



**HAL**  
open science

# Résolution d'un problème industriel de type pick & place d'un centre de tri postal

Cyrine Selma

► **To cite this version:**

Cyrine Selma. Résolution d'un problème industriel de type pick & place d'un centre de tri postal. Automatique / Robotique. Université de Nantes (UN), FRA., 2021. Français. NNT: . tel-03789818

**HAL Id: tel-03789818**

**<https://theses.hal.science/tel-03789818v1>**

Submitted on 27 Sep 2022

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

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

# THESE DE DOCTORAT

DE L'UNIVERSITE DE NANTES

ECOLE DOCTORALE N° 601

*Mathématiques et Sciences et Technologies*

*de l'Information et de la Communication*

Spécialité : Automatique, Productique et Robotique

Par

**« Cyrine SELMA »**

**« Résolution d'un problème industriel de type pick & place d'un centre de tri postal »**

Thèse présentée et soutenue à l'Université de Nantes, le 13 Juillet 2021

Unité de recherche : Laboratoire des Sciences du Numérique de Nantes (LS2N)

## Rapporteurs avant soutenance :

Samir LAMOURI	Professeur, Arts & Métiers ParisTech, Paris
Cyril BRIAND	Professeur, Université de Toulouse 3 Paul Sabatier, Toulouse

## Composition du Jury :

Président :	Hind BRIL	Professeur, ENSTIB, Epinal
Examineurs :	Hind BRIL	Professeur, ENSTIB, Epinal
	Nasser MEBARKI	Maître de Conférences-HDR, IUT de Nantes, Université de Nantes
	Dalila TAMZALIT	Maître de Conférences-HDR, IUT de Nantes, Université de Nantes
Dir. de thèse :	Olivier CARDIN	Maître de Conférences-HDR, IUT de Nantes, Université de Nantes

## Invité(s)

Didier THIERIOT	Ingénieur, Groupe La Poste
Loïc BRUGGEMAN	Ingénieur, Groupe La Poste

# Résolution d'un problème industriel de type pick & place d'un centre de tri postal

## Résumé

Les problèmes de type Pick & Place (PAP) sont très étudiés dans la littérature, mais, à notre connaissance, très peu de travaux étudient les systèmes de PAP dans un contexte industriel. L'objectif de cette thèse est la résolution d'un problème industriel de type PAP au sein d'un centre de tri postal, où des bacs remplis de courriers arrivent dynamiquement et dans un ordre inconnu, et où des opérateurs placent ces bacs dans des chariots en fonction de leur destination. Compte tenu de la diversité importante des destinations journalières, un équilibre doit être trouvé en temps réel entre les flux traités par les humains et par le robot.

Ce problème a été résolu en quatre phases. En premier lieu, des modèles à base de connaissance ont été proposés à partir de l'expérience de l'opérateur logistique. Le résultat de l'application de ces modèles sur une simulation du système réel est considéré comme une borne inférieure de la performance du système. En second lieu, un modèle mathématique du système a été établi, le relâchement de plusieurs contraintes permettant de traiter le problème comme un problème d'ordonnancement classique. Les résultats de cet ordonnancement, inapplicables sur le terrain, nous ont conduits à investiguer l'utilisation d'heuristiques en ligne. Une troisième étape a été de proposer un modèle heuristique, à base de règles dynamiques, évaluée en simulation. Enfin, un modèle multi-agents, intégrant ces règles de décisions, a été développé afin de valider l'applicabilité d'un tel système de pilotage sur le système réel.

**Mots clés :** pick-and-place, modèle multi-agent, robotique, heuristique, simulation, recherche opérationnelle



« We are what we repeatedly do. Excellence, then, is not an act, but a habit »

*Aristote*



# Remerciements

Je remercie Olivier CARDIN, mon directeur de thèse pour son soutien scientifique, pour sa disponibilité et pour ces précieuses directives. Je remercie également mes encadrants Nasser MEBARKI et Dalila TAMZALIT de m'avoir accompagné dans cette aventure scientifique. Je tiens à remercier Nasser pour sa réactivité, ses conseils et surtout ses connaissances dans le domaine de la Recherche Opérationnelle. Je remercie Dalila pour ses encouragements, son dynamisme et sa joie de vivre.

J'adresse mes sincères remerciements à Didier THIERIOT pour son implication dans la direction de ces travaux, sa bienveillance, et son intérêt à l'égard de ma recherche. Je remercie vivement Loïc BRUGGEMAN pour l'intérêt porté à mon travail. Qu'ils trouvent ici l'expression de toute ma gratitude.

Je remercie Hind BRIL, pour m'avoir fait l'honneur de présider mon jury de thèse. Je remercie également les rapporteurs Samir LAMOURI et Cyril BRIAND pour le temps et l'attention consacrés à l'étude de ces travaux et pour leurs interrogations pertinentes qui m'ont permis d'améliorer sa rigueur.

Il ne m'est pas possible d'oublier mes collègues de bureau. Merci Emmanuel pour ta gentillesse et pour ton sourire indéfectible, merci Artem pour ta serviabilité et pour les pauses café, merci Erica pour les discussions et pour tes expressions venues d'une autre planète ! merci Queen pour tes petits plats libanais, et finalement merci Nicolas pour ta bonne humeur.

Je remercie mes amis, en particulier Sabra pour son soutien moral et ses conseils avisés.

Je tiens ensuite à exprimer toute ma gratitude envers mes parents Monia et Abderraouf, à qui je dois beaucoup, pour le soutien quotidien indéfectible durant ces années de thèse et de m'avoir apporté la motivation nécessaire quand il le fallait. Je remercie aussi mes sœurs Assal, Emna et Darine pour l'amour, l'écoute et l'encouragement durant mes périodes de doute.

Je remercie mon cher époux pour son amour, pour son enthousiasme à l'égard de mes travaux et qui, avec cette question récurrente, « quand est-ce que tu soutiens ta thèse ? », bien qu'angoissante en période fréquente de doutes, m'a donné la force de persister et ne jamais dévier de mon objectif final.

Enfin, je remercie mes grands-parents pour leur soutien et leur amour inconditionnel. Particulièrement mon grand-père paternel qui a attendu ce moment avec impatience, mais sans pouvoir être parmi nous pour le fêter.

L'accomplissement de ce travail est aussi le vôtre.

**Table des matières**

1	Problématique .....	6
1.1	Fonctionnement actuel.....	6
1.2	Besoin .....	11
1.3	Sous système (Présentation de flexibac) .....	12
1.4	Conclusion .....	14
2	Etat de l'art.....	16
2.1	Présentation générale du problème de PAP.....	16
2.2	Les méthodes de résolution des problématiques de PAP dans la littérature ..	19
2.2.1	Résolution de problèmes de génération de trajectoire .....	20
2.2.2	Résolution de problèmes d'équilibrage de charge de travail .....	31
2.2.3	Résolution de problèmes d'optimisation du nombre de produits traités.....	35
2.3	Conclusion .....	37
3	Evaluations préliminaires : Système expert.....	43
3.1	Modèle à base de connaissance .....	44
3.1.1	Définition d'un système expert basé sur la connaissance .....	44
3.1.2	Développement des modèles à base de connaissance :.....	47
3.2	Expérimentations .....	51
3.3	Conclusion .....	59
4	Modèle basé sur les méthodes de recherche opérationnelle .....	61
4.1	Nombre de robots .....	61
4.2	Résolution du problème .....	63
4.2.1	Résolution hors-ligne .....	64
4.2.2	Résolution en ligne : Heuristiques .....	67
4.3	Expérimentations .....	69
4.3.1	Présentation du modèle de simulation .....	69
4.3.2	Analyse de sensibilité .....	71
4.4	Résultats et comparaisons.....	72
4.5	Conclusion .....	72



5	Implémentation par SMA .....	74
5.1	Agents industriels .....	75
5.1.1	Présentation des SMA et de la notion d'agent .....	75
5.1.2	Les SMA pour gérer la variabilité dans un environnement incertain .....	77
5.2	Méthodologies de conception du SMA .....	79
5.2.1	Analyse et comparaison des méthodologies .....	79
5.2.2	Application de la méthodologie sur notre SMA .....	83
5.3	Outil de modélisation.....	91
5.3.1	Choix de l'outil .....	91
5.3.2	Présentation de l'outil .....	92
5.4	Implémentation du SMA .....	95
5.4.1	Catégorisation des agents.....	96
5.4.2	Présentation des comportements des agents du système .....	97
5.4.3	Présentation des interactions entre les agents .....	104
5.5	Expérimentations et résultats.....	105
5.6	Conclusion .....	108
	Conclusion générale et perspectives .....	110
	Bibliographie .....	112

*Table des figures*

Figure 1 Conteneurs - format standardisé La Poste .....	3
Figure 2 Chariot de courrier - format standardisé de La Poste .....	3
Figure 3 Etiquette de destination d'un conteneur.....	3
Figure 4 Cellule de chargement automatique.....	4
Figure 5 Machine de tri petit format .....	7
Figure 6 Processus d'expédition de courrier (collecte, tri, distribution).....	8
Figure 7 Lignes de ventilation dans un centre de tri postal (depuis l'anneau de tri à gauche et depuis l'extrémité à droite). .....	9
Figure 8 Organisation d'un centre de tri postal.....	9
Figure 9 Parcours du flux de courrier entre les quais d'arrivées et les quais de départs.....	10
Figure 10 Quai de départ dans un centre de tri postal.....	11
Figure 11 Schéma simplifié de l'anneau de tri avec la cellule robotisée .....	12
Figure 12 Cellule de chargement automatique - Vue aval.....	13
Figure 13 Robot 6 axes de manipulation de contenants.....	13
Figure 14 Couvercle de sécurité.....	14
Figure 15 Emplacement des chariots dans la cellule de chargement .....	14
Figure 16 Cellule robotisée de type PAP avec convoyeurs parallèles (Humbert et al., 2015). .....	17
Figure 17 Configuration à flux unidirectionnel - configuration à flux opposé (Briand and Parlouar, 2019). .....	17
Figure 18 Différents types de robots de PAP (Mizuuchi et al., 1999). .....	18
Figure 19 Principe des algorithmes d'optimisation par colonies de fourmis. ....	21
Figure 20 Le jumeau numérique vs l'environnement réel (Ridhi et al., 2019) .....	22
Figure 21 Programmation par démonstration d'un robot pour une opération de PAP (Dimeas et al., 2019).....	23
Figure 22 Architecture orientée service (Švaco et al., 2011a). .....	25
Figure 23 Architecture multi-agent (Švaco et al., 2011c). .....	27
Figure 24 Tests de simulation avec l'outil développé dans (Humbert et al., 2015). .....	28
Figure 25 Système de type PAP multi-robot (Huang et al., 2015).....	31
Figure 26 Architecture de simulation avec les deux niveaux de stratégies (règles d'ordonnancement individuelles et stratégies collaboratives) (Humbert et al., 2015) .....	32
Figure 27 Ligne d'assemblage robotisée (Daoud et al., 2012) .....	32
Figure 28 Schéma simplifié du système étudié. ....	44
Figure 29 Architecture générale d'un système expert .....	46
Figure 30 Fichier Excel qui regroupe les données pour la journée du 17/10/2018.....	48
Figure 31 Le calcul des destinations les plus importantes pour chaque phase.....	49
Figure 32 Table comportant les données d'entrée sur FlexSim .....	53
Figure 33 Table représentant le plan de tri journalier sur FlexSim .....	53

Figure 34 Processus logique de création et gestion des lots sur le logiciel FlexSim .....	54
Figure 35 Processus logique de gestion des plans de tri sur le logiciel FlexSim .....	54
Figure 36 Capture d'écran du modèle de simulation sur FlexSim .....	55
Figure 37 Capture d'écran du module de chargement automatique sur FlexSim .....	55
Figure 38 Nombre de bacs sur Excel vs nombre de bacs sur FlexSim par rapport au temps – Journée du 19/10 .....	57
Figure 39 Nombre de bacs sur Excel vs nombre de bacs sur FlexSim par rapport au temps – Journée du 15/10 .....	57
Figure 40 Courbe représentant l'évolution du nombre de bacs par rapport au temps pour les journées : 19 et 17 – SEBC 2 (Bleu : sur Excel, Orange : sur FlexSim).....	58
Figure 41 Chargement et déchargement des chariots par destination par rapport au temps pour les stations 4 et 8 sur le robot – SEBC 1 – journée du 17. ....	59
Figure 42 Processus logique présentant les règles heuristiques .....	70
Figure 43 Vecteur de paramétrage .....	70
Figure 44 Schéma représentatif des interactions d'un agent dans son environnement (Guériaux, 2016).....	76
Figure 45 Les agents cognitifs vs réactifs (Reichgelt, 1990) .....	76
Figure 46 Modèle multi-agent pour le système multi-sites.....	79
Figure 47 Méthodologie de développement d'un SMA.....	82
Figure 48 Etapes de la méthodologie de conception adoptée .....	83
Figure 49 Représentation des blocs d'activités dans le diagramme d'activité.....	86
Figure 50 Modèle de Buts .....	87
Figure 51 Modèle de Rôles .....	88
Figure 52 Modèle d'agents.....	89
Figure 53 Diagramme d'activité .....	90
Figure 54 Évaluation de différents outils SMA (Garneau and Delisle, 2002) .....	92
Figure 55 Paradigme d'envoi de messages dans JADE .....	93
Figure 56 Architecture multi-agents – JADE (Metawei et al., 2012) .....	94
Figure 57 Éléments principaux de la plateforme multi-agents sur JADE et leurs relations. (Bellifemine et al., 2007).....	94
Figure 58 Modélisation UML de la hiérarchie des comportements dans JADE.....	95
Figure 59 Association à AD .....	100
Figure 60 Modifier le statut d'AC.....	100
Figure 61 Routage d'AC .....	101
Figure 62 Envoyer AC vers le robot .....	101
Figure 63 Affecter AC à une place sur le robot .....	101
Figure 64 Gestion de popularité d'AD .....	102
Figure 65 Vérifier si AD est toujours candidat .....	103
Figure 66 Envoyer AD au robot.....	103
Figure 67 Envoyer AD au chantier manuel.....	103
Figure 68 Se désassocier d'AD .....	103
Figure 69 Gérer la capacité de l'anneau de tri.....	103
Figure 70 Affecter AD à la station Flexibac appropriée .....	103

## Table des figures

---

Figure 71 Gérer la permission .....	104
Figure 72 Vérifier la disponibilité du convoyeur Flexibac .....	104
Figure 73 Ajouter AC à une station Flexibac .....	104
Figure 74 Retirer AC de la station Flexibac .....	104
Figure 75 Architecture d'implémentation de l'émulateur .....	106
Figure 76 Serveur OPC – paramètres Flexibac .....	107
Figure 77 Main-Container sur JADE .....	107

