n = 10$; *Layout 2* : $n = 20$ et *Layout 3* : $n = 30$). Elles représentent un environnement de manufacture, avec des obstacles statiques dispersés dans le milieu de déploiement du robot. Les emplacements des obstacles sont définis au hasard pour chaque configuration à travers la suppression d'un nombre déterminé de nœuds (avec les arêtes qui lui sont connectées) de la grille complètement remplie, initialement. Par conséquent, lors de la génération des trois configurations, une suppression de 10, 20 et 30 nœuds est effectuées pour chaque *layout 1*, *2* et *3* respectivement. Pour une meilleure compréhension, un exemple simplifié est illustré dans la figure 4-1 qui montre dix configurations différentes d'environnements de manufactures à travers des grilles (10×10) avec un retrait de 1 à 3 nœuds aléatoires et une sélection de 3 sommets d'intérêts aléatoires (en rouge) dans chaque configuration. Par ailleurs, l'information de friction est définie à travers trois matrices d'adjacences correspondantes à chaque *layout*. Ces matrices sont générées d'une manière aléatoire avec des coefficients de résistance au roulement dans l'intervalle $[0.01, 0.06]$ pour chaque arête [205]. Par exemple, la figure 4-2 montre deux "*grid-graphs*" (10×10) pondérés avec l'information de distance et de résistance au roulement, respectivement. Ces derniers sont utilisés comme entrées pour l'approche *EE-GPP*. Le graphe bidirectionnel de résistance au roulement à droite de la figure 4-2-b est coloré avec des dégradations de bleu, reflétant le degré de rugosité des surfaces par arête (le coefficient de résistance au roulement).

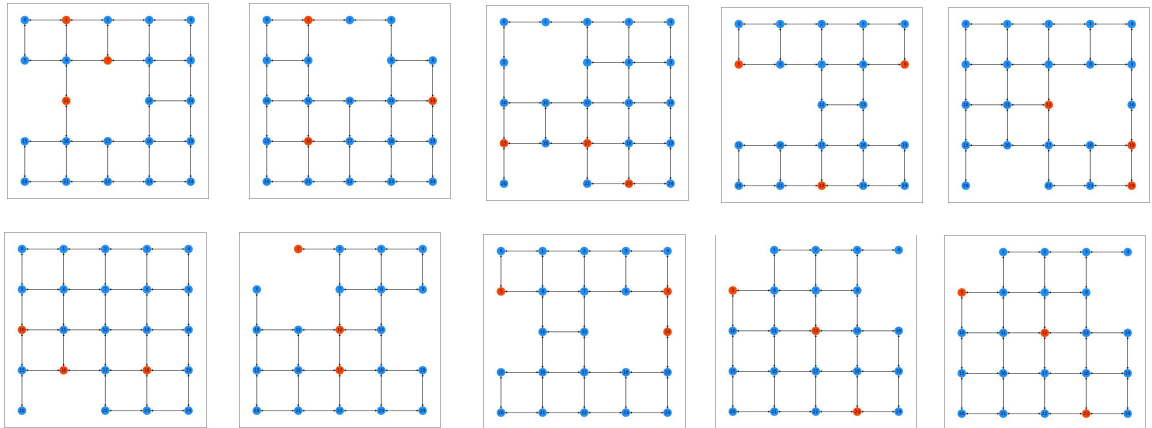


FIGURE 4-1 Exemple de génération aléatoire d'environnements de manufactures (10 configurations différentes présentées par une grille 10*10 avec un retrait de 3 nœuds aléatoires et une sélection de 3 sommets d'intérêts aléatoires (en rouge)).

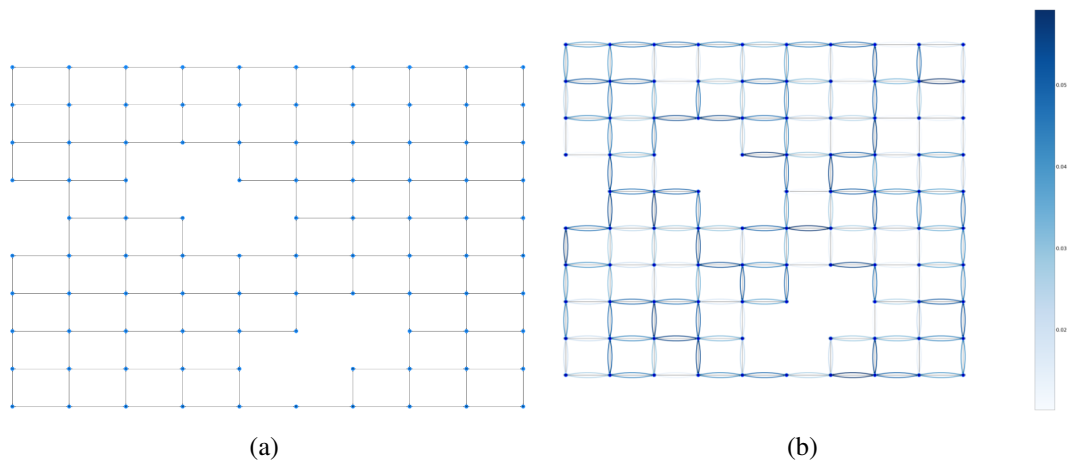


FIGURE 4-2 Informations sur l'environnement avec la structure du graphe en grille (a) graphe en grille avec l'information sur la distance (b) graphe en grille avec l'information sur la friction (la dégradation en bleu reflète la grandeur du coefficient de résistance au roulement).

4.1.2 Résultats de simulation

Afin d'évaluer la méthode proposée *EE-GPP* et l'effet de résistance au roulement, nous considérons les graphes présentés précédemment (trois graphes "distances" de chaque *layout* et trois graphes de résistances aux roulements correspondants (voir 4-1). Cependant, nous soulignons que *EE-GPP* peut être utilisé avec toutes autres configurations d'industries. L'im-

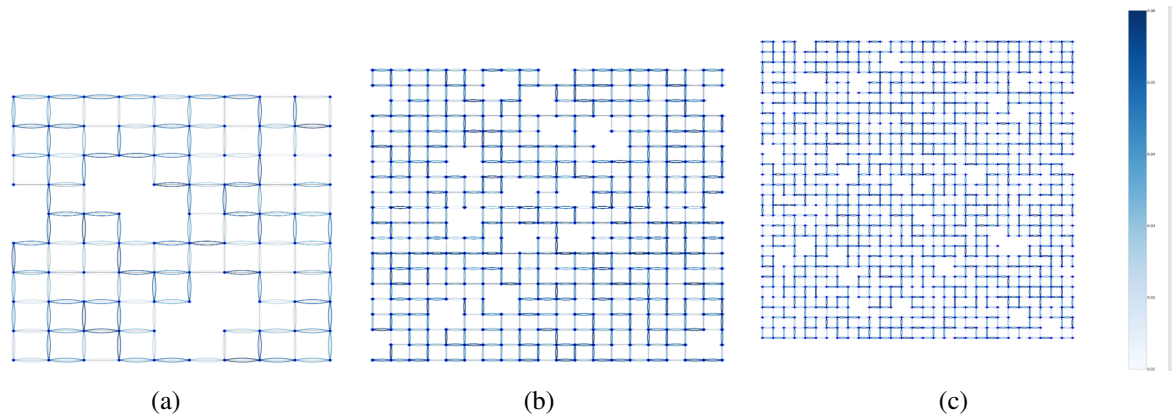


FIGURE 4-3 Représentation des configurations pour les simulations : *layout 1* (a) *grid-graph* "résistance au roulement" de *Layout 1* (b) *grid-graph* "résistance au roulement" de *Layout 2* (c) *grid-graph* "résistance au roulement" de *Layout 3*.

plémentation de la méthode et des simulations est faite en utilisant un ordinateur Intel Xeon W-2102 avec 128 GB DDR4 RAM et nous avons utilisé Python comme langage de programmation. Ces simulations ont pour objectif, la comparaison de la méthode proposée *EE-GPP* avec une méthode *SP-GPP* qui utilise *Dijkstra* traditionnelle pour la recherche dans un graphe. En outre, pour étudier le compromis temps/énergie, nous avons considéré une méthode *ECM-GPP*, la version "zéro" de notre approche qui considère seulement le critère énergétique dans la fonction objective ($c_{ij} = e_{ij}$ 3.15) ignorant le facteur distance/temps qui empêche d'avoir des chemins beaucoup plus longs que le *SP*. Ainsi, des simulations sont réalisées comme suit :
Pour chaque *Layout 1, 2 et 3* :

- Définir les coefficients de résistances au roulement au hasard pour chaque arête du graphe de la manufacture (pour chaque arête du graphe, générer une valeur de coefficients de résistance au roulement dans l'intervalle $[0.01, 0.06]$ d'une manière aléatoire (voir la figure 4-3)).

- Générer 100 instances de problèmes de planification de chemins *source-destination* (générer les points de départ et les points d'arrivée aléatoirement pour chaque instance de problème *GPP*) et les résoudre à l'aide de chacune des méthodes *SP-GPP*, *ECM-GPP* et *EE-GPP*. Ainsi, chaque instance correspond à un problème *GPP* avec un nœud *start* et un nœud *target* sélectionnés d'une manière aléatoire à partir de l'ensemble des nœuds d'intérêts ($I \in V$) représentant les différentes positions des stations de manutention dans la manufacture. Ensuite, les coûts en distance, énergie et argent des "*paths*" résultants de chaque méthode sont analysés.

L'énergie consommée par les chemins les plus courts (résultats de *SP-GPP*) est comparée à celle demandée par les *paths* donnés suite à l'application de la méthode *ECM-GPP* et *EE-GPP*, à travers le calcul de la différence entre chaque méthode et la *SP* et la déduction du pourcentage de gain en énergie de chacun d'eux. Parallèlement, la distance parcourue pour chaque *path* est calculée et une comparaison a été faite par rapport à la longueur du plus court chemin. Enfin, les coûts en dollars de tous les *paths* sont calculés afin d'unifier les valeurs et comparer toutes les méthodes. La figure 4-4 montre les résultats de simulations des trois *layouts* regroupés dans les boîtes à moustaches. Il s'agit des coûts des *paths* résultants suite à l'exécution de *SP-GPP*, *ECM-GPP* et *EE-GPP*, pour les trois configurations. Ces coûts sont calculés à l'aide de la fonction C_{ij} 3.15. Il s'agit de (i) calculer pour chaque chemin résultant de chaque méthode sa longueur par suite le temps requis pour le traverser et le multiplier par le coût d'une unité de temps, (ii) calculer pour chaque chemin résultant de chaque méthode l'énergie requise pour le déplacement et le multiplier par le coût d'une unité d'énergie. On constate bien que traverser les *paths* générés par la méthode proposée coûte toujours moins que les chemins donnés par *SP-GPP* et *ECM-GPP*. Par ailleurs, ces gains augmentent en fonction des *layouts* allant de plus petit au plus large. En effet, lorsque la superficie de la manufacture augmente, il y a plus de chance d'avoir des gains plus importants à cause des distances plus longues à parcourir et au graphe de réseau plus élargi (plus de possibilités de chemins). Pour une meilleure visibilité de résultats par *layout*, un graphique représentant les coûts des *paths* résultants de simulations de 100 instances d'exécutions de scénarios de *GPP* dans la première configuration est illustré dans la figure 4-5. On observe, également, que les coûts des chemins écoénergétiques sont toujours inférieurs ou égaux aux coûts des plus courts chemins.

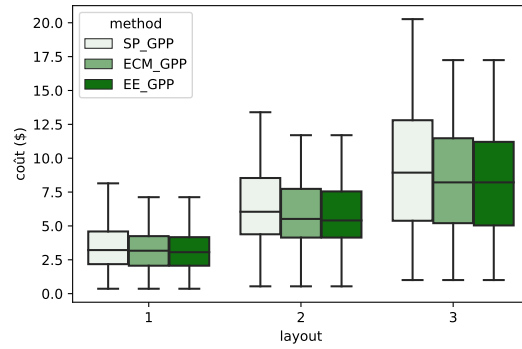


FIGURE 4-4 Résultats des simulations *EE-GPP* des trois *layouts*

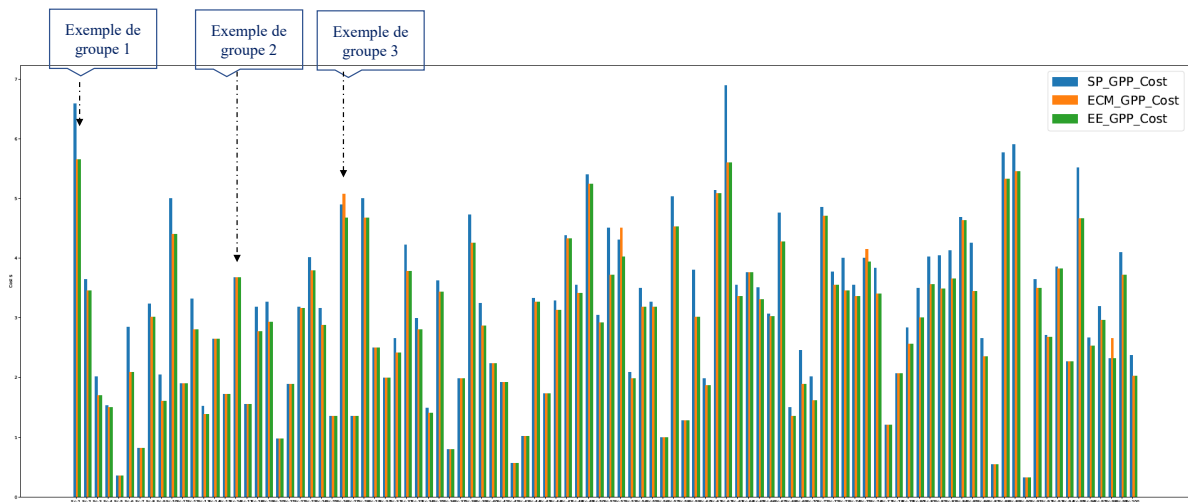


FIGURE 4-5 Résultats de simulations de 100 instances de *GPP* pour le *layout 1*

4.1.3 Analyse et discussion

En analysant les résultats présentés dans les figures 4-4 et 4-5, on constate qu'il existe trois cas de figure qui peuvent être classifiés en groupe de scénarios, comme suit :

- Groupe 1 : *ECM-GPP* apporte un gain énergétique sans impacter la distance parcourue (un autre chemin de même longueur que le *SP*).
 ⇒ *EE-GPP* sélectionne le même chemin que *ECM* ou bien un chemin avec le même coût.

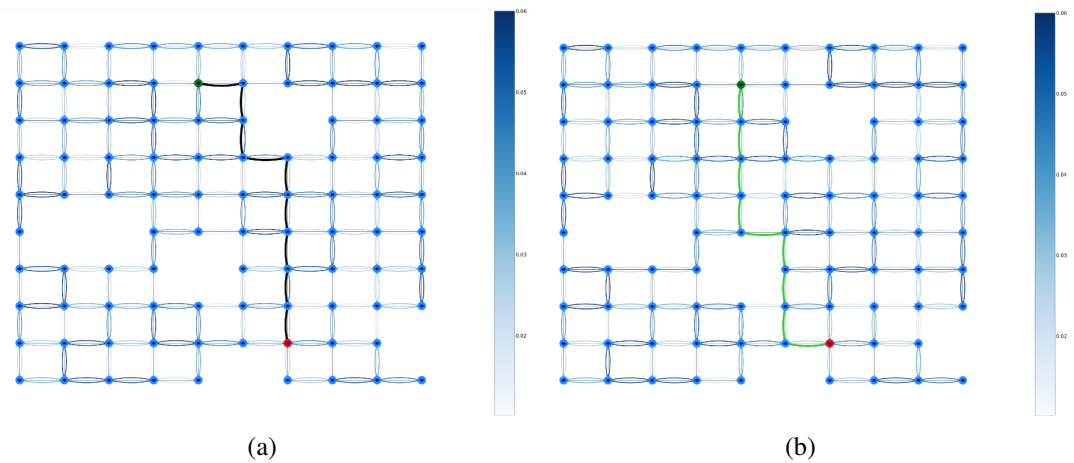


FIGURE 4-6 Exemple d'un scénario qui appartient au groupe 1 (a) *SP-GPP path* (b) *EE-GPP path*.

- Groupe 2 : *ECM-GPP* n'arrive pas à apporter un gain énergétique comparativement à *SP-GPP* (il n'existe pas une meilleure alternative).
 ⇒ *EE-GPP* sélectionne le même chemin que *ECM* et *SP* ou bien un chemin avec le même coût.
- Groupe 3 : *ECM-GPP* fournit un gain énergétique comparativement à *SP-GPP*. Cependant, le chemin donné est plus long que celui de plus court chemin *SP*.
 ⇒ Dépendement de la différence de longueur entre les deux chemins *ECM-GPP* et *SP-GPP* et du coût d'une unité de temps et d'une unité d'énergie, *EE-GPP* (qui essaie de trouver un compromis temps/énergie) peut sélectionner un de ces deux chemins ou un autre avec un meilleur compromis. Il peut sélectionner le même chemin que *ECM-GPP*, si la différence de longueur n'est pas assez grande, par exemple. Autrement, il peut

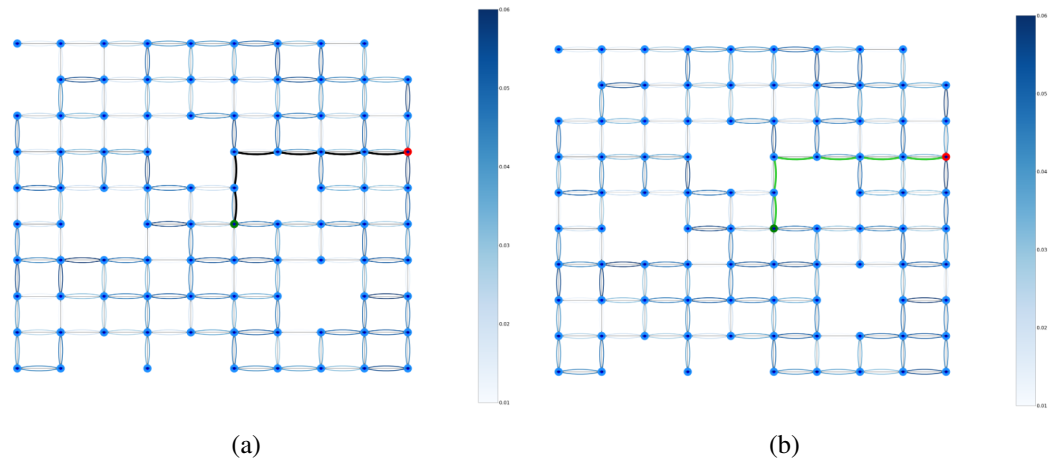


FIGURE 4-7 Exemple d'un scénario qui appartient au groupe 2 (a) *SP-GPP path* (b) *EE-GPP path*.

adopter le même chemin *SP-GPP* si ce n'est plus rentable de prendre un chemin plus long pour une petite économie en énergie, par exemple (le chemin le plus économe en énergie est très long par rapport au plus court chemin). Sinon, un autre chemin qui apporte un bon compromis entre les deux peut être choisi (ceci est grâce à la fonction objective définie précédemment tenant en compte le coût total en argent des deux critères temps/distance et énergie consommée).

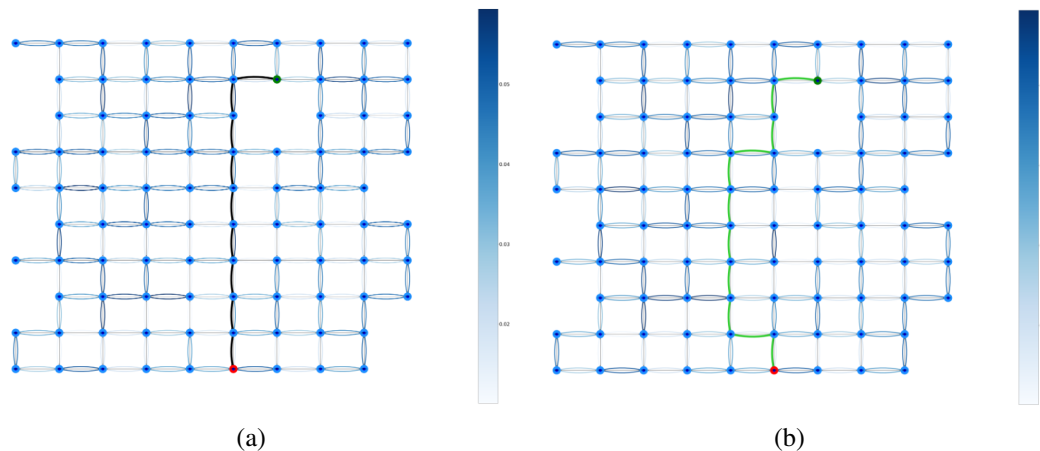


FIGURE 4-8 Exemple d'un scénario qui appartient au groupe 3 (a) *SP-GPP path* (b) *EE-GPP path*.

Le tableau 4-9 affiche un aperçu de résultat de compilation de plusieurs instances de *GPP* ainsi que les coûts des méthodes *ECM-GPP* et *EE-GPP* versus *SP-GPP* appliquées dans le

layout 1. Par exemple, la ligne 0-colonne 2 du tableau signifie que le *path-ECM* consomme moins 18.18% que celui de *SP*. Dans cet exemple, *EE-GPP* sélectionne le même chemin que le *ECM*.

	La distance parcourue du SP_Path (m)	Pourcentage de perte en distance si on utilise ECM (%)	Pourcentage de perte en distance si on utilise EE_GPP (%)	l' énergie dépensée du SP_path	l' énergie dépensée du ECM_path	l' énergie dépensée du EE_GPP_path	Pourcentage de gain energie si on utilise ECM (%)	Pourcentage de gain en energie si on utilise EE_GPP (%)	Pourcentage de gain en argent si on utilise EE_GPP (%)
0	1300.0	-18.18	-18.18	70.233224	48.268143	48.268143	31.27	31.27	10.69
1	400.0	0.00	0.00	22.304997	22.304997	22.304997	0.00	0.00	0.00
2	600.0	0.00	0.00	40.077774	29.362311	29.362311	26.74	26.74	13.33
3	600.0	0.00	0.00	36.063522	36.063522	36.063522	0.00	0.00	0.00
4	500.0	0.00	0.00	20.605414	13.661406	13.661406	33.70	33.70	12.81
...
95	300.0	0.00	0.00	14.875394	14.875394	14.875394	0.00	0.00	0.00
96	400.0	0.00	0.00	25.963146	25.963146	25.963146	0.00	0.00	0.00
97	300.0	0.00	0.00	20.168379	18.112694	18.112694	10.19	10.19	5.10
98	700.0	0.00	0.00	34.590551	31.029521	31.029521	10.29	10.29	4.36
99	200.0	0.00	0.00	11.443855	11.443855	11.443855	0.00	0.00	0.00

FIGURE 4-9 Un aperçu des résultats de compilation de 100 instances de *GPP* et des coûts des méthodes *ECM-GPP* et *EE-GPP* versus *SP-GPP*

Par ailleurs, les tableaux 4-1, 4-2 et 4-3 présente les pourcentages moyens de gain en énergie à l'utilisation de la méthode *ECM-GPP* vs *SP-GPP*, contre les pourcentages moyens des pertes en distance de l'application de la méthode *ECM-GPP* vs *SP-GPP* et les pourcentages moyens de gain en argent à l'utilisation de la méthode *EE-GPP* vs *SP-GPP* pour chacun des *layout* 1, 2, 3, respectivement. De plus, étant donné que le vecteur de 100 instances de simulations contient des zéros (lorsqu'il n'existe pas de meilleures alternatives de chemins, le coût du chemin *EE-GPP* et celui de *SP-GPP* sont les mêmes, alors le gain est nul) la moyenne de ces vecteurs en éliminant les zéros est également calculée (elle est affichée dans le tableau sous le nom "instances non nulles" (la moyenne des valeurs de gain non nul)). Cette valeur permet d'avoir une idée sur la moyenne des gains potentiels.

La deuxième partie de ces tableaux (séparée par le texte en vert) à son tour expose la comparaison avec l'approche proposée *EE-GPP*.

Les résultats de simulation ont prouvé que le chemin le plus court n'est pas toujours le plus efficace sur le plan énergétique. Ceci est expliqué par le fait qu'à faible vitesse, l'énergie due à la résistance au roulement est beaucoup plus importante que l'énergie aérodynamique.

<i>ECM-GPP</i>			
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	14.32	-3.38	5.04
Instances non nulles	18.36	-25.97	6.46
<i>EE-GPP</i>			
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	13.77	0.0	5.04
Instances non nulles	18.36	0.0	6.46

TABLEAU 4-1 Résultats relatifs au layout 1

<i>ECM-GPP</i>			
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	21.36	-4.27	7.7
Instances non nulles	22.48	-17.07	8.11
<i>EE-GPP</i>			
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	20.58	- 0.28	9.47
Instances non nulles	22.62	-13.89	10.41

TABLEAU 4-2 Résultats relatifs au layout 2

<i>ECM-GPP</i>			
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	22.78	-3.92	8.59
Instances non nulles	23.73	-17.84	8.95
<i>EE-GPP</i>			
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	21.66	-0.11	10.08
Instances non nulles	23.8	-10.53	11.08

TABLEAU 4-3 Résultats relatifs au layout 3

Par conséquent, afin d’optimiser la navigation d’un *AMR*, le planificateur *GPP* doit prendre en compte les obstacles présents dans l’espace de déploiement du robot, les dimensions du graphe de la manufacture ainsi que l’information sur la surface de celle-ci. Par ailleurs, nous allons évaluer l’effet de ce facteur dans la section qui suit.

4.2 Évaluation de la méthode *EE-GPP* avec un graphe topologique avec considération de l’effet du trafic

Mis à part la résistance au roulement, nous allons considérer dans cette partie l’effet du trafic sur la consommation énergétique et sur la décision du routage. Ici, nous étudions un graphe topologique présenté dans la figure 3-8-*b* du chapitre précédent (graphe correspondant à une manufacture réelle).

4.2.1 Configuration de l’espace et représentation des facteurs de la résistance au roulement et du trafic

Un graphe topologique de manufacture de disposition orthogonale d’arêtes est considéré. En suivant le même principe que l’évaluation précédente, deux graphes de coefficients de pénalités relatifs au volume du trafic sont définis : (i) graphe avec un trafic sur quelques arêtes relativement élevé (heure de pointe) défini aléatoirement dans un intervalle [1, 3.12] (tableau

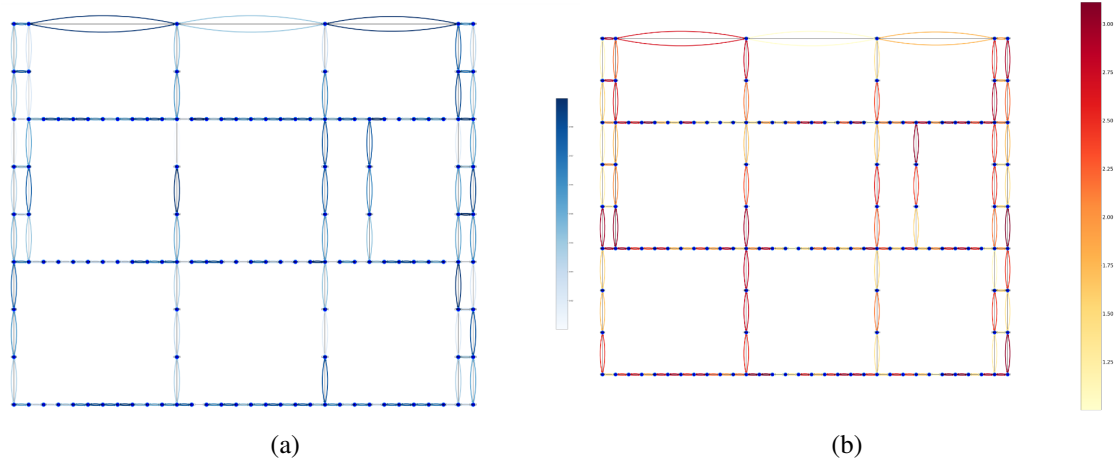


FIGURE 4-10 Informations sur l'environnement avec la structure topologique (a) graphe topologique avec l'information sur la friction (la dégradation en bleue reflète la grandeur du coefficient de résistance au roulement) (b) graphe topologique avec l'information du trafic (la dégradation entre le jaune et le rouge reflète la grandeur de la pénalité attribuée à chaque arc du graphe à cause du nombre moyen d'obstacles rencontrés (jaune :faible trafic, rouge :fort trafic)

3-1) (ii) graphe avec un trafic faible (heure moins achalandée) défini aléatoirement dans un intervalle $[1, 1.68]$ (tableau 3-1). La figure 4-10 montre le graphe représentant l'information de la résistance au roulement des différentes arêtes, ainsi que le graphe "trafic". La longueur de chaque arête est équivalente à la distance euclidienne des nœuds connectés. Ces coûts de distances forment le graphe "distance" associé à la configuration de la manufacture. Toutes ces informations sont utilisées par l'approche *EE-GPP* pour le calcul du chemin écoénergétique de l'*AMR*

4.2.2 Résultats de simulation

Une comparaison des méthodes *SP-GPP*, *ECM-GPP* et *EE-GPP* est effectuée à travers l'exécution de 100 scénarios de *GPP* avec des nœuds de départ et nœuds de destinations sélectionnés au hasard depuis différentes positions (nœud) de la manufacture, pour chaque étude de cas "*high traffic*" et "*low traffic*". De la même façon que la section 4.1, les résultats sont exprimés dans les deux figures 4-11 et 4-12 ainsi que les deux tableaux 4-4 et 4-5, pour les deux scénarios "*high traffic*" et "*low traffic*", respectivement.

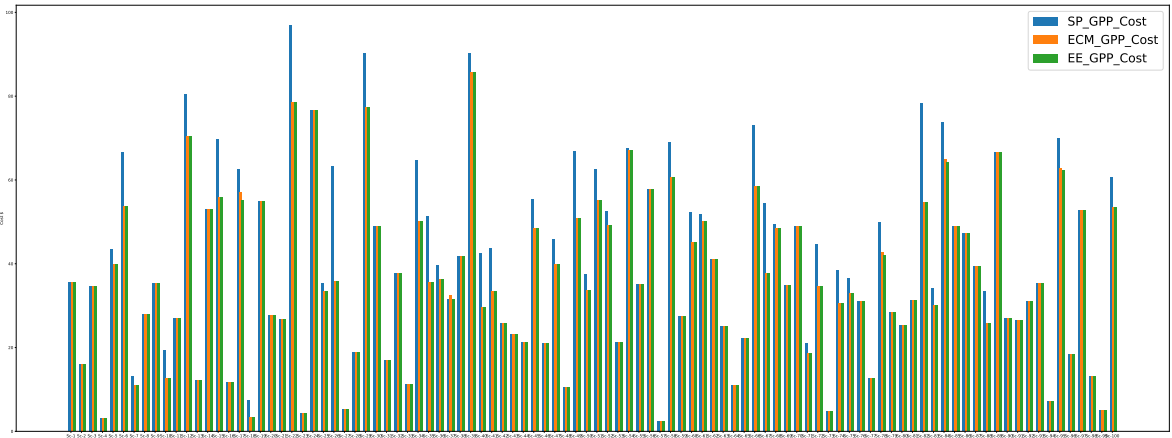
	<i>ECM-GPP</i>		
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	8,18	-4,15	7,11
Instances non nulles	19,02	-16,6	14,53
	<i>EE-GPP</i>		
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	8,15	-2,89	7,2
Instances non nulles	19,41	-14,44	17,13

TABLEAU 4-4 Résultats relatifs au graphe topologique avec information du trafic une heure de pointe.

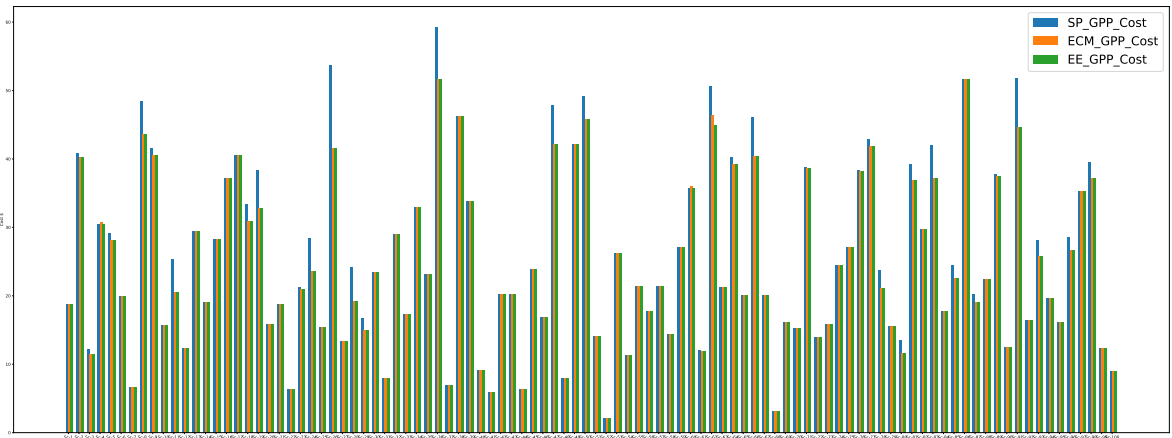
	<i>ECM-GPP</i>		
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	3,92	-3,63	2,83
Instances non nulles	10,6	-21,63	7,65
	<i>EE-GPP</i>		
	Moyenne de gain en énergie vs SP (%)	Moyenne de perte en distance vs SP (%)	Moyenne de gain en argent vs SP (%)
Toutes les instances	3,89	-3,14	2,88
Instances non nulles	11,11	-20,92	8,22

TABLEAU 4-5 Résultats relatifs au graphe topologique avec information du trafic faible.

[]

FIGURE 4-11 Résultats de simulation de *EE-GPP* (avec considération d'un trafic élevé)

[]

FIGURE 4-12 Résultats de simulation de *EE-GPP* (avec considération d'un faible trafic)

4.2.3 Analyse et discussion

On remarque que le coût de chemin *EE-GPP* est toujours inférieur ou égal au coût de prendre le chemin *SP*. D'autre part, on note que le coût des chemins *EE-GPP* est souvent égal au coût des chemins *EE-ECM* et que le nombre d'instances sans gain apporté est relativement élevé (68 instance sur 100). En effet, la forme du graphe considérée contient de longs couloirs et peu d'intersections. Par conséquent, le nombre de possibilités de chemins et la probabilité d'avoir des chemins concurrents baissent. Par contre, dans le cas d'existence d'un chemin alternatif plus économe énergétiquement, le gain donné par ce dernier est généralement élevé.

Par ailleurs, les résultats de simulations appliqués sur le graphe "*low traffic*" donnent des gains en énergie et en argent moins important que le premier cas de figure. Ceci est expliqué par le fait que les coefficients de pénalités sont faibles. Cependant, le facteur de résistance au roulement revient à être le plus important conducteur dans la décision de planification de chemin écoénergétique, dans ce cas.

En outre, si on compare les résultats de cette section versus la section précédente 4.1 (topologique versus *grid-graph*), le pourcentage de gain donné dans la section précédente (la configuration *grid-graph*) avec le facteur de la résistance au roulement est légèrement plus élevé que celui donné dans la configuration topologique considérée dans cette section (4-10). Cette observation est expliquée par le nombre d'obstacles plus élevés et leurs dispersions (plus de nombre d'intersections) ainsi que la taille et la nature du graphe *grid-graph* (plus de nombre d'arêtes) qui font en sorte que le nombre de possibilités de chemins augmente et que la probabilité d'avoir de meilleurs chemins écoénergétique accroît.

4.3 Évaluation de la deuxième méthode *EE-OPR*, avec variation de la masse

Dans cette partie, la problématique est différente, comme expliqué précédemment. Étant donné qu'on traite le prélèvement des commandes d'une liste donnée depuis leurs emplacements de stockage, la masse varie dans une tournée de collecte d'un *Ps-AMR*. Nous soulignons, alors, dans cette section l'effet de la masse et le séquençage de ramassage des articles sur la consommation énergétique et sur le coût en énergie et en argent, à travers des simulations de l'utilisation de l'approche *EE-OPR* et l'évaluation de sa performance.