*Liste des tableaux*

Table 1	Table de synthèse des domaines de résolution des problématiques de PAP .....	38
Table 2	Tableau comparatif des travaux antérieurs.....	40
Table 3	Règles d'affectation de l'expert .....	50
Table 4	Règles d'affectation - SEBC 2 - plan de tri de 24 heures.....	51
Table 5	Logiciels de simulation. Source (Sun et al., 2012) .....	52
Table 6	Paramétrage du système modélisé .....	53
Table 7	Vecteur de paramétrage pour la validation – Journée du 15/10 .....	56
Table 8	Pourcentage de bas vers le robot sur FlexSim et sur Excel pour le SEBC 2 .....	56
Table 9	Pourcentage de bacs vers le robot – SEBC 1.....	58
Table 10	Pourcentage de bacs vers le robot - SEBC 2 .....	58
Table 11	Pourcentage de bacs vers le robot - configuration 2Bis .....	59
Table 12	Sensibilité du SEBC 2 .....	60
Table 13	Taux d'utilisation des robots .....	63
Table 14	Résultats de la résolution hors-ligne .....	67
Table 15	Comparaison des règles heuristiques .....	69
Table 16	Variation du pourcentage de conteneurs par rapport à $I_{max}$ .....	71
Table 17	Variation du pourcentage de conteneurs par rapport à $\alpha$ .....	72
Table 18	Variation du pourcentage de conteneurs par rapport à $A$ .....	72
Table 19	Pourcentage de conteneurs avec : ( $\alpha = 8$ , $A = 16$ et $I_{max} = 200$ ).....	72
Table 20	Table de comparaison des méthodologies de conception de SMA .....	80
Table 21	Tableau de catégorisation des agents selon le type de perturbation .....	97
Table 22	Tâches réalisées par chaque agent et le type de comportement sur JADE .....	97
Table 23	Table représentatif des interactions entre les agents.....	105
Table 24	Comparaison des résultats .....	108

# Introduction générale

## I. Contexte

Cette thèse a été réalisée dans le cadre d'un contact CIFRE bilatéral entre l'entreprise La Poste et le Laboratoire des Sciences du Numérique de Nantes (LS2N) de l'Université de Nantes. La Poste est une société anonyme française leader dans le secteur de livraison. Les activités du groupe La Poste sont organisées en cinq branches : Services Courrier Colis, GeoPost, Réseau La Poste, La Banque Postale et Branche numérique. Les travaux de thèse ont été faits en collaboration entre le département Avant-projet innovant au sein de la Direction technique de La Poste branche Services Courrier Colis sous la responsabilité de Didier THIERIOT et Loïc BRUGGEMAN, et l'équipe Pilotage des Systèmes Industriels (PSI) du laboratoire LS2N sous la codirection d'Olivier CARDIN, Nasser MEBARKI et Dalila TAMZALIT.

## II. Problématique industrielle

L'industrie 4.0 se définit comme la nouvelle mutation technologique des systèmes automatiques basée sur l'augmentation remarquable de la vitesse de traitement de l'information, de la capacité de stockage numérique et du développement massif des systèmes de communication. Cette révolution prometteuse a pour but de changer le mode de fonctionnement des usines par l'introduction massive des nouvelles technologies d'information et de communication, pour n'en citer que quelques-unes : l'Internet des objets (IoT), l'intelligence artificielle (IA), le Cloud Computing, le Big Data et les systèmes cyber-physiques. Le concept de l'usine du futur n'est pas seulement basé sur l'optimisation des solutions existantes, mais il tend aussi à intégrer de réelles transformations conduisant à des solutions innovantes dans la société, l'économie, l'interaction homme-machine et la manière dont le travail est planifié et organisé dans une chaîne de production.

Aujourd'hui, les entreprises logistiques, comme toutes les autres entreprises, sont confrontées à certains défis dans l'utilisation des nouvelles technologies pour assurer leurs transitions vers l'industrie 4.0. Parmi ces entreprises, on peut citer La Poste.

Depuis sa création au XVII<sup>e</sup> siècle, La Poste a subi beaucoup de changements groupés principalement en trois axes de recherche : le domaine du transport, y compris la collecte et la livraison ; le domaine du tri, y compris la reconnaissance des caractères manuscrits ; et le domaine de l'ordonnement du personnel (Selma et al., 2018). Aujourd'hui, La Poste est confrontée à de profonds défis : la baisse du volume de courrier, accompagnée d'un nombre

croissant de concurrents sur un marché mondial bien développé qui regorge de nouvelles technologies de communication telles que Facebook, SMS, e-mails, fax, etc. Pour assurer sa place sur le marché, La Poste a décidé de mettre en œuvre les paradigmes de l'Industrie 4.0 en employant de plus en plus de solutions innovantes dans ses centres de tri.

La problématique principale rencontrée par les centres de tri concerne la grande variabilité des courriers à traiter chaque jour. Même si certaines tendances peuvent apparaître en fonction du jour de la semaine, il est nécessaire que les centres s'adaptent aux différents volumes de courriers déposés par les clients. Pour résoudre ce problème, les centres sont organisés suivant des plannings horodatés rigides, permettant de synchroniser les différentes tâches (arrivées du courrier, aiguillage vers le tri, tri, aiguillage vers les camions, départ).

Si ces plannings ont prouvé leurs performances dans des conditions normales de fonctionnement, ils ne tolèrent que peu de perturbations (pannes, retards routiers, etc.). Si de telles perturbations apparaissent, ce sont les compétences métier des opérateurs et des gestionnaires de lignes qui permettent d'amortir l'effet de la perturbation sur le fonctionnement global du système. Un levier d'amélioration de la performance industrielle de l'opération de tri est donc son adaptabilité aux conditions imprévues.

En parallèle, le groupe encourage la robotisation de procédés de manutention pour des raisons de santé au travail. L'utilisation de robots a depuis longtemps été réservée au secteur industriel pour des applications nécessitant beaucoup d'opérations répétitives. Aujourd'hui, l'industrie logistique emploie de plus en plus de solutions robotiques afin d'offrir plus de confort aux opérateurs en remplaçant les tâches manuelles les plus fastidieuses, répétitives et sources d'erreurs et de troubles musculosquelettiques. Parmi celles-ci, les tâches de prise-dépose, autrement appelées « pick-and-place » (PAP) sont spécifiquement étudiées ici.

La Poste rencontre ces tâches dans ses centres de tri et de distribution de courriers, où des opérateurs manipulent chaque jour des milliers de conteneurs contenant les courriers (près de 30 000 conteneurs par jour en moyenne). La **Figure 1** présente les deux types de conteneurs standardisés de La Poste : les conteneurs contenant les courriers de petit format (couleur orange) et ceux contenant les courriers de grand format (couleur grise). Ces conteneurs ont un poids pouvant dépasser les 20 kg. Les opérateurs doivent les déposer selon leurs destinations (affichées sur des étiquettes collées sur les conteneurs, **Figure 3**) dans des chariots standardisés mobiles (**Figure 2**). Un chariot plein contient 21 conteneurs de courriers de petit format, 15 conteneurs de courriers de grand format. Il peut aussi contenir un mélange des deux types de conteneurs. Dans ce cas, il ne faut pas dépasser la hauteur maximale du chariot.

Les chariots sont ensuite déplacés manuellement vers des zones dites « faux camions » près des quais de départ. Cette opération facilitera par la suite leur chargement dans des camions vers d'autres centres de tri ou de distribution de courriers. L'organisation des opérations de chargement des conteneurs sur les chariots a un impact direct sur les centres de distributions et donc sur toute la chaîne logistique.



**Figure 1** Conteneurs - format standardisé La Poste



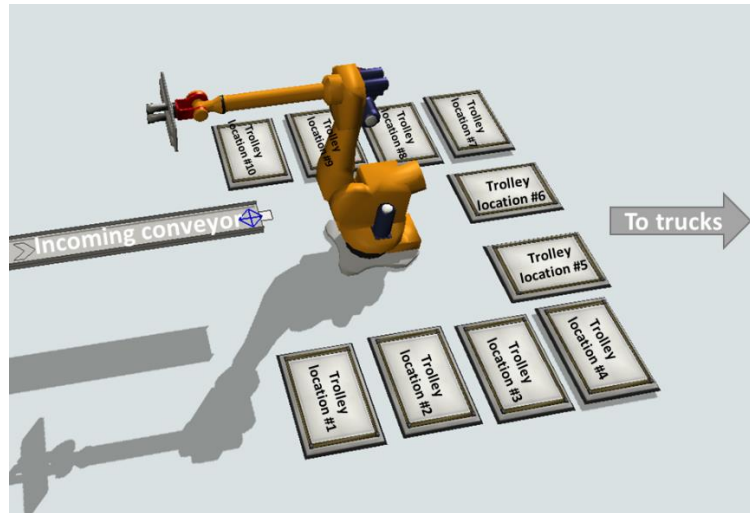
**Figure 2** Chariot de courrier - format standardisé de La Poste



**Figure 3** Etiquette de destination d'un conteneur

Pour réduire la pénibilité de la tâche répétitive du chargement des conteneurs manuellement dans les chariots, La Poste a décidé d'installer un système robotisé de type PAP pour automatiser une partie du processus de chargement des conteneurs du convoyeur vers les chariots. Ce système robotisé est constitué d'un robot poly-articulé qui peut desservir jusqu'à 10 stations de chargement (**Figure 4**). Chaque station contient un chariot qui correspond à une destination.





**Figure 4** Cellule de chargement automatique

La difficulté rencontrée par le partenaire industriel se concentre sur le pilotage du système robotisé, et notamment l'affectation des destinations aux stations de chargement. Quand un chariot atteint sa capacité maximale de chargement, il est vidé et remplacé manuellement par un autre chariot vide. Comme un centre de tri postal envoie en moyenne 30 000 conteneurs par jour, avec plusieurs centaines de destinations différentes potentielles, La Poste doit déterminer quels conteneurs à affecter à la cellule robotisée afin de maximiser la charge de travail du robot. Cette affectation doit prendre en compte plusieurs contraintes industrielles, parmi lesquelles :

- i) Deux conteneurs ayant deux destinations différentes ne peuvent pas être dans le même chariot ;
- ii) Capacité de stockage limitée de l'anneau de tri ;
- iii) Temps de remplacement de chariot ;
- iv) Seuil de remplissage minimal et maximal d'un chariot.

Or, face à la variabilité du volume de courrier entrant pour chaque destination, il est difficile de procéder par planning rigide pour maîtriser la rotation des destinations affectées aux stations. De plus, le système étant totalement autonome, il ne peut profiter de l'amortissement des perturbations généralement amené par l'expertise des opérateurs.

Le problème rencontré par La Poste est un problème scientifique dénommé « problème de type pick-and-place ». La première partie de ces travaux propose un état de l'art des méthodes de résolution de ce problème dans la littérature. Cet état de l'art montre que ce problème est traditionnellement résolu par des règles simples basées sur des méthodes de recherche opérationnelle. Nous allons montrer dans la seconde partie de ces travaux que cette approche ne satisfait pas aux exigences du partenaire du fait de la taille du problème et de l'incertitude sur les données d'entrée quotidienne, mais que certaines heuristiques peuvent être performantes. Nous avons alors intégré ces heuristiques sous la forme de règles de décisions dans un modèle de simulation de flux. Les résultats ont montré l'intérêt de cette approche par rapport aux plannings traditionnels du partenaire. Un modèle multi-agents, intégrant ces règles de décisions simulées précédemment, a été développé. L'originalité de ces travaux est de

démontrer la pertinence de ces solutions en allant jusqu'à une programmation pouvant être couplée au système réel. La résolution du problème industriel de type PAP d'un centre de tri postal par système multi-agent constitue la principale contribution des travaux consignés dans cette thèse.

### III. Organisation du document

Ce document est divisé en cinq chapitres :

- Le chapitre 1 présente une explication détaillée de la problématique industrielle, ainsi qu'une présentation du fonctionnement actuel du système étudié.
- Le chapitre 2 présente un état de l'art sur les systèmes de type PAP. Il détaille les principales problématiques liées à l'utilisation des systèmes de PAP présentes dans la littérature ainsi que les solutions proposées pour les résoudre.
- Le chapitre 3 présente la première méthode choisie pour réaliser l'étude préliminaire, à savoir le modèle expert à base de connaissance. Les modèles construits sont évalués par simulations en utilisant des données réelles.
- Le chapitre 4 présente une résolution exacte basée sur les techniques de la recherche opérationnelle et une approche basée sur des heuristiques pour la détermination des règles d'affectation. Les différents avantages et inconvénients de ces méthodes seront présentés.
- Le chapitre 5 traite la résolution du problème PAP à l'aide d'un système multi-agent. L'implémentation du SMA ainsi que quelques pistes d'optimisation seront détaillées.

### IV. Publications

- Cyrine Selma, Dalila Tamzalit, Nasser Mebarki, Olivier Cardin, Loic Bruggeman, Didier Thiériot: **Industry 4.0 and Service Companies: The Case of the French Postal Service.** *SOHOMA 2018*: 436-447
- C. Selma, S. Thévenin, N. Mebarki, O. Cardin, D. Tamzalit, D. Thiériot, L. Bruggeman. **Heuristics for Robots-Humans Tasks Assignment in a Containers Loading Center.** In IMS 2019: 13th IFAC Workshop on Intelligent Manufacturing Systems, août 2019, Oshawa, Canada.
- Niati, A., Selma, C., Tamzalit, D., Bruneliere, H., Mebarki, N., Cardin, O., 2020. Towards a digital twin for cyber-physical production systems: a multi-paradigm modeling approach in the postal industry, in: Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings, MODELS '20. Association for Computing Machinery, New York, NY, USA, pp. 1–7. <https://doi.org/10.1145/3417990.3421438>

# Chapitre 1

## 1 Problématique

L'un des principaux enjeux de l'industrie 4.0 est l'implication du client dans l'expérience de production. L'être humain est considéré comme le cœur de la quatrième révolution industrielle. C'est ainsi que l'amélioration de l'expérience de l'humain dans son lieu de travail est devenue une obligation et non plus une option au sein de l'industrie 4.0. Les usines du futur ont besoin de systèmes de contrôle de la production qui est à la fois scalable, flexible et reconfigurable. Des systèmes de contrôle qui leur permettent de renforcer la communication entre l'homme et les machines intelligentes. Cela, permettra aux humains de superviser les missions et de collaborer plus facilement avec les machines grâce à des interfaces plus sophistiquées, d'améliorer les conditions de travail et de résoudre plus rapidement les problèmes. Aujourd'hui, avec les nouvelles technologies de l'information et de la communication, les entreprises de logistique comme La Poste commencent à intégrer de plus en plus de solutions robotiques. La robotisation reste l'une des technologies les plus prometteuses pour améliorer les processus de production. Les robots sont utilisés pour remplacer les tâches manuelles les plus pénibles, répétitives et dangereuses. Le but est d'améliorer la santé et la sécurité des opérateurs.

Dans ce chapitre, nous allons présenter brièvement la problématique industrielle traitée dans cette thèse. La première partie détaillera le fonctionnement actuel d'un centre de tri postal. La problématique ainsi que le besoin de l'industriel seront présentés dans la deuxième partie.

### 1.1 Fonctionnement actuel

La Poste, premier distributeur de courrier en France, possède aujourd'hui une trentaine de centres de tri appelés aussi Plateformes Industrielles Courriers (PIC). Ces PICs représentent la plus grande chaîne d'acheminement de courrier en France. Elles se situent principalement à proximité des grandes villes. Elles ont été créées dans le but d'industrialiser la distribution de courrier. Des milliers de courriers de différents formats sont traités tous les jours par les machines de tri dans les PICs.

Différents types de machines pour traitement de courriers de petit ou grand format existent dans un centre de tri. Leurs nombres dépendent de plusieurs paramètres comme les dimensions de chaque centre de tri, la capacité de tri, l'emplacement géographique, les liaisons avec les autres centres de tri, etc. Dans le centre de tri de Paris sud Wissous par exemple, il

existe 11 machines de tri du courrier de petit format MTIPF (**Figure 5**), 3 machines de tri du courrier de grand format MTIGF, 2 machines de tri préparatoire MTP et 12 machines de tri séquentiel de distribution TTF pour les courriers de petit format (tri fin selon la tournée de facteur).

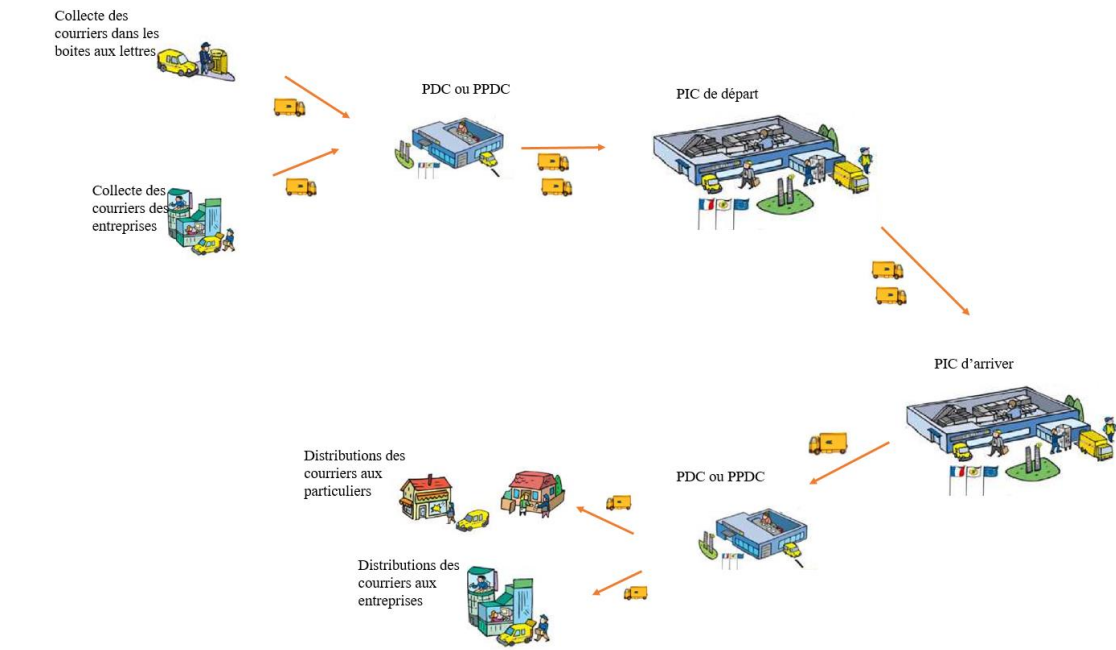


**Figure 5** Machine de tri petit format

Le processus de tri, comme le montre la **Figure 6**, commence par la collecte de courrier par le facteur dans les boîtes aux lettres et dans les locaux des entreprises pour le courrier professionnel. En respectant les délais, ces courriers sont acheminés vers les Plateformes de Distribution de Courriers PDC qui existent en plusieurs milliers dans tout le pays ou vers les Plateformes de Préparation de de Distribution de Courrier PPDC les plus proches. Dans la phase de départ, la PDC a pour unique mission de collecter le courrier et l'expédier au PPDC. Ni la préparation ni le contrôle des flux de courrier ne font partie des missions assignées au PDC.

Après leurs transports au PPDC, les courriers subissent un premier tri préparatoire. Dans la phase, les courriers sont séparés par destination, par format (petit, grand ou hors format) et par qualité d'expédition (jour de livraison J+1, J+2, J+4 ou J+7).

La mission principale de la PPDC dans la phase de départ est la préparation des courriers et leur transport vers les PICs. Les PPDC existent en plusieurs centaines dans toute la France.



**Figure 6** Processus d'expédition de courrier (collecte, tri, distribution)

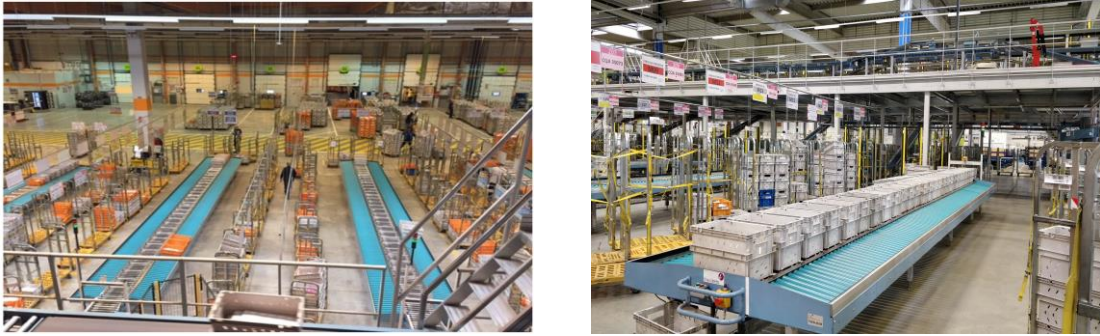
Dans un centre de tri PIC, les courriers sont triés en fonction de leur date de livraison : les courriers standards (J+4/7) par exemple ne sont triés que la veille de leur jour de livraison. Dans le cas contraire, les bureaux de distribution, qui disposent généralement de petits entrepôts, seraient saturés.

Plus de 90 % des envois postaux sont mécanisés (triés par des machines) dans une PIC, et plus de 50 % sont triés finement et séquencés dans l'ordre de distribution à l'aide des différents types de machines. Toutes les machines sont équipées d'un lecteur optique de caractères capable de lire les adresses manuscrites en ligne dans les courriers postaux. En cas de non-reconnaissance de l'adresse, la machine envoie en temps réel une image de l'adresse vers une salle de vidéo-codage locale où l'adresse est vérifiée par des humains et envoyée au système après avoir été reconnue.

À l'intérieur du centre de tri du courrier, le processus de tri du courrier est programmé jour par jour et dépend du volume et du type de flux de courrier. Après l'arrivée des envois par camion, la première étape consiste à les expédier en séparant les petits formats des grands formats. Pour les petits formats, le cachet des lettres doit être vérifié à l'aide du MTP (si cela n'a pas été fait dans le centre de préparation du courrier), puis tous les petits formats sont stockés selon leur plan de tri non loin des MTIPF.

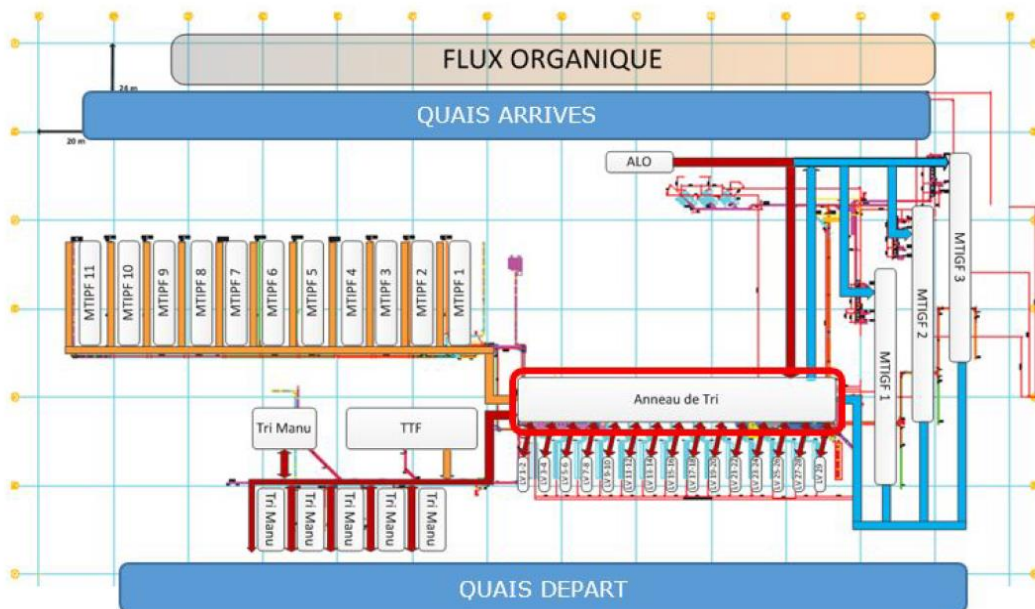
Les lettres de grand format sont également stockées selon leur plan de tri à côté des MTIGF et les courriers triés manuellement sont expédiés vers le chantier manuel.

À la sortie des machines de tri, les courriers sont disposés dans des conteneurs, appelés bacs contenant les courriers. Ces bacs sont disposés sur un convoyeur en boucle, appelé anneau de tri (**Figure 9**), et dirigés vers des emplacements sur une des lignes de descente, appelées Convoyeur de Ventilation CV selon leurs destinations.



**Figure 7** Lignes de ventilation dans un centre de tri postal (depuis l’anneau de tri à gauche et depuis l’extrémité à droite).

Le nombre des CV dépend de la taille de la PIC et de l’importance de son flux de courrier journalier. Chaque CV est constitué de 2 convoyeurs, appelés Lignes de Ventilation (LV), comportant chacune jusqu’à 15 emplacements physiques de prise pour les opérateurs (**Figure 7**). Ceux-ci manutentionnent les conteneurs vers des chariots standards à roulettes. Par conséquent, ils facilitent leur transport en camion vers d’autres plateformes. Le cas échéant, les conteneurs seront de retour vers des machines de tri pour des opérations de tri plus complexes nécessitant de multiples passes ou vers des chantiers de tri manuels en cas de problème sur l’automatisation du tri (adresse illisible par exemple). Le débit de l’anneau de tri est de 12000 conteneurs /heure et celui d’une ligne de ventilation est de 1200 conteneurs /heure.

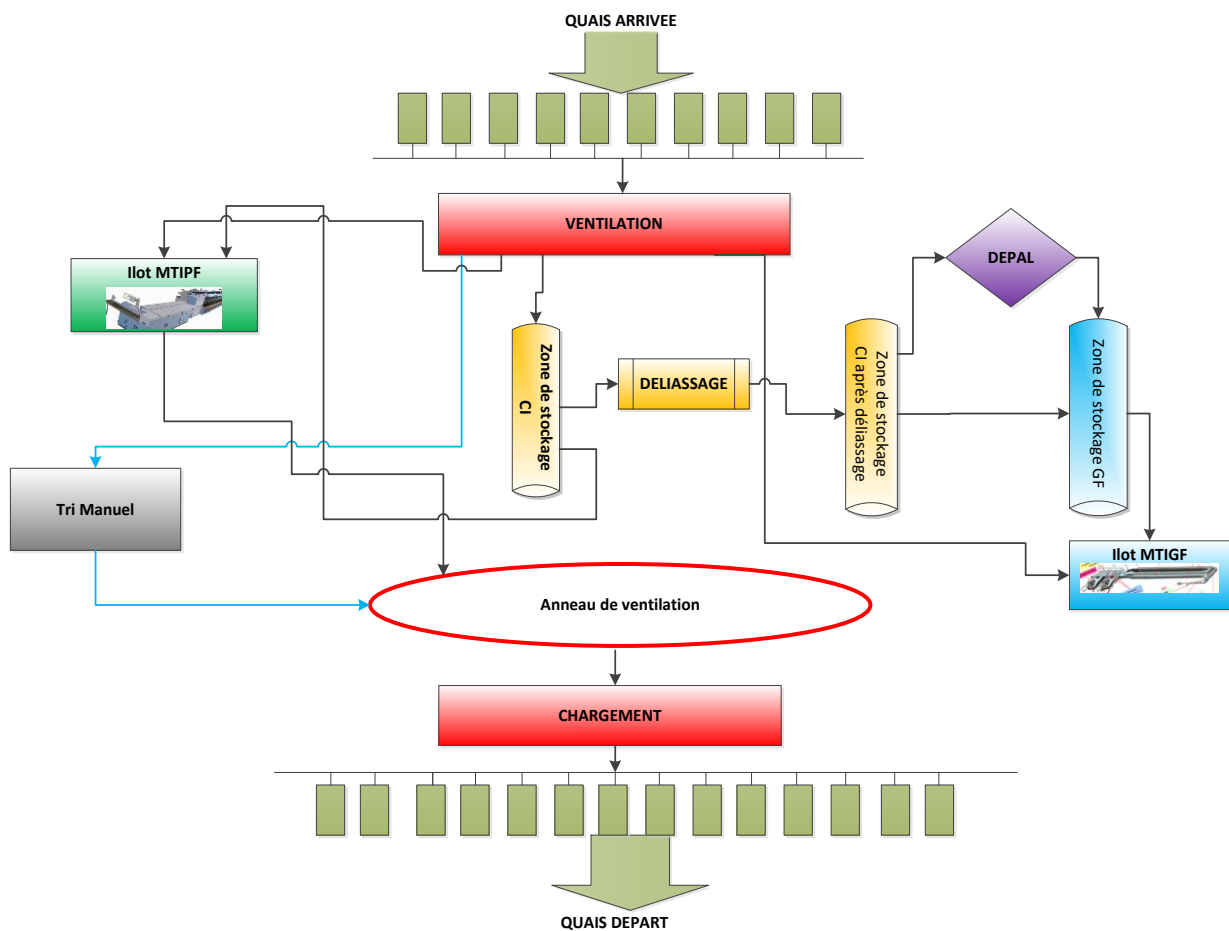


**Figure 8** Organisation d’un centre de tri postal

Dans le fonctionnement actuel, qui demeure totalement manuel, les opérateurs viennent charger les conteneurs vers le bon chariot selon leurs destinations, qui sont directement

indiquées sur une feuille attachée au-dessus de l'emplacement de chaque chariot. Les opérateurs répètent cette tâche tout au long de la journée postale. Un plan de transport journalier organise l'évacuation des machines de tri plusieurs fois par jour avant le départ du camion correspondant. De ce fait, beaucoup de conteneurs peuvent sortir sur les LV dans un intervalle de temps très court. Cela, accroît encore plus la pénibilité de cette tâche pour les opérateurs.