4.3.1 Configuration de l'entrepôt

Afin d'évaluer la méthode proposée, nous considérons un entrepôt unibloc comportant g allées, avec l positions de prélèvement horizontales, comme le montre la figure 4-13. Il s'agit de la structure la plus courante étudiée dans la littérature. Cependant, nous soulignons que *EE-OPR* peut être adopté à toute autre configuration d'entrepôt.

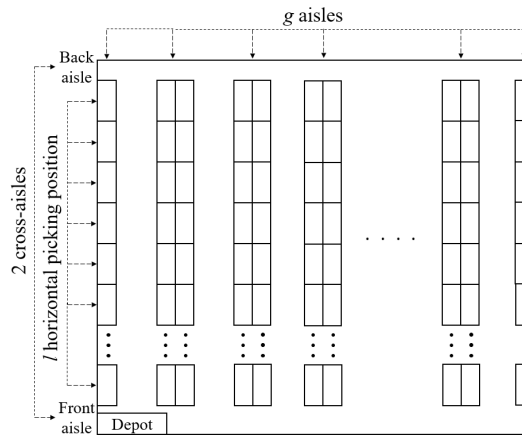


FIGURE 4-13 La configuration de l'entrepôt considérée pour les études de cas

4.3.2 Résultats de simulation

L'efficacité de *EE-OPR* est évaluée et comparée à une méthode que nous appelons approche de minimisation du temps de parcours (*SP-OPR*). Le *SP-OPR* proposé par Letchford et al. (2013)[179] est une approche qui vise à minimiser uniquement le temps de parcours des tournées de sélection. Elle est basée sur une formulation compacte de flux à une seule marchandise. *SP-OPR* est également utilisé par Pansart et al.[177] pour résoudre un programme linéaire mixte en nombres entiers. Ce dernier n'utilise que le critère de temps t_{ij} dans sa fonction objectif. ($z^* = \min \sum_{(i,j) \in A} (t_{ij} x_{ij})$). Cette technique est utilisée pour générer la tournée de ramassage la plus courte pour chaque instance. De plus, afin de la comparer à *EE-OPR*, la somme du coût de chaque arc de la séquence de tournées résultant des deux méthodes est calculée à l'aide de la fonction c_{ij} 3.15, ce qui donne le coût en dollars de l'exécution de chaque chemin (puisque *SP-OPR* n'utilise pas c_{ij} comme fonction de coût, le coût en dollars des tournées résultant de *SP-OPR* est calculé après avoir effectué le *SP-OPR* en utilisant c_{ij}).

Autrement dit, l'exécution de *SP-OPR* donne une tournée donnée dont le coût en argent est calculé en sommant le coût de temps écoulé et le coût d'énergie dépensée, comme suit : Le temps de voyage requis pour la traverser (connue) est multipliée par le coût d'une unité de temps et l'énergie requise pour la parcourir est calculé (en sommant les coûts de transition de chaque arc donné par le chemin *SP-OPR* à l'aide de l'équation 3.13) et multiplié par le coût en argent par kilo-joule. Il est important de mentionner que *SP-OPR* ne peut pas intégrer la consommation d'énergie puisque e_{ij} est une fonction de M_i^L qui n'est pas définie par *SP-OPR* selon sa formulation.

D'autre part, le *SP-OPR* représente tous les algorithmes exacts qui visent à optimiser le temps ou la distance sans tenir compte de l'aspect énergétique par une formulation de programmation dynamique qui n'est pas adaptée à l'ajout de contraintes supplémentaires [177]. Plus précisément, pour évaluer la performance de notre approche, nous considérons des scénarios de demande aléatoire avec une distribution uniforme de la demande dans tout l'entrepôt avec des politiques de stockage aléatoires. Cette évaluation est réalisée à travers la variation de :

- la taille de la liste de prélèvement,
- les emplacements de prélèvement nécessaires dans l'entrepôt (liste de prélèvement),
- les masses des articles à prélever dans les différentes positions et,
- forme de l'entrepôt.

Comme le montre la figure 4-13, le nombre d'allées verticales et transversales est variable. Par conséquence, les simulations sont générées pour quatre configurations différentes : Configuration 1= (40 × 50), Configuration 2= (25 × 80), Configuration 3= (20 × 100), et Configuration 4= (10 × 200). Les distances entre les allées et les emplacements de prélèvement sont respectivement de 4 mètres et de 1 mètre pour tous les agencements. Le choix de la forme de l'entrepôt s'inspire de Wang et al. [183]. Chaque configuration est explorée dans le cadre de cinq scénarios avec différentes tailles de listes de prélèvement, à savoir 8, 10, 12, 14 et 16, respectivement. Pour chaque scénario, 100 tournées de prélèvement sont simulées avec des emplacements et des masses arbitraires afin de fournir une analyse statistique fiable. Pour chaque tournée, une liste de cas est choisie au hasard en utilisant une distribution uniforme.

En pratique, les articles peuvent représenter des lots qui regroupent plusieurs commandes dans une liste de prélèvement pour être séparés dans la station d'emballage [183]. Les articles d'une liste de prélèvement ont des masses différentes qui sont également générées de manière aléatoire entre 10 kg et M_{\max} kg (capacité maximale du véhicule). La somme des masses des articles de la liste de sélection ne doit pas dépasser la capacité maximale du véhicule, comme suit :

$$m_{i,j}^s \sim \text{Unif}(10, M_{\max}) \quad (4.1)$$

avec s, i, j représentent le scénario, l'instance (tournée de ramassage) et le lieu de ramassage, respectivement.

La simulation réalisée dans le présent travail est effectuée pour un AMR ayant une vitesse maximale de 1.2 m/s, un poids de 1600 kg, et une charge maximale supportée de 1200 kg.

La figure 4-15 présente les résultats de l'étude comparative de notre approche et de la méthode *SP-OPR* basée sur la moyenne et la variance du coût de chaque scénario. Comme le montre la figure 4-15, *EE-OPR* surpasse *SP-OPR* dans les vingt paramètres (quatre configurations ayant chacune cinq scénarios différents). En particulier, l'efficacité de *EE-OPR* augmente en se rapprochant du cinquième scénario pour toutes les configurations par rapport à *SP-OPR*. En d'autres termes, nous constatons une moindre augmentation du coût lorsque la taille de la liste de sélection augmente. La figure 4-15 montre une augmentation de la différence entre les valeurs moyennes du coût de la tournée dans les scénarios de chaque disposition. Ce résultat est dû à la capacité du *EE-OPR* à minimiser l'énergie, qui a tendance à augmenter pour les *pick-lists* plus larges avec une plus grande possibilité d'accumulation de masse. De plus, nous observons une variance plus faible du coût de routage dans *EE-OPR* par rapport à *SP-OPR*, en particulier pour les scénarios 4 et 5 (avec un nombre d'articles plus élevé). Ainsi, *EE-OPR* est moins sensible aux variations de l'emplacement et de la masse de la liste de prélèvement. Les résultats prouvent ainsi que la minimisation simultanée de l'énergie et du temps peut réduire efficacement le coût d'une tournée de prélèvement.

Nous remarquons également que dans certaines situations, *SP-OPR* et notre approche aboutissent exactement au même résultat. Ceci est logique et s'explique par le fait que nous

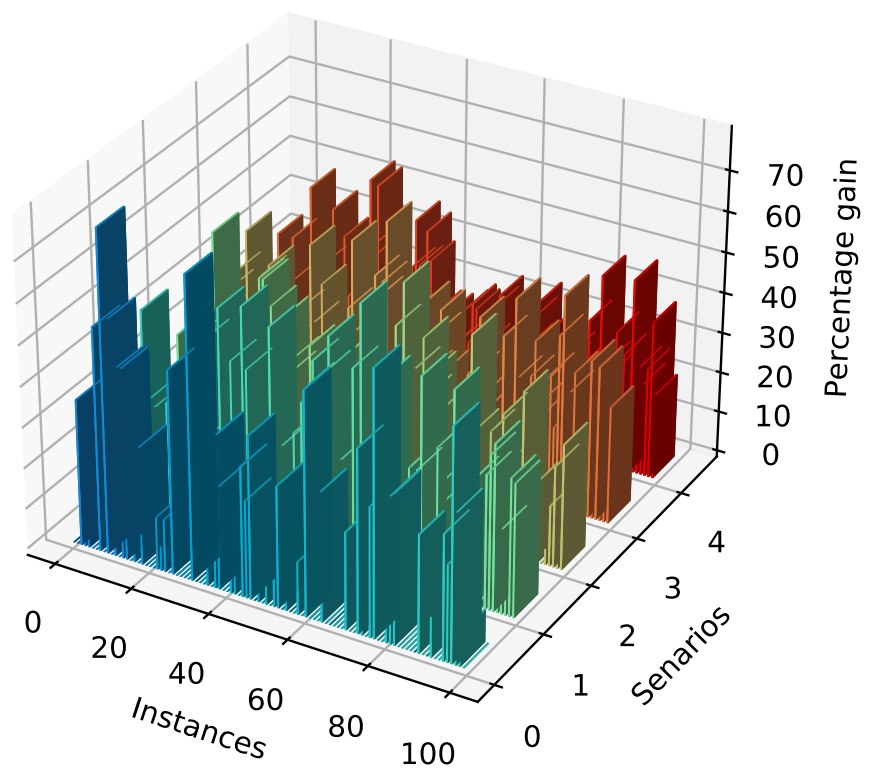


FIGURE 4-14 Pourcentage de gain qu'EE-OPR donne versus les *SP-OPRs*

nous concentrons sur l'effet de la masse sur la consommation d'énergie et la prise de décision de routage. Ainsi, si la variation de masse des articles est faible ou nulle, minimiser l'énergie consommée revient à minimiser la distance parcourue. En outre, si la situation de passer deux fois sur une position n'est pas présente et que la direction de la tournée de traversée est heureusement la même, l'avantage donné par notre approche n'est plus valable.

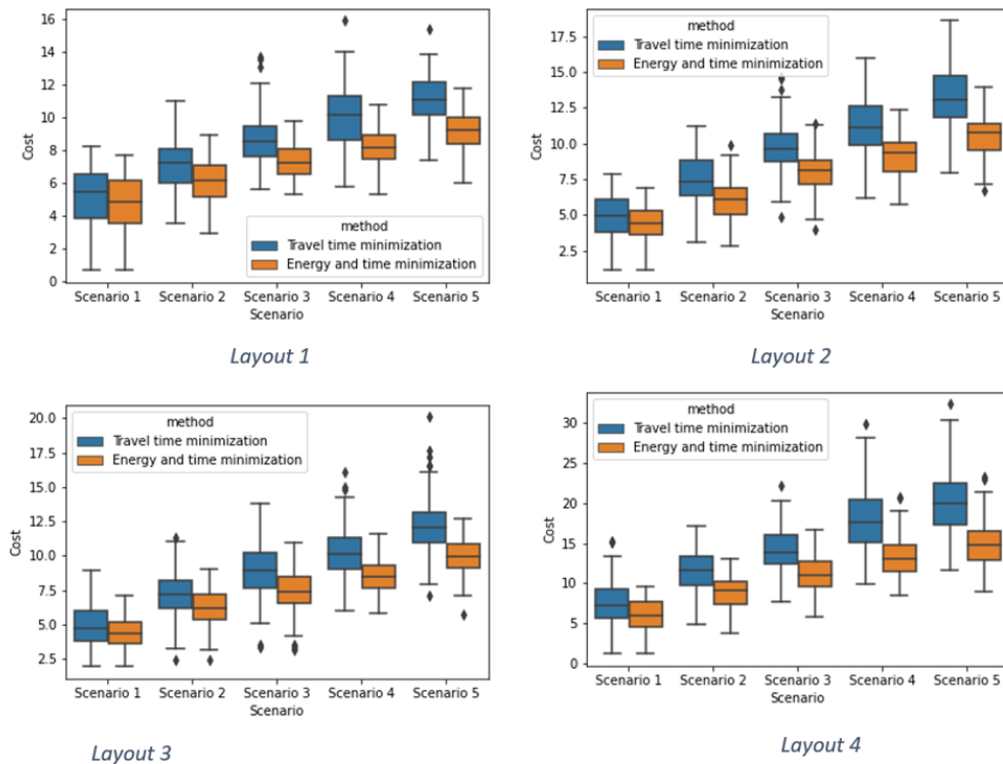


FIGURE 4-15 Comparaison des tours résultants après application des approches SP-OPR (a) et EE-OPR (b) appliquées sur différentes configurations.

Pour mieux illustrer comment notre approche permet d'apporter un gain en énergie et argent, comparativement aux approches *SP-OPRs*, un exemple de problème de préparation de commande d'un graphe simple est utilisé (Figure 4-16). Le graphe est composé de 20 sommets dont 4 sont obligatoires. Ces sommets requis représentent les sommets de la *pick-list* et ils sont représentés par des cercles rouges dans le graphe. Les sommets requis sont 18, 19, 15, et 16 qui ont des masses de 100, 10, 70, et 20, respectivement.

Afin de résoudre le problème de préparation de commandes OPR, *SP-OPR* (Figure 4-16,a) and *EE-OPR* (Figure 4-16,b) sont appliqués. Les tournées obtenues par chaque méthode sont

représentées sur la figure par des flèches rouges. Notez que les segments constituant le chemin obtenu par les deux approches sont exactement les mêmes. En d'autres termes, les chemins résultants des deux approches sont similaires en termes de distance parcourue et aussi de la forme des chemins rouges. Cependant, l'application des approches diffère dans la direction de la trajectoire ainsi que dans l'ordre de prélèvement des articles dans les positions requises.

Soit v_i la position du sommet v et soit $v_i \sim > v_j$ le plus court chemin entre deux sommets v_i et v_j et nous supposons qu'un AMR part d'une position v_0 , prend les charges aux positions requises (cercles rouges) puis retourne à v_0 (le caractère $\sim >$ signifie le chemin le plus court de v_i à v_j). Ainsi, l'application de *SP-OPR* donne le résultat suivant :

v_0 (charge transportée = 0 kg)

$\sim > v_{15}$ (charge transportée = 0 + 70 kg)

$\sim > v_{18}$ (charge transportée = 70 + 100 kg) (ramasser l'article v_{18} lors du premier passage)

$\sim > v_{19}$ (charge transportée = 70 + 100 + 10 kg)

$\sim > v_{18}$ (charge transportée = 70 + 100 + 10 kg) (passer à travers v_{18})

$\sim > v_{16}$ (charge transportée = 70 + 100 + 10 + 20 kg)

$\sim > v_0$ (charge transportée kg)

Cependant, l'application de *EE-OPR* conduit au résultat suivant :

v_0 (charge transportée = 0 kg)

$\sim > v_{16}$ (charge transportée = 0 + 20 kg)

$\sim > v_{18}$ (charge transportée = 0 + 20 kg) (passer à travers v_{18} mais ne pas ramasser l'article dedans)

$\sim > v_{19}$ (charge transportée = 20 + 10 kg)

$\sim > v_{18}$ (charge transportée = 20 + 10 + 100 kg) (ramasser l'article v_{18} lors du second passage)

$\sim > v_{15}$ (charge transportée = 20 + 10 + 100 + 70 kg)

$\sim > v_0$ (charge transportée = 200 kg)

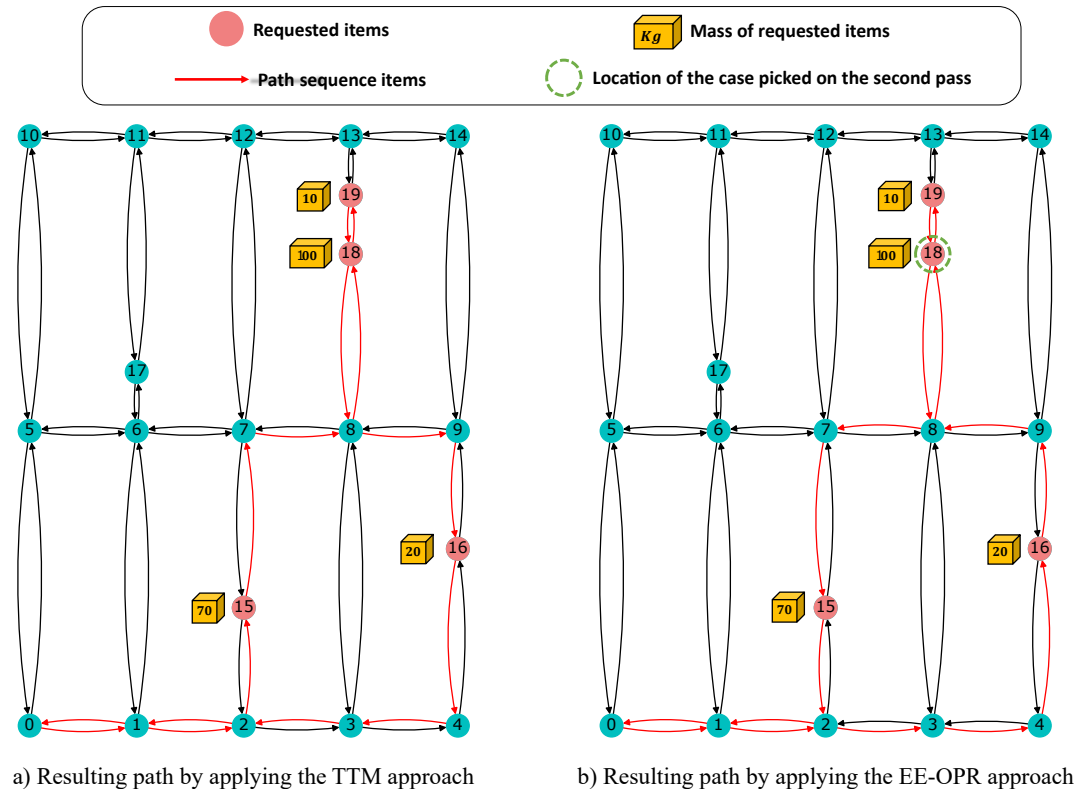


FIGURE 4-16 Exemple de comparaison des tournées de ramassage de commandes résultants de l'approche SP-OPR (à gauche) et de l'approche EE-OPR (à droite)

Comme le montrent les résultats, contrairement à l'approche *EE-OPR*, le chemin résultant du *SP-OPR* traîne plus longtemps (une plus longue distance) les masses lourdes en raison de la direction de la tournée. De plus, *EE-OPR* permet au véhicule de passer par une position requise sans l'obligation de ramasser le ou les articles lors du premier passage. Par exemple, contrairement à *SP-OPR*, *EE-OPR* passe par le sommet 18 sans ramasser la charge (afin de ne pas la traîner inutilement). Ensuite, il se déplace vers le sommet 19 pour collecter la charge correspondante, puis revient au sommet 18 pour collecter sa charge. En d'autres termes, les objets situés au sommet 18 ne sont pas ramassés lors du premier passage, mais lors du second. On observe également, dans cet exemple, que le chemin résultant de *EE-OPR* est différent de celui donné par *SP-OPR* (direction et moment du ramassage des articles), mais avec la même longueur/durée (chemin court). De plus, le chemin résultant de *EE-OPR* a la même demande énergétique donnée par *ECM* (avec la plus faible consommation d'énergie). Il est important de noter que cette situation (*EE-OPR* sélectionne un chemin qui est à la fois le plus efficace sur le plan énergétique et le plus court) est très courante, surtout lorsque la taille de la *pick-list* est

petite.

Par conséquent, l'application de *EE-OPR* entraîne une réduction de 25% du coût total en dollars de la tournée de ramassage par rapport à *SP-OPR*. Or, ce gain en termes de coût peut être beaucoup plus important lorsque le plus court chemin entre les sommets v_{18} et v_{19} vaut des kilomètres, par exemple dans les entrepôts de grande surface.

Ainsi, le *EE-OPR* minimise la perte d'énergie associée au déplacement de charges lourdes. En effet, si possible, il retarde la récupération de certains articles afin de réduire la distance à parcourir avec ces charges et réduire ainsi le cumul de la masse avec les commandes suivants à ramasser, tout en respectant la contrainte de temps de parcours.

4.3.3 Discussion

L'application de notre approche *EE-OPR* a permis une économie moyenne de 18% sur le coût total de la tournée de préparation de commandes. Ce pourcentage de réduction des coûts dépend de nombreux facteurs, tels que la politique d'affectation du stockage, le type et l'aménagement de l'entrepôt, le type et le poids des produits, la taille de l'AMR, etc. D'ailleurs, selon Shetty et al. [206], la stratégie de routage, le stockage, le zonage, la mise en lots (*Batching*) et le mode de validation des commandes sont des composantes du niveau décisionnel (*policy level/ operational policies*) dans les opérations et la gestion d'entrepôt. Ce niveau dépend fortement du niveau stratégique (caractéristiques du système), qui inclut le cycle de commande, le niveau de mécanisation, la dimension de l'entrepôt et la disponibilité de l'information (voir la figure 4-17 qui exprime la complexité des systèmes de préparation des commandes). Par conséquent, le taux de rentabilité apporté par notre algorithme de routage dépend à la fois du niveau stratégique et des autres décisions opérationnelles. D'autre part, l'efficacité de calcul de notre algorithme dépend des emplacements à visiter (c'est-à-dire si ces emplacements sont proches, éloignés ou dispersés), qui affecte la vitesse de recherche de l'état cible.

Par ailleurs, *EE-OPR* donne plusieurs avantages tels que (i) la flexibilité du graphe à traiter (ii) l'économie en énergie et en argent dépensés grâce à la considération des deux aspects "*Direction*" et "*SecondPass*", (iii) la considération du compromis entre le temps et l'énergie

□

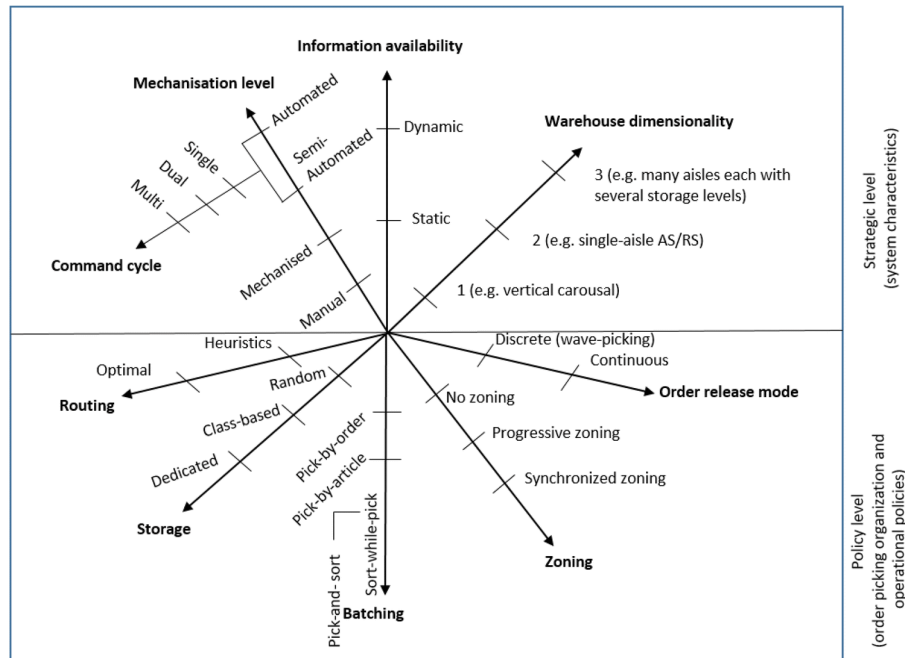


FIGURE 4-17 Les différents éléments qui affectent l'efficacité de la stratégie de routages [186]

consommés. Néanmoins, cette approche possède des limites qui peuvent être améliorées dans des travaux futurs. Nous allons discuter tous ces points dans les sous-sections qui suivent.

4.3.3.1 Flexibilité du graphe d'entrepôt et la différence entre la méthode *EE-OPR* et les méthodes basées sur *RRs*

Contrairement à la majorité des travaux qui traitent le problème de OPR en se basant sur le principe de la méthode de Ratliff et Rosenthal *RR* (l'approche la plus connue pour la résolution exacte du problème OPR [172]), *EE-OPR* traite le graphe d'entrepôt indépendamment de sa forme (il peut aborder même les entrepôts non conventionnels). En effet, comme précédemment décrits dans la section 2, ces méthodes appliquent la programmation dynamique pour trouver la tournée de préparation de commandes la plus courte et ils se focalisent sur le fait que les entrepôts réguliers ont une configuration rectangulaires à allées parallèles pour définir les sous-problèmes de leurs formulations de la DP (des possibilités spécifiques de mouvements horizontaux et verticaux (voir figure 2-16)) [172], [173], [177]. Les deux approches *RR-based* et *EE-OPR* utilisent le principe de la programmation dynamique. Par contre, les deux

procédures sont totalement différentes. Discutons alors cette différence et l'avantage que *EE-OPR* donne versus les méthodes *RR-based*.

L'approche *EE-OPR* prend, simplement, en entrée un graphe dirigé pondéré avec les distances entre les positions des sommets et une *pick-list* bien déterminées (qui permet de définir les sommets Steiner et les sommets '*visiting vertices*' de ce graphe), quelles que soient leurs positions spatiales. Justement dans la vraie vie, il existe des entrepôts conventionnels, multi-blocs et non conventionnels tel que les exemples illustrés dans la figure 4-18. Par conséquent, l'approche *EE-OPR* donne une flexibilité élevée en termes de l'entrée du graphe spatial (n'importe quel nombre d'allées verticales et transversales) et en termes de la position d'arrivée qui peut être différente de la position initiale. En outre, la méthode proposée utilise une manière

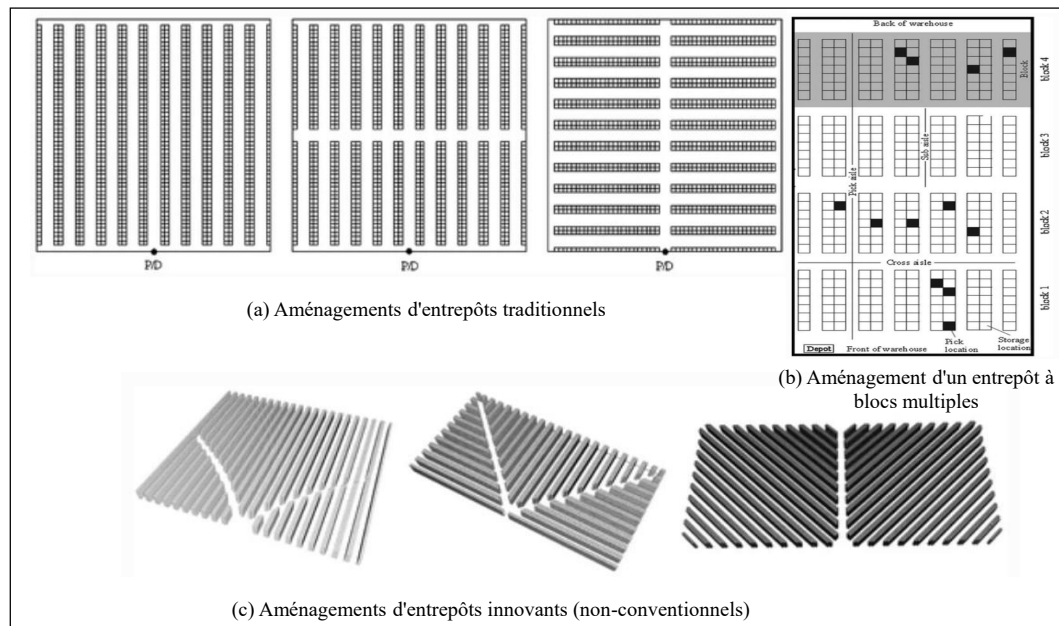


FIGURE 4-18 Les différents aménagements (configurations) possibles d'un entrepôt ou d'un DC (figure adaptée de [207](a) [208],(b)[171],(c)[209])

différente de celles explorées par *RR* pour construire les sommets et les transitions du graphe dynamique. Cette différence réside principalement dans l'incorporation de l'information de masse de chaque sommet d'état lors de la prise de décision de routage. Plus précisément,

les sommets sont construits sur la base des informations suivantes : (a) la position de l'AMR dans l'entrepôt (sommet de l'emplacement) et (b) la masse des charges transportées à chaque étape avec les détails des articles déjà prélevés. En outre, les arcs de transition sont définis en fonction de la possibilité de ramasser l'article d'un sommet voisin et sont ajoutés sur la base du graphe original de l'entrepôt. Plus importants encore, ils sont pondérés par le coût en temps et en énergie lié à la distance parcourue et à la masse transportée. Un tel processus dynamique conduit à la création progressive d'un graphe dirigé acyclique, pour lequel le problème de planification de chemin est résolu avec l'objectif de minimiser le temps et l'énergie tout en atteignant le sommet "état" cible (position finale avec tous les articles demandés) en partant de la position initiale (la position sans aucun article). Par conséquent, le cadre développé est capable non seulement de prendre en compte la distance de la tournée de prélèvement, mais aussi de traiter sa demande en énergie. En fait, lier l'utilisation de l'énergie de l'AMR à la fois à la distance et à la masse peut aboutir à un chemin écoénergétique différent de celui offert par la seule première dépendance (la distance). En effet, les études de recherche n'ont pas pris en considération l'aspect énergétique de l'OPR. Elles sont limitées par les principes de l'approche RR, qui ne considère pas la masse et ne permet pas son intégration, et donc, contrairement au schéma proposé, leur solution est uniquement optimale en termes de distance.

De plus, notre approche peut être utilisée pour différents aménagements d'entrepôts avec des points de départ et d'arrivée arbitraires d'une tournée. Il suffit de définir un graphe contenant les segments de passage possibles (arcs et sommets), le point de départ (v_s), le point d'arrivée (v_t), et les points requis (*pick-list*). Alors l'état initial sera (v_s, a_s) et l'état final sera (v_t, a_t). De sorte que a_s et a_t sont respectivement '*empty arrangement*' et '*full arrangement*'.

4.3.3.2 Économie d'énergie et d'argent

Grâce à la considération des deux aspects *Direction* et *SecondPass* dans *EE-OPR* des chemins plus intéressants énergiquement sont découverts. Ci-dessous, on présente deux exemples plus concrets appliqués sur la configuration de l'article [177] et l'article de base RR [172], respectivement. La première, présente une configuration avec deux blocs et la seconde est uniblocs. On constate que seulement avec la modification du sens de direction de trajectoire les chemins *EE-OPR* pour les deux situations apportent un gain énergétique versus le SP-OPR

sans aucune perte du temps. Cependant, c'est une constatation réalisée après l'application des algorithmes et ne présente pas le fond de l'algorithme, car la différence de directions n'est pas toujours présente et peut l'être dans un ou des sous-sous-graphe(s) seulement. Par exemple, dans la figure 4-19(a), un changement de masse d'un ou des trois article(s) peut engendrer une modification dans la décision de routage. Effectivement, dans la vie réelle, les SKUs rangés dans une position de stockage particulière peuvent changer de place ou être remplacés par de nouveaux produits. D'où l'utilité et l'adaptabilité de l'approche *EE-OPR*.

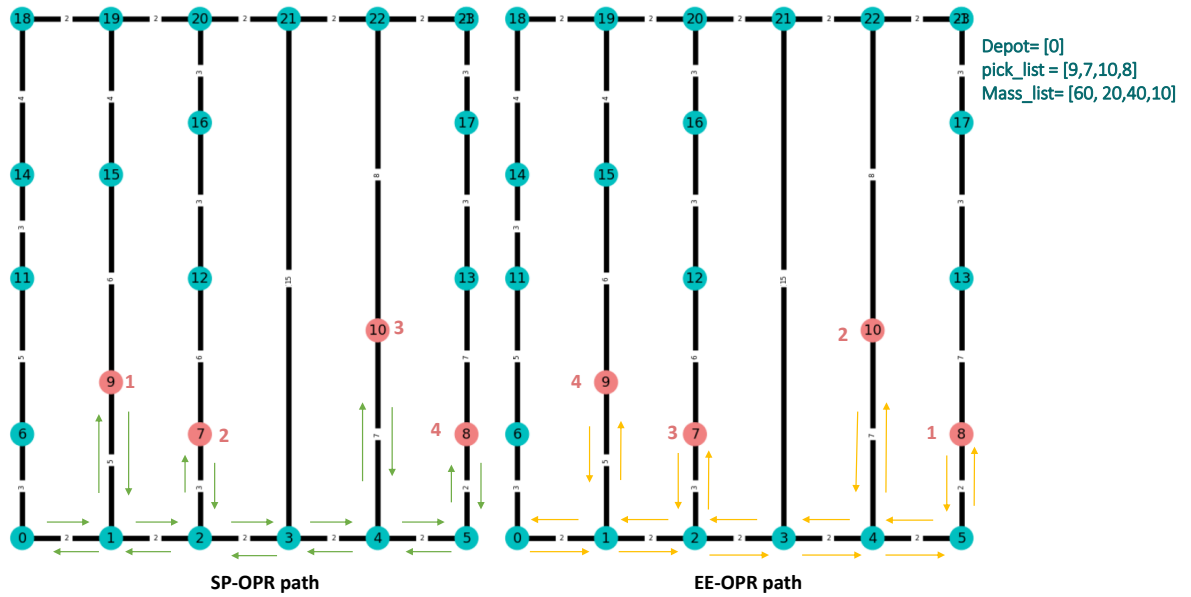
4.3.3.3 Importance de la considération du compromis temps/énergie

Afin de discuter le compromis du coût entre le temps et l'énergie, avec les mêmes préfixes de la section précédemment, mais appliqué au problème de l'*OPR*, trois approches différentes sont définies : la méthode de minimisation du temps de parcours (*SP-OPR*) qui ne prend en compte que la distance/temps, la méthode de minimisation de la consommation d'énergie (*ECM-OPR*) qui ne prend en compte que l'énergie, et notre méthode proposée *EE-OPR* qui prend en compte le temps et l'énergie simultanément. Nous utilisons pour ces trois méthodes les notations suivantes :

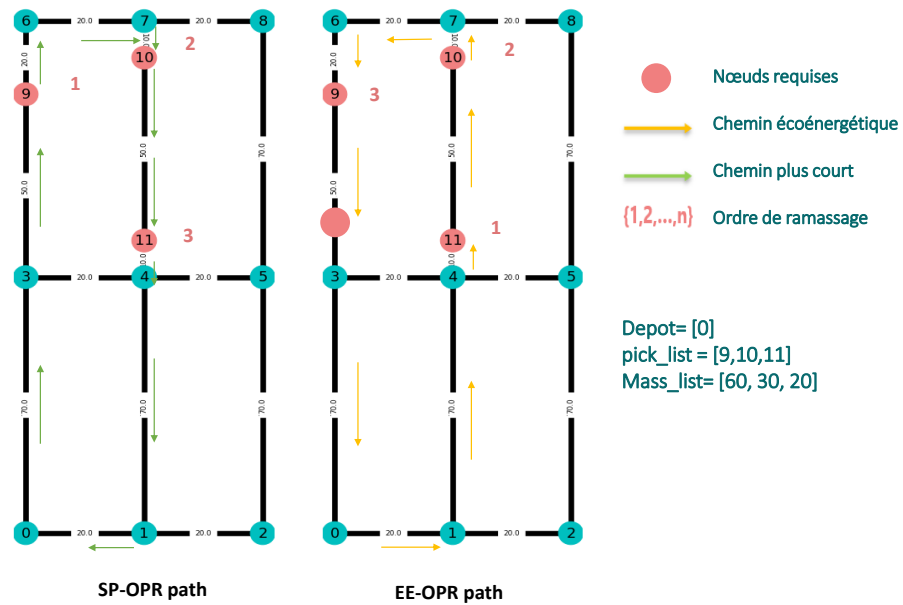
- *SP-OPR* : *SP-OPR-path*, $T - cost_{SP-OPR}$, $E - cost_{SP-OPR}$, $C - cost_{SP-OPR}$, désignent respectivement le chemin résultant, le temps utilisé, l'énergie consommée et le coût en dollars (temps et énergie), correspondants à l'utilisation de l'approche *SP-OPR*.
- *ECM-OPR* : *ECM-OPR-chemin*, $T - cost_{ECM-OPR}$, $E - cost_{ECM-OPR}$, $C - cost_{ECM-OPR}$, représentent respectivement le chemin résultant, le temps dépensé, l'énergie consommée, et le coût en dollars (temps et énergie) à l'utilisation de la méthode *ECM-OPR* .
- *EE-OPR* : *EE-OPR-chemin*, $T - cost_{EE-OPR}$, $E - cost_{EE-OPR}$, $C - cost_{EE-OPR}$, correspondent au chemin résultant, au temps passé, à l'énergie consommée, et au coût en dollars (temps et énergie), respectivement.