En attendant l'heure de départ camion, les opérateurs amènent les chariots pleins pour les déposer sur des zones de stockage temporaire, proches des quais de départ (**Figure 8**), appelées « faux camions » (**Figure 10**). Cette zone a la même surface qu'un camion ce qui permet de gagner le temps, d'éviter la surcharge des camions et de bien répartir les chariots dans les camions.



**Figure 9** Parcours du flux de courrier entre les quais d'arrivées et les quais de départs



**Figure 10** Quai de départ dans un centre de tri postal

### 1.2 Besoin

Pour réduire la pénibilité de la tâche de chargement manuel des bacs dans les chariots et diminuer le temps total de chargement, La Poste propose d'automatiser le chargement des chariots pour les destinations les plus chargés en nombre de conteneurs qui représentent une partie non négligeable du trafic journalier (20 % des directions représentent 50 % du trafic sur une journée-type pour le centre de tri de Paris Sud Wissous.).

Un appel d'offres a été lancé. Trois modèles de robots de changement automatiques ont été proposés par différentes entreprises. L'offre retenue selon le meilleur rapport de performance/prix est celle proposée par la société OSE.

La solution proposée consiste à implanter des modules de chargement automatique sur différentes lignes de ventilation. Chaque module est constitué d'un bras robotique et de 8 à 10 emplacements de chariots en supplément des emplacements déjà prévus pour le chargement manuel. Les robots doivent remplir les chariots des destinations de plus fort trafic, ce qui implique une rotation dynamique des destinations assignées aux robots compte tenu du fait que le trafic des destinations fluctue fortement au cours de la journée. Des opérateurs sont chargés d'évacuer les chariots pleins et les remplacer par des chariots vides au niveau des emplacements robotisés.

Si les flux sont relativement stables et que des journées-type puissent être identifiées (des effets de saisonnalité existent.), la variabilité quotidienne rend la tâche d'optimisation hors ligne difficile et très sensible à ces variations autour de la journée-type.

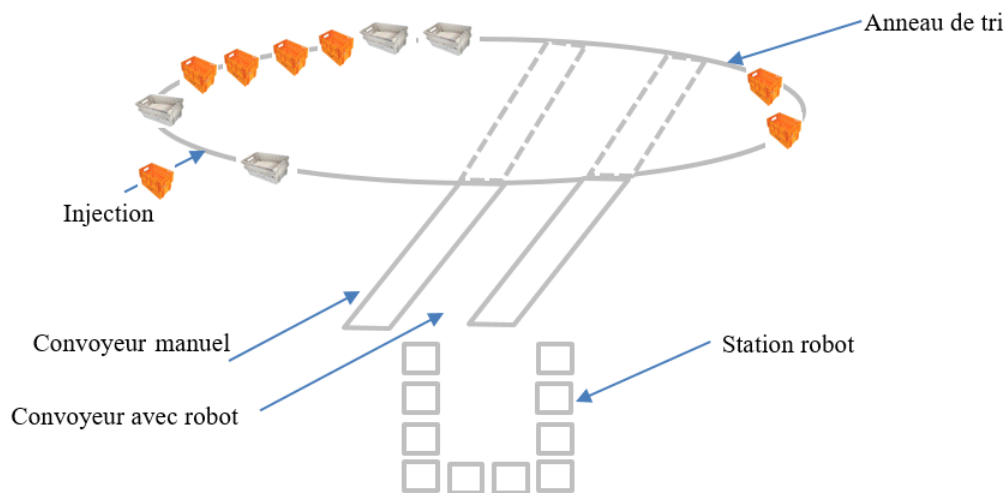


Afin de rentabiliser le fonctionnement de ces modules de chargement automatique et d'augmenter le trafic journalier pris en charge par les robots, les règles d'affectation des destinations à robotiser doivent être soigneusement choisies.

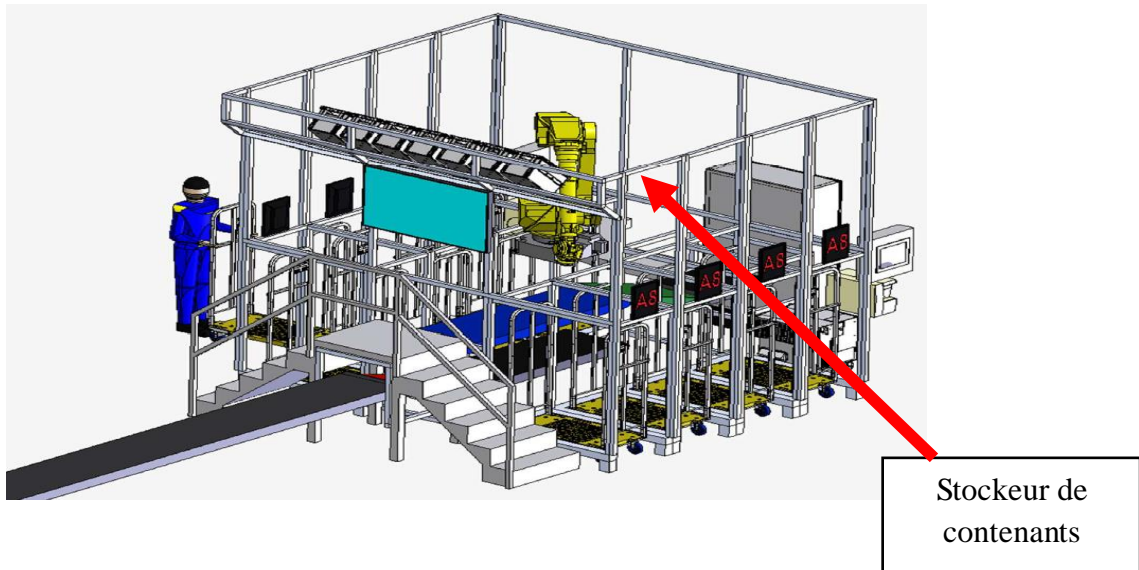
L'objectif est de se rapprocher de 50 % des flux robotisés grâce à une unité de chargement automatique. Le programme d'affectation à définir doit prendre en compte l'effort et le temps que les opérateurs mettent pour changer les chariots à chaque fois que la destination change. Pour cela, le robot doit charger au maximum les chariots avant de changer de destination afin de diminuer le nombre de changements.

### 1.3 Sous système (Présentation de flexibac)

Lorsqu'un conteneur sort de l'anneau de tri, il arrive dans la cellule en passant de côté de la flèche rouge (**Figure 12**). Dès lors son étiquette est lue pour déterminer s'il est destiné à être chargé par le robot ou bien manuellement par un opérateur (**Figure 11**). Dans le premier cas, le contenant reste en position et sera donc chargé automatiquement par le robot à 6 axes (**Figure 13**) dans l'un des chariots des deux côtés du convoyeur, si ce n'est pas le cas, l'aiguilleur vert se met en position basse et le contenant continuera son chemin sur le convoyeur pour être chargé manuellement par la suite par l'un des opérateurs.

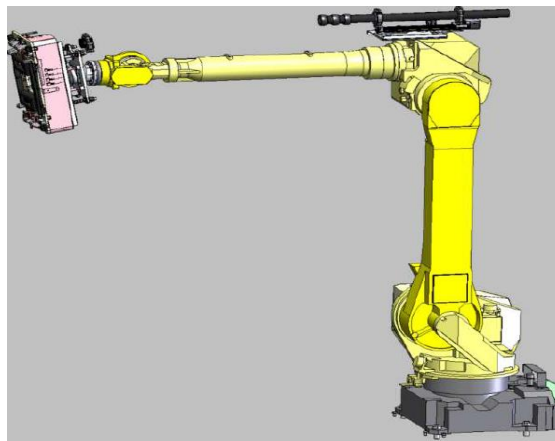


**Figure 11** Schéma simplifié de l'anneau de tri avec la cellule robotisée



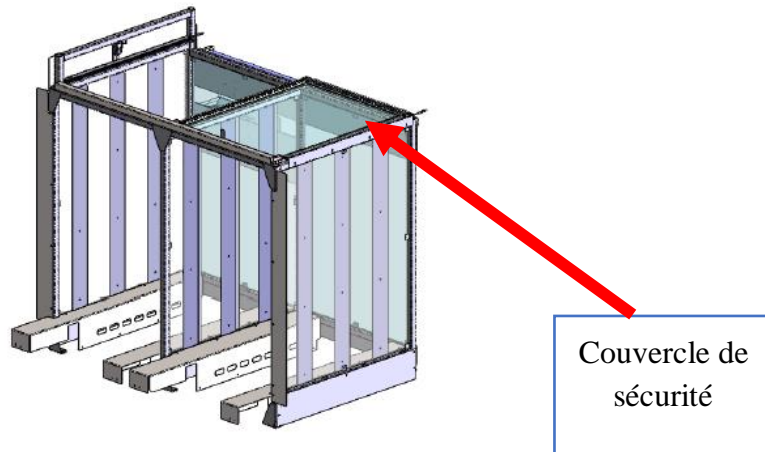
**Figure 12** Cellule de chargement automatique - Vue aval

Quand un chariot est plein, le robot dépose verticalement un couvercle sur son emplacement (**Figure 14**). Ensuite, la porte guillotine qui sert aussi à sécuriser l'environnement de travail s'ouvre pour permettre à l'opérateur de remplacer le chariot plein par un autre vide (**Figure 15**). Pendant ce temps, le robot continue à charger normalement les autres chariots. Le numéro et la destination de chaque chariot sont affichés sur la station d'indexage par un afficheur basique.



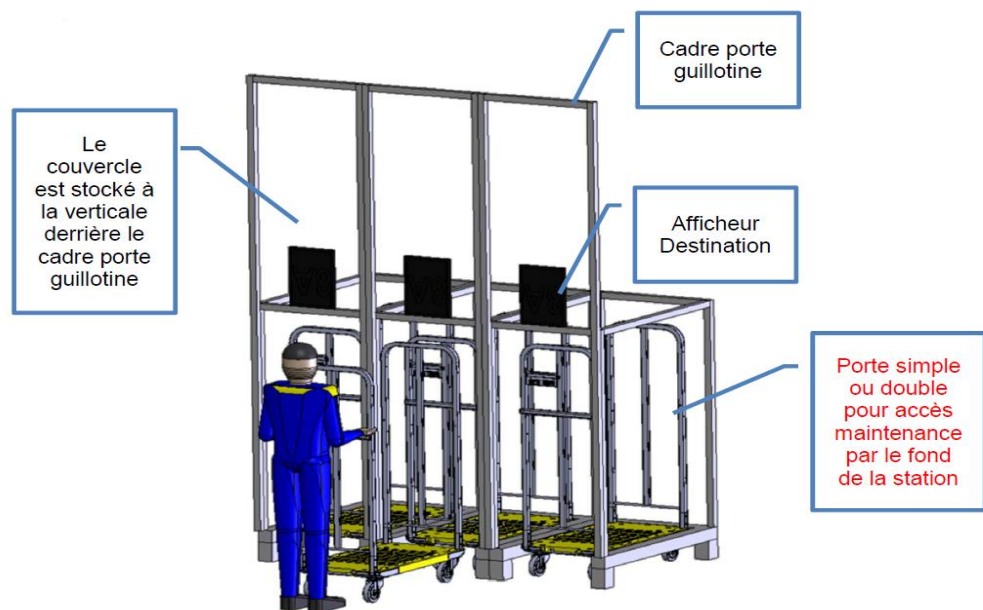
**Figure 13** Robot 6 axes de manipulation de contenants

Lorsqu'un contenant destiné à un chariot en cours d'évacuation arrive dans la cellule, il sera stocké dans un espace de stockage précis (**Figure 12**) grâce à la capacité de stockage de la cellule de 7 contenants. Le robot vide son stock par priorité.



**Figure 14** Couvercle de sécurité

À la coupure partielle ou totale d'un plan de tri sur l'une des machines à cause d'un proche départ camion, il est possible d'avoir un pic de contenants sur l'anneau de tri. Dans ce cas si la capacité d'absorption de la cellule est inférieure au flux entrant, la cellule enverra un signal à l'anneau de tri pour qu'il conserve une partie du flux ou bien si cela est possible, d'orienter une partie du flux vers d'autres brins de ventilation.



**Figure 15** Emplacement des chariots dans la cellule de chargement

## 1.4 Conclusion

Outre la réduction des tâches manuelles répétitives, la robotisation permet d'améliorer la santé et la sécurité des opérateurs. Cependant, les robots sont coûteux, et ils doivent être utilisés à haute capacité pour être rentables.

Contrairement aux robots, les humains sont flexibles et capables d'effectuer un large éventail de tâches. Par conséquent, les entreprises modernes utilisent des robots pour gérer le flux régulier d'opérations, tandis que les travailleurs humains gèrent dans des situations plus particulières (pics de production, pannes, changement de la planification, etc.). Dans ce contexte, l'affectation des tâches aux robots et aux humains doit être soigneusement gérée.

Dans ce chapitre, nous avons présenté en détail le centre de tri postal ainsi que le problème de gestion des flux de courriers traités dans ce dernier. Le chapitre suivant présentera un état de l'art détaillé sur les problématiques et les principales méthodes employées pour la résolution des systèmes robotiques de PAP.

## Chapitre 2

### 2 Etat de l'art

Au sein des flux logistiques, la manutention répétée de charges lourdes par les opérateurs est un problème identifié pouvant mener rapidement au développement de différents troubles de santé. Afin de limiter ces risques, l'une des solutions proposées est d'intégrer des systèmes robotisés permettant de traiter tout ou partie de ces flux de manutention. Parmi ces solutions robotiques, on distingue les systèmes de manutention de type Pick-And-Place (PAP). Ces systèmes sont de plus en plus utilisés dans l'industrie logistique, en particulier dans les centres de tri et de distribution de marchandises.

Plusieurs études dans la littérature ont été menées sur la résolution des problèmes de type PAP. Ce chapitre va présenter une définition générique des systèmes de PAP ainsi qu'une revue détaillée des différentes problématiques et méthodes de résolution et d'optimisation des systèmes de PAP. La majorité de ces travaux sont réalisés dans un contexte académique. Une introduction à notre système industriel de PAP sera présentée à la fin de ce chapitre.