Tels que la section précédente, les résultats démontrent l'existence de trois différents groupes de scénarios, comme suit :

- Groupe 1 : $ECM-OPR-path \neq SP-OPR-path$, $E - cost_{ECM} < E - cost_{SP-OPR}$, $T - cost_{ECM} = T - cost_{SP-OPR}$. La figure 4-19 illustre deux exemples d'une telle situation.



(a)



(b)

FIGURE 4-19 Comparaison entre les tours de prélèvement des commandes fournis par les deux approches *SP-OPR* et *EE-OPR* (l'effet de l'aspect *Direction*) (a) Configuration d'entrepôt de *RR (unibloc)* (b) Configuration d'entrepôt à deux blocs

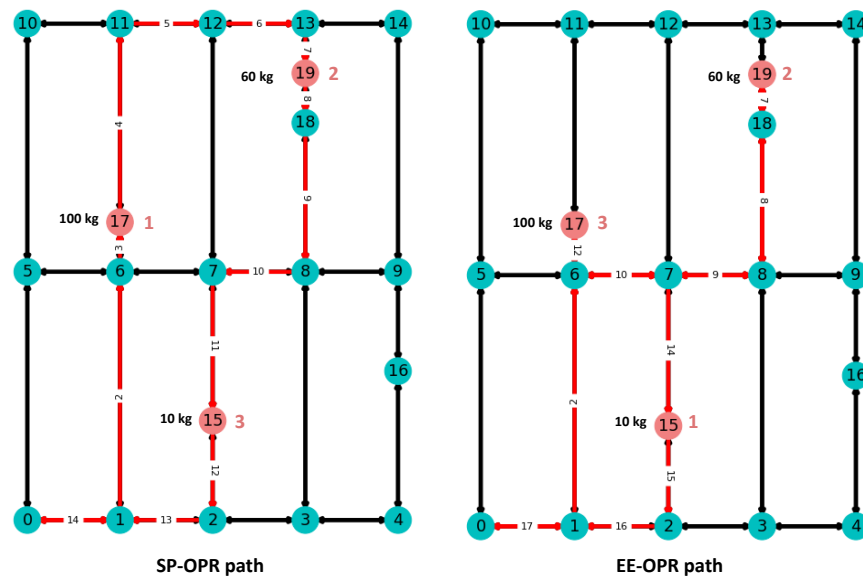


FIGURE 4-20 Un exemple concret de chemin plus économe en énergie, mais qui est plus long du chemin le plus court

Cette situation s'explique par la variation de la masse des éléments et la décision de les choisir ou non.

- Groupe 2 : $ECM-OPR-path = SP-OPR-path$, $E - cost_{ECM} = E - cost_{SP-OPR}$, $T - cost_{ECM} = T - cost_{SP-OPR}$. Ces situations s'expliquent soit par la faible (ou nulle) variation de la masse, soit par une situation où l' ECM ne trouve pas de meilleures alternatives que le $SP-OPR$.
- Groupe 3 : $ECM-OPR-path \neq SP-OPR-path$, $E - cost_{ECM-OPR} < E - cost_{SP-OPR}$, $T - cost_{ECM-OPR} > T - cost_{SP-OPR}$. La figure 4-21, ci-dessous, illustre un exemple d'une telle situation.

Par conséquent, dans le groupe 1, le coût d'exécution du ECM -path en dollars est évidemment inférieur à celui de $SP-OPR$ ($c_{cost_{ECM-OPR}} < c_{cost_{SP-OPR}}$) et le chemin $EE-OPR$ est égal au chemin $ECM-OPR$ -path ($c_{cost_{EE-OPR}} = c_{cost_{ECM-OPR}}$). Dans le groupe 2, $EE-OPR$ -chemin est égal aux chemins $ECM-OPR$ et $SP-OPR$ ($c_{cost_{EE-OPR}} = c_{cost_{ECM-OPR}} = c_{cost_{SP-OPR}}$). Néanmoins, dans le groupe 3, $EE-OPR$ -chemin peut être soit égal au $ECM-OPR$ -path, au chemin $SP-OPR$ -path

ou bien à un chemin optimal différent avec un meilleur compromis entre l'énergie et le temps consommés.

Par conséquent, notre approche *EE-OPR* consiste à trouver un compromis entre le temps et les coûts énergétiques dans une tournée de préparation de commandes. Cette méthode présente une fonction objectif qui additionne les coûts du temps de déplacement exprimé en secondes multiplié par son coût (coût en dollar par seconde) et la demande d'énergie calculée en joule multiplié par son coût (coût en dollar par joule). Par conséquent, le chemin sélectionné de l'*EE-OPR* dépend de plusieurs paramètres tels que l'emplacement des articles à prélever (la distance qui les sépare), les masses des articles et le coût en dollar du temps et de la consommation d'énergie. Nous notons ici que, afin de donner la priorité à un paramètre, nous lui attribuons un coût élevé.

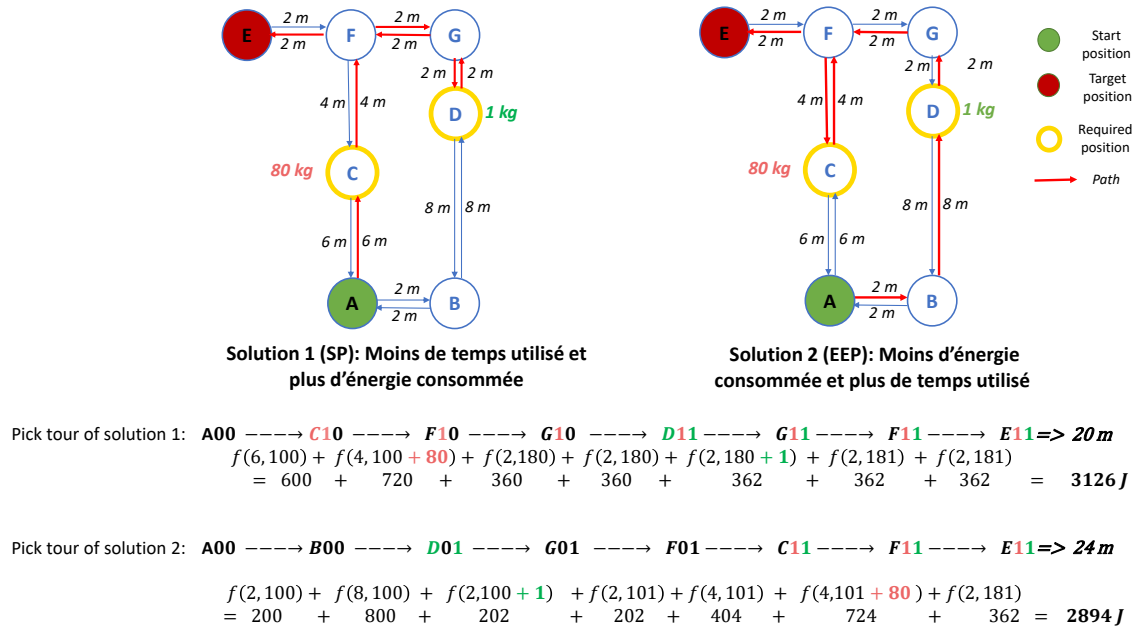


FIGURE 4-21 Un exemple concret de chemin plus économe en énergie, mais qui est plus long que le chemin le plus court (analyse).

Prenons l'exemple de la configuration présentée dans le chapitre introduction 1 à travers la figure 1-9. Ce scénario appartient au groupe 3. Le même exemple est reproduit dans la figure 4-21 avec des chiffres pour analyser les coûts résultants liés à la distance et à l'énergie de

SP-OPR et *ECM*, respectivement. Si on considère les valeurs de distance et de masse comme affichées dans la figure, l'*ECM* fournit un chemin plus long que le *SP-OPR*, mais consomme moins d'énergie. Ainsi, l'*ECM* évite de traîner 80 kg sur 8 mètres, contre un déplacement de 4 mètres de plus sans charge. En outre, si nous supposons que le coût du temps est de 1\$/h, le coût de l'énergie est de 0.012\$/wh; $c_{cost_{ECM}} > c_{cost_{SP-OPR}}$. Par conséquent, il est plus intéressant de choisir le chemin *ECM*. Cependant, si nous augmentons le coût du temps ou si nous considérons la même liste de prélèvement avec des masses différentes dans *D* (par exemple, 77 kg au lieu de 1 kg), il devient préférable de choisir le chemin *SP-OPR*. En d'autres termes, il ne vaut plus la peine que le robot choisisse un chemin plus long de 4 mètres, avec sa propre masse (100 kg), pour éviter de traîner 3 kg de plus sur 8 mètres.

La question qui se pose est de savoir quelle est la perte en temps si j'applique la méthode *ECM-OPR* en comparaison avec le temps optimum d'un tour de ramassage donné par la méthode *SP-OPR*. En outre, quel est le gain en énergie qui sera obtenu si on applique cette méthode. Pour cette raison nous présentons un schéma représentant ces aspects dans la figure 4-22. Ce diagramme en boîte, montre un exemple de simulation de 100 instances différentes avec la même taille de *pick-list* et le même *layout*, mais avec des positions et des masses différentes. La boîte à moustache à gauche représente le pourcentage de perte en temps lorsqu'on utilise l'approche *ECM-OPR* (en comparant avec le temps donné par *SP-OPR*). En contrepartie, la boîte à droite représente le pourcentage de gain en énergie donnée par cette méthode par rapport à la *SP-OPR*.

De même, le tableau 4-6 montre que le gain d'énergie et la perte de temps en pourcentage obtenus par l'*ECM-OPR* sont présentés dans les deux premières colonnes. Nous observons que 28% des tournées de prélèvement obtenues par l'*ECM-OPR* ont la même longueur que celles fournies par *SP-OPR*. De plus, la colonne 3 du tableau 4-6 présente le gain monétaire obtenu par *EE-OPR* (*trade-off*) par rapport à *SP-OPR*. En outre, les deux dernières colonnes du tableau présentent la différence en pourcentage entre le temps passé par *EE-OPR* et *SP-OPR* et la différence en pourcentage entre l'énergie consommée par *EE-OPR* et *ECM-OPR*. La moyenne de chaque colonne est présentée dans la dernière ligne. On peut noter que la méthode *EE-OPR* apporte un gain moyen de 13% par rapport au coût total des tournées de préparation de commandes.

Instances	Avec l'utilisation de la méthode <i>ECM-OPR</i>		Avec l'utilisation de la méthode <i>EE-OPR</i>		
	Le pourcentage de gain en énergie vs <i>SP-OPR</i> (%)	Le pourcentage de perte en temps vs <i>SP-OPR</i> (%)	Le pourcentage de gain en argent vs <i>SP-OPR</i> (%)	La différence entre le temps optimal donné par <i>SP-OPR</i> et le temps requis pour traverser <i>EE-OPR</i> - path (%)	La différence entre l'énergie optimal donné par <i>ECM-OPR</i> et l'énergie requise pour effectuer <i>EE-OPR</i> - path (%)
0	30.7	4.5	27.1	0.5	0.0
1	23.1	20.6	20.0	2.4	0.8
2	4.6	6.6	3.2	6.6	0.0
3	16.4	12.7	13.1	12.7	0.0
4	49.6	8.9	44.2	8.9	0.0
...	0.0
97	8.2	2.8	6.9	2.8	0.0
98	6.6	10.6	4.1	10.6	0.0
99	24.3	0.0	21.6	0.0	0.0
Moyenne	16.0	6.4	13.6	4.4	0.1

TABEAU 4-6 Les pourcentages de gain en énergies et de pertes en temps donnés par la méthode *ECM-OPR* (les colonnes colorées en vert) et par la méthode *EE-OPR* (les colonnes de couleur jaune) vs *SP-OPR*.

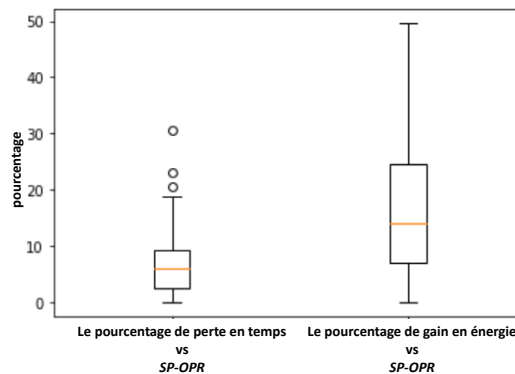


FIGURE 4-22 Les avantages et les inconvénients de la méthode *ECM-OPR*

Observation : On considère dans ce qui suit un cas de figure connexe/ inverse (réapprovisionnement des stocks dans différents points de l'entrepôt). Lorsque la matrice d'adjacence de distance (relative à l'environnement du robot) est symétrique, le chemin optimal du cas de livraison d'articles (réapprovisionnement des stocks) est égal au chemin optimal du cas de collecte des articles parcouru dans l'ordre inverse. Cependant, une instruction qui oblige le planificateur de déposer les articles lors de la première visite doit être ajoutée. En d'autres termes, le chemin optimal du cas de livraison est égal à l'inverse du chemin optimal du cas de collecte, mais le moment du dépôt des articles est toujours en arrivant à la position requise (première visite contrairement à *EE-OPR*).

Chapitre 5 - Conclusions et recommandations

L'intralogistique présente un levier crucial pour la réussite d'une chaîne d'approvisionnement d'une entreprise, mais aussi son goulot d'étranglement [210]. Ainsi, toute optimisation dans les opérations intralogistiques ou réduction de coûts de ces derniers aura une répercussion directe sur l'efficacité de la chaîne d'approvisionnement et le profit de l'entreprise. Par exemple, l'utilisation des *AMR* de différents types dans les milieux industriels pour le transport horizontal des biens offre divers avantages du point de vue opérationnel avec des opérations intralogistiques plus efficaces et moins coûteuses et du point de vue écologique avec des *AMR/AGV* majoritairement électriques [28]. Afin d'accroître ces avantages et pour suivre le processus de rendre les usines et les entrepôts plus éco-responsable, une meilleure gestion énergétique de ces systèmes est requise. Dans cette optique, et partant du fait que l'autonomie énergétique de ces véhicules est limitée par la quantité d'énergie contenue dans leurs batteries, cette thèse adresse l'optimisation de l'efficacité énergétique des *AMR* industriels à travers l'optimisation de leur planification de chemin afin de prolonger leurs durées de service, accroître la productivité, économiser leur consommation énergétique et réduire les déchets de leurs batteries en fin de vie.

En effet, les recherches actuelles liées à la planification de chemin des *AMR* se concentrent principalement sur l'optimisation de la distance et du temps de déplacement, tandis que l'aspect énergétique est rarement pris en compte lors de la planification globale. Contrairement à la planification de chemin locale économe en énergie qui est très abordée dans la littérature actuelle. Les rares travaux qui s'intéressent à la *GPP* avec considération d'énergie sont dédiés aux robots dans des environnements extérieurs et/ ou qui ignorent un ou plusieurs paramètre(s) qui influence(nt) la consommation énergétique d'un chemin versus l'autre. C'est pourquoi nous avons proposé une approche de planification de chemin globale écoénergétique qui

cherche à minimiser simultanément le temps et l'énergie consommée par un *AMR* lors de l'exécution de ces tâches de transport et de manutentions, en considérant différents paramètres de l'environnement pouvant influencer la consommation énergétique d'un *AMR*. En outre, la prise en compte de la consommation d'énergie nécessaire au déplacement de l'*AMR* dans le cadre d'une *OPP*, nécessite la considération de la masse transportée tout au long de sa tournée de préparation de commandes. En effectuant son tour de ramassage, l'*AMR* se déplace d'une position à l'autre dans l'entrepôt en consommant une énergie mécanique en translation, mais il s'arrête également pour ramasser les commandes dans un ordre bien déterminé. Cette addition de masse affecte la prochaine énergie en translation. Ainsi, l'accumulation des masses de chaque commande, avec un ordre déterminé, affecte l'énergie totale nécessaire pour effectuer une tournée de préparation de commandes. Par exemple, il est possible de réaliser des économies d'énergie en retardant le ramassage des charges les plus lourdes (en minimisant le temps nécessaire pour traîner ces charges). Il est donc essentiel d'intégrer la masse dans la définition de l'*OPP* et par conséquent dans la prise de décision.

5.1 Synthèse des travaux

5.1.1 *EE-GPP*

La première approche étudie le problème de *GPP* écoénergétique, pour qu'un *AMR* navigue d'un point à l'autre dans un environnement contenant des obstacles. Dans cette approche, nous avons souligné en premier lieu, l'effet de la résistance au roulement sur la consommation des différentes possibilités de chemins et en second lieu nous avons considéré l'effet de la congestion des différents segments de rue du graphe de l'industrie sur la consommation énergétique des chemins alternatifs. Comme tout planificateur de chemin global, *EE-GPP* nécessite deux étapes principales à savoir la représentation de l'espace et la stratégie de planification. *EE-GPP*, propose une représentation de la carte de l'environnement non seulement métrique, mais aussi avec l'information de la résistance au roulement et du trafic avec des matrices d'adjacence superposées. En outre, l'approche *EE-GPP* propose un algorithme de recherche de chemin capable de gérer ces informations de la carte et de considérer le modèle dynamique

du véhicule pour une meilleure sélection de chemin tout en assurant un bon compromis entre le temps et l'énergie nécessaires pour l'exécution du chemin.

Les résultats obtenus à travers différentes simulations effectuées pour l'évaluation de la méthode *EE-GPP* indiquent qu'elle donne toujours un coût de transport inférieur ou égal à celui de *ECM-GPP* et *SP-GPP* avec des moyennes dans l'intervalle [2.89, 10.08] % de gain en pourcentage vs *SP-GPP* et en des moyennes dans l'intervalle [6.46, 17.13] % si on considère seulement les instances avec zéro gain vs le plus court chemin. Ces gains permettent d'étendre la durée de fonctionnement des AMR et d'allonger la durée de vie de la batterie qui a aussi un coût (non considéré dans cette étude).

Ceci présente l'une des limitations de l'approche *EE-GPP*. En outre, cette dernière peut générer des chemins avec plusieurs virages angulaires qui obligent le robot à ralentir et faire des virages serrés. Une telle situation est observée dans l'exemple de la figure 5-1-a où on voit le chemin *EE-GPP* avec plus de virages que le *SP-GPP*. Cependant, à ce stade, c'est le rôle du planificateur local de générer des trajectoires plus lisses satisfaisant une certaine contrainte de continuité et de sécurité tout en suivant les directions données par le planificateur global. Un exemple d'un chemin local qui suit les *waypoints* donnés par le *GPP* tout en lissant la trajectoire est dessiné dans la même figure à droite 5-1-b. Rappelons que le rôle d'un planificateur global est de trouver un chemin hors ligne qui relie une configuration de départ à une configuration de destination tout en évitant les obstacles présents dans l'environnement du robot. Il génère un chemin global relié par des segments de ligne droite et par des points de passage étalonné dans le temps. Ensuite, le planificateur de chemin local (*LPP*) convertit le chemin conçu à une trajectoire qui optimise le profil de vitesse afin que l'*AMR* puisse suivre la trajectoire souhaitée avec des commandes de contrôles stables. De plus, le *LPP* s'occupe de contournement d'obstacles dû à l'environnement dynamique à travers l'ajustement des trajectoires locales. En outre, cette limitation peut être atténuée avec l'utilisation d'une représentation spatiale différente telle qu'un graphe de visibilité comme celui présenté dans la figure 5-6-b, ou encore avec chemin qui coupe les coins (voir figure 5-2). Cet aspect présente un axe de recherche à part entière et le traitement de ce dernier avec la considération de l'économie énergétique présente une direction intéressante de future recherche.

5.1.2 EE-OPR

La deuxième approche aborde le problème d'*OPR* écoénergétique d'un *Ps-AMR* étant donné son importance majeure dans la chaîne d'approvisionnement et son effet significatif sur l'efficacité opérationnelle des entrepôts et des *DCs*. [23].

Cette approche résout l'*OPR* indépendamment de la structure et l'aménagement de l'entrepôt et, par conséquent, peut être utilisée avec n'importe quelles configurations d'environnements, contrairement à la majorité des travaux dans la littérature résolvant l'*OPR* qui s'attachent à la configuration rectangulaire à allées parallèles des entrepôts, et *DCs* pour concevoir leurs méthodes. En effet, l'approche *EE-OPR* se base sur le fait que l'*OPR* est vu comme un problème de vendeur de commerce *Steiner (STSP)*, mais en considérant l'efficacité énergétique. Elle nécessite uniquement la définition d'un graphe pondéré (avec des distances) et les sommets de départ et d'arrivée, les sommets obligatoires (visiteurs) et les sommets de passage. Contrairement aux autres travaux de la littérature, non seulement le graphe de l'entrepôt pondéré avec l'information de distance est défini, mais aussi l'information de la masse de chaque sommet à visiter au regard de son effet sur la demande énergétique des chemins de prélèvement. Par suite, un algorithme dynamique est développé. Il transforme l'*OPR* en un problème de plus court chemin (*SPP*) en créant un graphe d'états dynamique acyclique et en effectuant un processus de recherche dans ce dernier. Ce graphe dynamique est composé de sommets d'états et d'arcs de transitions. La structure des sommets d'états couvre les informations spatiales (position) et l'état de prélèvement (articles prélevés et par conséquent la masse actuelle du robot). Par ailleurs, les arcs de transition définissent la connectivité entre les différents états et le coût en temps et en énergie nécessaire pour passer d'un état à l'autre. Ce graphe est créé progressivement et il est utilisé comme entrée à *SPP* avec l'objectif de partir d'une position de départ en portant zéro kilogramme et d'arriver à la position cible avec toutes les charges demandées par la liste de prélèvement (*pick-list*). Cette procédure permet de minimiser simultanément le temps et l'énergie d'une tournée de préparation de commandes.

En outre, l'approche proposée met en évidence deux observations majeures négligées dans la littérature. Premièrement, contrairement aux travaux existants dans la littérature qui

imposent de ramasser les objets une fois que le robot est arrivé à une position requise, *EE-OPR* n'impose pas une telle contrainte. En d'autres termes, *EE-OPR* donne la possibilité à un robot de passer par un sommet sans ramasser d'objets. Cette option permet, si nécessaire, de récupérer un élément requis lors du second passage. En fait, les études existantes se sont concentrées sur la minimisation de la distance/temps de la tournée de ramassage dans le but de collecter tous les éléments en parcourant la plus courte distance. Cependant, si on considère simultanément le temps et l'énergie, dans le cas où le plus court sous-graphe de la tournée de prélèvement doit passer deux fois sur un sommet requis, il est plus efficace d'un point de vue énergétique de récupérer l'article dans cette position sur le chemin du retour pour éviter de le traîner tout le long du trajet. Deuxièmement, la plupart des études qui traitent l'*OPP* se concentrent sur la détermination du plus court sous-graphe de la tournée en indiquant les arêtes qui construisent la tournée de prélèvement optimale et elles ne donnent pas de détails sur ce post-traitement qui donne la direction de ces arêtes pour suivre le tour au complet. En revanche, la façon dont *EE-OPR* aborde l'*OPP* permet de construire la tournée optimale avec la direction optimale écoénergétique.

Différentes simulations avec des situations diverses ont été effectuées afin d'évaluer *EE-OPR*. Ces derniers montrent que l'utilisation de cette approche conduit toujours à de meilleurs résultats par rapport à l'approche basée uniquement sur la minimisation du temps de parcours avec un gain moyen de 18%. À moyen et long terme, des gains potentiels d'économie d'énergie peuvent être réalisés. Ces gains peuvent être la diminution du coût de l'électricité, l'augmentation du temps opérationnel des AMR, la réduction du coût de maintenance ou de remplacement de la batterie.

Parmi les limitations de l'approche *EE-OPR* on retrouve le fait que le nombre de lieux de prélèvement à visiter dans une seule tournée de prélèvement est censé être inférieur à 18, en respectant la capacité de calcul. En effet, la complexité de l'algorithme est exponentielle en fonction de la taille du *pick-list*. Cela implique que le nombre de stops peut aller jusqu'à 18 dans une tournée de ramassage de commandes. Par contre, la quantité d'articles requis dans chaque emplacement peut être supérieure à un dans chaque emplacement. Par exemple, la mise en lots des commandes (*Order batching*) est une technique qui permet de regrouper un ensemble de commandes en lots[212], par emplacement de stockage. Ainsi, la quantité d'articles requis

dans chaque emplacement peut varier en fonction de cette technique de regroupement des commandes. Par conséquent, la taille de la *pick-list* ne reflète pas nécessairement le nombre d'articles collectés dans un tour de ramassage, mais plutôt le nombre de stops dans ce dernier (si on suppose que le nombre d'articles dans un sommet est dix dans une *pick-list* de taille 18, alors le nombre d'articles prélevés dans ce tour de ramassage est 180). Toutefois, l'amélioration de cette limite de taille de la *pick-list* reste requise et peut être abordée dans des recherches futures. En outre, cette limitation ne présente plus une contrainte, si la taille de la liste de prélèvement est très restreinte par défaut. Par exemple, dans les entrepôts où les articles sont lourds et encombrants¹ (tels que les gros appareils électroniques, les tapis ou tout autre article massif et volumineux), la taille de la *pick-list* est souvent petite, ce qui rend la limitation de la complexité élevée de notre approche moins contraignante. D'ailleurs, la capacité du véhicule limiterait le nombre d'articles à ramasser par tour de ramassage.

En outre, les hypothèses mises en place dans ce travail peuvent être modifiées dans des travaux futurs pour considérer plus qu'une problématique en même temps. Par exemple, cette thèse considère la problématique de planification de chemin d'un seul robot. La problématique de gestion multirobots avec le problème VRP écoénergétique peut présenter une bonne extension de ce travail, cependant avec plus de contraintes et beaucoup plus de complexité spatiales et temporelles. De plus, étant la dépendance entre le problème de routage de préparation de commandes et le problème de la mise en lots, la considération de la fusion des deux problématiques dans le but de l'amélioration de l'efficacité énergétique, sera une bonne amélioration de *EE-OPR*. Par ailleurs, l'hypothèse relative à la vitesse considérée constante facilite la résolution du problème et peut être acceptable pour les raisons énoncées. Cependant, pour une meilleure prédiction de la demande énergétique, la considération de la variation de la vitesse linéaire et angulaire est préférable. D'autre part, étant donné que la définition du problème de prélèvement dans cette étude offre la possibilité d'un mouvement bidirectionnel des AMR dans une allée, *EE-OPR* peut convenir à des centres de distribution à faible et moyen débit. Le cas échéant, un problème de congestion dans les allées peut se poser. Toutefois, un

1. Les PS-AGV/AMR sont souvent utilisés pour les articles lourds et volumineux. Cependant, le prélèvement des commandes par PS-AGV n'est pas limité à ces types d'articles. Les PS-AGV peuvent également être appliqués pour assister les préparateurs de commandes dans un environnement de zonage et de mise en lots.[25]

mouvement unidirectionnel peut être requis en changeant simplement le graphe de représentation de l'entrepôt. Par ailleurs, le changement de la stratégie de stockage ainsi que la stratégie de mise en lot ont, sans aucun doute, un effet sur le gain que *EE-OPR* peut fournir par rapport à *SP-OPR*. Cet effet peut être étudié dans des recherches futures.

5.2 Recommandations

Dans cette section, nous allons citer quelques pistes intéressantes en relation avec le travail présenté dans cette thèse qui peuvent être des futurs projets potentiels :

- Quelques travaux de recherche fusionnent l'optimisation du routage de ramassage de commandes (pour la minimisation de la distance/ temps de voyage) avec l'optimisation de la gestion des politiques de stockages des items dans les entrepôts et les *DCs*. Ces travaux peuvent être enrichis par l'amélioration de l'efficacité énergétique de ces opérations, surtout que l'emplacement des articles dans les différentes positions de l'entrepôt a certes un impacte directe sur la gestion des déplacements de différentes masses, influençant l'énergie consommée. Par exemple, les figures 5-3 et 5-4 montrent des exemples de stratégies de stockage qui peuvent certainement affecter l'efficacité d'une politique de routage. Par conséquent, une étude sur l'efficacité d'*EE-OPR* avec différentes stratégies de stockage de commandes peut être réalisée. Par exemple, Tarczynski et al. [213] effectue une étude sur l'impact des emplacements des étagères, de la taille des listes de prélèvement et du nombre d'allées de prélèvement sur le choix de la méthode de routage de la préparation des commandes ainsi que l'effet de tous ces paramètres sur les performances du processus d'*OP*.
- Comme précédemment annoncée, l'amélioration du graphe de l'industrie présente une bonne piste d'amélioration de l'efficacité énergétique du routage de l'*AMR*. Par exemple, le graphe de visibilité peut être utilisé comme moyen alternatif pour estimer les distances entre les *waypoints* du graphe d'entrepôt ou de la manufacture [211] (voir figure 5-6). De plus, les méthodes proposées dans cette thèse sont capables de gérer cette nouvelle configuration (c'est le graphe "distance" qui change).

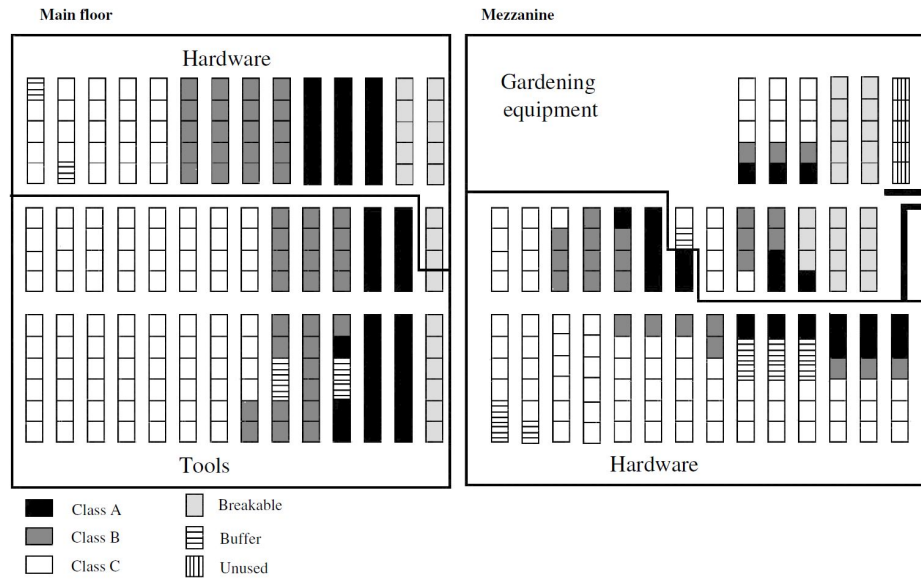


FIGURE 5-3 Un exemple de stratégie de stockage combiné (la ligne continue indique la répartition des types des produits sur les emplacements)[214].

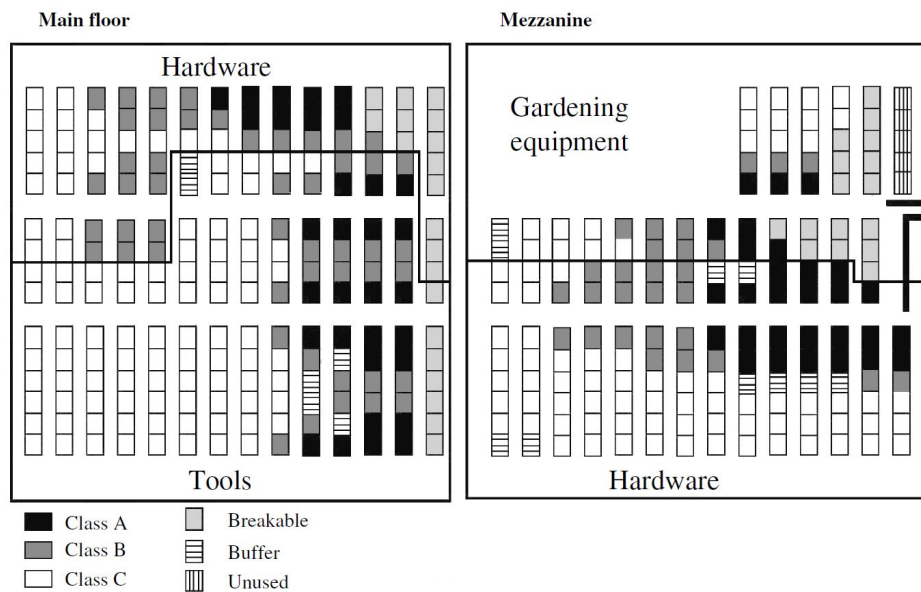


FIGURE 5-4 Un exemple de stratégie de stockage *Midpoint* (la ligne continue indique la répartition des types des produits sur les emplacements) [214].

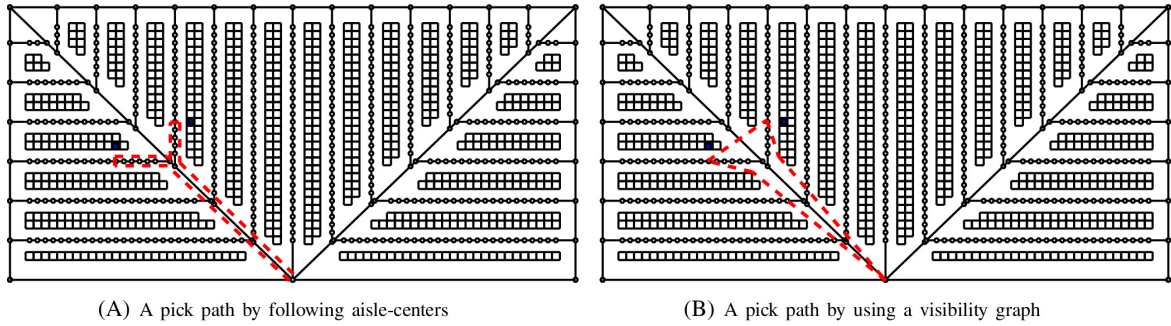


FIGURE 5-5 Deux solutions de chemins de prélèvement de deux articles représentés par des points en gras qui commencent et retournent au dépôt (en bas au centre.) [211]

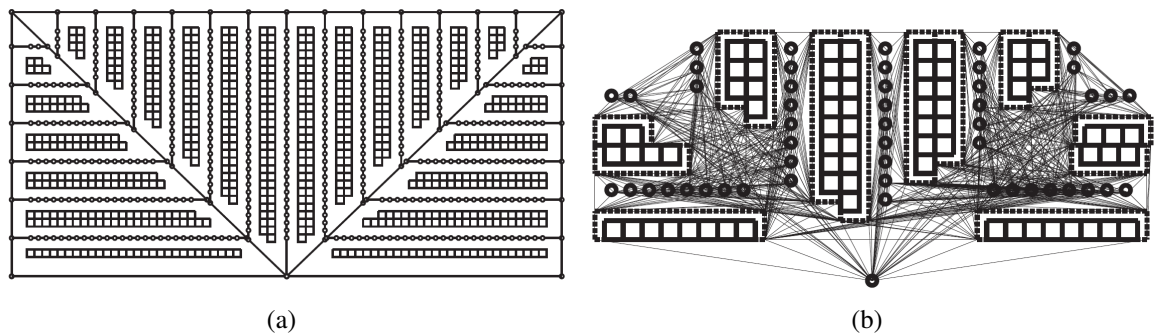


FIGURE 5-6 (a) Représentation d'un entrepôt avec la configuration « fishbone » utilisant la méthode définie par Gue et al. [209] (b) Graphe de visibilité d'un entrepôt avec la configuration « fishbone » [211]

- La réalisation d’une étude comparative entre *EE-OPR* et les différentes stratégies de routage heuristiques.
- Une étude expérimentale qui utilise les approches proposées dans un contexte réel.
- Étude sur l’influence des approches *EE-PP* sur le planificateur local *LPP* et l’amélioration de ces derniers simultanément. En effet, une fois le chemin écoénergétique est défini (la séquence de positions à visiter), il est communiqué au planificateur *LPP*. Il est intéressant que ce dernier soit à son tour efficace énergiquement en utilisant les contraintes cinématiques et cinétiques des robots, mais aussi qui considère l’effet de la position de la charge sur la dynamique de l’*AMR*, surtout qu’avec *EE-OPR*, l’information de la masse entre les stops de ramassage de commandes est connue. Par exemple, [68] et al. proposent une nouvelle approche pour améliorer l’efficacité énergétique du planificateur local du robot en prenant en compte la position et le poids de la charge déposée sur l’*AMR*.