#### 2.1 Présentation générale du problème de PAP

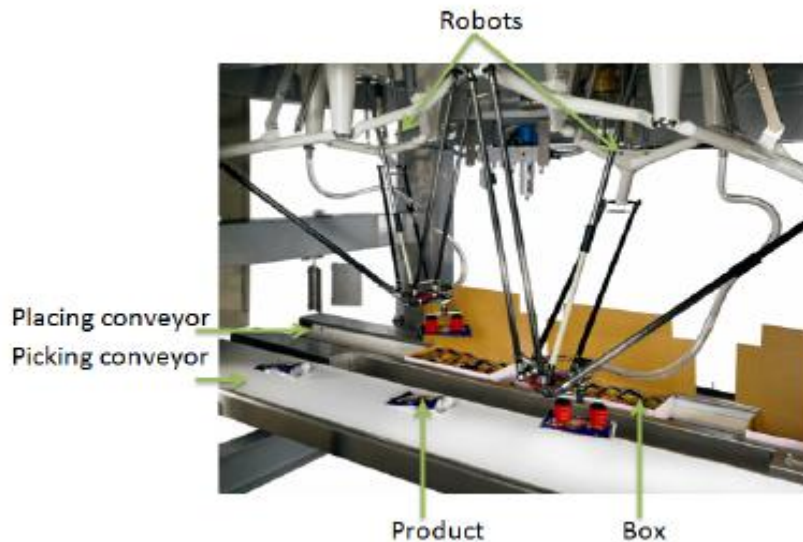
Avec la diminution des coûts d'investissement liés à la robotisation, les champs d'application se sont étendus et ont vu une augmentation des opérations dites de PAP. Ce type d'application a été étudié dans de nombreux travaux, plus fréquemment pour le placement à grande vitesse et avec une grande précision des composants électroniques sur les cartes de circuits imprimés (PCB). Ces cartes sont utilisées dans la majorité d'équipements industriels.

Une application de PAP est un système qui, en utilisant un robot, saisit de manière sélective un ou plusieurs objets d'un emplacement A et les dépose dans un emplacement B (Drezner and Nof, 1984 ; Schubert, 2000).

Le problème associé a été présenté dans les travaux de (Mattone et al., 2000) comme étant un système robotique composé d'un ou de plusieurs robots installés en chaînes les uns après les autres. Les robots saisissent des produits sur un convoyeur en mouvement et les déposent dans des emballages sur des points de dépôt. Les produits arrivent sur une ou plusieurs voies selon la configuration des convoyeurs (Humbert et al., 2015) (**Figure 16**).

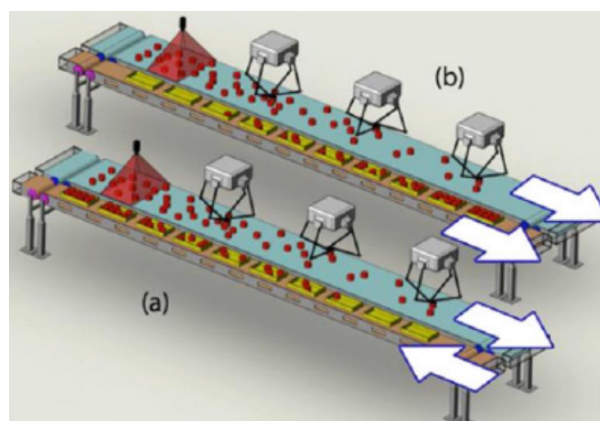
Les convoyeurs peuvent avoir deux configurations selon le type d'application : i) configuration parallèle ; ii) configuration circulaire (Briand and Parlouar, 2019). La

configuration circulaire offre plus de flexibilité au système puisqu'elle permet au produit/emballage de passer plusieurs fois devant le robot. Cela augmente les chances pour que le produit soit pris par le robot. Toutefois, cette configuration est la moins utilisée à cause de son coût élevé.



**Figure 16** Cellule robotisée de type PAP avec convoyeurs parallèles (Humbert et al., 2015).

Quant à la configuration parallèle, elle est moins complexe, mais elle requiert plus de gestion pour bien s'assurer du rangement du produit dans l'emballage adéquat. Pour cette configuration, il existe deux sous-configurations selon le flux : i) les convoyeurs à flux unidirectionnels ; ii) les convoyeurs à flux opposés. Les convoyeurs à flux opposés (**Figure 17**) sont plus avantageux, puisqu'ils permettent d'augmenter le nombre de produits collectés.

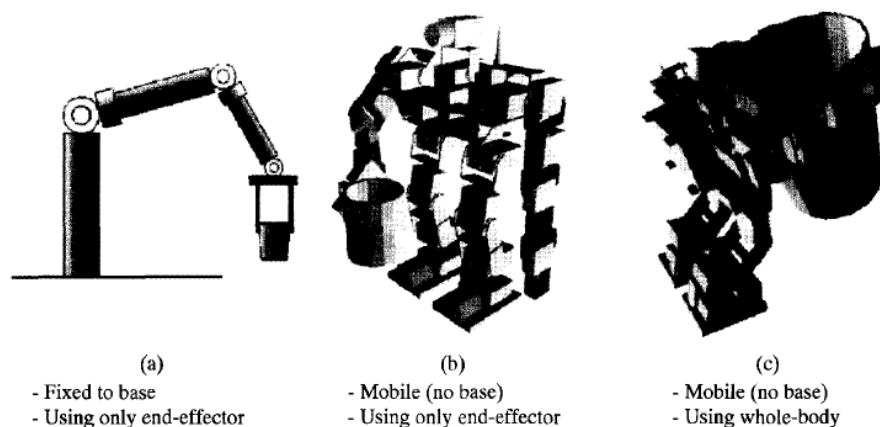


**Figure 17** Configuration à flux unidirectionnel - configuration à flux opposé (Briand and Parlouar, 2019).

Selon (Mizuuchi et al., 1999), il existe deux grandes familles de robots pour les opérations de PAP :

- Robots manipulateurs ou de manutention
- Robots humanoïdes

Comme le montre la **Figure 18 (a)**, dans la première famille, le robot a une base fixe et ne peut utiliser que son effecteur final pour réaliser une tâche. Au sein de la seconde famille, comme le montre la **Figure 18 (b, c)**, le robot n'a aucune base fixe et peut utiliser soit son corps entier soit seulement ses effecteurs finaux pour manipuler un objet.



**Figure 18** Différents types de robots de PAP (Mizuuchi et al., 1999).

Dans la famille des robots de manutention, il existe cinq types de robots selon (Smys and Ranganathan, 2019) :

- Bras robotisé : c'est le type le plus répandu de robot de PAP. Un bras robotisé à 5 axes est généralement utilisé pour les applications standards de PAP. Pour des applications plus complexes, les robots à 6 axes sont utilisés. En effet, plus le nombre d'axes augmente, plus le système devient flexible et plus la variété des tâches effectuées par le robot augmente.
- Cartésien : il a trois axes principaux de mouvement suivant des coordonnées prises dans un repère cartésien. Ce robot est capable de manipuler des pièces lourdes avec une grande précision de positionnement.
- Delta : également connu sous le nom de « Robot araignée ». Il a un bras de manipulation formé de parallélogrammes articulés reliés à une base commune. Les robots Delta sont très rapides (la légèreté de ses bras de manutention réduit considérablement l'inertie due à la masse, ce qui augmente la vitesse.). Ils sont dotés de technologies de vision avancées qui leur permettent de distinguer différentes tailles, formes et couleurs d'objets. Cela explique leur grande utilisation dans les industries alimentaire, pharmaceutique et électronique.
- Collaboratif : ils présentent de nombreuses similitudes avec les bras robotisés traditionnels. La seule différence est que les robots collaboratifs sont conçus pour travailler dans le même environnement de travail que les opérateurs humains. Ces

derniers peuvent généralement intervenir dans un mode dit « pantin » pour guider le robot dans chaque tâche.

- Prise rapide (Fast Pick) : ce type de robot est idéal pour des opérations de PAP de moyens ou grands volumes. Le robot est extrêmement rapide avec des temps de cycle pouvant atteindre 200 cycles par minute. Il est généralement utilisé pour la manutention de pièces légères et petites.

Bien que la majorité des travaux présents dans la littérature utilise des robots de manutention pour les opérations de types PAP, quelques travaux se sont intéressés à l'utilisation des robots humanoïdes. Les auteurs dans (Mizuuchi et al., 1999) ont ainsi créé un robot humanoïde de PAP avec un comportement adaptatif de prise d'objets. Le robot utilise son corps entier pour soulever et poser un objet sans avoir besoin de connaître son poids sa taille ou sa rigidité en assurant un comportement adaptatif, autonome et robuste durant toute l'opération.

Indépendamment du type de robot utilisé, la performance d'un système de PAP dépend en grande partie des stratégies et algorithmes de contrôle employés. Selon (Mattone et al., 1998), la liste des produits à saisir peut être considérée comme une file d'attente de clients à servir. Avec un temps de service qui varie dynamiquement, ces clients doivent être servis avant de quitter l'espace de travail du robot. Dans une situation standard où un seul robot est utilisé, et sous des conditions optimales (pas d'arrêt des convoyeurs, les emballages sont toujours disponibles, pas de retard d'arriver des produits, etc.), des règles simples de planification comme FIFO (*First In First Out*), LIFO (*Last In First Out*) et SPT (*Shortest Processing Time*) peuvent être utilisées vu leur facilité de mise en œuvre et la faiblesse du temps de calcul.

Toutefois, dans d'autres situations (Exemples : dysfonctionnement d'un robot dans un contexte multi-robots, temps d'attente important sur le convoyeur de dépose, retard d'arrivée des produits, etc.) des algorithmes plus complexes que la règle FIFO sont nécessaires pour l'ordonnancement des tâches de PAP, comme le démontrent (Mattone et al., 2000) sur l'influence de l'ordre du prélèvement des objets sur le convoyeur et sur la performance du système lors d'une opération de PAP.

Plusieurs stratégies de commande pour les systèmes de PAP sont présentes dans la littérature. Ces stratégies peuvent être séparées en trois catégories selon le type de problème rencontré :

- L'équilibre de la charge de travail entre les robots dans un système multi-robots
- L'optimisation du nombre de produits traités
- Génération de trajectoires.

## **2.2 Les méthodes de résolution des problématiques de PAP dans la littérature**

Cette section propose un focus sur une sélection d'articles de la littérature sur les systèmes de PAP utilisés dans un contexte académique (la majorité des articles) ou industriel. À notre connaissance (Hou et al., 2009) est le seul article qui étudie les systèmes de PAP dans un contexte industriel. Dans cette étude, toutes les opérations de prélèvement sont effectuées manuellement. Les opérateurs humains étant plus flexibles que les robots, le problème



d'ordonnement étudié est moins contraignant, et les résultats obtenus ne s'appliquent pas à un système robotique.

Nous présentons dans cette section les principales problématiques liées à l'utilisation des systèmes de PAP ainsi que les solutions proposées pour les résoudre. Les algorithmes cités dans cette section peuvent être séparés en trois catégories selon le type du problème étudié : (1) problème de génération de trajectoires ; (2) problème d'équilibrage de la charge de travail ; (3) problème d'optimisation du nombre de produits traités.

## **2.2.1 Résolution de problèmes de génération de trajectoire**

Dans cette section, nous présentons quelques méthodes issues de la littérature pour résoudre les problèmes de génération de trajectoires des systèmes de PAP.

### **2.2.1.1 Algorithmes génétiques pour le calcul des trajectoires sur une situation statique**

Les algorithmes génétiques ont été initiés par J. Holland dans les années 1970. Ces algorithmes sont dérivés des mécanismes d'évolution de la nature et ont pour but de trouver une solution approchée à un problème d'optimisation. Le temps de calcul est un paramètre important dans l'évaluation de ces algorithmes. En effet, comme le calcul de la fonction d'évaluation est répété plusieurs fois, le temps de calcul pour la recherche de la solution optimale doit être raisonnablement court. Les algorithmes génétiques sont généralement utilisés pour réduire au minimum le temps nécessaire à la réalisation des tâches dans un système de PAP.

(Premachandra et al., 2020) ont utilisé un algorithme génétique pour optimiser les séquences de prise et dépose dans un système de PAP qui trie les produits en deux catégories selon leurs couleurs et leurs tailles et les place sur deux convoyeurs. Le but est de garantir que le processus de tri soit effectué le plus rapidement possible. Les coordonnées de positions statiques sont utilisées pour calculer l'adaptation (fitness). Le croisement et la mutation en un point sont appliqués avec la sélection élitiste lorsque la génération actuelle évolue vers la génération suivante. L'algorithme génétique garantit que la séquence de sélection converge vers la meilleure adaptation en un nombre minimum de générations, ce qui réduit considérablement le temps de calcul. L'algorithme développé peut être utilisé pour une optimisation en temps réel d'un système industriel.

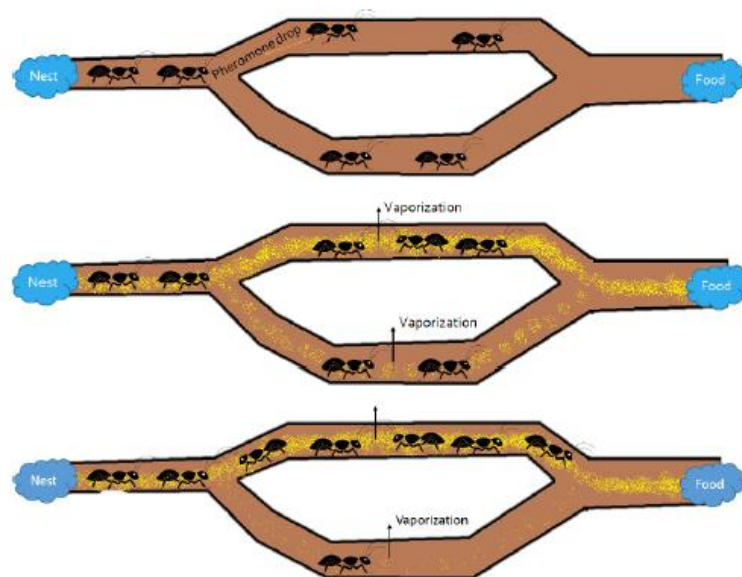
De la même manière dans (Garcia-Najera and Brizuela, 2005), les auteurs ont utilisé un algorithme génétique pour résoudre des problèmes d'affectation et de séquençement des tâches de PAP pour un système d'assemblage de cartes électroniques PCB. Le but est de diminuer le temps total d'assemblage. L'algorithme proposé fait appel à des opérateurs de croisement, de mutation et de sélection. La combinaison de ceux-ci permet à l'algorithme de générer de meilleures solutions au problème. L'élitisme a été choisi comme méthode de sélection. Les expérimentations montrent que l'algorithme utilisé améliore le temps d'assemblage d'un circuit imprimé par rapport aux méthodes utilisées dans l'industrie de 13.93 % en moyenne.