5.3 Travaux en cours

Nous avons discuté dans le chapitre 1 que la flexibilité croissante des manufactures et des entrepôts rend le processus de planification de chemins de plus en plus complexe. Nous avons exposé les différentes variations de l’environnement du robot qui peuvent augmenter la consommation énergétique des différentes alternatives des chemins disponibles. Ces augmentations sont inconnues à priori, ce qui donne au problème un aspect stochastique qui va être étudié dans les travaux futurs. En effet, (i) le coût du chemin optimal peut varier en fonction du temps (le même chemin avec un coût différent) et (ii) le chemin optimal peut changer en fonction de temps. La figure 5-7 exprime ces deux aspects à travers un exemple de routes urbaines pris de *GoogleMap*.

De plus, la modélisation de la consommation énergétique avec un apprentissage automatique de la carte améliore la précision des prévisions de la consommation énergétique des segments de rues tous en introduisant l’aspect stochastique de l’environnement.

Dans cette optique, nous proposons une approche qui permet l’apprentissage de la carte énergétique d’une industrie. Cette dernière se base sur la création de la carte d’indicateurs

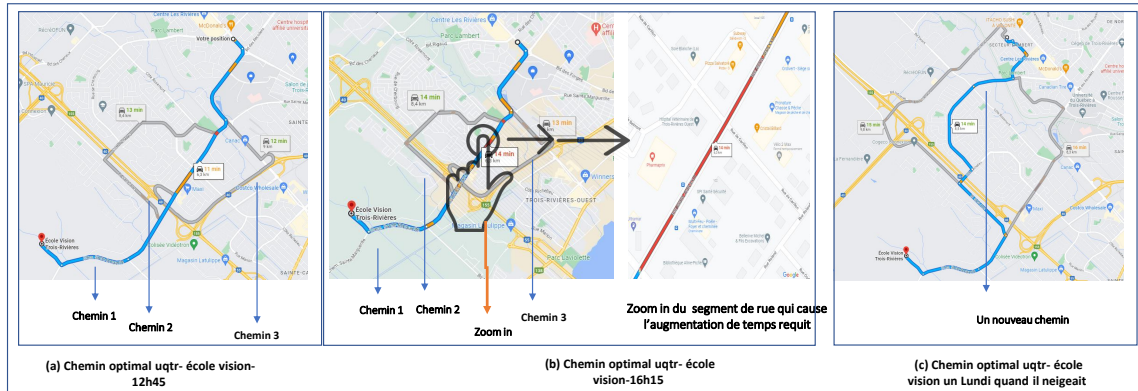


FIGURE 5-7 Explication de l'effet du temps sur les décisions du routage à travers un exemple de planification de chemin *source-target* sur une route urbaine utilisant *GoogleMap*

énergétique à travers l'exécution de plusieurs tâches de transport avec différents robots. Pour se déplacer d'un point d'intérêt à l'autre, le robot passe par plusieurs arêtes dans son parcours. La consommation énergétique de chaque arête est mesurée et enregistrée dans la banque de données de cette dernière (l'arête). Après une heure de travail, on se ramasse avec l'historique de la consommation énergétique d'un sous-ensemble de graphe de rue, avec un nombre variable de passages par arête. Pour une arête (u, v) , un vecteur de consommation énergétique historique est créé. En parallèle, une valeur de consommation énergétique de référence est calculée. Cette valeur est obtenue à l'aide de l'équation dynamique du robot en fonction de la masse transportée en partant du nœud u . Par la suite le rapport entre la valeur d'énergie consommée mesurée et la valeur d'énergie consommée référence est calculée pour chaque arête. Ce coefficient est appelé "indicateur énergétique". Nous utilisons ce dernier pour construire une carte d'indicateurs énergétiques d'une manière dynamique selon le jour et la période de la journée. En effet, l'indicateur énergétique se détache de toute les paramètres imprécis qui influencent la consommation sur un segment donné. Par exemple, si l'AMR a consommé 5 wh en passant sur un segment donné, ce chiffre dépend d'une multitude de paramètres. Par conséquent, la base de données d'une arête n'est pas censée enregistrer ce chiffre et juger si ce segment est énergivore ou non (car il dépend de la masse transportée et autres). Cependant, le calcul d'indicateurs énergétiques permet d'avoir une idée si le passage sur cette arête est énergivore ou non (par exemple à cause de la congestion sur ce segment de rue). La création graduelle de l'historique d'indicateurs énergétique par arête permet de construire une carte

énergétique pour chaque heure de la journée analogiquement à l'exemple présenté dans la figure 5-7. Cette carte est utilisée comme une nouvelle entrée pour nos planificateurs éco-énergétiques. Un exemple de résultat préliminaire de carte énergétique colorée est représenté dans la figure 5-8 où on représente les segments de rue les plus énergivores en rouge et graduellement les segments dont la valeur de l'énergie mesurée est proche de l'énergie de référence en jaune pâle. Nous constatons qu'à l'aide de cette carte, le planificateur global réussit à trouver un chemin plus écoénergétique (vert) que celui de plus courte distance (noir), sans pour autant affecter le temps de voyage.

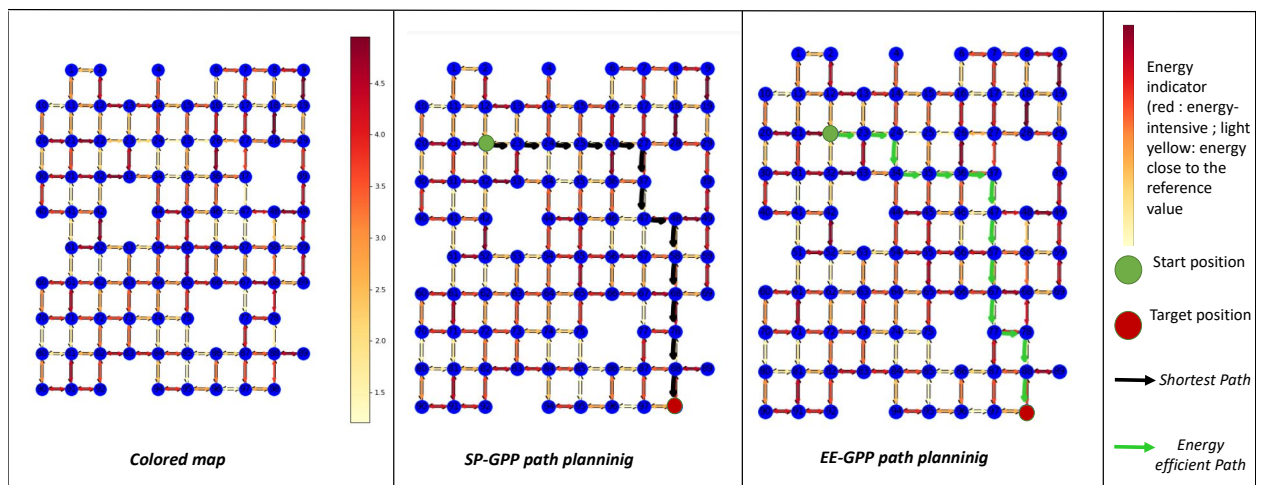


FIGURE 5-8 Carte colorée d'indicateur énergétique et son effet sur la décision de routage

5.4 Conclusion

Plusieurs travaux de recherche récents ont été établis pour encourager le déploiement d'une industrie verte et durable. En tant qu'équipement de manutention typique des manufactures, entrepôts et centres de distribution modernes, la consommation d'énergie des AMR représente une part importante du gaspillage énergétique total de l'entrepôt. Cependant, les recherches actuelles liées à la planification de chemin source-destination et de préparation des commandes se concentrent principalement sur l'optimisation de la distance et du temps de déplacement, tandis que l'aspect énergétique est rarement pris en compte. Pourtant, dans de nombreuses situations, l'énergie est aussi importante que le temps, surtout pour les robots lourds, les articles

lourds et les surfaces d'industrie vaste. Un moyen efficace de réduction de la consommation des *AMR* est d'améliorer leur efficacité opérationnelle et leur routage, ce qui présente l'objectif de cette thèse. Une bonne stratégie de planification de chemins pour les robots de manutention autonomes permet non seulement d'économiser beaucoup de temps, mais aussi de réduire la consommation énergétique, les périodes d'arrêts de recharge improductive, l'usure de la batterie de ces véhicules et leurs déchets en fin de vie, l'investissement en capital de ces robots, sans oublié l'aspect environnemental. Ainsi, chaque petit pas peut participer à sauver notre planète.

Bibliographie

- [1] M INDRI, A GRAU, M. R. I. T. o. INDUSTRIAL et u. 2018, « Guest editorial special section on recent trends and developments in industry 4.0 motivated robotic solutions, » *ieeexplore.ieee.org*, adresse : <https://ieeexplore.ieee.org/abstract/document/8331177/>.
- [2] L. CUSTODIO et R. MACHADO, « Flexible automated warehouse : a literature review and an innovative framework, » *The International Journal of Advanced Manufacturing Technology*, t. 106, n° 1, p. 533-558, 2020.
- [3] G FRAGAPANE, R. D. KOSTER, F. S. E. J. o. . . . et u. 2021, « Planning and control of autonomous mobile robots for intralogistics : Literature review and research agenda, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0377221721000217>.
- [4] *Service Robotics Market Size, Share & Trends 2021-2026*.
- [5] G. ULLRICH et T. ALBRECHT, « Automated Guided Vehicle Systems, » 2023. DOI : 10.1007/978-3-658-35387-2. adresse : <https://link.springer.com/10.1007/978-3-658-35387-2>.
- [6] *Global Automated Guided Vehicle Market Report 2016-2021 - Analysis, Technologies & Forecasts - Key Vendors : ABB, Siemens, Doerfer Corp - Research and Markets | Business Wire*.
- [7] P. MATTABONI, « Autonomous mobile robot, » 1987. adresse : <https://www.osti.gov/biblio/6721410>.

- [8] D AGARWAL, P. B. J. o. S. MANAGEMENT, et u. 2019, « Computation of cause and effect relationship for acceptance of autonomous mobile robots in industries, » *Taylor & Francis*, t. 22, n° 2, p. 237-256, fév. 2019. DOI : 10.1080/09720510.2019.1580903. adresse : <https://www.tandfonline.com/doi/abs/10.1080/09720510.2019.1580903>.
- [9] J. VENTURA, S PAZHANI, A. M. I. J. o. PRODUCTION et u. 2015, « Finding optimal dwell points for automated guided vehicles in general guide-path layouts, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0925527315000754>.
- [10] A. LOGANATHAN et N. S. AHMAD, « A systematic review on recent advances in autonomous mobile robot navigation, » *Engineering Science and Technology, an International Journal*, t. 40, p. 101 343, avr. 2023. DOI : 10.1016/J.JESTCH.2023.101343.
- [11] M LOM, O PRIBYL, M. S. S. C. SYMPOSIUM et u. 2016, « Industry 4.0 as a part of smart cities, » *ieeexplore.ieee.org*, adresse : <https://ieeexplore.ieee.org/abstract/document/7501015/>.
- [12] *How are Fleet Management Systems implemented in different robotic companies ?*
- [13] FETCHROBOTICS, *AMR-assisted Case Picking*.
- [14] *Warehousing Automation - Vecna Robotics*.
- [15] *Préparateurs de commandes autonomes | STILL France*.
- [16] *Fulfillment Solutions & Warehouse Automation | 6 River Systems*.
- [17] *Automated Warehouse Robots | Robotic Process Automation*.
- [18] *Autonomous Warehouse Robots that Improve Productivity from Fetch Robotics*.
- [19] R. BOUCHER, « Le commerce électronique au Québec, » 2020. adresse : <https://www.jll.ca/en/trends-and-insights/investor/who-funds-the-next-wave-of-warehouse-automation>.

- [20] Z. GHELICHI et S. KILARU, « Analytical models for collaborative autonomous mobile robot solutions in fulfillment centers, » *Applied Mathematical Modelling*, t. 91, p. 438-457, 2021. DOI : <https://doi.org/10.1016/j.apm.2020.09.059>. adresse : <https://www.sciencedirect.com/science/article/pii/S0307904X20305801>.
- [21] « Who funds the next wave of warehouse automation ?, » adresse : <https://www.jll.ca/en/trends-and-insights/investor/who-funds-the-next-wave-of-warehouse-automation>.
- [22] P. FAGER, F. SGARBOSSA et M. CALZAVARA, « Cost modelling of onboard cobot-supported item sorting in a picking system, » *International Journal of Production Research*, p. 1-16, 2020.
- [23] S. WINKELHAUS, E. H. GROSSE et S. MORANA, « Towards a conceptualisation of Order Picking 4.0, » *Computers & Industrial Engineering*, t. 159, p. 107 511, sept. 2021. DOI : 10.1016/J.CIE.2021.107511.
- [24] K. AZADEH, D. ROY et M. B. M. de KOSTER, « Dynamic Human-Robot Collaborative Picking Strategies, » *Available at SSRN 3585396*, 2020.
- [25] N. BOYSEN, R. de KOSTER et F. WEIDINGER, « Warehousing in the e-commerce era : A survey, » *European Journal of Operational Research*, t. 277, n° 2, p. 396-411, 2019.
- [26] R. MELLER, D. NAZZAL et L. THOMAS, « Collaborative Bots in Distribution Centers, » *Progress in Material Handling Research*, jan. 2018. adresse : https://digitalcommons.georgiasouthern.edu/pmhr_2018/17.
- [27] S. VAIDYA, P. AMBAD et S. BHOSLE, « Industry 4.0 – A Glimpse, » *Procedia Manufacturing*, t. 20, p. 233-238, jan. 2018. DOI : 10.1016/J.PROMFG.2018.02.034.
- [28] D BECHTSIS, N TSOLAKIS, D VLACHOS, E. I. J. o. CLEANER et u. 2017, « Sustainable supply chain management in the digitalisation era : The impact of Automated Guided Vehicles, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0959652616316675>.

- [29] T PAKSOY, C. KOCHAN et S. ALI, *Logistics 4.0 : digital transformation of supply chain management*. 2020.
- [30] S. LANG, M. WARBURTON et A. A. QUETTAWALA, « Global Autos : Don't Believe the Hype – Analyzing the Costs & Potential of Fuel-Efficient Technology, » *Bernstein Black Book - Global Autos : Don't Believe the Hype - Analyzing the Costs & Potential of Fuel-Efficient Technology*, p. 1-475, 2011. adresse : <http://proxy.uqtr.ca/login.cgi?action=login&u=uqtr&db=ebsco&ezurl=http://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=70255755&site=ehost-live>.
- [31] T. KAWAKAMI et S. TAKATA, « Battery Life Cycle Management for Automatic Guided Vehicle Systems, » *Design for Innovative Value Towards a Sustainable Society*, p. 403-408, 2012. DOI : 10.1007/978-94-007-3010-6{_}77.
- [32] V. ANAND, S. LEE et V. V. PRABHU, « Energy-aware models for warehousing operations, » in *IFIP International Conference on Advances in Production Management Systems*, Springer, 2014, p. 390-397.
- [33] F. MARTEL, Y. DUBÉ, S. KELOUWANI, J. JAGUEMONT et K. AGBOSSOU, « Long-term assessment of economic plug-in hybrid electric vehicle battery lifetime degradation management through near optimal fuel cell load sharing, » *Journal of Power Sources*, t. 318, p. 270-282, juin 2016. DOI : 10.1016/J.JPOWSOUR.2016.04.029.
- [34] F. MARTEL, S. KELOUWANI, Y. DUBÉ et K. AGBOSSOU, « Optimal economy-based battery degradation management dynamics for fuel-cell plug-in hybrid electric vehicles, » *Journal of Power Sources*, t. 274, p. 367-381, jan. 2015. DOI : 10.1016/J.JPOWSOUR.2014.10.011.
- [35] A. BARRÉ, B. DEGUILHEM, S. GROLLEAU, M. GÉRARD, F. SUARD et D. RIU, « A review on lithium-ion battery ageing mechanisms and estimations for automotive applications, » *Journal of Power Sources*, t. 241, p. 680-689, nov. 2013. DOI : 10.1016/J.JPOWSOUR.2013.05.040.

- [36] S KELOUWANI, K AGBOSSOU, Y DUBÉ, L. B. J. o. P. SOURCES et u. 2013, « Fuel cell plug-in hybrid electric vehicle anticipatory and real-time blended-mode energy management for battery life preservation, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0378775312012906>.
- [37] M SHAMS-ZAHRAEI, A. KOUZANI, S. K. J. o. p. SOURCES et u. 2012, « Integrated thermal and energy management of plug-in hybrid electric vehicles, » *Elsevier*, adresse : https://www.sciencedirect.com/science/article/pii/S037877531200924X?casa_token=xzq0HzAjdSgAAAAA:qclwr7iMey-k2iFLu3wJ8eUjdEOtKI2HXekyAPxsLhZaEOc36F9rQ96M6fE8yY.
- [38] A. MILLNER, « Modeling lithium ion battery degradation in electric vehicles, » *2010 IEEE Conference on Innovative Technologies for an Efficient and Reliable Electricity Supply, CITRES 2010*, p. 349-356, 2010. DOI : 10.1109/CITRES.2010.5619782.
- [39] M ECKER, N NIETO, S KÄBITZ, J. S. J. o. P. ... et u. 2014, « Calendar and cycle life study of Li (NiMnCo) O₂-based 18650 lithium-ion batteries, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0378775313016510>.
- [40] S. PETERSON, J APT, J. W. J. o. P. SOURCES et u. 2010, « Lithium-ion battery cell degradation resulting from realistic vehicle and vehicle-to-grid utilization, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0378775309017443>.
- [41] N OMAR, M. MONEM, Y FIROUZ, J SALMINEN, J. S. A. ENERGY et u. 2014, « Lithium iron phosphate based battery—Assessment of the aging parameters and development of cycle life model, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0306261913007393>.
- [42] K. DIVYA, J. E. p. s. RESEARCH et u. 2009, « Battery energy storage technology for power systems—An overview, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0378779608002642>.
- [43] F. MARTEL, Y. DUBE, S. KELOUWANI et K. AGBOSSOU, « Economy-Focused PHEV Battery Lifetime Management through Optimal Fuel Cell Load Sharing, » *2015 IEEE*

- Vehicle Power and Propulsion Conference, VPPC 2015 - Proceedings*, déc. 2015. DOI : 10.1109/VPPC.2015.7352934.
- [44] Q. KABIR, Y. S. C. I. ENGINEERING et u. 2018, « Increasing manufacturing flexibility through battery management of automated guided vehicles, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0360835218300330>.
- [45] E. G. TSARDOULIAS, A. ILIAKOPOULOU, A. KARGAKOS et L. PETROU, « A Review of Global Path Planning Methods for Occupancy Grid Maps Regardless of Obstacle Density, » *Journal of Intelligent & Robotic Systems 2016 84 :1*, t. 84, n° 1, p. 829-858, mai 2016. DOI : 10.1007/s10846-016-0362-z. adresse : <https://link.springer.com/article/10.1007/s10846-016-0362-z>.
- [46] M. MEISSNER et L. MASSALSKI, « Modeling the electrical power and energy consumption of automated guided vehicles to improve the energy efficiency of production systems, » *International Journal of Advanced Manufacturing Technology*, t. 110, n° 1-2, p. 481-498, sept. 2020. DOI : 10.1007/s00170-020-05796-8/FIGURES/27. adresse : <https://link.springer.com/article/10.1007/s00170-020-05796-8>.
- [47] A. ALAJLAN, K. ELLEITHY, M. ALMASRI et T. SOBH, « An Optimal and Energy Efficient Multi-Sensor Collision-Free Path Planning Algorithm for a Mobile Robot in Dynamic Environments, » *Robotics 2017, Vol. 6, Page 7*, t. 6, n° 2, p. 7, mars 2017. DOI : 10.3390/ROBOTICS6020007. adresse : <https://www.mdpi.com/2218-6581/6/2/7/htmhttps://www.mdpi.com/2218-6581/6/2/7>.
- [48] L. XIE, C. HENKEL, K. STOL et W. XU, « Power-minimization and energy-reduction autonomous navigation of an omnidirectional Mecanum robot via the dynamic window approach local trajectory planning, » <https://doi.org/10.1177/1729881418754563>, t. 15, n° 1, jan. 2018. DOI : 10.1177/1729881418754563. adresse : <https://journals.sagepub.com/doi/full/10.1177/1729881418754563>.
- [49] R. d. K. L. TRANSPORT, et u. 2018, « Automated and robotic warehouses : developments and research opportunities, » *bibliotekanauki.pl*, adresse : <https://bibliotekanauki.pl/articles/504415.pdf>.

- [50] B. K. PATLE, G. BABU L, A. PANDEY, D. R. PARHI et A. JAGADEESH, « A review : On path planning strategies for navigation of mobile robot, » *Defence Technology*, t. 15, n° 4, p. 582-606, août 2019. DOI : 10.1016/J.DT.2019.04.011.
- [51] K. AZADEH, R. DE KOSTER et D. ROY, « Robotized warehouse systems : Developments and research opportunities, » *ERIM report series research in management Erasmus Research Institute of Management*, n° ERS-2017-009-LIS, 2017.
- [52] Y. MEI, Y. H. LU, Y. C. HU et C. S. LEE, « Energy-efficient motion planning for mobile robots, » *Proceedings - IEEE International Conference on Robotics and Automation*, t. 2004, n° 5, p. 4344-4349, 2004. DOI : 10.1109/ROBOT.2004.1302401.
- [53] Z. ZHANG, L. WU, W. ZHANG, T. PENG et J. ZHENG, « Energy-efficient path planning for a single-load automated guided vehicle in a manufacturing workshop, » *Computers & Industrial Engineering*, t. 158, p. 107397, 2021. DOI : <https://doi.org/10.1016/j.cie.2021.107397>. adresse : <https://www.sciencedirect.com/science/article/pii/S0360835221003016>.
- [54] V. USENKO, J. ENGEL, J. STUCKLER et D. CREMERS, « Direct visual-inertial odometry with stereo cameras, » *Proceedings - IEEE International Conference on Robotics and Automation*, t. 2016-June, p. 1885-1892, juin 2016. DOI : 10.1109/ICRA.2016.7487335.
- [55] H. ZHANG, Y. ZHANG et T. YANG, « A survey of energy-efficient motion planning for wheeled mobile robots, » *Industrial Robot*, t. 47, n° 4, p. 607-621, juin 2020. DOI : 10.1108/IR-03-2020-0063/FULL/HTML.
- [56] *Car Weight vs. MPG.*
- [57] R. A. NEWCOMBE, S. IZADI, O. HILLIGES et al., « KinectFusion : Real-time dense surface mapping and tracking, » *2011 10th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2011*, p. 127-136, 2011. DOI : 10.1109/ISMAR.2011.6092378.

- [58] C. CLARAMUNT et S. WINTER, « Structural Salience of Elements of the City, » *https://doi.org/10.1068/b32099*, t. 34, n° 6, p. 1030-1050, juill. 2016. DOI : 10.1068/B32099. adresse : https://journals.sagepub.com/doi/abs/10.1068/b32099?casa_token=tFQGq1Ef-iwAAAAA%3AU4mR3pSfnvZh4ODTgX55sHb8nbzxK_O9kAlz_WVy4_zDMt_ig1q_lRlW96wrIbWR4DjviI1yYEG4g.
- [59] F. BOENZI, S. DIGIESI, F. FACCHINI, G. MOSSA et G. MUMMOLO, « GREENING ACTIVITIES IN WAREHOUSES : A MODEL FOR IDENTIFYING SUSTAINABLE STRATEGIES IN MATERIAL HANDLING, » *Annals of DAAAM & Proceedings*, t. 26, n° 1, 2015.
- [60] F. FACCHINI, G. MUMMOLO, G. MOSSA, S. DIGIESI, F. BOENZI et R. VERRIELLO, « Minimizing the carbon footprint of material handling equipment : Comparison of electric and LPG forklifts, » *Journal of Industrial Engineering and Management (JIEM)*, t. 9, n° 5, p. 1035-1046, 2016.
- [61] P. ZAJAC et T. ROZIC, « Energy consumption of forklift versus standards, effects of their use and expectations, » *Energy*, t. 239, p. 122 187, jan. 2022. DOI : 10.1016/J.ENERGY.2021.122187.
- [62] P. ZAJAC et P. SKORUPSKI, « Modeling of the Energy Consumption of a Forklift Truck Using the Matlab Simulink System, » *Lecture Notes in Mechanical Engineering*, p. 851-857, juin 2018. DOI : 10.1007/978-3-030-04975-1_{_}99. adresse : https://link.springer.com/chapter/10.1007/978-3-030-04975-1_99.
- [63] F. BASSO, L. D. EPSTEIN, R. PEZOA et M. VARAS, « An optimization approach and a heuristic procedure to schedule battery charging processes for stackers of palletized cargo, » *Computers & Industrial Engineering*, t. 133, p. 9-18, juill. 2019. DOI : 10.1016/J.CIE.2019.04.045.
- [64] B. ZOU, X. XU, Y. Y. GONG et R. DE KOSTER, « Evaluating battery charging and swapping strategies in a robotic mobile fulfillment system, » *European Journal of Operational Research*, t. 267, n° 2, p. 733-753, juin 2018. DOI : 10.1016/J.EJOR.2017.12.008.

- [65] K. ALIEV, E. TRAINI, M. ASRANOV, A. AWOUDA et P. CHIABERT, « Prediction and estimation model of energy demand of the AMR with cobot for the designed path in automated logistics systems, » *Procedia CIRP*, t. 99, p. 116-121, jan. 2021. DOI : 10.1016/J.PROCIR.2021.03.036.
- [66] Y. MEI, Y.-H. LU, Y. C. HU et C. S. G. LEE, « Deployment of mobile robots with energy and timing constraints, » *IEEE Transactions on Robotics*, t. 22, n° 3, p. 507-522, 2006.
- [67] S. LIU et D. SUN, « Minimizing energy consumption of wheeled mobile robots via optimal motion planning, » *IEEE/ASME Transactions on Mechatronics*, t. 19, n° 2, p. 401-411, 2013.
- [68] M. MOHAMMADPOUR, L. ZEGHMI, S. KELOUWANI, M. A. GAUDREAU, A. AMAMOU et M. GRABA, « An Investigation into the Energy-Efficient Motion of Autonomous Wheeled Mobile Robots, » *Energies 2021, Vol. 14, Page 3517*, t. 14, n° 12, p. 3517, juin 2021. DOI : 10.3390/EN14123517. adresse : <https://www.mdpi.com/1996-1073/14/12/3517/html><https://www.mdpi.com/1996-1073/14/12/3517>.
- [69] M. MONTAZERIJOUYBARI, L. BARON et S. KELOUWANI, « Kinematics of 2-DOF AGVs with Differential Driving Wheels and Caster Wheels Modeling, » *CISM International Centre for Mechanical Sciences, Courses and Lectures*, t. 601, p. 495-502, 2021. DOI : 10.1007/978-3-030-58380-4_{_}59.
- [70] S. BAKSHI, T. FENG, Z. YAN, Z. MA et D. CHEN, « Energy-Conscientious Trajectory Planning for an Autonomous Mobile Robot in an Asymmetric Task Space, » *Journal of Intelligent and Robotic Systems : Theory and Applications*, t. 101, n° 1, p. 1-14, jan. 2021. DOI : 10.1007/S10846-020-01288-9/METRICS. adresse : <https://link.springer.com/article/10.1007/s10846-020-01288-9>.
- [71] L. LIU, R. ZHONG, A. WILLCOCK, N. FISHER et W. SHI, « An Open Approach to Energy-Efficient Autonomous Mobile Robots, » *Proceedings - IEEE International Conference on Robotics and Automation*, t. 2023-May, p. 11 569-11 575, 2023. DOI : 10.1109/ICRA48891.2023.10161110.

- [72] H. Y. ZHANG, W. M. LIN et A. X. CHEN, « Path Planning for the Mobile Robot : A Review, » *Symmetry* 2018, Vol. 10, Page 450, t. 10, n° 10, p. 450, oct. 2018. DOI : 10.3390/SYM10100450. adresse : <https://www.mdpi.com/2073-8994/10/10/450/html><https://www.mdpi.com/2073-8994/10/10/450>.
- [73] M. A. CONTRERAS-CRUZ, V. AYALA-RAMIREZ et U. H. HERNANDEZ-BELMONTE, « Mobile robot path planning using artificial bee colony and evolutionary programming, » *Applied Soft Computing*, t. 30, p. 319-328, mai 2015. DOI : 10.1016/J.ASOC.2015.01.067.
- [74] J.-C. LATOMBE, *Robot motion planning*. Kluwer Academic Publishers, 1991, p. 651.
- [75] T. LOZANO-PÉREZ, « Spatial Planning : A Configuration Space Approach, » *IEEE Transactions on Computers*, t. C-32, n° 2, p. 108-120, 1983. DOI : 10.1109/TC.1983.1676196.
- [76] D HAGER, Z DODDS et D. MOCHA, « 6-735, Howie Choset with slides from G, » adresse : <http://www.cs.cmu.edu/~motionplanningHowieChoset><http://www.cs.cmu.edu/~choset>.
- [77] A. KOUBAA, H. BENNACEUR, I. CHAARI et al., « Introduction to mobile robot path planning, » *Studies in Computational Intelligence*, t. 772, p. 3-12, 2018. DOI : 10.1007/978-3-319-77042-0{_}1.
- [78] P. LI, X. HUANG et M. WANG, « A novel hybrid method for mobile robot path planning in unknown dynamic environment based on hybrid DS_m model grid map, » <https://doi.org/10.1080/0952813X.2010.506283>, t. 23, n° 1, p. 5-22, mars 2011. DOI : 10.1080/0952813X.2010.506283. adresse : <https://www.tandfonline.com/doi/abs/10.1080/0952813X.2010.506283>.
- [79] H. NIU, Y. LU, A. SAVVARIS et A. TSOURDOS, « An energy-efficient path planning algorithm for unmanned surface vehicles, » *Ocean Engineering*, t. 161, p. 308-321, août 2018. DOI : 10.1016/J.OCEANENG.2018.01.025.

- [80] H. ZHANG, Y. ZHANG, C. LIU et Z. ZHANG, « Energy efficient path planning for autonomous ground vehicles with ackermann steering, » *Robotics and Autonomous Systems*, t. 162, p. 104-366, avr. 2023. DOI : 10.1016/J.ROBOT.2023.104366.
- [81] K. OSANLOU, C. GUETTIER, A. BURSUC, T. CAZENAVE et E. JACOPIN, « Learning-based Preference Prediction for Constrained Multi-Criteria Path-Planning, » août 2021. adresse : <https://arxiv.org/abs/2108.01080v1>.
- [82] K. OTSU et T. KUBOTA, « Energy-aware terrain analysis for mobile robot exploration, » *Springer Tracts in Advanced Robotics*, t. 113, p. 373-388, 2016. DOI : 10.1007/978-3-319-27702-8_{_}25/COVER. adresse : https://link.springer.com/chapter/10.1007/978-3-319-27702-8_25.
- [83] T. ERSAL, Y. KIM, J. BRODERICK et al., « Keeping Ground Robots on the Move Through Battery and Mission Management, » *Mechanical Engineering*, t. 136, n° 6, p. 1-6, juin 2014. DOI : 10.1115/6.2014-JUN-4.
- [84] R. BELLMAN, « On the Theory of Dynamic Programming, » *Proceedings of the National Academy of Sciences of the United States of America*, t. 38, n° 8, p. 716-719, 1952. adresse : <http://www.jstor.org/stable/88493>.
- [85] S. G. TZAFESTAS, « Introduction to mobile robot control, » p. 687, 2014.
- [86] D. HSU, L. E. KAVRAKI, J.-C. LATOMBE, R. MOTWANI et S. SORKIN, « On finding narrow passages with probabilistic roadmap planners, » *books.google.com*, adresse : https://books.google.fr/books?hl=fr&lr=&id=ucO2DwAAQBAJ&oi=fnd&pg=PT151&dq=Hsu,+Kavraki,+Latombe,+Motwani,+%26+Sorkin,+1998&ots=t-iaQe8L_7&sig=g3igB6S-QfhemHy9_LecWFq_7YQ.
- [87] STEVEN M. LAVALLE, *Planning Algorithms*, CAMBRIDGE UNIVERSITY PRESS, éd. 2006.
- [88] A. GASPARETTO, P. BOSCARIOL, A. LANZUTTI et R. VIDONI, « Path planning and trajectory planning algorithms : A general overview, » *Mechanisms and Machine Science*, t. 29, p. 3-27, jan. 2015. DOI : 10.1007/978-3-319-14705-5_{_}1/

FIGURES/9. adresse : https://link.springer.com/chapter/10.1007/978-3-319-14705-5_1.

- [89] DE HOWIE CHOSET, KEVIN M. LYNCH, SETH HUTCHINSON, GEORGE A. KANTOR et WOLFRAM BURGARD, *Principles of Robot Motion : Theory, Algorithms, and Implementations*, MIT PRESS, éd. 2005.
- [90] J BRUCE, M. V. I. i. c. ON et u. 2002, « Real-time randomized path planning for robot navigation, » *ieeexplore.ieee.org*, adresse : <https://ieeexplore.ieee.org/abstract/document/1041624/>.
- [91] M. DE BERG, O. CHEONG, M. VAN KREVELD et M. OVERMARS, « Computational geometry : Algorithms and applications, » *Computational Geometry : Algorithms and Applications*, p. 1-386, 2008. DOI : 10.1007/978-3-540-77974-2.
- [92] H. CHOSET, « Coverage for robotics - A survey of recent results, » *Annals of Mathematics and Artificial Intelligence*, t. 31, n° 1-4, p. 113-126, 2001. DOI : 10.1023/A:1016639210559.
- [93] P RAJA, S. P. I. j. o. PHYSICAL et u. 2012, « Optimal path planning of mobile robots : A review, » *academicjournals.org*, t. 7, n° 9, p. 1314-1320, 2012. DOI : 10.5897/IJPS11.1745. adresse : <https://academicjournals.org/journal/IJPS/article-full-text-pdf/93D9A3918801.pdf>.
- [94] A. KOUBAA, H. BENNACEUR, I. CHAARI et al., « Robot Path Planning and Cooperation, » *Studies in Computational Intelligence*, t. 772, 2018. DOI : 10.1007/978-3-319-77042-0. adresse : <http://link.springer.com/10.1007/978-3-319-77042-0>.
- [95] J. CHOI, M. CHOI, S. Y. NAM et W. K. CHUNG, « Autonomous topological modeling of a home environment and topological localization using a sonar grid map, » *Autonomous Robots*, t. 30, n° 4, p. 351-368, mai 2011. DOI : 10.1007/S10514-011-9223-6.
- [96] S. C. WONG, « Qualitative Topological Coverage of Unknown Environments by Mobile Robots, » 2006.

- [97] B KUIPERS, Y. B. AAAI et u. 1988, « A Robust, Qualitative Method for Robot Spatial Learning., » *cs.utexas.edu*, adresse : <https://www.cs.utexas.edu/users/ai-lab/pubs/Kuipers+Byun-aaai-88.pdf>.
- [98] M. J. MATARIC, « Integration of Representation Into Goal-Driven Behavior-Based Robots, » *Routledge Library Editions : Artificial Intelligence*, t. 9-10, p. 165-186, juin 2021. DOI : 10.4324/9781351001885-5/INTEGRATION-REPRESENTATION-GOAL-DRIVEN-BEHAVIOR-BASED-ROBOTS-MAJA-MATARI. adresse : <https://www.taylorfrancis.com/chapters/edit/10.4324/9781351001885-5/integration-representation-goal-driven-behavior-based-robots-maja-matari%C4%87>.
- [99] S. T. A. INTELLIGENCE et u. 1998, « Learning metric-topological maps for indoor mobile robot navigation, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S0004370297000787>.
- [100] N TOMATIS, I NOURBAKHS, R. S. R. SYSTEMS, AUTONOMOUS et u. 2003, « Hybrid simultaneous localization and map building : a natural integration of topological and metric, » *Elsevier*, adresse : <https://www.sciencedirect.com/science/article/pii/S092188900300006X>.
- [101] G. KLANČAR, A. ZDEŠAR, S. BLAŽIČ et I. ŠKRJANC, *Chapter 7 Autonomous Guided Vehicles*. 2017, p. 502.
- [102] E. W. DIJKSTRA, « A note on two problems in connexion with graphs, » *Numerische Mathematik*, t. 1, n° 1, p. 269-271, déc. 1959. DOI : 10.1007/BF01386390/METRICS. adresse : <https://link.springer.com/article/10.1007/BF01386390>.
- [103] D. DELLING, P. SANDERS, D. SCHULTES et D. WAGNER, « Engineering route planning algorithms, » *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, t. 5515 LNCS, p. 117-139, 2009. DOI : 10.1007/978-3-642-02094-0_{_}7/COVER. adresse : https://link.springer.com/chapter/10.1007/978-3-642-02094-0_7.

- [104] P. E. HART, N. J. NILSSON et B. RAPHAEL, « A Formal Basis for the Heuristic Determination of Minimum Cost Paths, » *IEEE Transactions on Systems Science and Cybernetics*, t. 4, n° 2, p. 100-107, 1968. DOI : 10.1109/TSSC.1968.300136.
- [105] C. SOMMER, « Shortest-path queries in static networks, » *ACM Computing Surveys*, t. 46, n° 4, 2014. DOI : 10.1145/2530531.
- [106] M. SNIEDOVICH, « Dijkstra's algorithm revisited : the dynamic programming connexion, » *Control and Cybernetics*,
- [107] T. H. CORMEN, C. E. LEISERSON, R. L. RIVEST et C. STEIN, *Introduction to Algorithms - Selected Solutions*. 2009.
- [108] M. DIETZFELBINGER, « Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, Algorithms, McGraw Hill, Boston (2007), p. x+320, Paperback \$33.75, ISBN : 978-007352340-8 Jon Kleinberg, Éva Tardos, Algorithm Design, Pearson/Addison Wesley, Boston (2006), p. xxiii+838, Hardcover \$103, ISBN : 978-032129535-4, » *Computer Science Review*, t. 2, n° 2, p. 131-136, août 2008. DOI : 10.1016/J.COSREV.2008.03.001. adresse : <https://linkinghub.elsevier.com/retrieve/pii/S1574013708000130>.
- [109] C. ZHOU, B. HUANG et P. FRÄNTI, « A review of motion planning algorithms for intelligent robots, » *Journal of Intelligent Manufacturing*, t. 33, n° 2, p. 387-424, fév. 2022. DOI : 10.1007/S10845-021-01867-Z/TABLES/7. adresse : <https://link.springer.com/article/10.1007/s10845-021-01867-z>.
- [110] *Dijkstra's algorithm*.
- [111] A. STENTZ, « The D* Algorithm for Real-Time Planning of Optimal Traverses, » 1994.
- [112] D. FERGUSON et A. STENTZ, « Using Interpolation to Improve Path Planning : The Field D * Algorithm, » 2006. DOI : 10.1002/rob.20109. adresse : www.interscience.wiley.com.

- [113] K. DANIEL, A. NASH, S. KOENIG et A. FELNER, « Theta* : Any-Angle Path Planning on Grids, » *Journal Of Artificial Intelligence Research*, t. 39, p. 533-579, jan. 2014. DOI : 10.1613/jair.2994. adresse : <http://arxiv.org/abs/1401.3843><http://dx.doi.org/10.1613/jair.2994>.
- [114] O. KHATIB, « Real-time obstacle avoidance for manipulators and mobile robots, » *Proceedings - IEEE International Conference on Robotics and Automation*, p. 500-505, 1985. DOI : 10.1109/ROBOT.1985.1087247.
- [115] R. VOLPE et P. KHOSLA, « Manipulator Control with Superquadric Artificial Potential Functions : Theory and Experiments, » *IEEE Transactions on Systems, Man and Cybernetics*, t. 20, n° 6, p. 1423-1436, 1990. DOI : 10.1109/21.61211.
- [116] C. I. CONNOLLY, J. B. BURNS et R. WEISS, « Path planning using Laplace's equation, » p. 2102-2106, 1990. DOI : 10.1109/ROBOT.1990.126315.
- [117] J. O. KIM et P. K. KHOSLA, « Real-Time Obstacle Avoidance Using Harmonic Potential Functions, » *IEEE Transactions on Robotics and Automation*, t. 8, n° 3, p. 338-349, 1992. DOI : 10.1109/70.143352.
- [118] E. RIMON et D. E. KODITSCHKEK, « Exact Robot Navigation using Artificial Potential Functions, » *IEEE Transactions on Robotics and Automation*, t. 8, n° 5, p. 501-518, 1992. DOI : 10.1109/70.163777.
- [119] F. ARAMBULA COSÍO et M. A. PADILLA CASTAÑEDA, « Autonomous robot navigation using adaptive potential fields, » *Mathematical and Computer Modelling*, t. 40, n° 9-10, p. 1141-1156, nov. 2004. DOI : 10.1016/J.MCM.2004.05.001.
- [120] J. SFEIR, M. SAAD et H. SALIAH-HASSANE, « An improved Artificial Potential Field approach to real-time mobile robot path planning in an unknown environment, » *ROSE 2011 - IEEE International Symposium on Robotic and Sensors Environments, Proceedings*, p. 208-213, 2011. DOI : 10.1109/ROSE.2011.6058518.
- [121] D. H. KIM, « Escaping route method for a trap situation in local path planning, » *International Journal of Control, Automation and Systems 2009 7 :3*, t. 7, n° 3,

- p. 495-500, mai 2009. DOI : 10.1007/s12555-009-0320-7. adresse : <https://link.springer.com/article/10.1007/s12555-009-0320-7>.
- [122] K. SUGIHARA et J. SMITH, « Genetic algorithms for adaptive motion planning of an autonomous mobile robot, » *Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation, CIRA*, p. 138-143, 1997. DOI : 10.1109/CIRA.1997.613850.
- [123] R. YAN, S. J. DUNNETT et L. M. JACKSON, « Reliability Modelling of Automated Guided Vehicles by the Use of Failure Modes Effects and Criticality Analysis, and Fault Tree Analysis, » *DROPS-IDN/6514*, t. 50, p. 1-2, août 2016. DOI : 10.4230/OASICS.SCOR.2016.2.
- [124] G. CONFESSORE, M. FABIANO et G. LIOTTA, « A network flow based heuristic approach for optimising AGV movements, » *Journal of Intelligent Manufacturing*, t. 24, n° 2, p. 405-419, avr. 2013. DOI : 10.1007/s10845-011-0612-7/METRICS. adresse : <https://link.springer.com/article/10.1007/s10845-011-0612-7>.
- [125] H. FAZLOLLAHTABAR et M. SAIDI-MEHRABAD, « Optimising a multi-objective reliability assessment in multiple AGV manufacturing system, » *International Journal of Services and Operations Management*, t. 16, n° 3, p. 352-372, 2013. DOI : 10.1504/IJSOM.2013.056768.
- [126] L. HU, B. ZHOU et Y. LI, « An Energy Saving Scheduling Method for Just in Time Material Handling in Mixed-Model Assembly Line, » juill. 2017. DOI : 10.1115/MSEC2017-2985. adresse : /MSEC/proceedings-abstract/MSEC2017/50756/268701.
- [127] B. MRUGALSKA et R. STETTER, « Health-Aware Model-Predictive Control of a Cooperative AGV-Based Production System, » *Sensors 2019, Vol. 19, Page 532*, t. 19, n° 3, p. 532, jan. 2019. DOI : 10.3390/s19030532. adresse : <https://www.mdpi.com/1424-8220/19/3/532/htm><https://www.mdpi.com/1424-8220/19/3/532>.

- [128] X. YANG et C. WUSHAN, « AGV path planning based on smoothing A* algorithm, » *International Journal of Software Engineering & Applications (IJSEA)*, t. 6, n° 5, 2015. DOI : 10.5121/ijsea.2015.6501.
- [129] C. WANG, L. WANG, J. QIN et al., « Path planning of automated guided vehicles based on improved A-Star algorithm, » *2015 IEEE International Conference on Information and Automation, ICIA 2015 - In conjunction with 2015 IEEE International Conference on Automation and Logistics*, p. 2071-2076, sept. 2015. DOI : 10.1109/ICINFA.2015.7279630.
- [130] R PRASERTTAWHEELAP, S. K. I. J. I. C. . . . et u. 2019, « Safety path planning with obstacle avoidance using particle swarm optimization for AGV in manufacturing layout, » *ijicic.org*, t. 15, n° 1, p. 351-368, 2019. DOI : 10.24507/ijicic.15.01.351. adresse : <http://www.ijicic.org/ijicic-150123.pdf>.
- [131] Z. ZHANG, Q. GUO, J. CHEN et P. YUAN, « Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classification, » *IEEE Access*, t. 6, p. 26 022-26 035, mars 2018. DOI : 10.1109/ACCESS.2018.2819199.
- [132] K. C. VIVALDINI, J. P. GALDAMES, T. S. BUENO et al., « Robotic forklifts for intelligent warehouses : Routing, path planning, and auto-localization, » *Proceedings of the IEEE International Conference on Industrial Technology*, p. 1463-1468, 2010. DOI : 10.1109/ICIT.2010.5472487.
- [133] T XU, P YANG et H GUO, « Energy Efficiency Analysis on Robotic Mobile Fulfillment System, » in *2019 IEEE 6th International Conference on Industrial Engineering and Applications (ICIEA)*, 2019, p. 145-149. DOI : 10.1109/IEA.2019.8714923.
- [134] R. MCHANEY, « Modelling battery constraints in discrete event automated guided vehicle simulations, » <http://dx.doi.org/10.1080/00207549508904859>, t. 33, n° 11, p. 3023-3040, 2007. DOI : 10.1080/00207549508904859. adresse : <https://www.tandfonline.com/doi/abs/10.1080/00207549508904859>.
- [135] M. RAINERI, S. PERRI et C. GUARINO LO BIANCO, « Safety and efficiency management in LGV operated warehouses, » *Robotics and Computer-Integrated Manufacturing*, t. 57, p. 73-85, juin 2019. DOI : 10.1016/J.RCIM.2018.11.003.

- [136] J. VITTEK, P. BUTKO, B. FTOREK, P. MAKYŠ et L. GOREL, « Energy Near-Optimal Control Strategies for Industrial and Traction Drives with a.c. Motors, » *Mathematical Problems in Engineering*, t. 2017, 2017. DOI : 10.1155/2017/1857186.
- [137] H. ZHENG, R. R. NEGENBORN et G. LODEWIJKS, « Predictive path following with arrival time awareness for waterborne AGVs, » *Transportation Research Part C : Emerging Technologies*, t. 70, p. 214-237, sept. 2016. DOI : 10.1016/J.TRC.2015.11.004.
- [138] R. ABOUSLEIMAN et O. RAWASHDEH, « Energy-efficient routing for electric vehicles using metaheuristic optimization frameworks, » *Proceedings of the Mediterranean Electrotechnical Conference - MELECON*, p. 298-304, 2014. DOI : 10.1109/MELCON.2014.6820550.
- [139] R. ABOUSLEIMAN et O. RAWASHDEH, « Tabu search based solution to the electric vehicle energy efficient routing problem, » *2014 IEEE Transportation Electrification Conference and Expo : Components, Systems, and Power Electronics - From Technology to Business and Public Policy, ITEC 2014*, juill. 2014. DOI : 10.1109/ITEC.2014.6861787.
- [140] R. ABOUSLEIMAN et O. RAWASHDEH, « Electric vehicle modelling and energy-efficient routing using particle swarm optimisation, » *IET Intelligent Transport Systems*, t. 10, n° 2, p. 65-72, mars 2016. DOI : 10.1049/IET-ITS.2014.0177. adresse : <https://onlinelibrary.wiley.com/doi/full/10.1049/iet-its.2014.0177><https://onlinelibrary.wiley.com/doi/abs/10.1049/iet-its.2014.0177><https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/iet-its.2014.0177>.
- [141] A ARTMEIER, J HASELMAYR, M LEUCKER, M. S. KI et u. 2010, « The Shortest Path Problem Revisited : Optimal Routing for Electric Vehicles., » *Springer*, adresse : <https://link.springer.com/content/pdf/10.1007/978-3-642-16111-7.pdf#page=322>.
- [142] M. BAUM, J. DIBBELT, T. PAJOR et D. WAGNER, « Energy-optimal routes for electric vehicles, » *GIS : Proceedings of the ACM International Symposium on Advances in*

- Geographic Information Systems*, p. 54-63, 2013. DOI : 10.1145/2525314.2525361.
adresse : <https://dl.acm.org/doi/10.1145/2525314.2525361>.
- [143] D. DELLING, A. V. GOLDBERG, T. PAJOR et R. F. WERNECK, « Customizable route planning, » *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, t. 6630 LNCS, p. 376-387, 2011. DOI : 10.1007/978-3-642-20662-7{_}32/COVER. adresse : https://link.springer.com/chapter/10.1007/978-3-642-20662-7_32.
- [144] T. JURIK, A. CELA, R. HAMOUCHE et al., « Energy optimal real-time navigation system, » *IEEE Intelligent Transportation Systems Magazine*, t. 6, n° 3, p. 66-79, 2014. DOI : 10.1109/MITS.2014.2324000.
- [145] M. SACHENBACHER, M. LEUCKER, A. ARTMEIER et J. HASELMAYR, « Efficient Energy-Optimal Routing for Electric Vehicles, » *Proceedings of the AAAI Conference on Artificial Intelligence*, t. 25, n° 1, p. 1402-1407, août 2011. DOI : 10.1609/AAAI.V25I1.7803. adresse : <https://ojs.aaai.org/index.php/AAAI/article/view/7803>.
- [146] U. F. SIDDIQI, Y. SHIRAISHI et S. M. SAIT, « Multi-constrained route optimization for Electric Vehicles (EVs) using Particle Swarm Optimization (PSO), » *International Conference on Intelligent Systems Design and Applications, ISDA*, p. 391-396, 2011. DOI : 10.1109/ISDA.2011.6121687.
- [147] T. M. SWEDA et D. KLABJAN, « Finding minimum-cost paths for electric vehicles, » *2012 IEEE International Electric Vehicle Conference, IEVC 2012*, 2012. DOI : 10.1109/IEVC.2012.6183286.
- [148] Y. WANG, J. JIANG et T. MU, « Context-aware and energy-driven route optimization for fully electric vehicles via crowdsourcing, » *IEEE Transactions on Intelligent Transportation Systems*, t. 14, n° 3, p. 1331-1345, 2013. DOI : 10.1109/TITS.2013.2261064.
- [149] K. AL-MUTEB, M. FAISAL, M. EMADUDDIN et al., « An autonomous stereovision-based navigation system (ASNS) for mobile robots, » *Intelligent Service Robotics*, t. 9, n° 3, p. 187-205, juill. 2016. DOI : 10.1007/S11370-016-0194-5/TABLES/14.

adresse : <https://link.springer.com/article/10.1007/s11370-016-0194-5>.

- [150] H. CHE, Z. WU, R. KANG et C. YUN, « Global path planning for explosion-proof robot based on improved ant colony optimization, » *Proceedings of 2016 Asia-Pacific Conference on Intelligent Robot Systems, ACIRS 2016*, p. 36-40, août 2016. DOI : 10.1109/ACIRS.2016.7556184.
- [151] H. WANG, Y. YU et Q. YUAN, « Application of Dijkstra algorithm in robot path-planning, » *2011 2nd International Conference on Mechanic Automation and Control Engineering, MACE 2011 - Proceedings*, p. 1067-1069, 2011. DOI : 10.1109/MACE.2011.5987118.
- [152] Y. ZHANG, G. GUAN et X. PU, « The Robot Path Planning Based on Improved Artificial Fish Swarm Algorithm, » *Mathematical Problems in Engineering*, t. 2016, 2016. DOI : 10.1155/2016/3297585.
- [153] E. FATNASSI et J. CHAOUACHI, « Scheduling automated guided vehicle with battery constraints, » *2015 20th International Conference on Methods and Models in Automation and Robotics, MMAR 2015*, p. 1010-1015, sept. 2015. DOI : 10.1109/MMAR.2015.7284017.
- [154] S. DOGRU et L. MARQUES, « Towards fully autonomous energy efficient Coverage Path Planning for autonomous mobile robots on 3D terrain, » *2015 European Conference on Mobile Robots, ECMR 2015 - Proceedings*, nov. 2015. DOI : 10.1109/ECMR.2015.7324206.
- [155] K. J. KALIŃSKI et M. MAZUR, « Optimal control of 2-wheeled mobile robot at energy performance index, » *Mechanical Systems and Signal Processing*, t. 70-71, p. 373-386, mars 2016. DOI : 10.1016/J.YMSSP.2015.09.047.
- [156] S. LIU et D. SUN, « Minimizing energy consumption of wheeled mobile robots via optimal motion planning, » *IEEE/ASME Transactions on Mechatronics*, t. 19, n° 2, p. 401-411, 2014. DOI : 10.1109/TMECH.2013.2241777.

- [157] L. XIE, W. HERBERGER, W. XU et K. A. STOL, « Experimental validation of energy consumption model for the four-wheeled omnidirectional Mecanum robots for energy-optimal motion control, » *2016 IEEE 14th International Workshop on Advanced Motion Control, AMC 2016*, p. 565-572, juin 2016. DOI : 10.1109/AMC.2016.7496410.
- [158] P. A. PLONSKI, J. V. HOOK et V. ISLER, « Environment and Solar Map Construction for Solar-Powered Mobile Systems, » *IEEE Transactions on Robotics*, t. 32, n° 1, p. 70-82, fév. 2016. DOI : 10.1109/TRO.2015.2501924.
- [159] X. ZHAO, Z. SU et L. DOU, « A path planning method with minimum energy consumption for multi-joint mobile robot, » *Proceedings of the 33rd Chinese Control Conference, CCC 2014*, p. 8326-8330, sept. 2014. DOI : 10.1109/CHICC.2014.6896396.
- [160] T. WANG, B. WANG, H. WEI, Y. CAO, M. WANG et Z. SHAO, « Staying-alive and energy-efficient path planning for mobile robots, » *Proceedings of the American Control Conference*, p. 868-873, 2008. DOI : 10.1109/ACC.2008.4586602.
- [161] K. MAALEJ, S. KELOUWANI, K. AGBOSSOU, Y. DUBÉ et N. HENAO, « Long-Trip Optimal Energy Planning With Online Mass Estimation for Battery Electric Vehicles, » *IEEE Transactions on Vehicular Technology*, t. 64, n° 11, p. 4929-4941, nov. 2015. DOI : 10.1109/TVT.2014.2376700.
- [162] A. SADRPOUR, J. JIN et A. G. ULSOY, « Experimental validation of mission energy prediction model for unmanned ground vehicles, » *Proceedings of the American Control Conference*, p. 5960-5965, 2013. DOI : 10.1109/ACC.2013.6580773.
- [163] A. SADRPOUR, J. JIN et A. G. ULSOY, « Mission energy prediction for unmanned ground vehicles, » *Proceedings - IEEE International Conference on Robotics and Automation*, p. 2229-2234, 2012. DOI : 10.1109/ICRA.2012.6224860.
- [164] O. TRIGUI, Y. DUBE, S. KELOUWANI et K. AGBOSSOU, « Comparative Estimation of Electric Vehicle Rolling Resistance Coefficient in Winter Conditions, » *2016 IEEE Vehicle Power and Propulsion Conference, VPPC 2016 - Proceedings*, déc. 2016. DOI : 10.1109/VPPC.2016.7791630.

- [165] *Principles of Robot Motion : Theory, Algorithms, and Implementations.*
- [166] M. DE RYCK, M. VERSTEYHE et F. DEBROUWERE, « Automated guided vehicle systems, state-of-the-art control algorithms and techniques, » *Journal of Manufacturing Systems*, t. 54, p. 152-173, jan. 2020. DOI : 10.1016/J.JMSY.2019.12.002.
- [167] M. N. A. WAHAB, S. NEFTI-MEZIANI et A. ATYABI, « A comparative review on mobile robot path planning : Classical or meta-heuristic methods ? » *Annual Reviews in Control*, t. 50, p. 233-252, jan. 2020. DOI : 10.1016/J.ARCONTROL.2020.10.001.
- [168] G. DANTZIG, R. FULKERSON et S. JOHNSON, « Solution of a large-scale traveling-salesman problem, » *Journal of the operations research society of America*, t. 2, n° 4, p. 393-410, 1954.
- [169] R. E. BURKARD, V. G. DEINEKO, R. VAN DAL, J. A. A. van der VEEN et G. J. WOEGINGER, « Well-solvable special cases of the traveling salesman problem : a survey, » *SIAM review*, t. 40, n° 3, p. 496-546, 1998.
- [170] B. FLEISCHMANN, « A cutting plane procedure for the travelling salesman problem on road networks, » *European Journal of Operational Research*, t. 21, n° 3, p. 307-317, sept. 1985. DOI : 10.1016/0377-2217(85)90151-1.
- [171] K.-J. ROODBERGEN, *Layout and Routing Methods for Warehouses*. [ERIM, Erasmus Research Institute of Management], mai 2001.
- [172] H. D. RATLIFF et A. S. ROSENTHAL, « Order-picking in a rectangular warehouse : a solvable case of the traveling salesman problem, » *Operations research*, t. 31, n° 3, p. 507-521, 1983.
- [173] K. J. ROODBERGEN et R. DE KOSTER, « Routing order pickers in a warehouse with a middle aisle, » *European Journal of Operational Research*, t. 133, n° 1, p. 32-43, 2001.
- [174] M. LÖFFLER, N. BOYSEN et M. SCHNEIDER, « Picker routing in AGV-assisted order picking systems, » *INFORMS Journal on Computing*, 2021.

- [175] M. MASAE, C. H. GLOCK et P. VICHITKUNAKORN, « Optimal order picker routing in a conventional warehouse with two blocks and arbitrary starting and ending points of a tour, » *International Journal of Production Research*, t. 58, n° 17, p. 5337-5358, 2020. DOI : 10.1080/00207543.2020.1724342. adresse : <https://doi.org/10.1080/00207543.2020.1724342>.
- [176] H. CAMBAZARD et N. CATUSSE, « Fixed-parameter algorithms for rectilinear Steiner tree and rectilinear traveling salesman problem in the plane, » *European Journal of Operational Research*, t. 270, n° 2, p. 419-429, oct. 2018. DOI : 10.1016/J.EJOR.2018.03.042.
- [177] L. PANSART, N. CATUSSE et H. CAMBAZARD, « Exact algorithms for the order picking problem, » *Computers & Operations Research*, t. 100, p. 117-127, 2018. DOI : <https://doi.org/10.1016/j.cor.2018.07.002>. adresse : <https://www.sciencedirect.com/science/article/pii/S0305054818301862>.
- [178] Y. SU, M. LI, X. ZHU et C. LI, « Steiner TSP based on aisle as a unit for order picking, » *Computers & Industrial Engineering*, t. 168, p. 108 026, juin 2022. DOI : 10.1016/J.CIE.2022.108026. adresse : <https://linkinghub.elsevier.com/retrieve/pii/S0360835222000961>.
- [179] A. N. LETCHFORD, S. D. NASIRI et D. O. THEIS, « Compact formulations of the Steiner traveling salesman problem and related problems, » *European Journal of Operational Research*, t. 228, n° 1, p. 83-92, 2013.
- [180] A. SCHOLZ, S. HENN, M. STUHLMANN et G. WÄSCHER, « A new mathematical programming formulation for the single-picker routing problem, » *European Journal of Operational Research*, t. 253, n° 1, p. 68-84, 2016.
- [181] M. GOETSCHALCKX et H DONALD RATLIFF, « Order picking in an aisle, » *IIE transactions*, t. 20, n° 1, p. 53-62, 1988.
- [182] R. W. HALL, « Distance approximations for routing manual pickers in a warehouse, » *IIE transactions*, t. 25, n° 4, p. 76-87, 1993.

- [183] M. WANG, R.-Q. ZHANG et K. FAN, « Improving order-picking operation through efficient storage location assignment : A new approach, » *Computers & Industrial Engineering*, t. 139, p. 106-186, 2020. DOI : <https://doi.org/10.1016/j.cie.2019.106186>. adresse : <https://www.sciencedirect.com/science/article/pii/S0360835219306552>.
- [184] R. DE KOSTER et E. Van der POORT, « Routing orderpickers in a warehouse : a comparison between optimal and heuristic solutions, » *IIE transactions*, t. 30, n° 5, p. 469-480, 1998.
- [185] M. MASAE, C. H. GLOCK et E. H. GROSSE, « Order picker routing in warehouses : A systematic literature review, » *International Journal of Production Economics*, t. 224, p. 107-564, 2020. DOI : <https://doi.org/10.1016/j.ijpe.2019.107564>. adresse : <https://www.sciencedirect.com/science/article/pii/S0925527319304050>.
- [186] R. de KOSTER, T. LE-DUC et K. J. ROODBERGEN, « Design and control of warehouse order picking : A literature review, » *European Journal of Operational Research*, t. 182, n° 2, p. 481-501, 2007. DOI : <https://doi.org/10.1016/j.ejor.2006.07.009>. adresse : <https://www.sciencedirect.com/science/article/pii/S0377221706006473>.
- [187] M LÖFFLER, N BOYSEN et M SCHNEIDER, « Picker routing in AGV-assisted order picking systems, » rapp. tech., 2018.
- [188] P. CORTÉS, R. A. GÓMEZ-MONTOYA, J. MUÑUZURI et A. CORREA-ESPINAL, « A tabu search approach to solving the picking routing problem for large-and medium-size distribution centres considering the availability of inventory and K heterogeneous material handling equipment, » *Applied Soft Computing*, t. 53, p. 61-73, 2017.
- [189] S. ENE, KÜÇÜKOĞLU, A. AKSOY et N. ÖZTÜRK, « A genetic algorithm for minimizing energy consumption in warehouses, » *Energy*, t. 114, p. 973-980, 2016. DOI : <https://doi.org/10.1016/j.energy.2016.08.045>. adresse : <https://www.sciencedirect.com/science/article/pii/S0360544216311586>.

- [190] T. ROJANAPITON et J. TEERAVARAPRUG, « Economic Order Picker Routing Considering Travel Time and Vehicular Energy Consumption with Varying Aisle Traffic, »
- [191] T. ROJANAPITON et J. TEERAVARAPRUG, « A computer simulation for economical order picker routing when considering travel distance and vehicle energy consumption, » *International Journal of Engineering & Technology*, t. 7, n° 2.28, p. 33-37, mai 2018. DOI : 10.14419/ijet.v7i2.28.12878. adresse : <https://www.sciencepubco.com/index.php/ijet/article/view/12878>.
- [192] S. LEE, H. W. JEON, M. ISSABAKHSH et A. EBRAHIMI, « An electric forklift routing problem with battery charging and energy penalty constraints, » *Journal of Intelligent Manufacturing*, 2021. DOI : 10.1007/s10845-021-01763-6. adresse : <https://doi.org/10.1007/s10845-021-01763-6>.
- [193] P. A. MAKRIS, A. P. MAKRI et C. G. PROVATIDIS, « Energy-saving methodology for material handling applications, » *Applied energy*, t. 83, n° 10, p. 1116-1124, 2006.
- [194] R ELBERT et J. P. MÜLLER, « The impact of item weight on travel times in picker-to-parts order picking : An agent-based simulation approach, » in *2017 Winter Simulation Conference (WSC)*, 2017, p. 3162-3173. DOI : 10.1109/WSC.2017.8248035.
- [195] X. LI, C. CLARAMUNT et C. RAY, « A grid graph-based model for the analysis of 2D indoor spaces, » *Computers, Environment and Urban Systems*, t. 34, n° 6, p. 532-540, nov. 2010. DOI : 10.1016/J.COMPENVURBSYS.2010.07.006.
- [196] O. TRIGUI, E. MEJRI, Y. DUBE, S. KELOUWANI et K. AGBOSSOU, « Energy efficient routing estimation in electric vehicle with online rolling resistance estimation, » *2017 IEEE Vehicle Power and Propulsion Conference, VPPC 2017 - Proceedings*, t. 2018-January, p. 1-6, avr. 2018. DOI : 10.1109/VPPC.2017.8330912.
- [197] M. W. P. SAVELSBERGH et M. SOL, « The General Pickup and Delivery Problem, » <https://doi.org/10.1287/trsc.29.1.17>, t. 29, n° 1, p. 17-29, fév. 1995. DOI : 10.1287/TRSC.29.1.17. adresse : <https://pubsonline.informs.org/doi/abs/10.1287/trsc.29.1.17>.

- [198] R. RAJAMANI, « Longitudinal Vehicle Dynamics, » p. 87-111, 2012. DOI : 10.1007/978-1-4614-1433-9_{_}4. adresse : https://link.springer.com/chapter/10.1007/978-1-4614-1433-9_4.
- [199] K. AZADEH, R. DE KOSTER et D. ROY, « Robotized and automated warehouse systems : Review and recent developments, » *Transportation Science*, t. 53, n° 4, p. 917-945, 2019.
- [200] F. BULUT et H. M. EROL, « A Real-Time Dynamic Route Control Approach on Google Maps using Integer Programming Methods, » *INTERNATIONAL JOURNAL OF NEXT-GENERATION COMPUTING*, t. 9, n° 3, nov. 2018. DOI : 10.47164/IJNGC.V9I3.426. adresse : <http://perpetualinnovation.net/ojs/index.php/ijngc/article/view/426>.
- [201] SUNITA et D. GARG, « Dynamizing Dijkstra : A solution to dynamic shortest path problem through retroactive priority queue, » *Journal of King Saud University - Computer and Information Sciences*, t. 33, n° 3, p. 364-373, mars 2021. DOI : 10.1016/J.JKSUCI.2018.03.003.
- [202] M. S. C. CYBERNETICS, et u. 2006, « Dijkstra's algorithm revisited : the dynamic programming connexion, » *eudml.org*, adresse : <https://eudml.org/serve/209437/accessibleLayeredPdf/0>.
- [203] A. RAITH et M. EHRGOTT, « A comparison of solution strategies for biobjective shortest path problems, » *Computers & Operations Research*, t. 36, n° 4, p. 1299-1331, avr. 2009. DOI : 10.1016/J.COR.2008.02.002.
- [204] M BARBEHENN, « A note on the complexity of Dijkstra's algorithm for graphs with weighted vertices, » *IEEE Transactions on Computers*, t. 47, n° 2, p. 263, 1998. DOI : 10.1109/12.663776.
- [205] O. BAWDEN, D. BALL, J. KULK, T. PEREZ et R. RUSSELL, « A lightweight, modular robotic vehicle for the sustainable intensification of agriculture, » 2014.

- [206] N. SHETTY, B. SAH et S. H. CHUNG, « Route optimization for warehouse order picking operations via vehicle routing and simulation, » *SN Applied Sciences*, t. 2, n° 2, p. 1-18, 2020.
- [207] G. DUKIC et T. OPETUK, « Warehouse layouts, » *Springer 4*, t. 9781447122746, p. 55-69, 2012. DOI : 10.1007/978-1-4471-2274-6{_}3/TABLES/1. adresse : https://link.springer.com/chapter/10.1007/978-1-4471-2274-6_3.
- [208] L. M. POHL, R. D. MELLER et K. R. GUE, « An analysis of dual-command operations in common warehouse designs, » *Transportation Research Part E : Logistics and Transportation Review*, t. 45, n° 3, p. 367-379, mai 2009. DOI : 10.1016/J.TRE.2008.09.010.
- [209] K. R. GUE et R. D. MELLER, « Aisle configurations for unit-load warehouses, » *IIE Transactions (Institute of Industrial Engineers)*, t. 41, n° 3, p. 171-182, 2009. DOI : 10.1080/07408170802112726.
- [210] M. FUSKO, M. RAKYTA et F. MANLIG, « Reducing of Intralogistics Costs of Spare Parts and Material of Implementation Digitization in Maintenance, » *Procedia Engineering*, t. 192, p. 213-218, jan. 2017. DOI : 10.1016/J.PROENG.2017.06.037.
- [211] S. G. OZDEN, A. E. SMITH et K. R. GUE, « A novel approach for modeling order picking paths, » *Naval Research Logistics (NRL)*, t. 68, n° 4, p. 471-484, juin 2021. DOI : 10.1002/NAV.21966. adresse : <https://onlinelibrary.wiley.com/doi/full/10.1002/nav.21966><https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.21966><https://onlinelibrary.wiley.com/doi/10.1002/nav.21966>.
- [212] J. ZHANG, X. WANG et K. HUANG, « Integrated on-line scheduling of order batching and delivery under B2C e-commerce, » *Computers & Industrial Engineering*, t. 94, p. 280-289, 2016. DOI : <https://doi.org/10.1016/j.cie.2016.02.001>. adresse : <https://www.sciencedirect.com/science/article/pii/S0360835216300171>.

- [213] G. TARCZYŃSKI, « Analysis of the impact of storage parameters and the size of orders on the choice of the method for routing order picking, » *Operations Research and Decisions*, t. 22, n° 4, p. 105-120, 2012.
- [214] R. DEKKER, M. B. DE KOSTER, K. J. ROODBERGEN et H. VAN KALLEVEEN, « Improving Order-Picking Response Time at Ankor's Warehouse, » <https://doi.org/10.1287/inte.1040.0083>, t. 34, n° 4, p. 303-313, août 2004. DOI : 10.1287/INTE.1040.0083. adresse : <https://pubsonline.informs.org/doi/abs/10.1287/inte.1040.0083>.

Annexe A - Articles publiés

Titre : Energy Efficient Order Picking Routing for a Pick Support Automated Guided Vehicle (Ps-AGV)

Journal : IEEE Access

Auteurs : Emna Mejri ; Sousso Kelouwani ; Yves Dubé ; Nilson Henao ; and Kodjo Agbossou

Date de publication : 06 October 2022

DOI : [10.1109/ACCESS.2022.3212797](https://ieeexplore.ieee.org/document/9913474) <https://ieeexplore.ieee.org/document/9913474>

Titre : Energy Efficient Path Planning for Low Speed Autonomous Electric Vehicle

Journal : IEEE Conference on Vehicle Power and Propulsion (VPPC)

Auteurs : Emna Mejri ; Sousso Kelouwani ; Yves Dubé ; Omar Trigui ; and Kodjo Agbossou

Date de publication : 05 April 2018

DOI : [10.1109/VPPC.2017.8330908](https://ieeexplore.ieee.org/document/8330908) <https://ieeexplore.ieee.org/document/8330908>

RESEARCH ARTICLE

Energy Efficient Order Picking Routing for a Pick Support Automated Guided Vehicle (Ps-AGV)

EMNA MEJRI¹, (Member, IEEE), SOUSSO KELOUWANI¹, (Senior Member, IEEE),
YVES DUBÉ¹, NILSON HENAO², AND KODJO AGBOSSOU², (Senior Member, IEEE)

¹Department of Mechanical Engineering, Hydrogen Research Institute, University of Quebec at Trois-Rivières, Trois-Rivières, QC G8Z 4M3, Canada

²Department of Electrical and Computer Engineering, Hydrogen Research Institute, University of Quebec at Trois-Rivières, Trois-Rivières, QC G8Z 4M3, Canada

Corresponding author: Emna Mejri (emna.mejri@uqtr.ca)

This work was supported in part by the Foundation of Université du Québec à Trois-Rivières and Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG).