Un algorithme génétique a été utilisé comme solution au problème de voyageur de commerce. Le but est de minimiser les temps d'assemblage afin d'optimiser le processus de

placement des composants électroniques. L'algorithme choisi permet d'optimiser plusieurs paramètres simultanément en utilisant une nouvelle approche ad-hoc appelée DSWS « *Distance Score with Weights Selection* ». Cette méthode est comparée à un algorithme génétique et un algorithme par essaim de particules pour trouver la solution optimale de séquençement des positions de placement. Il a été montré que l'approche proposée est plus efficace puisqu'elle est la moins coûteuse en temps de calcul.

### 2.2.1.2 Utilisation de colonies de fourmis à partir de données de vision

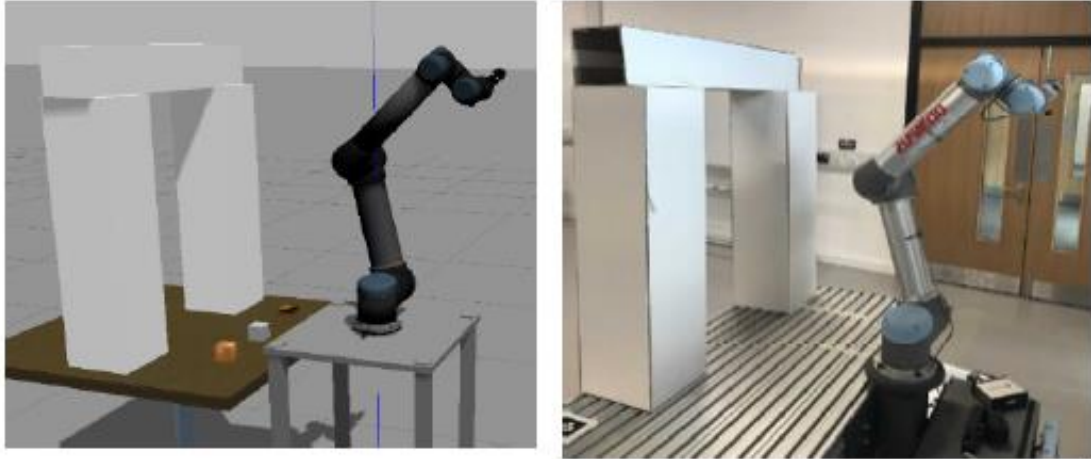
De nombreux travaux issus de la littérature se sont intéressés à l'application de l'optimisation par les algorithmes des colonies de fourmis sur les systèmes de PAP. Ces algorithmes s'inspirent du comportement des fourmis qui cherchent collectivement le chemin le plus court entre leur colonie et la source de nourriture (Coloni et al., 1992). Les fourmis déposent une quantité de phéromones proportionnelle à la longueur du trajet. L'évaporation de phéromone fait disparaître les mauvaises solutions (**Figure 19**). Les auteurs dans (Daoud et al., 2014a, 2014b) ont comparé trois méthodes d'optimisation : l'algorithme des colonies de fourmis ; l'algorithme génétique et l'algorithme d'optimisation par essaim particulaire. Le but est de minimiser le nombre de produits non collectés et de réduire le temps d'exécution des opérations. En testant plusieurs configurations, les résultats des calculs montrent que l'algorithme d'optimisation par colonies de fourmis donne la solution optimale avec un temps d'exécution très court.



**Figure 19** Principe des algorithmes d'optimisation par colonies de fourmis.

Les auteurs dans (Ridhi et al., 2019) utilisent l'algorithme des colonies de fourmis, implémenté dans un Jumeau Numérique, afin de trouver le chemin le plus court pour que le robot effectue une opération de PAP en évitant les obstacles existant dans l'environnement de production. Une fonction coût a été utilisée pour pénaliser l'algorithme à chaque fois qu'un obstacle n'a pas été détecté. L'utilisation du Jumeau Numérique a permis d'essayer de multiples

scénarios pour l'ordonnancement des tâches à un coût presque nul tout en garantissant la sécurité et en réduisant les déchets. Les résultats de l'algorithme d'optimisation sont transférés, après vérification et contrôle par des experts humains, dans le robot réel (**Figure 20**). Il a été démontré que l'algorithme proposé peut trouver l'ordonnancement optimal des tâches de PAP en évitant les collisions avec un minimum d'intervention humaine.



**Figure 20** Le jumeau numérique vs l'environnement réel (Ridhi et al., 2019)

(Wong et al., 2019) ont utilisé l'optimisation par l'algorithme des colonies de fourmis pour déterminer le trajet quasi optimal qui permet au bras robotisé d'accomplir une tâche de PAP en minimisant le temps total d'exécution de l'opération. L'objectif principal de la tâche de PAP est de contrôler le bras manipulateur afin qu'il reconnaisse de manière autonome tous les objets et de les poser dans la boîte correspondante dans un temps limité. Dans cet environnement expérimental, une caméra est montée au-dessus de la tête du robot pour capturer les objets sur la table de travail. Un système de segmentation basé sur la vision est utilisé pour extraire les informations d'image nécessaires pour reconnaître les coordonnées, le type et la couleur de chaque objet sur la table et l'emplacement dans la boîte.

### **2.2.1.3 Résolution par algorithme d'intelligence artificielle : algorithmes basés sur la logique floue**

La logique floue, créée par Lotfi Zadeh en 1965, est une extension de la logique booléenne. Elle a permis de répondre aux limites de la logique classique en remplaçant les valeurs de vérité d'une proposition  $\{vrai, faux\}$  par un degré de vérité, à choisir dans l'ensemble  $[0, 1]$ . La logique floue offre une flexibilité très appréciable dans la représentation des contraintes à l'aide d'ensembles flous, ce qui permet d'éviter d'avoir un ensemble vide de solutions.

Un contrôleur basé sur la logique floue a été développé par (Dewi et al., 2020). Le contrôleur commande un bras manipulateur utilisé pour la manipulation des tomates pendant la saison de la récolte. Il est utilisé pour améliorer la fluidité et la précision du mouvement de l'effecteur final du robot lors d'une tâche de PAP. Pour ce faire, le contrôleur prend en compte

les données fournies par les capteurs attachés à l'effecteur final. Cela permettra de définir la position optimale et le bon mouvement de ce dernier pour éviter d'endommager les tomates par l'exercice d'une force excessive ou le positionnement mal placé. Les paramètres estimés sont comparés aux résultats expérimentaux afin de montrer l'efficacité des résultats. Les différentes expérimentations montrent l'efficacité de la méthode proposée. En effet, la position réelle du robot, obtenue à partir des expériences, est similaire aux entrées de référence provenant de l'analyse de la cinématique inverse.

#### 2.2.1.4 La programmation par apprentissage kinesthésique (par démonstration)

La programmation des robots qui effectue des tâches répétitives de PAP dans un environnement industriel est une procédure très coûteuse en terme de temps. Elle exige du personnel spécialisé et représente une charge importante pour les entreprises. La programmation par démonstration est une méthode prometteuse qui peut réduire considérablement le temps de programmation et permettre à ces entreprises de passer à la production automatisée en remplaçant les tâches répétitives.

Dans ce contexte, un réseau de neurones bio-inspirés, présenté dans (De Rengervé et al., 2011), est utilisé dans un robot pour lui apprendre à réaliser des tâches de PAP démontrés par la manipulation passive et pour planifier ses actions en fonction des objectifs visuellement catégorisés. Le robot, en imitant l'opérateur, doit prendre les objets d'un endroit précis et les déposer dans deux différentes boîtes selon leurs couleurs. Comme le système est en apprentissage continu, l'opérateur peut toujours intervenir pour lui apprendre de nouvelles manières de manipulation d'objets. Cela est démontré en apprenant au robot à changer les objectifs pour les deux types d'objets.



**Figure 21** Programmation par démonstration d'un robot pour une opération de PAP (Dimeas et al., 2019)

Dans la même optique, les auteurs dans (Dimeas et al., 2019) utilisent le concept d'automatisation progressive par démonstration pour guider de manière kinesthésique un robot afin de lui apprendre à saisir des objets au fond d'un distributeur automatique de pièces et de

les placer dans le conteneur. La démonstration est répétée jusqu'à ce que le robot soit capable d'effectuer la tâche de manière autonome. Après que l'opérateur démontre la tâche robotique répétitive avec un enseignement kinesthésique (**Figure 21**), le robot doit exécuter la tâche démontrée, sans nécessité de connaissances préalables ou de préprogrammation.

Le robot deviendra progressivement proactif dans l'exécution des tâches de PAP. L'opérateur peut apporter des corrections et des ajustements aux tâches prédéfinies. Après quelques démonstrations, la tâche sera exécutée de manière autonome et l'utilisateur pourra cesser d'interagir avec le robot, formulant ainsi le concept d'automatisation progressive. Cette méthode permet de réduire considérablement le temps de programmation. D'autres méthodes basées sur la programmation par démonstration sont présentées dans les travaux de (Lin et al., 2013; Maeda et al., 2002; Skoglund et al., 2007).

### **2.2.1.5 Résolution par technologie multi-agent**

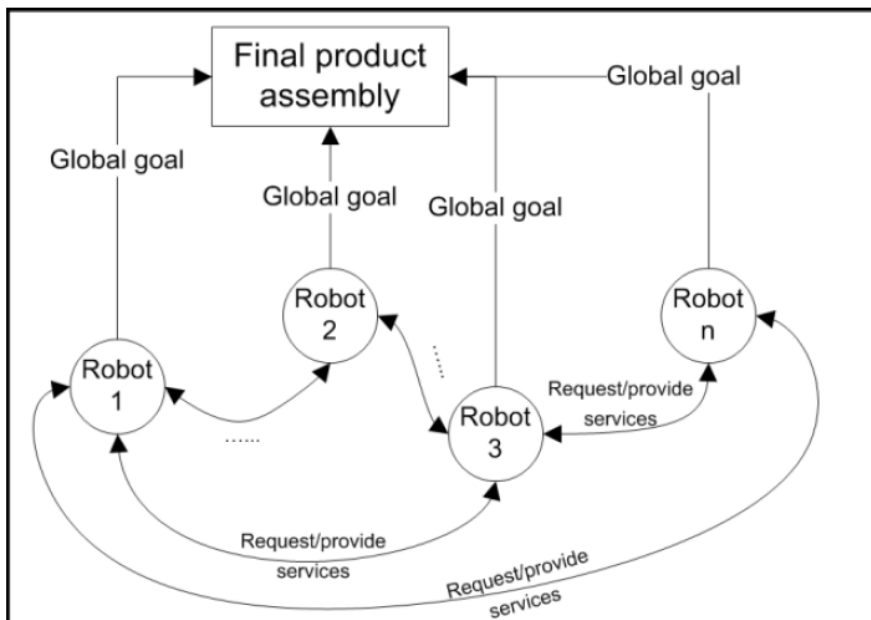
Malgré des recherches prometteuses dans le domaine de la robotique industrielle, les robots autonomes ont encore des difficultés à exécuter des tâches complexes dans un environnement turbulent. C'est particulièrement le cas des robots de PAP. Une petite déviation entre le modèle virtuel et le système réel provoque la défaillance du système. Pour parvenir à une exécution complètement autonome de tâches complexes dans un environnement non structuré, le bon équilibre entre l'autonomie du robot et l'interaction homme-robot doit être trouvé. Une architecture de contrôle multi-agents offre un environnement approprié pour la flexibilité de l'interaction humain-robot.

Les auteurs dans (Laengle and Wörn, 2001) présentent une interface homme-robot très simple et intuitive basée sur l'interaction homme-robot par contact direct afin d'accomplir une tâche de PAP dans un environnement incertain. Le principe consiste à partager le contrôle d'un manipulateur entre l'humain et le système de contrôle du robot en fonction de la complexité de la tâche, des capacités de l'humain et des capacités du robot. Une architecture de contrôle distribué à base de système multi-agent appelé KAMARA (Karlsruhe Multi-Agent Robot Architecture) est utilisée pour le contrôle des manipulateurs du robot KAMRO (Karlsruhe Autonomous Mobile Robot) afin d'assurer la coopération entre l'homme et le robot pour accomplir la tâche demandée. Lorsque le robot n'est pas capable d'accomplir la tâche requise de manière autonome (une erreur se produit, des paramètres manquent, etc.), il passe en mode semi-automatique et donne alors à l'opérateur la possibilité de l'aider. Cette coopération homme-robot est utile pour détecter l'état de l'environnement dynamique, éviter les collisions avec l'environnement et compenser les incertitudes qui le caractérisent.

Le mécanisme de communication des agents de l'architecture KAMARA est basé sur le blackboard (Laengle et al., 1997). La structure du blackboard contrôle la communication et l'échange d'informations entre les agents. Selon les auteurs, un agent est constitué de 3 parties : le communicateur (connecte la tête de l'agent aux autres agents d'un même niveau ou d'un niveau supérieur.) ; la tête (pour la planification et la sélection d'actions) et le corps (pour l'exécution des actions). Dans ce système, un agent n'est capable d'exécuter qu'une seule tâche à la fois puisque son corps est implémenté comme une procédure unique. Cette architecture fait une extension du concept de contrôle distribué par le recours à des agents exécutifs spéciaux

qui ont la même structure que les agents normaux. La seule exception est que leur corps est autorisé à allouer des corps d'autres agents, s'ils sont disponibles, et à les contrôler par des canaux de communication spéciaux avec des taux de transfert élevés. Pendant cette phase, les agents normaux n'ont pas accès à leurs corps, utilisés par les agents spéciaux. Ces derniers, ne doivent être utilisés que si aucun autre type de coopération ne convient.

L'architecture distribuée de KAMARA présente plusieurs avantages par rapport aux architectures centralisées. L'intégration d'un nouveau composant dans un système centralisé complexe est souvent très difficile, car il n'est pas évident de savoir où le système doit être modifié. Dans KAMARA, de nouveaux composants de système peuvent être ajoutés au système d'origine et toute amélioration (telle que la réduction du temps de réalisation d'une tâche) se fait sans aucun changement dans l'architecture du système. L'intégration est réalisée par le processus de négociation (Lueth and Laengle, 1994).