ABSTRACT Order picker routing refers to the process of collecting a set of products with the minimum travel time. Recently, a new generation of Automated Guided Vehicles (AGVs) has been developed to assist human order pickers in order to minimize their travel time. These vehicles are using battery as energy source. However, the routing energy efficiency aspect of these systems remains unexplored. Yet any improvement in power consumption will ultimately reduce the DOD (depth of discharge) of the battery and increase its lifespan. For example, in many real AGV applications incorporating the effect of load mass has been neglected, although its importance. In most studies, the methodology proposed for the order picking routing problem does not allow neither the integration of the mass of each Stock Keeping Unit (SKU) nor the calculation of associated energy costs. Those studies are generally limited to ensure that all the items requested by an order are picked up with minimum travel time/distance. In this paper, an Energy Efficient Order Picking Routing algorithm named *EE-OPR* is proposed to realize an efficient AGV tour with an acceptable trade-off between energy preservation and travel time minimization. The proposed approach takes into account the mass of loads and its accumulation throughout the pick tour since it intensifies the rolling resistance losses on flat ground, especially at lower speeds. In this regard, an optimization method by means of dynamic states graph is developed. This method is applied to different warehouse layouts. The performance of the suggested algorithm is evaluated by comparing it with an approach minimizing only travel time consumption. Results show that the optimized tours, offered by *EE-OPR* are effective and robust, with an 18% average saving on the total cost of picking tour.

INDEX TERMS Case picking, dynamic shortest path problem, energy efficient routing, load weights, order picking problem, pick support AGV (PS-AGV).

I. INTRODUCTION

With the rise of customization, the rapid growth of e-commerce, and labor shortage, the level of automation in warehouses and Distribution Centers (DCs) tends to increase in order to meet market requirements [1]. Picking the orders in some warehouses and DCs is done by Automated Guided Vehicles (AGVs) to minimize the pickers' unproductive walking time and improve the picking efficiency and ergonomics in picker-to-parts setup. These systems are

referred to as Pick Support Autonomous Guided Vehicles (PS-AGVs), AGV-assisted order picking systems, or simply Autonomous Mobile Robots (AMRs) [2]. An example of such systems is implemented by Fetch Robotics to optimize Case Picking [3]. Some of the existing research works focus on the economic aspects of PS-AGVs application. These works aim to study and provide PS-AGVs solutions that help in the business growth, match their solutions' operations and deliver a significant return on investment (ROI) [4], [5]. For instance, a crucial factor that is often overlooked in the literature and should be taken into account when using PS-AGVs is the energy cost of the vehicle exploitation as well as the cost

The associate editor coordinating the review of this manuscript and approving it for publication was Adamu Murtala Zungeru¹.

of battery maintenance and its exchange, especially when the battery is dead. This factor is critical for the adoption of AGVs and in particular, for the improvement of the deployment cost of this technology. In addition, the energy available on battery constraint is generally a limiting factor for the range and length of an AGV deployment [6]. Moreover, since an AGV uses several batteries throughout its life, the total cost of these batteries is significant [7].

Material handling consumes a notable amount of energy in warehouses [8]. Although electrification and green warehousing are receiving increasing attention due to environmental awareness [9], [10], additional efforts in terms of operational decisions should be considered. The number of operational decisions related to PS-AGV routing per day can notably influence the amount of required energy, particularly when AGVs are used in pallets or cases picking. These types of AGVs are known as order picking trucks or upgraded traditional forklifts [5]. It should be noted that forklifts are known as the most energy-consuming material handling equipment used in warehouses [11].

In order to enhance the energy efficiency of an AGV, its energy requirements for order picking must be studied. The process of collecting a set of products with the minimum cost is called Order Picking Problem (OPP), Picker Routing Problem (PRP), or picking problem for short [12]. This problem is the most challenging concern related to warehousing operations since it takes about 50% to 75% of the total operating cost associated with labor and time [13], [14]. Most of the current research works aim to optimize the routing time/distance of the order picking tour [2], [15] while ignoring the energetic aspect. However, for a better choice, it is essential to understand and adequately quantify the required energy of different possible paths to improve the order picking routing efficiency. In addition to the distance traveled and the time spent, the energy consumption of an AGV depends on many factors, such as its speed, weight, and the transported cargo quantity. Hence, the higher the payload weighs, the more power an AGV requires [16].

Therefore, it is crucial to take into account the different weights of requested items when planning a picking tour. It is also important to note that warehouses and DCs manage a large assortment of Stock Keeping Units (SKUs) (in terms of size and weight) on variable schedules [1], [17]. Often, PS-AGV has to deal daily with different quantities and heavy and bulky loads of a wide variety of items with varying requirements. However, they are not limited to this kind of item [5]. In addition, the positions of the products may change in the DC or may be replaced by a different product. This variation of the type of items and their masses has an impact on the required energy to complete a picking tour. In the case of an unexpected increase in the picking tour due to the growth in mass, the recharge time has to be updated to avoid aggressive discharge and extend the battery lifespan. In this work, the importance of having a precise estimation of the vehicle energy consumption considering cargo weights is highlighted to make better routing decisions.

Therefore, improving the energy efficiency of AGVs extends their productive operating time between recharging stops (autonomy). This energy improvement can also increase the number of picked items per day, resulting in short-term savings because the cost of electricity has decreased and also due to the autonomy of AGVs. Moreover, any improvement that reduces power consumption results in the reduction of the DOD of the battery and thus improves its lifespan [18], [19]. In addition, these improvements help with long-term savings by reducing the cost of maintenance and (or) battery change and the number of robots required on the floor, providing better autonomy per vehicle.

Considering the increased interest in the concept of green warehousing and its respective potential for energy saving, the main focus of this work is to answer the following question: Given an order sheet, start, and endpoints, how can we optimize the AGV's routing energy and time cost in order to collect all the requested items from different listed positions in the warehouse while minimizing energy and time simultaneously? It should be noted that in some settings, the energy-saving path can be longer than the travel-time saving. If a slightly longer trip can save energy, it could be interesting for long-term savings, especially for a large warehouse. Therefore, it is interesting to reach a trade-off between time and energy savings. Besides, it is required to specify whether time has high priority in a particular context, like having too many requirements in specific periods. Hence, this paper is concerned with the routing problem of an AGV in a bi-dimensional way to manage time and energy requirements for order picking movement in warehouses.

The rest of the paper is organized as follows. Section II provides the literature review. The methodology is explained in Section III. Section IV presents the empirical results and Section V discusses the benefits and the limitations of the proposed approach. Concluding remarks and some suggestions for future research directions are provided in Section VI.

II. LITERATURE REVIEW

A. PS-AGV ROUTING

1) ORDER PICKING PROBLEM DEFINITION

The single PS-AGV routing problem deals with the determination of the path which has to be traveled by the PS-AGV in order to collect a set of items requested by internal or external customers, in a distribution warehouse. This well-known problem is referred to as Order Picking Problem (OPP) in rectangular warehouses. Exact and heuristic methods have been widely utilized to deal with OPP in the literature. OPP is represented as a special case of the classical Traveling Salesman Problem (TSP), where the salesman is the AGV, and cities are items to collect [20]. This framework is about Steiner TSP (STSP) as not all cities are required but some of them (given the specific layout and location of items in a particular structure) [21]. The STSP is described by a directed graph $G = (V, E)$ where V is a set of vertices and E is a set of edges. In this graph, $P \subseteq V$ represents the required vertices and $V \setminus P$ depicts Steiner points. A Steiner tour of G is a closed walk that visits each vertex of P at least once. Therefore, there

are two differences between a Steiner tour and a traveling salesman tour. The first difference is that in a Steiner tour, the Steiner points do not have to be visited. The second difference is that a Steiner tour may contain some vertices more than once [22]. In the classical STSP, the goal is to minimize the length of a Steiner tour in the digraph G . In our study instead, we aim to find the Steiner tour minimizing energy and time consumption.

2) ROUTING POLICIES FOR ORDER PICKING OPERATIONS

In 1983, Ratliff and Rosenthal (RR) [23] presented a polynomial-time solution for an exact order picking strategy in a single-block warehouse. The method proposed by RR was extended by Roodbergen and De Koster for a two-block warehouse [24]. Subsequently, Löffler et al. and Masae et al. [15], [25] dealt with order picking problems with arbitrary starting and ending points by applying the concepts of RR. A new solution to the routing problem was suggested by Scholz et al. [26] based on new mathematical formulations in order to take into account the specificity of the warehouse layout. Their formulation has a main constraint that consists of imposing the picking of one unit on the first pass over a required vertex. However, if the energy consumption is considered, that constraint can result in a sub-optimal solution. Pansart et al. [12] present two exact algorithms for OPP. In the first algorithm, they demonstrate that the problem can be solved optimally with Mixed Integer Linear Programming (MILP) using a sparse formulation strengthened by pre-processing and valid inequalities. The problem is seen as a STSP and the authors used a compact single-commodity flow formulation which has been proposed by Letchford et al. [27]. Thus, the picker has to deposit a unit of items each time he picks one (flow principle). It is important to mention here that the direction of traversal is crucial in this setting yet the post-processing step to find the picking tour sequence from the resulting tour sub-graph is not detailed. In their second algorithm, Pansart et al. [12] proposed a dynamic programming approach extending RR known algorithms from two cross-aisles to any number of cross-aisles to deal with real-life applications. Nevertheless, this method cannot accommodate side constraints such as flow directions and precedence. References [12], [28] Also, heuristics methods have been proposed in the literature for the same purpose: traversal (or S-shape), largest-gap, return, midpoint, and composite [29], [30]. Heuristics are mainly used for solving OPP since the optimal route may seem illogical to a human operator [31]. However, in the case of a semi-autonomous or completely autonomous system where the robot is the leader, it is no longer a problem. Koster and Poort [32] present a practical comparison between exact algorithms and heuristics by comparing the S-shape method with dynamic programming. In the S-shape strategy, aisles are fully traveled if there are products to pick, otherwise, aisles are skipped. For example, when the picker starts from the lower-left corner of the warehouse (depot) and enters and leaves aisles from different sides (front and rear), it returns to the depot after finishing picking resulting in an S-shape

route. Koster and Poort conclude that despite the ease of use of the S-shape strategy, the optimal algorithms bring better savings in travel time. This result motivates the use of exact algorithms. For more details, Masae et al. [33] present a systematic literature review about order picker routing in warehouses.

Note that the S-shape routing is the most used method in PS-AGV systems. This method is near-optimal when there is an excessive pick density [14]. Löffler et al. [15] extend the RR algorithm for the problem of picking single order with given start and end locations. They also present an adaptation of the S-shape and gap strategies to fit with AGV-assisted order picking where start and end points could be different and are not limited to the depot location. However, this method, as well as other heuristics, are less suited to accommodate side constraints such as the energy consumption and the mass of the transported load.

It is important to mention that the main and common focus of the aforementioned studies is the travel time and/or distance. However, neither energy consumption nor the effect of masses on energy consumption have been considered and investigated.

B. AGV ENERGY PERFORMANCE

Due to environmental concerns, much research has been conducted on the energy performance of autonomous material handling vehicles in order to manage the greening process in factories and warehouses [34]. Some researchers have focused on the development of decision support tools for the selection of the type of material handling vehicles (such as Liquefied petroleum gas, diesel, or electric) in order to minimize the environmental impact of warehouse activities [35], [36]. Other research directions aimed to assess the factors that influence the energy needs of AGVs and to model these needs [37], [38]. Some researchers focused on charging optimization [39], [40] while others on improving the energy efficiency of mobile robots through motion planning and control [41], [42], [43], [44]. Finally, another interesting research works aim to improve the energy efficiency through routing decisions. Since the main goal of our work is to solve the order picking routing problem of an AGV in warehouses and DCs in an energy efficient way, we start by presenting some research studies that consider the minimization of energy consumption for the resolution of the OPP. Then, we present some other related works that focus mainly on energy efficient routing in the context of order picking.

1) ENERGY EFFICIENT ROUTING: RELATED WORKS

In the context of flexible manufacturing, Barak et al. [45] proposed an approach to modeling operation scheduling, machine allocation, and AGV scheduling while minimizing energy consumption. In particular, they used an adapted multi-objective particle swarm optimization method that takes into account both distance and load. Zhang et al. [46] proposed a path planning for a single-load AGV in a factory that efficiently uses energy. Some other research works

consider energy efficiency for Robotic Mobile Fulfillment Systems (RMFS), such as Li et al. [47], Xu et al. [48], and Zhou and Zhu [49]. Unlike the picker-to-part system, in RMFS, it is the items that move and not the workers in the factory. The RMFS is typically arranged in a grid with storage zones of inventory pods, picking stations, and replenishment stations. Robots lift and carry square shelving units called inventory pods with items from storage locations to replenishment or picking stations. Under such a setting, the workers can fill or pick items from inventory pods. The research works described above, consider the routing energy efficiency of different types of AGVs, some of which consider the effect of the mass. However, these works deal with problems different from ours.

In addition, in the transport field, PS-AGV routing is strongly related to the vehicle routing problem along with the pickup and delivery problems. In particular, PS-AGV routing can be considered as a generalization of the TSP. The classical vehicle routing problem consists of finding the best route defined for rather a fleet to reduce transport costs. Continue within the framework of green logistics, some studies have been carried out to reduce the consumption of energy for the vehicle routing problem. For example, a novel load-based cost objective for energy minimizing vehicle routing problem is proposed by Kara et al. [50]. The problem is presented as Capacitated Vehicle Routing Problem (CVRP) and defined using integer linear programming formulations for the delivery and collection cases.

2) ENERGY EFFICIENT ROUTING FOR ORDER PICKING

As mentioned in Section II-A2, the goal of most of the approaches related to OPP in the literature is to reduce the travel time and/or distance while overlooking the environmental performance of the warehouse [35], [51]. However, there are some studies that attempt to find a trade-off between travel time and energy consumption minimization in order to optimize order picking routing. For instance, Ene et al. [52] developed a genetic algorithm for order picking problem in warehouses. This algorithm aims to minimize the service time as well as energy consumption via order batching and picking routing optimization. The authors proved by examples the significant energy saving when applying their approach. However, their work is based only on the vertical and horizontal speed and the traveled distance of the forklift to estimate energy consumption. Also, they assume a constant energy consumption per unit time, overlooking load and resistance forces in their calculation. Rojanapitoon and Teeravarapug [53] introduced a new mathematical model for picker routing that minimizes the travel time and energy consumption given a variation of the level of traffic in a rectangular warehouse. Their mathematical model is then used on computer simulation software which has been presented in a previous work [54]. They compare their results with the time-staged model, and they validate them by the brute-force search strategy. Compared to the method that optimizes time, the authors reported that their proposed model optimizes both time and energy and saves up to 17% energy. However, they

excluded the possibility of picking up the requested item on the second pass (if there is any) and overlooked the direction of travel in the construction of a complete picking tour which can lead to sub-optimal solutions. Lee et al. [55] developed an integrated dynamic algorithm as a solution to the electric forklift routing problem w.r.t battery charging constraints. That is, the algorithm considers the electric forklift's picking/put-away routes and battery charging schedules along with the number of electric forklifts. In addition to that, their algorithm takes into account the consumption of electricity in the warehouse. Also, Makris et al. [56] addressed the OPP from an energy efficiency point of view. They present a TSP-based routing algorithm in order to achieve a trade-off between travel time and energy consumption of order picking in the warehouse. The authors, however, exclude the weight from their energy consumption evaluation. Similarly, most of the research works do not consider mass as a critical factor for vehicle routing decisions, especially when considering the energy saving. Elbert and Müller [57] investigate the impact of transported item weight on the velocity of the order picker and travel time in a manual picker-to-parts order picking. In their work, they focus on the problem of storage assignment and propose new weight class-based storage assignment policies to reduce travel time.

Therefore, two important points are overlooked in the aforementioned research studies, which are considered in this study. We discuss these two points in what follows. First, in the majority of research papers related to the resolution of the OPP, the applied constraints in the mathematical formulation either force the solution to pass only once on each required picking location or to take the products from the order on the first pass. These formulations lead to an optimal or near-optimal result which minimizes the travel time of order picking tour. However, considering the energy gain in the optimization formulation was not considered. This gain can be obtained by passing through a required point without picking up the load (so as not to drag it) and by picking it up on the way back to save energy. Figure 1a shows an example of considering the energy gain by not lifting the required item in the first pass. In this illustration, we suppose a bidirectional graph with a set of vertices X, Y, Z and a set of edges $(X, Y), (Y, Z)$. The wavy arrows represent the shortest paths between two nodes. The OPP here attempts to start from point X , to visit points Y and Z , and then to return to point X with minimal cost. In this example, the resulting shortest path requires the passage twice on the same node Y . Solution 1 does not consider the energy gain and hence carries the load in Y all the way from Y to Z , and then from Z to Y . However, Solution 2, which takes into consideration the energy gain, first picks the load in Z , and on the way back, it picks the load in Y . Second, most of the discussed works focus on the search of an optimal sub-tour and take the construction of a full tour for granted. Nevertheless, a change of direction of the same sub-tour can bring an energy gain if the loss of energy linked to the movement of heavy objects is delayed. Figure 1b gives an example of saving energy by changing the direction. We assume in this figure a different bidirectional graph with three vertices

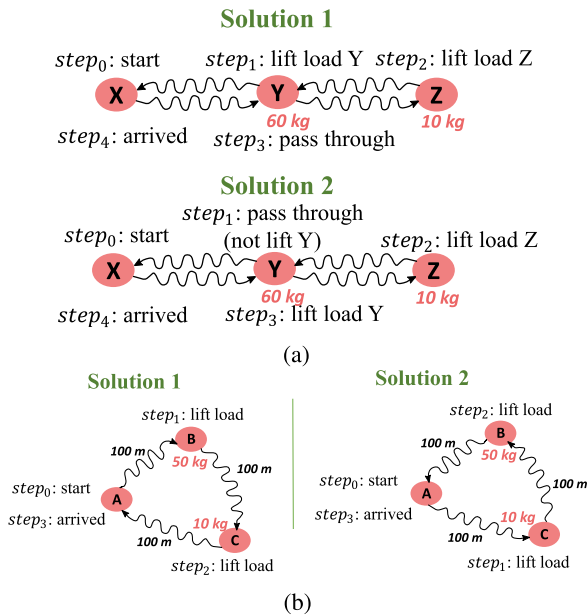


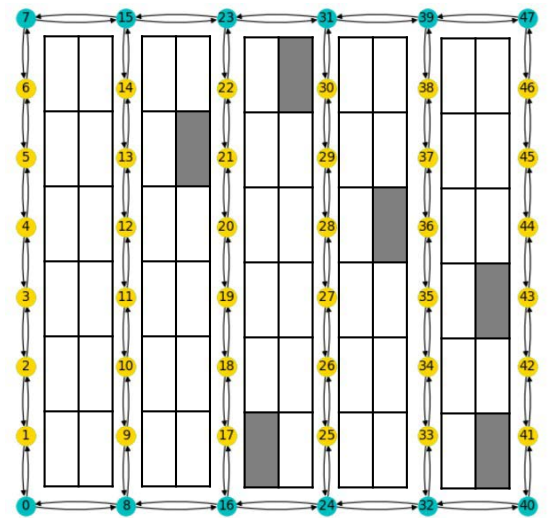
FIGURE 1. Concept of considering energy as well as time for picking loads in a warehouse. (a) illustrates an example of considering energy gain by not lifting the required item in the first pass (applied by Solution 2) while Solution 1 does not. (b) shows an example of saving energy by changing the direction by which the items are picked up. Solution 2 takes into account this saving while Solution 1 does not (B is heavier than C).

A, B, C, and three edges (A, B), (B, C) and, (C, A). Here, the OPP attempts to start from point A, picks up the loads from points B and C and returns back to point A with minimal cost. According to this example, the load in B is heavier than C. Solution 1 does not consider the change of direction and therefore, it first picks the heavier load and carries it all the way to C and goes back to A. However, Solution 2 first picks the lighter load in C. Then, it picks the heavier one in B and goes back to A. This change of direction can lead to energy savings.

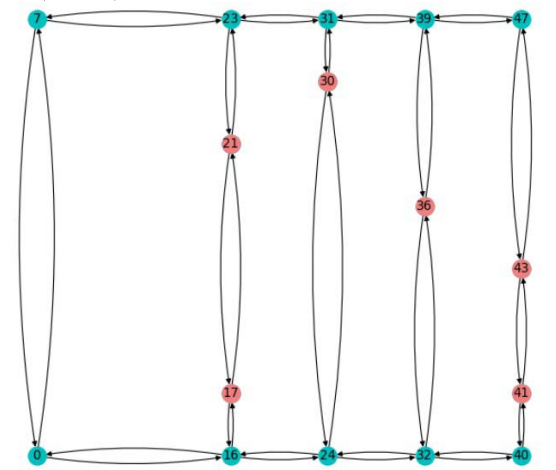
Hence, the main contribution of this work is to tackle these two not-well-studied points in the literature. Thus, in our approach called *Energy Efficient Order Picking Routing (EE-OPR)*, we consider time, energy, and mass in the planning decision in order to achieve an efficient order picking tour for an autonomous material handling vehicle. Moreover, unlike classical RR-based approaches which are mainly based on the particular structure of rectangular parallel-aisle warehouses for creating the subproblems of a dynamic program, the proposed approach can be used for other warehouses with different arrangements and layouts. In fact, in order to solve the OPP while simultaneously minimizing time and energy, a dynamic program is developed. This program considers the problem as an eco-energetic STSP and transforms it into a shortest path problem (SPP) by creating an acyclic dynamic state graph and performing a graph search process.

III. METHODOLOGY

In this section, we present the methodology we followed for *EE-OPR*.



(a) Layout of the warehouse and the graph representation $G_0(V_0, A_0)$



(b) Graph $G(V, A)$ associated with the order picking routing problem

FIGURE 2. Layout of the warehouse and graph representation.

A. WAREHOUSE LAYOUT REPRESENTATION

A conventional single-block parallel-aisle warehouse with single depot is considered as the classic configuration. Such a warehouse consists of g vertical aisles and 2 horizontal cross-aisles. Figure 2 presents an example of this case with six aisles and two cross-aisles. Aisles contain products on both sides, while cross-aisles make intersections through which the AGV can navigate. The warehouse structure can be described by the graph $G_0(V_0, A_0)$, where $V_0 = \{v_0, \dots, v_k\}$ is a set of k vertices (yellow and blue circles in Figure 2a) and A_0 presents a set of arcs denoting connections between vertices. In order to facilitate the selection of pick-list elements from their locations, the main set of vertices V_0 is divided into two subsets, denoted by V_I and V_L . V_I defines intersections vertices in cross-aisles (blue circles in Figure 2). V_L stands for other vertices, which account for possible picking locations (yellow circles in Figure 2a). An order given by a pick-list can be specified by another set $V_P \subseteq V_L$ with cardinality $p \geq 1$. This

subset contains vertices associated with p cases, which are described by their SKU and location in the warehouse (pink vertices in Figure 2b). This pick-list contains the order lines of a single customer (pick-by-order) or multiple customers (pick-by-batch). In the rest of the article, we use the term ‘case’ to present the total number of items or batches that are picked at the same pick location. In addition, we designate these cases by the vertex located in the corresponding position. For example, the picking location of the case i is expressed by v_i . Furthermore, the masses of the cases are given by the set $Y = \{m_1, \dots, m_p\}$ in which each element m_i represents the mass related to the case i , in kilogram (kg). Also, m_0 denotes the mass of an AGV without load.

The OPP is considered as an eco-energetic STSP that is stated in a directed graph, $G(V, A)$. Figure 2b presents an example of this graph with 16 vertices from which 6 are required. $G(V, A)$ is a sub-graph of $G_0(V_0, A_0)$, shown in Figure 2a, and contains only relevant locations. These locations account for Steiner, V_I , and pick-list, V_p , vertices and are represented by the set $V = \{v_0, \dots, v_n\}$. Consequently, the set A presents the arcs that connect adjacent vertices. The weight of each arc corresponds to the Euclidean distance cost, d_{ij} , between the i^{th} and the j^{th} vertices, connected by an arc. Accordingly, the picking routing problem can be determined as optimizing the AGV tour to collect all products in the pick-list by minimizing its time and energy consumption while traveling from initial to target position (depot or other predefined positions). The optimal tour also takes into account the mass of pick-list elements.

It can be noticed that the objective is to find an eco-energetic tour not necessarily a Hamiltonian one given that it is an eco-energetic Steiner TSP problem. This allows vertices and edges to be traversed more than once, if desired. In order to achieve this target, a new scheme is proposed that is applicable to any warehouse layout. *EE-OPR* is based on the creation of dynamic states through the bit-Masking method [58]. Additionally, it takes advantage of the dynamized Dijkstra algorithm for the graph search [59]. *EE-OPR* considers the energy consumption of the AGV and the effect of the transported cargo weight.

B. ASSUMPTIONS

In this section, we list the assumptions considered in *EE-OPR*.

- Only the energy consumption of the AGV travel movement is considered. The time and energy waste for picking stops can be assumed to be constant or negligible. Moreover, depending on the type of the robot, if it has to lift itself cases, the energy loss due to this work is not taken into account in the mathematical modeling. This is because the energy demand to overcome the gravity of the cases is almost the same and does not affect the routing decision.
- The minimization of the total travel time is equivalent to the minimization of the total tour length, given that the vehicle can move uniformly (constant speed (V) in both coordinates axis). Therefore, the energy consumption

caused by the process of acceleration and deceleration is not considered.

- The definition of the pick-list is done beforehand so that the energy onboard is sufficient to finish the tour and that the volume and the mass of the cases do not exceed the AGV’s capacity Q .
- The mass of all SKUs is available and by knowing the quantity required in each pick location, the mass of each case can be computed.
- The order picker continues to pick up all required items at the same storage location when the AGV stops for picking.
- The aisles are wide enough to allow two-way travel of AGVs. This assumption can be modified for parallel closed-end picking aisles by simply modifying the graph edges with unidirectional arcs representing the aisles and bidirectional arcs for edges of cross aisles.
- The depot location is displayed in the lower left-corner of the warehouse for simplicity. However, this assumption can easily be changed for any layout.
- A single AGV is considered in this case study. This can be easily extended to accommodate multiple AGVs.
- We assume that the robot can pass more than once on the same vertex during a picking tour.

C. DYNAMIC GRAPH CREATION

STSP is a combinatorial optimization process that can be formulated as a shortest path problem [58]. The solution to this problem is a path with the minimum cost that starts from the initial position with zero load and ends at the final position with all requested loads. *EE-OPR* is based on the dynamic creation of a state graph incrementally. Therefore, a new graph $G'(V', A')$, called state graph, is created in addition to the spatial graph $G(V, A)$. The set V' represents vertices states and the set A' defines arcs between two successive states. $G'(V', A')$ is weighted by a cost function considering the previous state and direct transition costs. This function determines the cost of time and energy that is required to travel the distance d_{ij} with a load m . Besides, the resolution of the shortest path problem is processed by means of the Bellman principle. These notions are detailed in what follows.

1) STATE VERTEX DEFINITION

As a particular case of TSP, the STSP can use the philosophy of bitmask arrangement emphasizing that each data is represented by a binary digit (0 or 1) with a particular permutation (bitmask) [58]. Let p vertices be the elements of a pick-list. For this list, the tuple $\mathbf{a}_i = (a_0 a_1 \dots a_p)$ with p binary digits is created with the same order. This tuple defines the cases that have been picked at a certain time. It presents them by the binary value 1 at their corresponding location. For instance, $\mathbf{a}_i = (0 1 \dots 0)$ shows that the second case of the pick-list has been picked by the robot. As a result, $\mathbf{a}_i = (1 1 \dots 1)$ signifies that all cases have been lifted. The total number of arrangements for \mathbf{a}_i is 2^p . Since the problem is presented in an incomplete network graph and only a subset of ‘visiting vertices’ [22] is covered by the tour, the vertex information

is also required to formulate the picking problem. In fact, the robot can be positioned in any vertex $v_i \in V$ (required or not) and transport 0 to p cases. Subsequently, spatial and temporal information is paired and referred to as states. A state is represented as a tuple $s_k = (v_i, a_j)$ in which v_i presents the vertex and a_j expresses the arrangement. The spatial complexity of the state representation is $n2^p$, where n is the cardinality of V . Generally, the overall execution time of a TSP based on the bitmask Dynamic Programming (DP) with n cities to visit is $O(n^2 * 2^n)$ [58]. Therefore, the resolution of Steiner TSP using DP bitmask is $O(n^2 * 2^p)$. Nevertheless, the time complexity of our work can decrease considering the sparsity of the warehouse graph and the particularity of the Steiner points. This advantage can be realized by *EE-OPR* which is capable of creating only necessary states and edges.

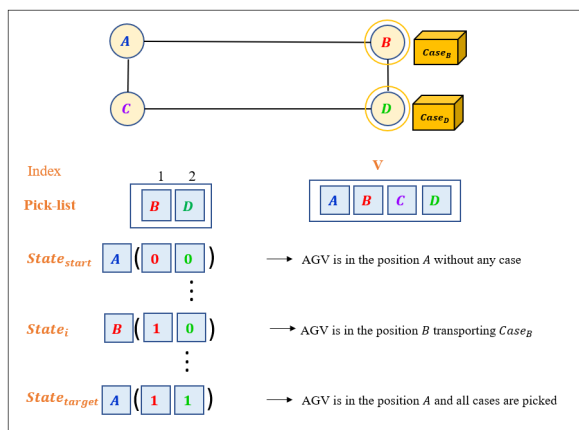


FIGURE 3. Explanatory example of the creation of states.

From this perspective, the targeted case can be presented in terms of a shortest path problem where $s_{start} = (v_{start}, a_{start})$ and $s_{target} = (v_{target}, a_{target})$ are starting and ending states, respectively. For these states, v_{start} and v_{target} stand for starting and ending vertices. In addition, $a_{start} = (a_1, a_2, \dots, a_p)$ where $a_i = 0 \forall i \in \{1, \dots, p\}$ and $a_{target} = (a_1, a_2, \dots, a_p)$ where $a_i = 1 \forall i \in \{1, \dots, p\}$ define starting arrangement and target arrangement, respectively. In the rest of the paper, we will refer to starting arrangement and target arrangement as ‘empty arrangement’ and ‘full arrangement’, respectively. Figure 3 simply illustrates the state creation process. In this Figure, the main graph includes 4 vertices namely: A, B, C, and D, where B and D are required. Given A as a starting and ending point at the same time and $P = \{B, D\}$ as a pick-list, the starting and target states can be expressed as (A, 00) and (A, 11), respectively. In order to avoid creating all possible states, a new Directed Acyclic Graph (DAG), G' , is considered. This dynamic graph is initiated by the start-state vertex. Afterward, it is expanded by state vertices and transition arcs based on the spatial graph G , the exploration, and the current load arrangement.

2) TRANSITION ARC DEFINITION

During the picking turn, the robot can move between vertices transporting 0 to p cases. This movement corresponds to a

transition between two states that is represented by an arc $a \in A'$. The adjacency matrix $n2^p \times n2^p$ can be used to define possible transitions over the entire states. However, this matrix includes impossible transitions and unattainable states. These unacceptable circumstances are:

- loading more than one item at the same time,
- decreasing load in the tour,
- turning on a bit in the arrangement (zero \rightarrow one) when its corresponding vertex does not exist in the pick-list,
- activating a bit in the arrangement (zero \rightarrow one) when its position does not relate to that of the vertex in the pick-list.

Consequently, *EE-OPR* is employed to generate the states and arcs of G' in a dynamic way. Additionally, the edge relaxation technique is used to update the paths associated with the existing states. The generation process is done during traversing the spatial graph G (the exploration phase).

3) STATE VERTICES AND TRANSITION ARCS CREATION

According to the above discussion, a state can be presented by several attributes consisting of name (v_i, a) , vertex v , arrangement a , mass m , predecessor $state_{i-1}$, and cost c (related to travelling from the starting vertex). The exploration phase involves the creation of states and transition arcs while passing through the graph G . At each step, a transition between two states is realized by a change of position (vertex) and an update of the arrangement (if possible). The vertex change can occur just between two adjacent vertices, $v_i, v_j \in V : (v_i, v_j) \in A$. Besides, changing the arrangement (state of loads) follows certain logic and there are two alternatives, described below.

- a_1 : the arrangement remains the same (Not carrying v_j).
- a_2 : the arrangement is updated by turning on the bit corresponding to v_j (carrying v_j) (see Figure 4).

By pairing the arrangements a_1 and a_2 with a neighbor vertex, two possible states can be created based on the following.

- (v_j, a_1) : This state expresses the AGV movement to the next vertex v_j with the same load.
- (v_j, a_2) : This state designates an AGV motion towards the next vertex v_j , with its current load and picking load v_j .

If v_j belongs to the pick-list, both possibilities are allowed. Otherwise, only the first case is permitted. Note that only one bit can change at every arrangement update.

Figure 5 illustrates an example of the creation of successor states for the state (A, 00) in the graph of Figure 3. It can be deduced that the neighbor B is an item on the pick-list since both possibilities are demonstrated. The choices are explained by a tuple in which the digit related to the neighbor vertex can be 0 (state₁) or 1 (state₂). In this figure, the ‘arrangement update’ refers to the masking process, described in Figure 4.

The state creation procedure is described in algorithm 1 and algorithm 2. The first algorithm involves the function *CreateState*₁. It takes as input the current state (its vertex u , arrangement a , mass m , and current cost J^*) as well as

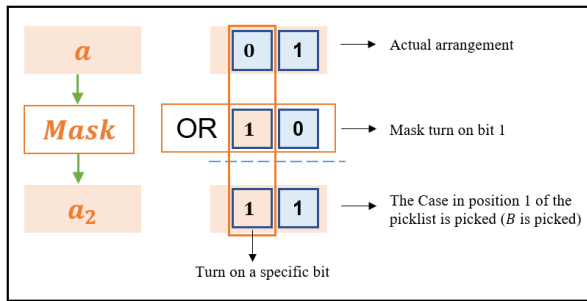


FIGURE 4. Applying a mask to turn on the bit corresponding to the requested vertex.

the neighbor spatial vertex $v \in V$. As output, it generates a tuple $state_1 = (v, a)$ that pairs the successor vertex v and the same arrangement and mass and the cumulative cost to move from the current state to $state_1$ using *transitionCost* function defined in the next subsection. Furthermore, the predecessor attribute is set to the current state.

Algorithm 1: CreateState1 (s^*, v)

```

 $u \leftarrow s^*.vertex; m \leftarrow s^*.mass; a \leftarrow s^*.arr; J^* \leftarrow s^*.cost;$ 
 $d \leftarrow dist(u, v)$ 
 $c \leftarrow J^* + Transition\_cost(d, m)$ 
 $s_1.vertex \leftarrow v$ 
 $s_1.arrangement \leftarrow a$ 
 $s_1.name \leftarrow (v, a)$ 
 $s_1.cost \leftarrow c$ 
 $s_1.mass \leftarrow m$ 
 $s_1.predecessor \leftarrow s^*$ 
Return  $s_1$ 
    
```

The second algorithm involves the function *CreateState2* which takes the same input information plus the pick-list. The determination of y , the position/index of v in the pick-list is required. Then, a mask is applied to the current arrangement to give a new arrangement a_2 . This mask is a binary number composed of p bits set all to zero except the y -th bit which is set to one. a_2 is then paired to the successor vertex v to create $state_2 = (v, a_2)$. In addition, a new mass m_2 is calculated which is the sum of the current mass and the mass associated with the vertex v . The cost of $state_2$ is calculated using the current cost J^* and the *transitionCost* function. This latter utilizes the current vertex u , the successor vertex v , and m_2 . These two functions are used in the main *EE-OPR* algorithm.

4) COST OF THE TRANSITION IN TERMS OF TIME AND ENERGY

By defining vertices states and transition edges, a new states graph $G'(V', A')$ is created. This graph (which is a DAG) starts from starting vertex $s_{start} = (v_{start}, a_{start})$ and branches out state vertices connected by transition arcs. The graph is weighted according to the criterion to be minimized (time only, energy only or both time and energy). We will call the methods that solve the OPP considering only the travel time, Travel Time Minimization (TTM) and those that consider

Algorithm 2: CreateState2 (s^*, v, V_p)

```

 $u \leftarrow s^*.vertex; m \leftarrow s^*.mass; a \leftarrow s^*.arr; J^* \leftarrow s^*.cost;$ 
 $d \leftarrow dist(u, v)$ 
 $y \leftarrow V_p.index(v)$ 
 $a_2 \leftarrow aORMask(y)$ 
 $m_2 \leftarrow m + M(y)$ 
 $c_2 \leftarrow J^* + Transition\_cost(d, m_2)$ 
 $s_2.vertex \leftarrow v$ 
 $s_2.arrangement \leftarrow a_2$ 
 $s_2.name \leftarrow (v, a_2)$ 
 $s_2.cost \leftarrow c_2$ 
 $s_2.mass \leftarrow m_2$ 
 $s_2.predecessor \leftarrow s^*$ 
Return  $s_2$ 
    
```

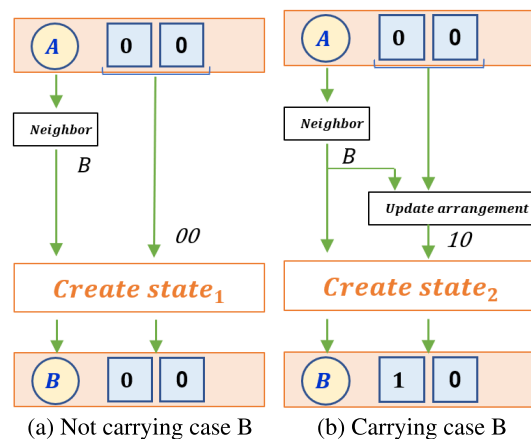


FIGURE 5. Two alternatives to carry or not a case in a required vertex: Create both $state_1$ and $state_2$.

only the energy, Energy Consumption Minimization (ECM). Let us define the model of the time required to move from state i to state j (t_{ij}) as follows:

$$t_{ij} = d_{(i.vertex, j.vertex)} / V \tag{1}$$

The computational function of the energy demand to move from state i to state j (e_{ij}) is defined based on the vehicle's longitudinal dynamics when moving along any arc (ij)

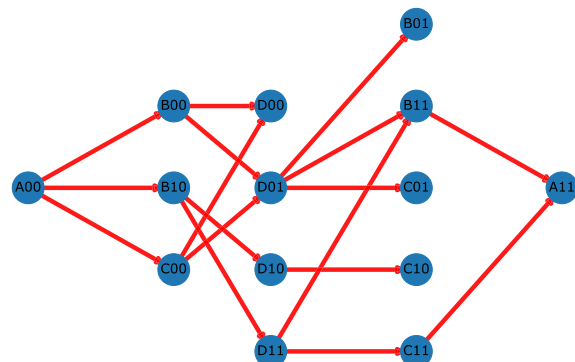


FIGURE 6. Dynamic states graph G' .

represented by:

$$F_T = M\dot{V} + \frac{1}{2}\rho AC_x (V - v_w)^2 + M g \mu \cos(\theta) + Mg \sin(\theta) \quad (2)$$

where F_T , M , V , ρ , A , C_x , v_w , g , μ and θ stand for the traction force at wheels, the vehicle total mass, the speed, the air density, the active aerodynamic surface of the vehicle, the drag coefficient, the wind speed, the gravity constant, the rolling resistance and the ground slop, respectively. Since most warehouses have flat ground, we have considered θ as null [46]. In addition, the wind speed in a warehouse can be neglected. Furthermore, since the vehicle is similar when moving along arcs, the aerodynamic force $\frac{1}{2}\rho AC_x (V - v_w)^2$ is similar on every arc. Moreover, the acceleration is limited to avoid sudden motion, jerk and tip over. Therefore, considering that AGV's velocity is constant, friction is the major external force applied to the AGV in an indoor context which is given by:

$$\hat{F}_T = Mg\mu \quad (3)$$

Assume that: M_i^R and M_i^L are the total masses of the AGV when it reaches vertex i .vertex and it leaves it, respectively. The total mass in kg, includes its own mass m_0 , the carrying SKUs (cases), and the picker mass (if it is driven to travel). For instance, if the case (load) in i .vertex is picked when the AGV reaches this vertex; $M_i^L = M_i^R + M[y_i]$ (current mass plus the mass of the case in position y_i of the pick-list). Otherwise, the mass remains the same; $M_i^L = M_i^R$. Therefore, the mass when leaving starting state $M_{s_{start}}^L$ is m_0 . Hence, the mechanical energy in translation from state i to j (e_{ij}) is given by \hat{F}_T multiplied by the distance between the position of i and the position of j as follows:

$$e_{ij} = (M_i^L g \mu) d(i.vertex, j.vertex) \quad (4)$$

It is important to mention that minimizing only e_{ij} (like ECMs) can lead to longer and more time-consuming tours than the shortest ones (given by TTMs). For instance, let's assume a directed graph C_n with 4 vertices A , C , D , and E and 5 weighed edges $(AC, 6)$, $(AD, 10)$, $(CD, 8)$, and $(ED, 6)$. The OPP consists of starting from A , picking C and D and then going to E . Assume that the masses m_0 of the AGV, m_C (in C) and m_D (in D) are: $100kg$, $80kg$, and $1kg$, respectively, and assume that $\epsilon = \mu g$:

- A path that consumes the minimum amount of time is: $P_1 = \{(AC), (CD), (DE)\}$ with a length of $20m$ and requiring $312.6J$ to execute the order picking tour ($E_1 = \epsilon[d_{AC}m_0 + d_{CD}(m_0 + m_C) + d_{DE}(m_0 + m_C + m_D)]$).
- A path that consumes the minimum amount of energy is: $P_2 = \{(AD), (DC), (CE)\}$ with a length of $24m$ and requiring $289.4J$ to execute the order picking tour ($E_2 = \epsilon[d_{AD}m_0 + d_{DC}(m_0 + m_D) + d_{CE}(m_0 + m_D + m_C)]$).

Thus, it is clear that P_2 overcomes P_1 in terms of energy. However, P_2 consumes more time than P_1 . It is therefore important to take into account the two criteria (time and energy) in order to reach a good trade-off between them and

this is what EE-OPR aims for. Since t_{ij} and e_{ij} have different units, let us define c_{ij} , the total cost in dollars to move from i to j , as follows:

$$c_{ij} = c_t(t_{ij}) + c_e(e_{ij}) \quad (5)$$

where c_t and c_e are two coefficients representing the time cost in dollar per second and the energy cost in dollar per joule, respectively. These two coefficients define the importance of each criterion. For example, in case of a large number of orders in a warehouse, the time factor is crucial and thus, it is recommended to increase the cost in dollars.

Algorithm 3: EE-OPR

```

initialization: set  $s_{start}$ 
 $s_{start}.mass \leftarrow M_0$ ;
 $cost[s_{start}] \leftarrow 0$ ;
 $pred[s_{start}] \leftarrow \text{undefined}$ 
 $Q \leftarrow [(0, s_{start})]$ 
while  $Q$  is not empty do
   $s^* \leftarrow \text{Extract-Min}(Q)$ 
   $u \leftarrow s^*.vertex$ ;
   $a \leftarrow s^*.arrangement$ ;
  for all  $v \in \Gamma_u^+$  do
    if  $v == s_{target}.vertex$   $a \neq s_{target}.arrangement$ 
      then
        | continue
      end
     $S_1 \leftarrow \text{CreateState1}(s^*, v)$ 
    if  $S_1 \in Q$  then
      | Relax( $s^*$ ,  $S_1$ )
    else
      |  $Q.insert(S_1)$ 
    end
     $cost[S_1] = S_1.cost$ 
     $pred[S_1] = S_1.pred$ 
    if  $v \in V_P$  then
      |  $S_2 \leftarrow \text{CreateState2}(s^*, v)$ 
      | if  $S_2 \in Q$  then
        | Relax( $s^*$ ,  $S_2$ )
      | else
        |  $Q.insert(S_2)$ 
      | end
      |  $cost[S_2] = S_2.cost$ 
      |  $pred[S_2] = S_2.pred$ 
    end
  end
end

```

D. GRAPH SEARCH PROBLEM

In addition to the construction of the states graph, the cost associated to each state is updated following Bellman's Principle of Optimality [60]. This technique is inspired by dynamized Dijkstra's algorithm [61] which is considered as a dynamic programming successive approximation procedure. Considering the graph $G'(V', A')$, each arc $(i, j) \in A'$ is

weighted by $c_{ij} \in \mathbb{R}_{\geq 0}$. Suppose that $s_{start}, s_{target} \in V'$ are the source and the destination vertices, respectively. We define by X a directed path of a sequence of vertices v_i such that: $X = \langle v'_1, \dots, v'_m \rangle$, where $v'_1 = s_{start}, v'_m = s_{target}, (v_k, v_{k+1}) \in A', \forall k \in \{1, \dots, m-1\}$. $c(X)$ is the path's cost that is, the sum of arcs' costs in X ; $c(X) = \sum_{i=1}^{m-1} c_{v_i, v_{i+1}}$. Obviously, the state vertices which constitute the resulting path are visited at most once. We specify that we are talking about state vertices and not the spatial vertices belonging to V . This means that we can pass twice on a vertex but with different arrangements.

Given the fact that eco-energetic STSP is seen as a problem of the shortest path between the state s_{start} and the state s_{target} , the following standard integer programming formulation is presented [62] (shortest path here designates optimal cost's path):

$$\text{Minimize } \sum_{(i,j) \in A'} (c_i(t_{ij}) + c_e(e_{ij}))x_{ij} \quad (6)$$

$$\text{subject to: } \sum_{j \in \Gamma_i^+} x_{ij} - \sum_{j \in \Gamma_i^-} x_{ji} = \quad (7)$$

$$\begin{cases} 1 & \text{if } i = s_{start} \\ 0 & \text{if } i \neq s_{start}, s_{target} \\ -1 & \text{if } i = s_{target} \end{cases} \quad (8)$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A' \quad (9)$$

where i and j denote states vertices, Γ_i^+ and Γ_i^- designate the sets of successor and predecessor vertices, respectively, x_{ij} represents the decision variable defining whether arc (ij) is part of the shortest path or not. It takes the value 1 in the first case and 0, otherwise. Constraints (1) and (4) specify that for each vertex that belongs to the shortest path, different from the start state and target state, must have the same number of incoming and outgoing arcs. In order to solve the presented eco-energetic STSP, *EE-OPR* is proposed. It follows the following steps: Create a queue of priority Q and initialize it with the start state with a cost of 0. While this latter is not empty, select the state with minimal cost, consider it as a current state and delete it from the priority queue. Then, define its composition (which vertex (u) and arrangement (a) it is composed of). Explore the vertex (u) by determining the neighboring vertices. These vertices will be used to create $state_1$ and $state_2$, and to add them to the queue if they don't already exist. Otherwise, check if the current computed cost can be decreased (optimized) by going to the next state through the current state. This is described by the RELAX function 4. In parallel, the cost list and the pred list are created to save the added states with their costs and predecessors, respectively. An update of these costs and predecessors can be done using the RELAX function which tries to minimize the cost of the paths. The final cost of the target state represents the optimal cost of the picking tour. In addition, the (pred list) allows to trace the optimal path from the target states arriving to the start state. Note that *EE-OPR* prevents movement to the target vertex without having retrieved all the items in the pick-list. This is done during the exploration phase by the prohibition of the creation of states and transitions when the neighboring

vertex is the target vertex but the arrangement is different from the '*full arrangement*' (see the first if statement in the algorithm).

Therefore, if the target vertex is different from the start vertex, the only state that uses this vertex is the state target (the total number of possible states is thus less than $(n-1)2^p + 1$. Otherwise, if the depot position represents both the starting point and the ending point, the two possible states in this position are the states having the '*empty arrangement*' (at the start) or the '*full arrangement*' (at the end). The total number of possible states is $(n-1)2^p + 2$. Consequently, only arcs leaving the start state and arcs entering the destination state are kept. Moreover, in case a neighboring vertex v of a current vertex u belongs to the pick-list and the current arrangement is already a_2 (the bit corresponding to v in the arrangement is equal to one). In this case, $state_2$ is equal to $state_1$. Hence, only $state_1$ is added.

Algorithm 4: RELAX (s^*, st)

```

 $m \leftarrow s^*.mass;$ 
 $d \leftarrow dist(s^*.vertex, st.vertex)$ 
 $w(s^*, st) \leftarrow s^*.cost + Transition\_cost(d, m)$ 
if  $st.cost > s^*.cost + w(s^*, st)$  then
  |  $st.cost \leftarrow s^*.cost + w(s^*, st)$ 
end

```

Now, we will discuss the time complexity of *EE-OPR*. As the priority queue Q is represented as a binary heap, where operations are performed in $O(\log(q))$ time (where q is the size of Q), the time complexity of *EE-OPR* is measured as follows. The time taken for each extract-min operation is $O(\log|V'|)$. Moreover, iterating over all vertices' neighbors and updating their *dist* values is executed a total of $O(|V'|)$ times and each vertex priority update takes $O(\log|V'|)$ time. Consequently, the total computational cost of both calculations takes $O(|V'| \times \log|V'|)$ time [63].

As a result, the overall time complexity of *EE-OPR* is $O(|V'| \times \log|V'|)$. Given that the maximum number of states that can be created is less than $n2^p$, then the overall time complexity of *EE-OPR* is $O(n2^p \times \log(n2^p))$.

IV. EMPIRICAL STUDIES

A. WAREHOUSE LAYOUT

As shown in Figure 7, we assume a single block warehouse with g aisles, for each of which l horizontal picking positions are considered. This warehouse graph is chosen for the evaluation of our proposed method as it is the most common structure studied in the literature. However, we note that *EE-OPR* can be adopted to any other warehouse configuration.

B. COMPUTATIONAL RESULTS

In this section, we explain how the simulations are performed. The implementation is done using an Intel Xeon W-2102 computer with 128 GB DDR4 RAM and we used Python as a programming language.

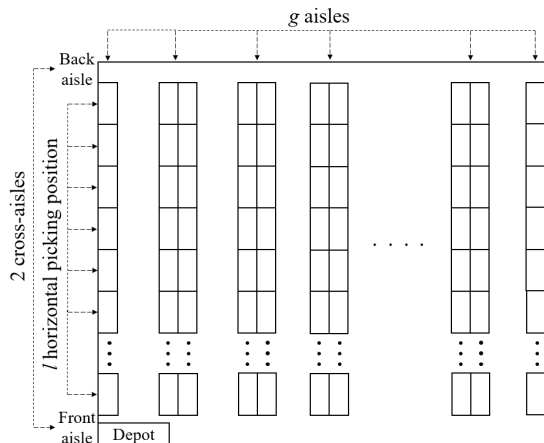


FIGURE 7. The warehouse layout considered for the case studies.

The effectiveness of *EE-OPR* is evaluated and compared with a method that we call Travel Time Minimizing approach (TTM). TTM proposed by Letchford et al. [27] is an approach that aims to only minimize the travel time of selection tours. It is based on a compact single-commodity flow formulation. TTM is also used by Pansart et al. [12] for solving a mixed-integer linear program. The latter uses only time criterion t_{ij} in its objective function ($z^* = \min \sum_{(i,j) \in A} (t_{ij} x_{ij})$). This technique is used to generate the shortest pickup tour for each instance. Moreover, in order to compare it with *EE-OPR*, the sum of the cost of each arc of the tour sequence resulting from the two methods is computed using the c_{ij} function 5, giving the cost in dollars of executing each path (since TTM does not use c_{ij} as cost function, the cost in dollars of TTM resulting tours is calculated after performing the TTM using c_{ij}). It is important to mention that TTM can not integrate the energy consumption since e_{ij} is a function of M_i^L which is not defined by TTM according to its formulation. On the other hand, TTM reflects all exact algorithms that aim to optimize time or distance without considering the energy aspect through dynamic programming formulation that is not suitable for adding additional constraints [12]. More specifically, to evaluate the performance of our approach, we consider random demand scenarios with uniform demand distribution throughout the warehouse with random storage policies. This assessment is realized through the variation of:

- the size of the pick-list,
- the required pick locations in the warehouse (pick-list),
- the masses of items to be picked up in different positions and,
- the shape of the warehouse.

As shown in Figure 7, the simulations are generated for four different configurations: Layout 1= (40 × 50), Layout 2= (25 × 80), Layout 3= (20 × 100), and Layout 4= (10 × 200). The distances between the aisles and the pick locations are 4 meters and 1 meter, respectively for all the setups. The choice of the warehouse shape is inspired by [31]. Each structure is explored under five scenarios with different pick-list sizes namely, 8, 10, 12, 14, and 16, respectively. For

every scenario, 100 picking tours are simulated with arbitrary locations and masses in order to provide reliable statistical analysis. For each tour, a selection list of cases is chosen at random using a uniform distribution. In practice, cases can represent batches that group several orders in a pick-list to be separated in the packing station [31]. Cases in a selected list have various masses that are also generated at random between 10 kg and M_{max} kg (maximum vehicle capacity). The sum of the masses of the pick-list items should not exceed the maximum vehicle capacity as follows:

$$m_{i,j}^s \sim \text{Unif}(10, M_{max}) \tag{10}$$

where s, i, j represent the scenario, instance (picking tour), and pick-up location, respectively. The simulation performed in the present work is carried out for an AGV with a maximum speed of 1.2 m/s, a weight of 1600 Kg, and a maximum supported load of 1200 kg.

Figure 8 presents the results of the comparative study of our approach and the TTM method based on the mean and variance of the cost of each case. As shown in Figure 8, *EE-OPR* overcomes TTM in all the twenty settings (four layouts each of which having five different scenarios). In particular, the effectiveness of *EE-OPR* increases by moving towards the fifth scenario for all layouts when compared to TTM. In other words, we note a lower increase in cost as the size of the pick-list increases. Figure 8 shows an increase in the difference between the mean values of the tour cost in the scenarios of each layout. Such a result is achieved due to the *EE-OPR* ability to minimize the energy, which tends to increase for larger pick-lists with higher possibility of mass accumulation. Moreover, we observe a lower variance of the routing cost in *EE-OPR* by comparison to TTM, especially for scenarios 4 and 5 (with higher number of items). Thus, *EE-OPR* is less sensitive to variations in pick-list location and mass. Hence, the results prove that minimizing energy and time simultaneously can effectively decrease the cost of a picking tour.

We also notice that there are situations where TTM and our approach lead to the exact same result. This is logical and it is explained by the fact that our focus is on the effect of mass on energy consumption and routing decision making. Thus, if the mass variation of the items is low or zero, minimizing the energy consumed amounts to minimizing the distance covered. On the other hand, if the situation of passing over a position twice is not present and the direction of the traversal tour is luckily the same, the advantage given by our approach is no longer valid.

To better illustrate how our approach overcomes TTM, an example of an order picking problem of a simple graph is used (Figure 9). The graph is composed of 20 vertices of which 4 are required. These required vertices represent the pick-list vertices and are represented in red circles in the graphs. The required vertices are 18, 19, 15, and 16 which have the masses 100, 10, 70, and 20, respectively.

To solve this picking problem, TTM (Figure 9,a) and *EE-OPR* (Figure 9,b) are applied. The resulting tours of each method are plotted in the figure using red arrows. Note

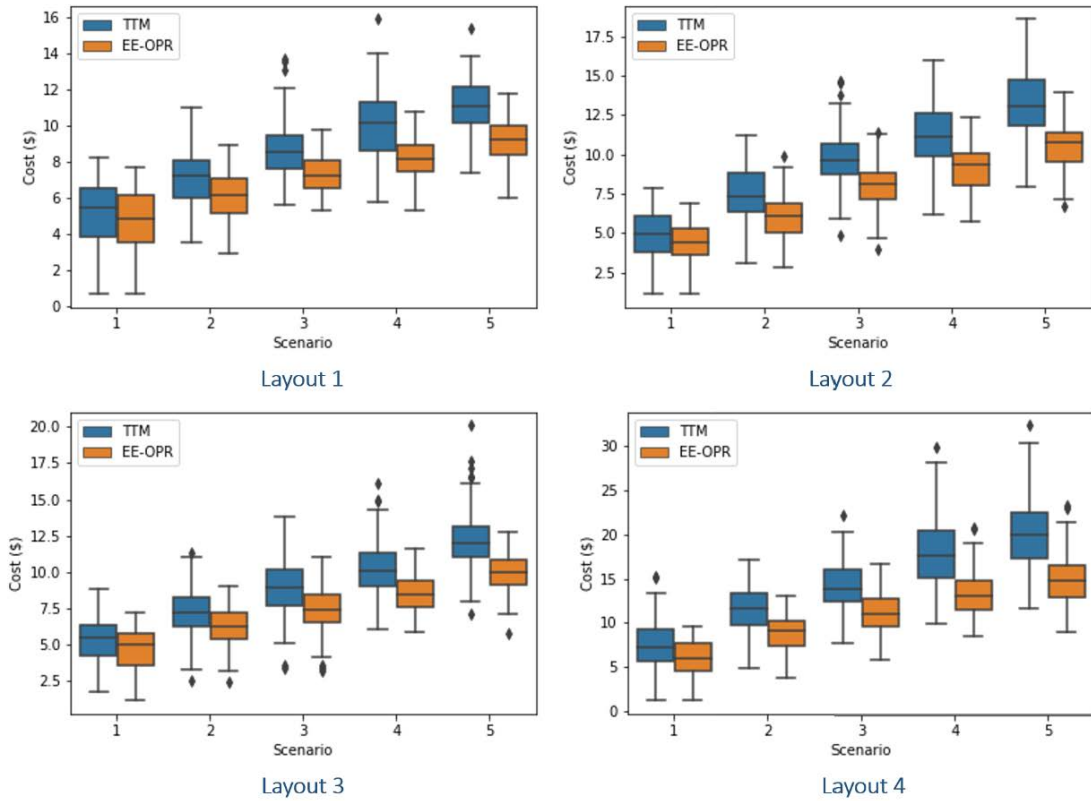


FIGURE 8. Comparison of the cost of the resulting routes after applying TTM and EE-OPR approaches. Four different layouts are used, each including five scenarios.

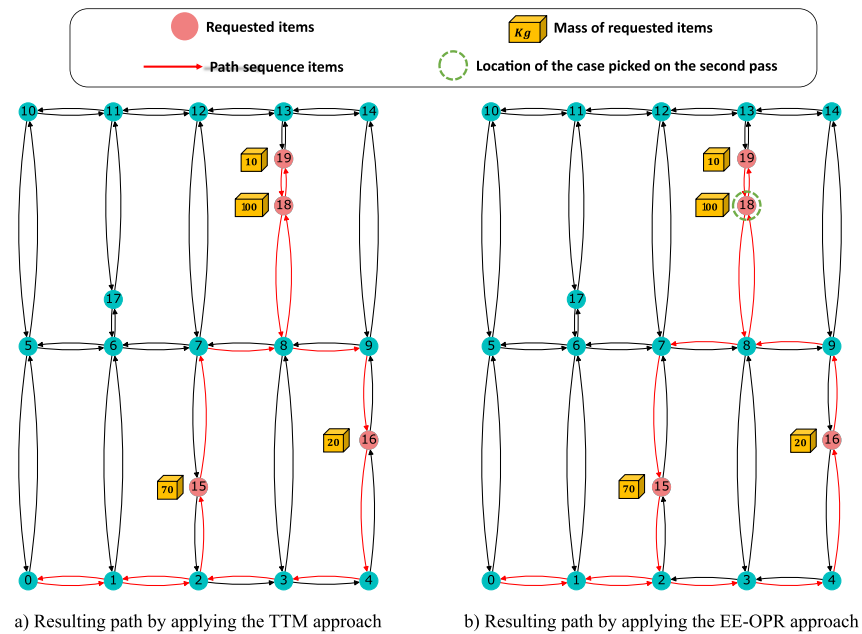


FIGURE 9. Comparison of the cost of resulting routes of TTM approach and EE-OPR approach.

that the segments constituting the path obtained by both approaches are exactly the same. In other words, the resulting paths of both approaches are similar in terms of distance traveled and also the shape of the red paths. However, the application of the approaches differs in the direction

of the trajectory as well as the order of picking the items at the required positions.

Let v_i be the position of vertex v and let $v_i \rightsquigarrow v_j$ be the shortest path between two vertices v_i and v_j and we assume that an AGV starts from a position v_0 , picks the loads at the

required positions (red circles) then goes back to v_0 . Thus, applying *EE-OPR* leads to the following result:

$$\begin{aligned} v_0(\text{load} = 0 \text{ kg}) &\sim v_{16}(\text{load} = 0 + 20 \text{ kg}) \\ &\sim v_{19}(\text{load} = 20 + 10 \text{ kg}) \\ &\sim v_{18}(\text{load} = 20 + 10 + 100 \text{ kg}) \\ &\sim v_{15}(\text{load} = 20 + 10 + 100 + 70 \text{ kg}) \\ &\sim v_0(\text{load} = 200 \text{ kg}) \end{aligned}$$

Now applying TTM leads to the following result:

$$\begin{aligned} v_0(\text{load} = 0 \text{ kg}) &\sim v_{15}(\text{load} = 0 + 70 \text{ kg}) \\ &\sim v_{18}(\text{load} = 70 + 100 \text{ kg}) \\ &\sim v_{19}(\text{load} = 70 + 100 + 10 \text{ kg}) \\ &\sim v_{16}(\text{load} = 70 + 100 + 10 + 20 \text{ kg}) \\ &\sim v_0(\text{load} = 200 \text{ kg}) \end{aligned}$$

As shown in the results and by contrast to *EE-OPR*, TTM drags unnecessary masses longer because of path direction. Furthermore, *EE-OPR* allows the vehicle to pass through one required position without the obligation of picking up the item(s) on the first pass. For instance, unlike TTM, *EE-OPR* passes through vertex 18 without collecting the load (so as not to drag it). Afterwards, it moves to vertex 19 to collect its corresponding load and then returns back to vertex 18 to collect its load. In other words, the items located in vertex 18 are not picked on the first pass but on the second. It is also observed, in this example, that the path resulting from *EE-OPR* is different from that given by TTM (direction and time of items withdrawal) but with the same length/duration (shorted path). Moreover, *EE-OPR* resulting path has the same energy demand given by ECM (with the lowest energy consumption). It is important to note that this situation (*EE-OPR* selects a path that is both the most energy efficient and the shortest) is very common, especially when the size of the pick-list is small.

Consequently, applying *EE-OPR* leads to a 25% of reduction in total cost in dollars of the pickup tour by comparison to TTM. Now, this gain in terms of cost can be more significant when the shortest path between the vertices v_{18} and v_{19} is worth kilometres (in large area warehouses).

Hence, *EE-OPR* minimizes energy loss associated with moving heavy objects. In fact, if possible, it delays the pickup of a certain item in order to reduce the distance to be covered with this item and therefore reduces the cumulative amount of the next load while respecting the travel time constraint.

V. DISCUSSION

Applying our approach led to an average of 18% saving in the total picking-tour cost. However, the number of pick locations to visit in a single picking tour is assumed to be less than 18, respecting the capacity of calculation. This implies that the number of stops can be up to 18 in a picking tour, but the quantity of required items in each location can vary according to orders' batching. Order batching is a technique for grouping a set of orders into batches [64]. We keep the

improvement of the limited number of 18 stops in a tour for future research.

However, this limitation is irrelevant for warehouses of heavy and bulky items.¹ That is, in such warehouses, items like large consumer electronics, carpets, or any other heavy items cannot be carried by the picker all the way back to the depot. Consequently, in these warehouses, the size of the pick-list is often small making the limitation of our approach w.r.t. the number of stops in a picking tour irrelevant.

It is important to note here that we assume that a single vehicle is used in the case study, which can be updated to accommodate multiple order pickers. We also assume that there are no obstacles on the travel path, thus enabling uninterrupted travel.

Note that the amount of the cost reduction achieved by our approach depends on many factors, such as the storage assignment policy, the type and the layout of the warehouse, the type and the weight of products, the size of the AGV, etc.

It is also important to mention that the routing, storage strategy, batching, zoning, and order release mode are components of the policy level which is highly dependent on the strategic level. In other words, the strategic level represents the system characteristics such as command cycle, mechanization level, warehouse dimensionality, and information availability [13].

Therefore, the efficiency of our routing algorithm depends on these characteristics. On the other hand, the computational efficiency of our algorithm depends on the locations to be visited (i.e., whether these locations are close, far or Scattered), which affects the speed of finding the target state.

Moreover, our approach can be used for different warehouse layouts with arbitrary starting and ending points of a tour. It suffices to define a Steiner graph containing the possible passage segments (arcs and vertices), the starting point (v_s), the ending point (v_t), and the required points (pick-list). Then the starting state will be (v_s, a_s) and the target state will be (v_t, a_t) . Such that a_s and a_t are 'empty arrangement' and 'full arrangement', respectively.

On the other hand, given that the definition of the picking problem in this study offers the possibility of a bidirectional movement of the AGVs in an aisle, *EE-OPR* can be suitable for low and medium throughput DCs. As if necessary, a congestion problem in the aisles may arise. However, a unidirectional movement may be required by simply changing the graph warehouse representation.

VI. CONCLUSION

Warehouses might represent a real threat to the environment as they might contribute to the rise of greenhouse gas emissions in supply chains. Consequently, many recent research works have been established to encourage the deployment of green and sustainable warehousing.

As typical material handling equipment in modern warehouses and distribution centers, the energy consumption of

¹PS-AGVs are often applied for heavy and bulky items. However, PS-AGVs picking is not bound to heavy and bulky goods. PS-AGVs can also be applied to support pickers in a zoning and batching environment. [5].

PS-AGVs represents a major part of the warehouse's total energy waste. However, current research related to order picking routing focuses mainly on travel distance and time optimization, while energy aspect is rarely considered. However, in many situations, energy is as important as time, especially during a period of low demand. An effective way to reduce the consumption of AGVs is to improve their operational efficiency and routing. In this study, an eco energetic routing for an PS-AGV was established. The approach called *EE-OPR* allows the robot to start from its depot to collect the items of orders from different locations in the storage area and to transport them to a defined position (the depot, packing station, or other defined target) while minimizing travel time and energy consumption simultaneously. We focus specifically on the effect of the mass transported for the decision-making of the order picking route to improve the AGV's energy efficiency without impacting the operating time. Moreover, unlike the RR-based methods, widely used in the literature, which are mainly based on the rectangular configuration of the warehouse (possible movements in a parallel aisles warehouse) to create their dynamic program, the *EE-OPR* is suitable for any type of warehouse layout. It is based on the dynamic creation of a state graph taking into account the energy demands of the vehicles and the weight of the transported cargo.

The solution to this problem is a path with the minimum cost that starts from the initial position with zero load and ends at the final position with all requested loads. First, an exploration phase is involved. It consists of the creation of states and transition arcs while passing through the spatial graph representing the warehouse. These states incorporate the information of (a) the position of the AGV in the warehouse (vertex of location) and (b) the mass of loads transported in each step with details of the items already picked. Besides, transition arcs are defined according to the possibility of picking up a neighboring vertex's item and are added based on the original graph of the warehouse. These arcs are weighted by a cost function based on the previous state and the AGV's energy consumption model having among the main factors the mass transported. In parallel, graph search phase is proceeded for the resolution of the shortest path problem using the Bellman principle.

Results obtained through different simulations indicate that *EE-OPR* always leads to better results compared to the approach based only on the minimization of travel time with an average gain of 18%. In the medium and long-term run, the potential for energy-saving gains can be achieved. Such gains can be the decrease in the cost of electricity, increasing the operational time of AGVs, and reducing the cost of maintenance or replacement of the battery. The change of storage strategy as well as batching strategy have, definitely, an effect on the gain that *EE-OPR* can provide by comparison to TTM. This effect can be studied in future research.

ACKNOWLEDGMENT

The authors would like to thank the Foundation of Université du Québec à Trois-Rivières, the AGV/AMR manufacturer Noovelia and Conseil de recherches en sciences naturelles

et en génie du Canada (CRSNG) for their financial support. The authors would like to thank Sayed Saeed Hosseini and Karima Makhoul for their valuable suggestions to improve the quality of the paper.

REFERENCES

- [1] L. Custodio and R. Machado, "Flexible automated warehouse: A literature review and an innovative framework," *Int. J. Adv. Manuf. Technol.*, vol. 106, nos. 1–2, pp. 533–558, Jan. 2020.
- [2] Z. Ghelichi and S. Kilaru, "Analytical models for collaborative autonomous mobile robot solutions in fulfillment centers," *Appl. Math. Model.*, vol. 91, pp. 438–457, Mar. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0307904X20305801>
- [3] Fetchrobotics. *AMR-Assisted Case Picking*. Accessed: Jun. 9, 2022. [Online]. Available: <https://fetchrobotics.com/case-picking/>
- [4] P. Fager, F. Sgarbossa, and M. Calzavara, "Cost modelling of onboard cobot-supported item sorting in a picking system," *Int. J. Prod. Res.*, vol. 59, no. 11, pp. 1–16, 2020.
- [5] N. Boysen, R. de Koster, and F. Weidinger, "Warehousing in the e-commerce era: A survey," *Eur. J. Oper. Res.*, vol. 277, no. 2, pp. 396–411, Sep. 2019.
- [6] S. Lang, M. Warburton, and A. A. Quettawala, "Global autos: Don't believe the hype—Analyzing the costs & potential of fuel-efficient technology," in *Bernstein Black Book—Global Autos: Don't Believe the Hype—Analyzing the Costs & Potential of Fuel-Efficient Technology*. New York, NY, USA: Bernstein Global Wealth Management, 2011.
- [7] H. A. Kiehne, *Battery Technology Handbook*, vol. 118. Boca Raton, FL, USA: CRC Press, 2003.
- [8] J. Freis, P. Vohlidka, and W. Günthner, "Low-carbon warehousing: Examining impacts of building and intra-logistics design options on energy demand and the CO₂ emissions of logistics centers," *Sustainability*, vol. 8, no. 5, p. 448, May 2016.
- [9] M. Bartolini, E. Bottani, and E. H. Grosse, "Green warehousing: Systematic literature review and bibliometric analysis," *J. Cleaner Prod.*, vol. 226, pp. 242–258, Jul. 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0959652619311321>
- [10] R. Carli, S. Digiesi, M. Dotoli, and F. Facchini, "A control strategy for smart energy charging of warehouse material handling equipment," *Proc. Manuf.*, vol. 42, pp. 503–510, Jan. 2020.
- [11] V. Anand, S. Lee, and V. V. Prabhu, "Energy-aware models for warehousing operations," in *Proc. IFIP Int. Conf. Adv. Prod. Manag. Syst.* Ajaccio, France: Springer, 2014, pp. 390–397.
- [12] L. Pansart, N. Catusse, and H. Cambazard, "Exact algorithms for the order picking problem," *Comput. Oper. Res.*, vol. 100, pp. 117–127, Dec. 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0305054818301862>
- [13] N. Shetty, B. Sah, and S. H. Chung, "Route optimization for warehouse order picking operations via vehicle routing and simulation," *Social Netw. Appl. Sci.*, vol. 2, no. 2, pp. 1–18, Feb. 2020.
- [14] R. de Koster, T. Le-Duc, and K. J. Roodbergen, "Design and control of warehouse order picking: A literature review," *Eur. J. Oper. Res.*, vol. 182, no. 2, pp. 481–501, 2007. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0377221706006473>
- [15] M. Löffler, N. Boysen, and M. Schneider, "Picker routing in AGV-assisted order picking systems," *Inform. J. Comput.*, vol. 34, no. 1, pp. 440–462, Jan. 2022.
- [16] M. Meißner and L. Massalski, "Modeling the electrical power and energy consumption of automated guided vehicles to improve the energy efficiency of production systems," *Int. J. Adv. Manuf. Technol.*, vol. 110, nos. 1–2, pp. 481–498, Sep. 2020. [Online]. Available: <https://link.springer.com/article/10.1007/s00170-020-05796-8>
- [17] T. van Gils, A. Caris, K. Ramaekers, K. Braekers, and R. B. de Koster, "Designing efficient order picking systems: The effect of real-life features on the relationship among planning problems," *Transp. Res. E, Logistics Transp. Rev.*, vol. 125, pp. 47–73, May 2019.
- [18] F. Martel, Y. Dubé, S. Kelouwani, J. Jaguemont, and K. Agbossou, "Long-term assessment of economic plug-in hybrid electric vehicle battery lifetime degradation management through near optimal fuel cell load sharing," *J. Power Sources*, vol. 318, pp. 270–282, Jun. 2016.
- [19] F. Martel, S. Kelouwani, Y. Dubé, and K. Agbossou, "Optimal economy-based battery degradation management dynamics for fuel-cell plug-in hybrid electric vehicles," *J. Power Sources*, vol. 274, pp. 367–381, Jan. 2015.