**Figure 22** Architecture orientée service (Švaco et al., 2011a).

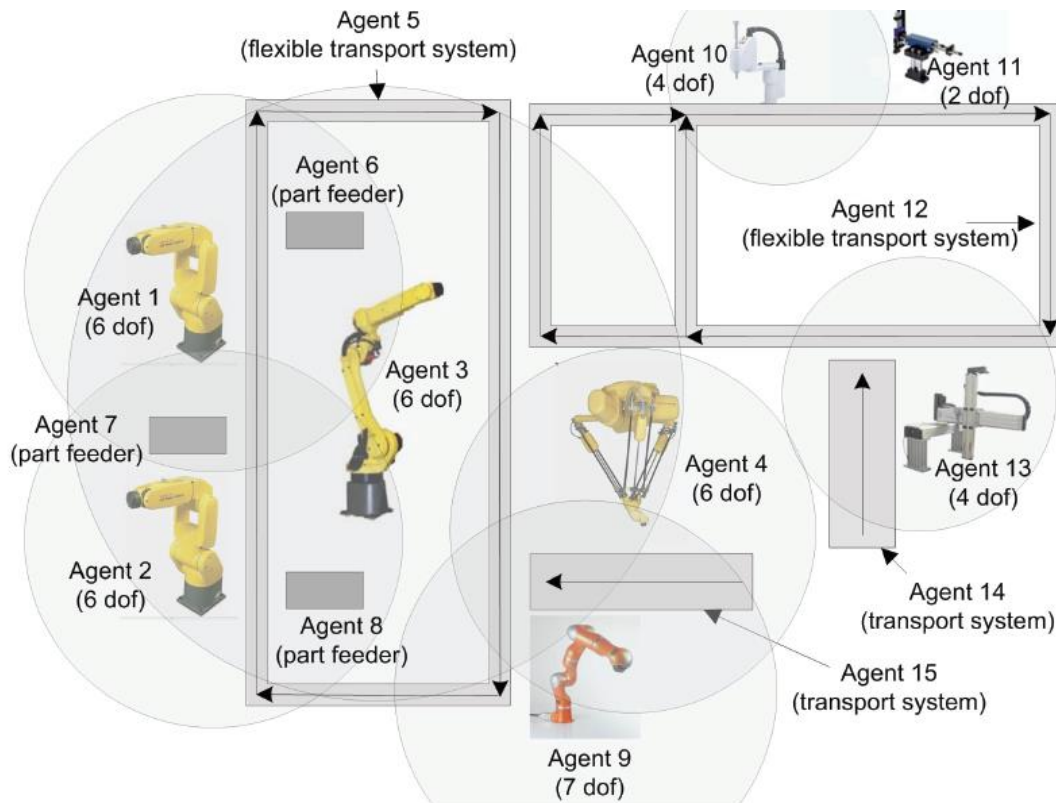
Une architecture multi-agents (**Figure 22**) a été développée dans (Švaco et al., 2011a, 2011b) pour résoudre les problèmes de calibrage spatial et positionnement des robots dans un système d'assemblage multi-robots. Les agents du système sont des entités autonomes capables de prendre des décisions et de négocier avec d'autres agents pour atteindre l'objectif global commun qui est l'assemblage de produits. En inspectant leur état réel, l'état de l'environnement et l'avancement du processus d'assemblage, un agent choisit ses futures actions. Un agent peut fournir et demander des services, par exemple : la manipulation, les opérations de prise et de dépose d'objets, le transport, etc. Un agent de planification décompose l'objectif global en tâches qui peuvent être exécutées individuellement ou bien en demandant de l'aide à des agents supplémentaires. Le voisin inactif le plus proche est sollicité. Si l'agent est occupé dans l'exécution d'une autre tâche, une demande de service est ajoutée au blackboard. Le meilleur candidat est sélectionné. Lors du traitement d'une tâche, un agent envoie des données concernant l'espace occupé pour éviter les collisions. Les agents inactifs répètent ce processus

afin d'acquiescer les tâches disponibles. Le système multi-agent développé a été testé sur un système de PAP composé de 4 robots industriels. Le système développé peut résoudre des problèmes avec des conditions d'assemblage imprévisibles et peut s'adapter aux changements dynamiques de l'environnement. La principale limitation de l'approche développée est le temps de cycle prolongé du processus.

Un système d'assemblage robotisé adaptatif basé sur la technologie multi-agent est présenté dans les travaux de (Švaco et al., 2012, 2011c). Le système est composé d'agents collaboratifs qui partagent le même objectif. Une solution pour le traitement des objets inconnus a été développée et mise en œuvre sur plusieurs agents du système. En cas de détection d'objet inconnu par le système, un agent envoie une alerte à tous les agents voisins impliqués dans son espace de travail. À la réception de l'alerte, la vigilance de tous les agents augmente et leur vitesse de fonctionnement diminue considérablement. En utilisant une méthode de segmentation, l'agent concerné tente d'identifier la géométrie de la pièce pour le comparer à des objets similaires dans la base de données. Si le modèle correspondant est un défaut d'une pièce connue, l'agent installe un outil approprié pour le manipuler. S'il n'y a pas d'outil approprié et qu'aucun autre agent voisin ne peut éliminer la pièce inconnue, une demande d'annulation manuelle est envoyée. Une fois le problème résolu, tous les agents voisins sont informés et le système reprend son fonctionnement à vitesse normale.

Le système est capable de produire les résultats souhaités dans des conditions environnementales variables. Afin de répondre à des changements imprévus, les agents collectent des informations sur l'environnement à l'aide de capteurs spéciaux et en communiquant avec d'autres agents. Le protocole de communication CNP (Contract Net Protocol) est utilisé pour allouer les opérations à l'agent le mieux qualifié. Pour ce faire, les agents étudient leurs états actuels et l'état de l'environnement dans le but de calculer la quantité de ressources nécessaires à l'accomplissement d'une tâche. Ce calcul est converti en une valeur accessible à tous les agents du système. L'agent dont l'offre est la plus basse se voit déléguer la tâche en question. L'architecture multi-agents développée, présentée dans la **Figure 23**, a été testée et vérifiée sur une installation industrielle réelle composée de 8 robots industriels. Le système est capable de réaliser une opération d'assemblage complète pour deux produits

différents.



**Figure 23** Architecture multi-agent (Švaco et al., 2011c).

### 2.2.1.6 Validation par simulation

La littérature présente également d'autres méthodes appliquées comme l'optimisation basée sur la simulation. Les techniques d'optimisations citées ci-dessus sont coûteuses en temps de calcul et ne sont généralement pas adaptées à une application en temps réel où le cycle de calcul est d'environ une milliseconde. La simulation permettra de comparer les résultats hors-ligne et de valider les différents algorithmes afin de trouver la meilleure combinaison de règles avant d'appliquer les résultats sur le système réel.

#### 2.2.1.6.1 La Simulation de flux

Un outil de simulation a été développé dans (Humbert et al., 2015) (**Figure 24**), combinant une simulation comportementale de plusieurs robots et des flux de produits. Cet outil de simulation, basé sur la simulation 3D en temps réel, permet de représenter le comportement des robots et leurs aspects collaboratifs et de visualiser et vérifier si l'effecteur final se déplace correctement afin d'assurer la performance globale souhaitée du système. Cette performance est assurée en choisissant les bons paramètres qui permettent le bon dimensionnement du système, comme le nombre de robots requis et la vitesse optimale du convoyeur, ainsi que les bonnes stratégies de contrôle.

Dans (Humbert et al., 2016) les auteurs ont utilisé le même outil de simulation pour essayer différentes stratégies de contrôle avant leur implémentation sur le système réel. Cela permettra d'éviter l'arrêt de la chaîne de production pour la réalisation des essais et permet de



gagner du temps en faisant ces essais en parallèle. Une extension de ces travaux dans (Humbert et al., 2017) montre que l'outil développé permet :

- Une simulation comportementale des robots,
- Une simulation de l'environnement de travail,
- Une simulation de différentes stratégies de collaboration de plusieurs robots,
- La possibilité de passer de la simulation à l'expérimentation.



**Figure 24** Tests de simulation avec l'outil développé dans (Humbert et al., 2015).

Dans (Comba et al., 2013) les auteurs ont étudié le comportement des lignes d'emballage en évaluant leurs performances dans le cas de différentes stratégies de PAP en utilisant un simulateur à événement discret développé en utilisant MATLAB. Le simulateur développé permet de définir un très grand nombre de paramètres dans le but de modéliser les différentes configurations possibles afin de choisir la configuration optimale.

Dans (Johari et al., 2007) Workspace5 a été utilisé pour simuler hors-ligne et visualiser en 3D le comportement d'un bras robotisé. Cela a permis d'identifier toute interférence entre le robot et les autres objets dans l'environnement et de bien définir le temps de cycle pour vérifier le délai de réalisation d'une tâche donnée par le robot. L'étude a montré que plus le temps de cycle est réduit plus le système est productif.

Dans (B. Wan Muhamad Raziff et al., 2018) les auteurs présentent la conception et l'analyse de la simulation d'un système de PAP à l'aide du logiciel Solidworks. Ce logiciel est utilisé en premier temps pour concevoir en 3D les différentes parties d'un système de PAP. En deuxième temps, le logiciel permet l'assemblage des différentes parties afin de concevoir le système complet. La conception sera analysée en utilisant l'étude Xpress de simulation et l'étude de mouvement. Pendant le processus de simulation, toute erreur dans la section d'assemblage est rapidement détectée. L'étude Xpress de simulation est conçue pour vérifier la capacité des pièces à être créées et pour analyser la pression appliquée aux pièces créées afin de connaître le niveau de sécurité de la conception pour l'utilisateur.

Le même logiciel a été utilisé dans (Sam et al., 2012) pour la conception d'un système robotique de PAP composé d'un robot cartésien, des bras robotiques articulés, des convoyeurs et des produits. Le logiciel a permis de faciliter la simulation du système et de détecter plus rapidement les erreurs dues à la conception des pièces. L'étude de mouvement a permis à l'utilisateur de développer le mouvement le plus efficace en minimisant le déplacement pour raccourcir le temps nécessaire pour atteindre les positions ciblées. L'analyse de l'étude Xpress de simulation et de l'étude de mouvement ont permis de raccourcir la période de conception et d'améliorer l'efficacité du système robotique modélisé.

Le processus de dimensionnement du nombre de manipulateurs requis pour l'exécution des tâches de PAP est un processus complexe. En effet, le système de PAP peut contenir un grand nombre de variables, souvent inconnues et incertaines, surtout dans les phases initiales du processus de conception. Ces variables dépendent de l'environnement de production, des caractéristiques des robots et des algorithmes affectation des produits aux robots.

Pour surmonter ces difficultés, les auteurs dans (Castelli et al., 2019) ont développé en utilisant MATLAB un outil simple de simulation capable de calculer le nombre de robots requis pour la réalisation de tâches de PAP dans une ligne de production en prenant en considération le plus grand nombre possible des variables. Cet outil répète la simulation avec les paramètres d'entrée fournis par le concepteur en ajoutant progressivement des stations robotisées jusqu'à atteindre les valeurs requises des indices de performance. L'utilisation de l'outil ne requiert aucune connaissance spécifique de la programmation et peut donner des résultats en quelques minutes. L'outil permet de simuler les différentes distributions de produits dans un environnement multi-robots. Il permet aussi de simuler les stratégies de PAP requises afin de tester plusieurs situations de production et d'évaluer la précision du dimensionnement.

### **2.2.1.6.2 Le réseau de Petri**

Le réseau de Petri a été utilisé dans plusieurs travaux de la littérature pour modéliser les systèmes de type PAP. Dans (Yasuda, 2012, 1999) le réseau de Petri a été utilisé pour mettre en œuvre un système de programmation et d'exécution permettant de programmer facilement un système de contrôle multi-robot. Le réseau de Petri développé permet de représenter les opérations, décrire les flux de produits, concevoir et mettre en œuvre les lois de contrôle pour facilement détecter et corriger toute erreur.

(Sciomachen et al., 1990) ont développé un émulateur de machines de PAP basé sur le réseau de Petri qui permet d'estimer le temps de cycle des machines pour un produit donné qui est une étape nécessaire pour la prévision de la capacité de la ligne d'assemblage et la planification du processus de production. Cet émulateur est capable de modéliser dans un environnement dynamique la synchronisation et le contrôle simultanés des opérations de PAP du système afin d'avoir un modèle complet tout en étant facile à comprendre et à modifier et exécutable tout en ayant une bonne performance à l'exécution.

Dans travaux similaires dans (Grotzinger and Sciomachen, 1988), le réseau de Petri a été utilisé pour fournir un cadre général dans lequel les systèmes de PAP peuvent être caractérisés. Le réseau de Petri fournit un cadre suffisamment flexible et formel pour décrire la

synchronisation et le contrôle simultanés des opérations qui affectent le fonctionnement de la cellule de travail.

Dans (Hanna et al., 1994) les auteurs utilisent une approche de modélisation basée le réseau de Petri et la logique floue dans un système basé sur la connaissance. Des situations imprévisibles et incertaines pourraient survenir lorsque le robot reçoit des données imprécises du système de vision. Dans ces cas, le robot peut se tromper lors de l'identification de l'objet ce qui augmente le risque d'erreur pour la prise de décision. Pour faire face à ce genre d'incertitude, une architecture de contrôle basée sur la logique floue a été construite en utilisant le réseau de Petri afin de contrôler l'action du robot et prendre des décisions efficaces.

Les auteurs dans (Zhou and Leu, 1991) ont utilisé le réseau de Petri dans une première étape vers la conception d'un système d'assemblage de cartes de circuits imprimés à grande vitesse et avec une haute flexibilité. Le système étudié est composé de deux bras robotisés qui partagent le même espace de travail. Une synchronisation des activités des deux bras robotisés a été réalisée afin d'éviter la collision. Le réseau de Petri permet de :

- Comprendre graphiquement les caractéristiques du système (les relations séquentielles, les activités, les conflits, etc.).
- Analyser les performances du système pour différents scénarios en essayant différentes combinaisons de paramètres.
- Utiliser les modèles pour aider à générer automatiquement des programmes de contrôle en temps-réel.

### **2.2.1.7 Autres méthodes**

D'autres travaux dans la littérature se sont intéressés à des méthodes alternatives pour la résolution de problèmes d'optimisation du système robotique global comme les heuristiques stochastiques basées sur le Recuit Simulé présenté dans (Su and Fu, 1998). Cette approche permet d'ordonner la séquence d'assemblage et d'assigner les emplacements des composants afin de raccourcir les temps de cycles et d'optimiser le fonctionnement du système robotique.

(Paola Blanco Rendon, 2013) a montré l'efficacité de la modélisation en utilisant les équations différentielles pour un système de type PAP. Trois approches différentes sont proposées pour décrire la dynamique du système à différents niveaux de complexité.