- [20] G. Dantzig, R. Fulkerson, and S. Johnson, "Solution of a large-scale traveling-salesman problem," *J. Oper. Res. Soc. Amer.*, vol. 2, no. 4, pp. 393–410, 1954.
- [21] R. E. Burkard, V. G. Deineko, R. van Dal, J. A. A. van der Veen, and G. J. Woeginger, "Well-solvable special cases of the traveling salesman problem: A survey," *SIAM Rev.*, vol. 40, no. 3, pp. 496–546, Jan. 1998.
- [22] B. Fleischmann, "A cutting plane procedure for the travelling salesman problem on road networks," *Eur. J. Oper. Res.*, vol. 21, no. 3, pp. 307–317, Sep. 1985.
- [23] H. D. Ratliff and A. S. Rosenthal, "Order-picking in a rectangular warehouse: A solvable case of the traveling salesman problem," *Oper. Res.*, vol. 31, no. 3, pp. 507–521, Jun. 1983.
- [24] K. J. Roodbergen and R. de Koster, "Routing order pickers in a warehouse with a middle aisle," *Eur. J. Oper. Res.*, vol. 133, no. 1, pp. 32–43, Aug. 2001.
- [25] M. Masae, C. H. Glock, and P. Vichitkunakorn, "Optimal order picker routing in a conventional warehouse with two blocks and arbitrary starting and ending points of a tour," *Int. J. Prod. Res.*, vol. 58, no. 17, pp. 5337–5358, Sep. 2020, doi: [10.1080/00207543.2020.1724342](https://doi.org/10.1080/00207543.2020.1724342).
- [26] A. Scholz, S. Henn, M. Stuhlmann, and G. Wäscher, "A new mathematical programming formulation for the single-picker routing problem," *Eur. J. Oper. Res.*, vol. 253, no. 1, pp. 68–84, Aug. 2016.
- [27] A. N. Letchford, S. D. Nasiri, and D. O. Theis, "Compact formulations of the Steiner traveling salesman problem and related problems," *Eur. J. Oper. Res.*, vol. 228, no. 1, pp. 83–92, Jul. 2013.
- [28] Y. Su, M. Li, X. Zhu, and C. Li, "Steiner TSP based on aisle as a unit for order picking," *Comput. Ind. Eng.*, vol. 168, Jun. 2022, Art. no. 108026. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0360835222000961>
- [29] M. Goetschalckx and H. D. Ratliff, "Order picking in an aisle," *IIE Trans.*, vol. 20, no. 1, pp. 53–62, Mar. 1988.
- [30] R. W. Hall, "Distance approximations for routing manual pickers in a warehouse," *IIE Trans.*, vol. 25, no. 4, pp. 76–87, Jul. 1993.
- [31] M. Wang, R.-Q. Zhang, and K. Fan, "Improving order-picking operation through efficient storage location assignment: A new approach," *Comput. Ind. Eng.*, vol. 139, Jan. 2020, Art. no. 106186. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0360835219306552>
- [32] R. D. Koster and E. V. D. Poort, "Routing orderpickers in a warehouse: A comparison between optimal and heuristic solutions," *IIE Trans.*, vol. 30, no. 5, pp. 469–480, May 1998.
- [33] M. Masae, C. H. Glock, and E. H. Grosse, "Order picker routing in warehouses: A systematic literature review," *Int. J. Prod. Econ.*, vol. 224, Jun. 2020, Art. no. 107564. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0925527319304050>
- [34] K. Azadeh, R. De Koster, and D. Roy, "Robotized warehouse systems: Developments and research opportunities," Erasmus Res. Inst. Manag. (ERIM), Rotterdam, The Netherlands, Tech. Rep. ERS-2017-009-LIS, 2017.
- [35] F. Boenzi, S. Digiesi, F. Facchini, G. Mossa, and G. Mummolo, "Greening activities in warehouses: A model for identifying sustainable strategies in material handling," *Ann. DAAAM Proc.*, vol. 26, no. 1, p. 16, 2015.
- [36] F. Facchini, G. Mummolo, G. Mossa, S. Digiesi, F. Boenzi, and R. Verriello, "Minimizing the carbon footprint of material handling equipment: Comparison of electric and LPG forklifts," *J. Ind. Manage. Optim.*, vol. 9, no. 5, pp. 1035–1046, 2016.
- [37] P. Zajac and T. Rozic, "Energy consumption of forklift versus standards, effects of their use and expectations," *Energy*, vol. 239, Jan. 2022, Art. no. 122187.
- [38] P. Zajac and P. Skorupski, "Modeling of the energy consumption of a forklift truck using the MATLAB simulink system," in *Proc. Int. Conf. Comput. Aided Eng.*, in Lecture Notes in Mechanical Engineering, Jun. 2018, pp. 851–857. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-04975-1_99
- [39] F. Basso, L. D. Epstein, R. Pezoa, and M. Varas, "An optimization approach and a heuristic procedure to schedule battery charging processes for stackers of palletized cargo," *Comput. Ind. Eng.*, vol. 133, pp. 9–18, Jul. 2019.
- [40] B. Zou, X. Xu, Y. Y. Gong, and R. De Koster, "Evaluating battery charging and swapping strategies in a robotic mobile fulfillment system," *Eur. J. Oper. Res.*, vol. 267, no. 2, pp. 733–753, Jun. 2018.
- [41] Y. Mei, Y.-H. Lu, Y. C. Hu, and C. S. G. Lee, "Deployment of mobile robots with energy and timing constraints," *IEEE Trans. Robot.*, vol. 22, no. 3, pp. 507–522, Jun. 2006.
- [42] S. Liu and D. Sun, "Minimizing energy consumption of wheeled mobile robots via optimal motion planning," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 2, pp. 401–411, Apr. 2014.
- [43] M. Mohammadpour, L. Zeghmi, S. Kelouwani, M.-A. Gaudreau, A. Amamou, and M. Graba, "An investigation into the energy-efficient motion of autonomous wheeled mobile robots," *Energies*, vol. 14, no. 12, p. 3517, Jun. 2021. [Online]. Available: <https://www.mdpi.com/1996-1073/14/12/3517>
- [44] M. Montazerijouybari, L. Baron, and S. Kelouwani, "Kinematics of 2-DOF AGVs with differential driving wheels and caster wheels modeling," in *Proc. Symp. Robot Design, Dyn. Control*, in CISM International Centre for Mechanical Sciences, vol. 601, 2021, pp. 495–502.
- [45] S. Barak, R. Moghdani, and H. Maghsoudlou, "Energy-efficient multi-objective flexible manufacturing scheduling," *J. Cleaner Prod.*, vol. 283, Feb. 2021, Art. no. 124610. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0959652620346540>
- [46] Z. Zhang, L. Wu, W. Zhang, T. Peng, and J. Zheng, "Energy-efficient path planning for a single-load automated guided vehicle in a manufacturing workshop," *Comput. Ind. Eng.*, vol. 158, Aug. 2021, Art. no. 107397. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0360835221003016>
- [47] X. Li, G. Hua, A. Huang, J.-B. Sheu, T. C. E. Cheng, and F. Huang, "Storage assignment policy with awareness of energy consumption in the kiva mobile fulfillment system," *Transp. Res. E, Logistics Transp. Rev.*, vol. 144, Dec. 2020, Art. no. 102158. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1366554520308036>
- [48] T. Xu, P. Yang, and H. Guo, "Energy efficiency analysis on robotic mobile fulfillment system," in *Proc. IEEE 6th Int. Conf. Ind. Eng. Appl. (ICIEA)*, Apr. 2019, pp. 145–149.
- [49] B. Zhou and Z. Zhu, "Multi-objective optimization of greening scheduling problems of part feeding for mixed model assembly lines based on the robotic mobile fulfillment system," *Neural Comput. Appl.*, vol. 33, no. 16, pp. 9913–9937, Aug. 2021, doi: [10.1007/s00521-021-05761-w](https://doi.org/10.1007/s00521-021-05761-w).
- [50] I. Kara, B. Y. Kara, and M. K. Yetis, "Energy minimizing vehicle routing problem," in *Proc. Int. Conf. Combinat. Optim. Appl.* Xi'an, China: Springer, 2007, pp. 62–71.
- [51] P. Cortés, R. A. Gómez-Montoya, J. Muñozuri, and A. Correa-Espinal, "A Tabu search approach to solving the picking routing problem for large- and medium-size distribution centres considering the availability of inventory and k heterogeneous material handling equipment," *Appl. Soft Comput.*, vol. 53, pp. 61–73, Apr. 2017.
- [52] S. Ene, İ. Küçükoglu, A. Aksoy, and N. Öztürk, "A genetic algorithm for minimizing energy consumption in warehouses," *Energy*, vol. 114, pp. 973–980, Nov. 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0360544216311586>
- [53] T. Rojanapitoon and J. Teeravarapug, "Economic order picker routing considering travel time and vehicular energy consumption with varying aisle traffic," *Int. J. Intell. Eng. Syst.*, vol. 13, no. 2, pp. 1–12, Feb. 2020.
- [54] T. Rojanapitoon and J. Teeravarapug, "A computer simulation for economical order picker routing when considering travel distance and vehicle energy consumption," *Int. J. Eng. Technol.*, vol. 7, no. 2.28, pp. 33–37, 2018.
- [55] S. Lee, H. W. Jeon, M. Issabakhsh, and A. Ebrahimi, "An electric forklift routing problem with battery charging and energy penalty constraints," *J. Intell. Manuf.*, vol. 33, no. 6, pp. 1761–1777, Aug. 2022, doi: [10.1007/s10845-021-01763-6](https://doi.org/10.1007/s10845-021-01763-6).
- [56] P. A. Makris, A. P. Makri, and C. G. Provatidis, "Energy-saving methodology for material handling applications," *Appl. Energy*, vol. 83, no. 10, pp. 1116–1124, Oct. 2006.
- [57] R. Elbert and J. P. Müller, "The impact of item weight on travel times in picker-to-parts order picking: An agent-based simulation approach," in *Proc. Winter Simul. Conf. (WSC)*, 2017, pp. 3162–3173.
- [58] F. Bulut and H. M. Erol, "A real-time dynamic route control approach on Google maps using integer programming methods," *Int. J. Next-Gener. Comput.*, vol. 9, no. 3, pp. 189–202, Nov. 2018. [Online]. Available: <http://perpetualinnovation.net/ojs/index.php/ijngc/article/view/426>
- [59] Sunita and D. Garg, "Dynamizing dijkstra: A solution to dynamic shortest path problem through retroactive priority queue," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 33, no. 3, pp. 364–373, Mar. 2021.
- [60] R. Bellman, "On the theory of dynamic programming," *Proc. Nat. Acad. Sci. USA*, vol. 38, no. 8, pp. 716–719, 1952. [Online]. Available: <http://www.jstor.org/stable/88493>

- [61] MSc Cybernetics. (2006). *Dijkstra's Algorithm Revisited: The Dynamic Programming Connexion*. [Online]. Available: <https://eudml.org/serve/209437/accessibleLayeredPdf/0>
- [62] A. Raith and M. Ehrgott, "A comparison of solution strategies for biobjective shortest path problems," *Comput. Oper. Res.*, vol. 36, no. 4, pp. 1299–1331, Apr. 2009.
- [63] M. Barbehenn, "A note on the complexity of Dijkstra's algorithm for graphs with weighted vertices," *IEEE Trans. Comput.*, vol. 47, no. 2, p. 263, Feb. 1998.
- [64] J. Zhang, X. Wang, and K. Huang, "Integrated on-line scheduling of order batching and delivery under B2C e-commerce," *Comput. Ind. Eng.*, vol. 94, pp. 280–289, Apr. 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0360835216300171>



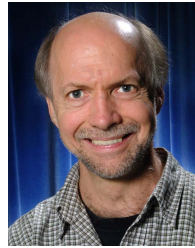
EMNA MEJRI (Member, IEEE) received the M.Sc. degree in mathematics and computer science in 2015. She is currently pursuing the Ph.D. degree with the Mechanical Engineering Department, University of Quebec at Trois-Rivières (UQTR), Trois-Rivières, QC, Canada. Her research interests include autonomous material handling vehicles routing, intelligent autonomous mobile robots (AMRs) path planning, global path planning (GPP), modeling of the energy consumed

by a path traversed, energy efficient order picking routing for picking support autonomous guided vehicles (Ps-AGVs), combinatorial optimization, and energy efficient navigation system in manufacturing plants and warehouses.



SOUSSO KELOUWANI (Senior Member, IEEE) received the Ph.D. degree in robotics systems from Ecole Polytechnique de Montreal, in 2011, and the Postdoctoral Internship in fuel cell hybrid electric vehicles from the Université du Québec à Trois-Rivières (UQTR), in 2012. He was a holder of the Canada Research Chair in energy optimization of intelligent transport systems and the Divel Research Chair in intelligent navigation of autonomous industrial vehicles. He developed

expertise in the optimization and the intelligent control of vehicular applications. He has been a Full Professor of mechatronics at the Department of Mechanical Engineering, since 2017. He is a member of the Hydrogen Research Institute. He holds four patents at the U.S. and Canada, in addition to having published more than 100 scientific articles. His research interests include optimizing energy systems for vehicle applications, advanced driver assistance techniques, and intelligent vehicle navigation taking into account Canadian climatic conditions. In 2019, his team received the 1st Innovation Prize in partnership with DIVEL, awarded by the Association des Manufacturiers de la Mauricie et Center-du-Québec, for the development of an autonomous and natural navigation system. In 2017, he received the Environment Prize at the Gala des Grands Prix d'excellence en Transport from the Association québécoise du Transport (AQTr), for the development of hydrogen range extenders for electric vehicles. He was a winner of the Canada General Governor Gold Medal, in 2003. He is a member of the Order of Engineers of Quebec. He was the Co-President and the President of the Technical Committee of the IEEE International Conferences on Vehicular Power and Propulsion in Chicago, USA, in 2018, and in Hanoi, Vietnam, in 2019.



YVES DUBÉ received the B.Sc. degree in physical engineering, the M.Sc. degree in mechanical engineering, and the Ph.D. degree in simulation and control from the Université Laval, Quebec, QC, Canada, in 1977, 1979, and 1985, respectively. He is a co-holder of the Divel Research Chair in intelligent navigation of autonomous industrial vehicles. He is currently an Associate Professor at the Department of Mechanical Engineering, Université du Québec at Trois-Rivières,

Trois-Rivières, QC, Canada. He has authored or coauthored more than 100 publications and holds one patent. His current research interests include renewable energy, hydrogen-combustion engine development, and control of mechanical systems. He has been a member of the Ordre des Ingénieurs du Québec and a Professional Engineer, since 1977.



NILSON HENAO received the B.S. degree in electronics engineering from the Universidad de los Llanos, Villavicencio, Colombia, in 2010, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Quebec at Trois-Rivières (UQTR), Trois-Rivières, QC, Canada, in 2013 and 2018, respectively. His research interests include statistical and machine learning methods with applications to residential energy management, distributed optimization, multi-agent control, smart grids, intelligent energy planning, energy storage, and load monitoring.

control, smart grids, intelligent energy planning, energy storage, and load monitoring.



KODJO AGBOSSOU (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electronic measurements from the Université de Nancy I, France, in 1987, 1989, and 1992, respectively. He is currently an Hydro-Québec Research Chair holder at the Transactive Management of Power and Energy in the Residential Sector, and the Chair at the Smart Energy Research and Innovation Laboratory, Université du Québec à Trois-Rivières (UQTR). He was the Head at the Engineering School, UQTR (2011–2017). He was the Head at the Department of Electrical and Computer Engineering, UQTR (2007–2011). He was also the Director of graduate studies in electrical engineering of the UQTR (2002–2004). He was a Postdoctoral Researcher (1993–1994) at the Electrical Engineering Department, UQTR, and a Lecturer (1997–1998) at the same department. He is the author of more than 325 publications and has four patents and two Patents Pending. His current research interests include renewable energy, the use of hydrogen, home demand side management, integration of energy production, storage and electrical energy generation systems, connection of electrical vehicle to the grid, and control and measurements. He is a member of Hydrogen Research Institute and Research Group "GREI" of UQTR. Since 2015, he has been the Sub-Committee Chair on "Home and Building Energy Management of Smart Grid Technical Committee" and IEEE Industrial Electronics Society (IES).

He was the Head at the Department of Electrical and Computer Engineering, UQTR (2007–2011). He was also the Director of graduate studies in electrical engineering of the UQTR (2002–2004). He was a Postdoctoral Researcher (1993–1994) at the Electrical Engineering Department, UQTR, and a Lecturer (1997–1998) at the same department. He is the author of more than 325 publications and has four patents and two Patents Pending. His current research interests include renewable energy, the use of hydrogen, home demand side management, integration of energy production, storage and electrical energy generation systems, connection of electrical vehicle to the grid, and control and measurements. He is a member of Hydrogen Research Institute and Research Group "GREI" of UQTR. Since 2015, he has been the Sub-Committee Chair on "Home and Building Energy Management of Smart Grid Technical Committee" and IEEE Industrial Electronics Society (IES).

...

Energy efficient path planning for low speed autonomous electric vehicle

Emna Mejri, Souso Kelouwani, Yves Dubé, Omar Trigui, Kodjo Agbossou
Hydrogen Research Institute (IRH)

Université du Québec à Trois-Rivières QC G9A 5H7

Email: Emna.Mejri, Souso.Kelouwani, Yves.Dube, Omar.Trigui, Kodjo.Agbossou @uqtr.ca

Abstract—This work presents an energy efficient approach for autonomous electric vehicle path planning. When the vehicle is moving at low speed, the rolling resistance losses can be more than the aerodynamic losses, on a flat ground. In particular, for warehouse low-speed electric vehicles, different road-tire frictions may lead to varying rolling resistance which impacts the energy consumption. The minimization of the energy consumed by a vehicle is important in the context where the number of charging stations is limited. The proposed method aims at planning the most energy efficient path by taking into account the rolling resistance and the path length. Unlike most studies reported in the literature, this energy efficient path planner can achieve a good trade-off between preserving battery energy and not extending to much the path length. The preliminary results obtained through extensive simulation indicates that the optimized path planner is effective and robust.

Index Terms—energy consumption; battery management; path planning; autonomous vehicles; rolling resistance; vehicle dynamics

I. INTRODUCTION

DURING the previous years, mobile robots and autonomous vehicles have witnessed a remarkable evolution. These vehicles play an unquestionable role in the military, manufacturing and industrial fields [1]. They are also getting more and more included in other fields of study. A lot of progress has been achieved in the navigational aspect and platform security. However, the autonomy remains energetically unexplored. Nonetheless, the batteries represent an important part of the cost of the equipment and its autonomy remains limited. In fact, the use of the batteries as a source of nourishment in a given electric vehicle raises several challenges such as the recharging period which is relatively long or the limited lifetime of the battery [2]. In order to contest with other conventional inner burning engines, there should be a development in these aspects [3]. An average daily use of these platforms usually produces a full discharge with negatively affect the batteries lifespan [4]. Actually, the lifetime of the battery is directly linked to the way it is used (the number or charge/discharge cycles and the depth-of-discharge) [4], [5]. It is recommended to have the battery in a given interval of State-of-Charge (SOC) in order to 1) maximize its performance and its range 2) minimize its depth discharge (DOD) to optimize its performance and lifetime.

The number of charging stations is generally weak compared to the number of the functional electric vehicles in a warehouse. The solution that consists in increasing the

batteries capacities would boost the costs: in terms of vehicle costs, recharging energy costs, supplementary energy costs related to the weight of the batteries as well as the maintenance costs. Even if we opt for this solution and we increase the capacity of batteries, the risk of the vehicles failure to meet its objectives will be more probable if we do not take into account the environmental factors.

However, the level of the used energy is straightly linked to the path planning. It is fundamental to take into consideration the energy standard in path planning. Consequently, implementing a planner energetically optimal which will contest with traditional planners generally based on distance standard. Many parameters influence the used battery energy. They can be classified in two categories: (i) intrinsic parameters essentially linked to the structure of the platform: the type of the energy, the powertrain architecture, the mechanical structure and aerodynamic, etc; (ii) extrinsic parameters due to platform-environment interaction on one hand (Slope, friction, number of turn, etc) and on the other hand due to driving style. Since the intrinsic parameters cannot be modulated easily, understanding and quantifying the losses associated with the extrinsic parameters becomes necessary in order to identify key point for energy efficiency.

In this paper, we will be tackling the effects of rolling resistance for the purpose of making an energetically optimal path planning.

- 1) Estimation of the consumed energy for the different sub-trajectory in mind the change in the features of the land.
- 2) The use of an algorithm specialized in path planning with the energy standard.

Many researchers worked on the estimation of consumed energy through a platform either mobile robot or generally speaking electric vehicle [2], [6]. However, some people neglected crucial parameters for a good prediction such as the weight, the friction of the floor, etc, which affect the recharging planning and the success of the mission.

On the other hand, many researchers worked on path planning by considering the shortest path criteria for autonomous navigation. Few reported works have included the energetic cost associated the selection of trajectories. In fact, the majority of analysis only consider the shortest path as a performance indicator in order to move point to point [7]–[11].

The problem of the optimal path selection is one of the most typical problems in graph theory, including Floyd algorithm,

Johnson algorithms, Dijkstra algorithm, etc. In this paper, Dijkstra's algorithm is used for its ability to explore the entire environment.

The rest of the paper is organized as follows: First, we describe the proposed approach. We explain in this section the representation of the environment, the discretization of space, the graph representation, the approach for solving the optimization problem. Then, in section III, we present the results and the discussion. Finally, in Section IV, we provide a short conclusion.

II. ENERGY EFFICIENT PATH PLANNING

In this work, we deal with the off-line optimal path planning needed for guiding an autonomous vehicle to move from its actual position to the destination, to be used later by a controller system [12]. Among different frameworks available to tackle the planning problem, the graph representation has some nice features: synthetic representation between the navigation features and the reachable poses in the workspace. So in the sequel, we present how this representation is used to solve the energy optimal path planning problem.

A. Graph representation

1) *Space configuration*: To find trajectory in a configuration space, discrete and continuous approaches are used in literature, but both share the same data and they only differ on how to find a solution. We consider the discrete space configuration of the vehicle, introduced in [13] constituted by all the triplets of values $= (x, y, \theta)$ where x , y and θ represent the vehicle abscissa, ordinate and orientation, respectively. However, for simplification, we consider only a point representation of the vehicle using its pose $= (x, y)$. Therefore the configuration space is the set of all reachable (x, y) .

2) *Map representation*: The configuration space is represented with a map and in this work, the metric map is used. An example of such a map is shown in Fig. (1). All cells have the same dimension (uniform cell decomposition). Cells occupied by obstacles have a black color where others have a white color. Therefore, the set of white cells is the workspace or the configuration space.

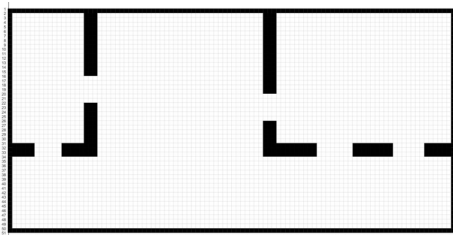


Fig. 1. Occupancy grid (matrix of 50*100 cells)

3) *Building a graph representation*: We assume that the vehicle can move to any adjacent cell of the configuration space (a maximum of 8 possible moves at any time) [14]. By connecting each configuration space cell to its neighbors a lattice or a mathematical structure called a graph is constructed.

The unoccupied cells are nodes, and links between adjacent nodes are edges. The vehicle moves between nodes (the white dots on Fig.(2)) which represent the points reachable by the vehicle.

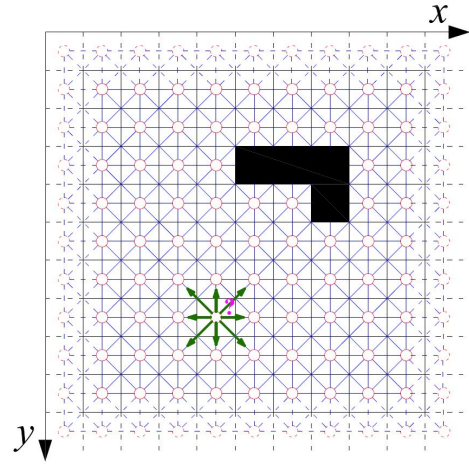


Fig. 2. Example of graph representation in the configuration space (each cell / node has Cartesian coordinates)

B. Energy optimal path planning problem

Assume that the vehicle can move uniformly (constant speed in both coordinates axis) from one admissible cell to another following cell (node). Given:

- the graph representation of the configuration,
- the initial pose of the vehicle,
- the final or destination pose

how can we compute the path with the shortest traveling length (minimum time) and the minimum energy consumption?

Before presenting the optimization strategy, let introduce the adjacency matrix.

1) *Adjacency matrix*: In our approach, we tried to minimize two different criteria: distance criteria that also reflects traveling time and energy consumption criteria by considering the impact of the rolling resistance. So, we define two adjacency matrices A_d and A_e as follows:

- If there is connexion between cells (x_1, y_1) (represented by i) and (x_2, y_2) (represented by j) then $A_d(i, j)$ is the euclidean distance between (x_1, y_1) and (x_2, y_2) , otherwise $A_d(i, j) = \infty$.
- If there is connexion between cells (x_1, y_1) and (x_2, y_2) then $A_e(i, j)$ is the energy used by the vehicle to move from (x_1, y_1) to (x_2, y_2) .

Fig. 3 illustrates a distance based adjacency matrix in which a vertical or horizontal motion have a distance of 1, whereas a diagonal motion distance is $\sqrt{2}$. The left figure of Fig. 3 represents the computed distance-based graph.

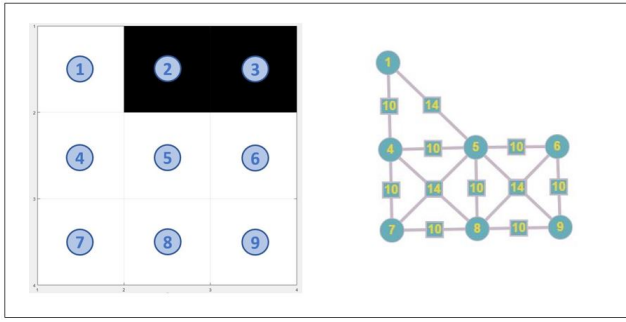


Fig. 3. Representation of distance-based graph: the left figure is the configuration space in which white cells are obstacle-free and black cells are occupied by obstacles. The right figure is the computed distance-based graph.

To build the energy-based adjacency matrix, the vehicle dynamic needs to be used and we consider the most used longitudinal dynamic model for ground vehicles. This model takes into account the tire/road rolling Resistance. Therefore, a rolling resistance coefficient map is required. For each cell in this paper, we assume that this rolling resistance coefficient exist. In [?], [6], methods for identifying the rolling resistance coefficient as the vehicle is moving are proposed.

2) *Computing energy consumption between two adjacent points:* The consumed energy estimation step is elaborated in order to build the energy-based adjacency matrix. From the configuration space, each cell rolling resistance coefficient is retrieved. Assume that the speed is constant, the path from to is discretized into small steps and is the timestamp. Therefore, the electric energy consumed to move from to is given by:

$$(1)$$

where is the duration between two consecutive timestamps and where represents the electric power delivered by the battery pack (see eq. (6)).

The vehicle discrete longitudinal dynamics [15] is:

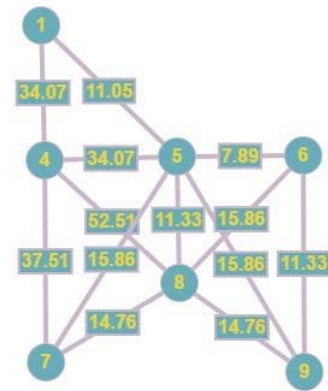
$$(2)$$

where , and represent the timestamp, the vehicle total mass and longitudinal speed and where , , and represent the longitudinal traction force provided by the powertrain, the aerodynamic force (see eq. (3)), the gravity force on a road slope (see eq. (4)) and the rolling resistance force (see eq. (5)), respectively.

The aerodynamic force defined in equation (3) is the force that opposes the vehicle's advancement in the air. It depends to the density of air in / , to the front surface of the vehicle in , drag coefficient and the square of vehicle speed

–

(3) Fig. 4. Representation of energy-based graph using the rolling resistance map



where , , and represent the density of air, the active front surface of the vehicle, the drag coefficient and the vehicle speed relatively to the air.

$$(4)$$

where is the road grade (road slope) and where is the gravity constant.

$$(5)$$

where is the given rolling resistance at the vehicle position

$$(6)$$

Using the rolling resistance coefficient map is shown in Fig. (5), we showed in Fig.(4) an example of energy-based adjacency matrix for the configuration space of Fig. (3).

To highlight the impact of the rolling resistance on the consumed energy, we showed in Fig. (5) an example of optimal path selection. It can be observed that the blue path is the shortest length path, however, the red path is the energy efficient one.

TABLE III

SIMULATION RESULTS OF DIJKSTRA'S APPROACH AND ENERGY EFFICIENT APPROACH

		Path 1	Path 2	Difference
Scenario 1	Energy consumption (J)	2341.75	1615.91	31 %
	Distance (m)	803.85	1054.56	-31 %
Scenario 2	Energy consumption (J)	1446.24	1252.49	13 %
	Distance (m)	683.55	851.42	-25 %
Scenario 3	Energy consumption (J)	2090.36	1128.51	46 %
	Distance (m)	621.42	761.42	-23 %
Scenario 4	Energy consumption (J)	747.79	633.62	15 %
	Distance (m)	143.14	154.85	-8 %

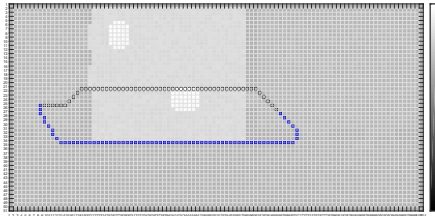


Fig. 8. Comparison of Dijkstra's approach and energy-efficient approach (scenario 3)

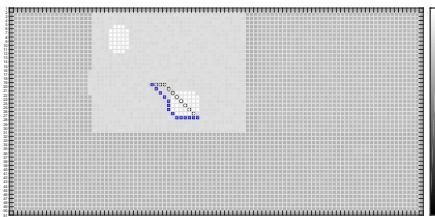


Fig. 9. Comparison of Dijkstra's approach and energy-efficient approach (scenario 4)

The comparative analysis between using Dijkstra method and energy-efficient approach is presented in Table (III). Scenario 2 (Fig.(7)) shows a limitation of our approach. Indeed, we notice a low energy gain compared to the high percentage of travel distance. On the other hand, scenario 3 shows a high-energy gain of 46% is observed. In Scenario 4, we assumed that there is a wall along the street (barred street) to test the ability of our tool to bypass the sludge. Here again, our method provides satisfactory results.

The proposed approach deals with global (offline) planning. Therefore, only static obstacles included in maps are considered. If a moving or unexpected obstacle appears or if the soil conditions change (e.g. winter context), a local re-planning must be carried out. In addition, other factors can influence energy consumption, such as slope and number of gyration. We will address these aspects in a future work.

IV. CONCLUSION

In this work, we proved that the shortest path is not always the most energy efficient one. Since we know that at low speed, the energy due to rolling resistance is much more important than aerodynamic energy, it is, therefore, valuable to select the path accordingly. In addition, for an autonomous vehicle, an energy efficient path planner should take into account the obstacles. Using the optimal planning framework, we select the Dijkstra method as the basis of finding an energy-efficient path. Therefore, the proposed smart planning algorithm used:

- 1) The environment map which includes the workspace.
- 2) The environment rolling resistance map.
- 3) The vehicle dynamics.

Through extensive simulation, the preliminary results suggested that the proposed approach can select the energetically optimal path without increasing too much the whole motion duration. Furthermore, the flexibility of the path planner can be applied to all type autonomous vehicles for unmarked navigation environments either inside of a building (museum, hotel, store, etc.) or outside a building (desert, port, etc.). The future works will focus on integrating other criteria (battery degradation for an electric vehicle, path slope, etc.) to the smart path planner.

V. ACKNOWLEDGEMENT

This work was supported by "Bureau de l'efficacité et de l'innovation énergétiques, "Ministère des ressources naturelles" et de la faune du Québec", "Natural sciences and engineering research council of Canada" and "Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG)".

REFERENCES

- [1] M. József, "Concept of an UGV with Arduino device," 2014.
- [2] K. Maalej, S. Kelouwani, K. Agbossou, Y. Dubé, and N. Henao, "Long-Trip Optimal Energy Planning With Online Mass Estimation for Battery Electric Vehicles," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 11, pp. 4929–4941, 2015.
- [3] S. G. Wirasingha and A. Emadi, "Classification and Review of Control Strategies for Plug-In Hybrid Electric Vehicles," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 1, pp. 111–122, 2011.
- [4] F. Martel, Y. Dubé, L. Boulon, and K. Agbossou, "Hybrid electric vehicle power management strategy including battery lifecycle and degradation model," in *2011 IEEE Vehicle Power and Propulsion Conference*, 2011, pp. 1–8.
- [5] Usbattery, "No Title." [Online]. Available: <http://usbattery.com/>
- [6] A. Sadrpour, J. Jin, and A. G. Ulsoy, "Experimental validation of mission energy prediction model for unmanned ground vehicles," in *American Control Conference (ACC)*, 2013. IEEE, 2013, pp. 5960–5965.
- [7] K. Al-Muteb, M. Faisal, M. Emaduddin, M. Arafah, M. Alsulaiman, M. Mekhtiche, R. Hedjar, H. Mathkooor, M. Algabri, and M. A. Bencherif, "An autonomous stereovision-based navigation system (ASNS) for mobile robots," *Intelligent Service Robotics*, vol. 9, no. 3, pp. 187–205, 2016.
- [8] H. Che, Z. Wu, R. Kang, and C. Yun, "Global path planning for explosion-proof robot based on improved ant colony optimization," in *2016 Asia-Pacific Conference on Intelligent Robot Systems (ACIRS)*, 2016, pp. 36–40.
- [9] E. G. Tsardoulis, A. Iliakopoulou, A. Kargakos, and L. Petrou, "A Review of Global Path Planning Methods for Occupancy Grid Maps Regardless of Obstacle Density," *Journal of Intelligent & Robotic Systems*, vol. 84, no. 1, pp. 829–858, 2016.

- [10] Y. Zhang, G. Guan, and X. Pu, "The Robot Path Planning Based on Improved Artificial Fish Swarm Algorithm," *Mathematical Problems in Engineering*, vol. 2016, 2016.
- [11] W. Huijuan, Y. Yuan, and Y. Quanbo, "Application of Dijkstra algorithm in robot path-planning," in *2011 Second International Conference on Mechanic Automation and Control Engineering*, 2011, pp. 1067–1069.
- [12] S. Kelouwani, C. Ouellette, and P. Cohen, "Stable and Adaptive Control for Wheeled Mobile," vol. 2013, no. November, pp. 391–405, 2013.
- [13] T. Perez-Lozano, "Spatial planning: A configuration space approach," *IEEE Trans. Computers*, vol. 32, 1983.
- [14] K.-Y. Chang, G. E. Jan, and I. Parberry, "A method for searching optimal routes with collision avoidance on raster charts," *The Journal of Navigation*, vol. 56, no. 03, pp. 371–384, 2003.
- [15] R. Rajamani, "Longitudinal Vehicle Dynamics," in *Vehicle Dynamics and Control*. Boston, MA: Springer US, 2006, pp. 95–122.
- [16] E. W. Dijkstra, "A note on two problems in connexion with graphs:(numerische mathematik, _1 (1959), p 269-271)," 1959.
- [17] R. Bellman, "On the Theory of Dynamic Programming," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 38, no. 8, pp. 716–719, 1952.