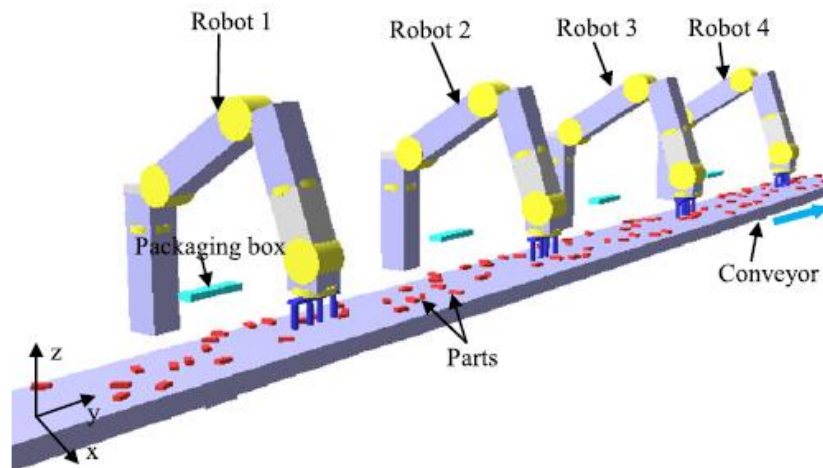
Dans (Ahmadi and Mamer, 1999), les auteurs ont utilisé des heuristiques de routage pour examiner le problème du séquençement et de placements des composants afin de minimiser le temps total d'assemblage d'une carte de circuit imprimé . Ces heuristiques ont été implémentées dans un système de contrôle dans une grande usine d'assemblage de cartes électroniques qui produit environ 12 000 cartes par jour.

Les auteurs dans (Wilhelm et al., 2007) ont étudié une nouvelle approche de génération de colonnes pour optimiser les opérations de dépose d'une machine de placement dans le but d'améliorer l'efficacité des opérations d'assemblage.

Dans un contexte similaire dans (Hop and Tabucanon, 2001), les auteurs ont présenté différents modèles de systèmes d'assemblage de circuits imprimés tels que : Pick-and-Place Dynamique (DPP) et Pick-and-Place Dynamique Etendu (EDPP) qui déterminent les coordonnées des points de prélèvement et de placement à partir de la prise en compte de la relation entre les points dans le système. Dans cet article, les auteurs montrent que le modèle EDPP est meilleur que le modèle DPP pour la diminution du temps total d'assemblage.

### 2.2.2 Résolution de problèmes d'équilibrage de charge de travail

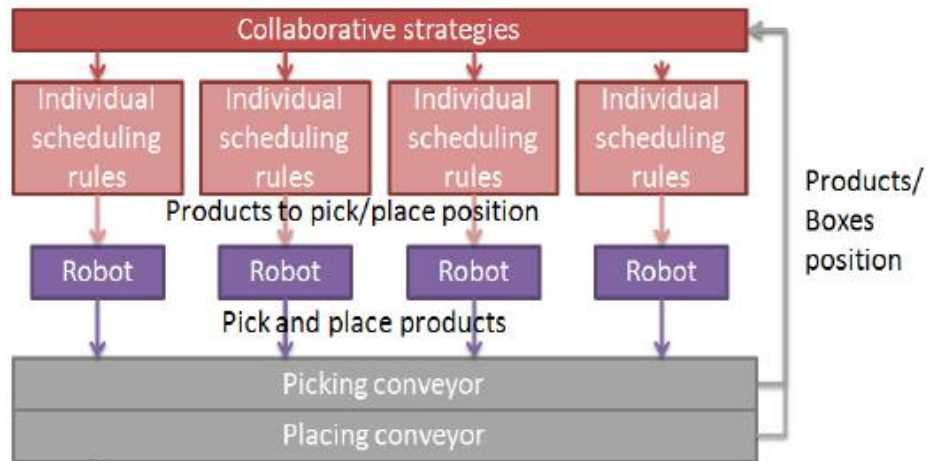
Pour un système PAP multi-robots (**Figure 25**) composé de plusieurs robots partageant un espace de travail commun, le problème de la coordination multi-robots deviendra très complexe. En effet, la répartition équilibrée de la charge de travail entre les robots tout en maximisant le nombre de produits collectés n'est pas une tâche évidente. L'action d'un robot est affectée par l'action de ses voisins. Pour déterminer la combinaison appropriée de règles de répartition de flux de pièces qui maximise le taux de prélèvement, le temps de calcul peut être très important.



**Figure 25** Système de type PAP multi-robot (Huang et al., 2015)

#### 2.2.2.1 Résolution par heuristiques : stratégies de gestion de stock

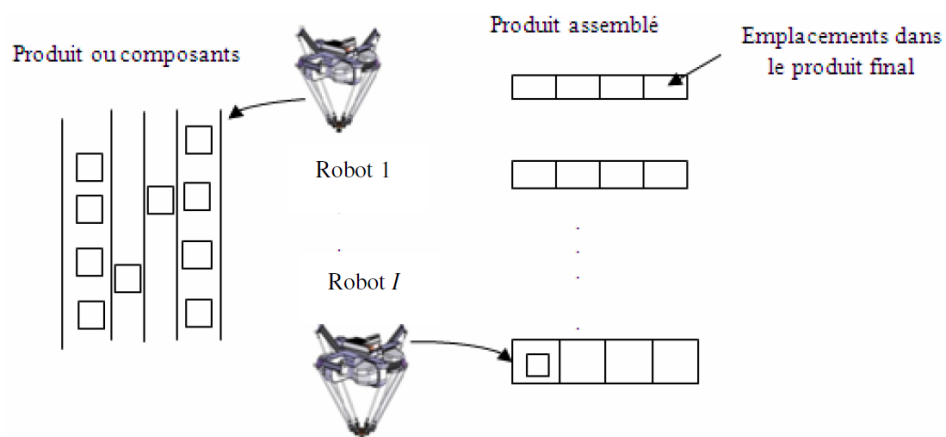
Les auteurs dans (Humbert et al., 2017, 2016, 2015) ont testé et comparé différentes stratégies de contrôle pour un système multi-robots : des stratégies individuelles pour chacun des robots (FIFO, LIFO, STP) et collaboratives entre les robots, avant leur implémentation sur le système réel (**Figure 26**). L'étude a montré que pour un système multi-robot, l'utilisation d'une stratégie collaborative est plus pertinente que l'utilisation des stratégies individuelles pour chaque robot puisqu'elle permet d'équilibrer la charge de travail entre tous les robots, de réduire le nombre de produits non pris et de réduire le temps de cycle.



**Figure 26** Architecture de simulation avec les deux niveaux de stratégies (règles d'ordonnancement individuelles et stratégies collaboratives) (Humbert et al., 2015)

### 2.2.2.2 Utilisation de colonies de fourmis à partir des données de vision

Dans (Daoud et al., 2012) les auteurs s'adressent au problème d'équilibrage de charge des lignes d'assemblages robotisées composées de plusieurs robots de PAP (**Figure 27**). Ils proposent des méta-heuristiques hybridées avec une recherche locale à savoir : colonies de fourmis, un algorithme génétique et un algorithme par essaim particulaire. Le but est d'optimiser les performances de ces lignes en équilibrant la charge de travail entre les différents robots tout en respectant les contraintes techniques et en réduisant le temps de cycle. Les performances des méthodes développées ont été évaluées en les comparant avec deux méthodes exactes développées en utilisant le solveur CPLEX et la méthode d'énumération complète. Les résultats des expérimentations prouvent l'efficacité des méthodes développées à trouver la meilleure solution avec un temps de réponse suffisamment court (une réduction moyenne du temps de 90 % par rapport aux méthodes exactes).



**Figure 27** Ligne d'assemblage robotisée (Daoud et al., 2012)

### 2.2.2.3 Résolution par algorithme d'intelligence artificielle

Les algorithmes de contrôle basés sur l'intelligence artificielle, comme le contrôle à base de la logique floue et le système expert sont utilisés pour contrôler les robots de PAP dans des environnements incertains. Nous présentons dans cette sous-section quelques exemples issus de la littérature sur l'utilisation de ces algorithmes.

### **2.2.2.3.1 Système expert**

Afin d'améliorer l'efficacité de la préparation des commandes dans les centres de distribution où l'affectation des tâches est traditionnellement assurée par le planificateur, les auteurs dans (Hou et al., 2009) ont cherché le plan d'affectation optimal pour équilibrer la charge de travail des opérateurs de préparation des commandes pour le système de préparation des commandes assisté par convoyeurs. Les règles empiriques pour l'affectation des tâches ont été extraites de l'expertise humaine. Des modèles quantitatifs sont utilisés pour convertir les règles empiriques en indices empiriques. Ensuite, une des deux méthodes, à savoir la méthode d'exhaustion ou la méthode d'échantillonnage aléatoire, est sélectionnée pour générer des plans d'affectation candidats. Enfin, les indices empiriques quantitatifs peuvent être appliqués aux plans d'affectation candidats afin d'identifier un plan d'affectation optimal.

Il a été prouvé que la méthodologie proposée fonctionne mieux que la répartition traditionnelle des tâches effectuée par le planificateur. Un cas réel, assez similaire à celui de La Poste, est également présenté dans ces travaux pour vérifier la performance de la méthodologie proposée sur le centre de distribution de Nung Hsueh Co. à Taiwan. Ce dernier est le plus grand centre logistique de l'industrie de l'imprimerie. Il distribue les livres et magazines nouvellement publiés aux librairies et autres commerçants.

### **2.2.2.3.2 Algorithmes basés sur la logique floue**

Pour contrôler les robots dans des environnements incertains, les algorithmes de contrôle basés sur le comportement et les algorithmes basés sur la logique floue sont parmi les méthodes les plus utilisées.

Les auteurs dans (Edan et al., 2004) ont comparé différents algorithmes appliqués à un système d'assemblage multi-robot où chaque robot doit assembler deux pièces. Le but est de maximiser le nombre de pièces assemblées et d'équilibrer la charge de travail entre les robots.

Les méthodes présentées sont :

- Un simple « Prends ce que tu peux » algorithme : cet algorithme est le plus simple à implémenter, il est utilisé comme base de comparaison avec les autres algorithmes. Pour cet algorithme, le robot assemble toutes les pièces disponibles.
- Un algorithme « Moyenne » : il assure une utilisation égale de tous les robots. Le nombre moyen d'assemblages de chaque robot doit être connu par les autres.
- Un algorithme « Occupation » : dans le cas où les robots auraient des temps de cycles différents, cet algorithme cherche à réduire le retard causé par les robots les plus lents. Le but est d'égaliser le temps d'attente entre les robots.
- Algorithme basé sur le comportement : cherche à égaliser l'utilisation des robots. Afin de décider si le robot doit prendre une pièce ou la laisser, deux

comportements ont été définis : satisfaction (productivité élevée) et frustration (faible productivité).

- Algorithme basé sur la logique floue : cet algorithme tente de maintenir une utilisation égale en cas d'incertitude et de perturbations du taux d'arrivée des pièces. La sortie floue, définie comme une décision de prendre ou de laisser une pièce dépend de plusieurs variables floues.

Différentes expérimentations ont été menées avec les algorithmes, tout en modifiant la fréquence d'arrivée des pièces et la vitesse du convoyeur. Une comparaison des résultats a permis de conclure que l'algorithme « Prends ce que tu peux » est plus performant en ce qui concerne le pourcentage de pièces assemblées. L'algorithme basé sur la logique floue permet de mieux équilibrer la charge de travail entre les différents robots. Par conséquent, il peut réduire la durée moyenne entre les pannes.

#### **2.2.2.4 Validation par simulation : simulation de flux**

Dans (Stumm et al., 2014) les auteurs ont présenté deux méthodes de simulation i) l'approche ABB SmartComponents ii) l'approche RobotStudio Picking PowerPac pour permettre une représentation réaliste des flux de produits. Ces méthodes offrent une intégration directe de la planification des opérations de type PAP dans un logiciel de simulation.

Selon les auteurs, une représentation réaliste du système PAP intégrée dans un logiciel de simulation pour la planification de la production permet de :

- Concevoir des flux de production efficaces.
- Déterminer le nombre optimal de robots.
- Diminuer le risque relié à la complexité de la gestion du flux de produits dans les applications PAP.
- Équilibrer la charge de travail pour une utilisation efficace de chaque robot.
- Analyser les différentes situations en variant aléatoirement les flux de produits et en changeant dynamiquement l'approvisionnement de produit, par exemple au cours du démarrage ou l'arrêt de la production, pour identifier le meilleur réglage, rentabiliser la production et optimiser la ligne sans interruption d'une production réelle.
- Valider efficacement tous les concepts de PAP et les changements dynamiques qui en découlent au sein du flux de produits.

Un algorithme amélioré à évolution différentielle *ED* a été utilisé dans (Zhu and Chen, 2009) pour résoudre un problème de type PAP pour une machine de montage multi-têtes en équilibrant la charge de travail entre toutes les têtes de la machine. Une comparaison de cette méthode avec un algorithme génétique a montré que l'algorithme *ED* est plus efficace que ce dernier. En effet, l'algorithme *ED* converge plus rapidement, et il est plus facile à utiliser et nécessite moins de variables de contrôle.

### **2.2.3 Résolution de problèmes d'optimisation du nombre de produits traités**

Dans cette section, nous présentons les principales méthodes trouvées dans la littérature, utilisés pour maximiser le nombre de produits collectés dans un système de PAP.

#### **2.2.3.1 Résolution par heuristiques : Stratégies de gestion de stock**

(Mattone et al., 2000, 1998) étudient l'influence de l'ordre d'ordonnement des objets à prélever sur la performance du système de PAP avec des files d'attente dynamiques (files d'attente où les temps de service des clients changent en fonction de l'évolution d'un système dynamique). Afin de trouver une stratégie optimale qui permettra la maximisation du taux de produits collectés, les auteurs étudient la performance de deux règles d'ordonnement standard pour les files d'attente statiques, à savoir, FIFO et SPT. Les résultats des expérimentations ont démontré que malgré le temps de calcul très faible, l'heuristique FIFO s'avère inefficace dans les situations de surcharge. D'autre part, l'algorithme SPT aboutit à de meilleurs résultats que le FIFO. En effet, en sélectionnant à chaque étape le temps le plus court pour saisir un objet, le nombre de produits collectés augmente. Toutefois, l'inconvénient de cet algorithme est que la durée de saisie des produits n'est pas flexible, ce qui peut entraîner dans certains cas la perte de l'objet. En effet, si la durée consacrée pour la saisie d'un produit est dépassée, le produit sort de l'espace de travail sans être prélevé par le robot.

Pour optimiser la performance du système, deux règles innovantes d'ordonnement en ligne ont été développées : FIFO améliorée et SPT améliorée. Ces règles ont la même complexité que les règles standards considérées, mais avec des performances améliorées dans le cas de files d'attente dynamiques.

Afin de bien définir la position de deux robots et de trouver une combinaison appropriée des algorithmes de tri et de prélèvement pour les objets sur une bande de convoyeurs en mouvement pour un système de PAP avec deux robots, les auteurs dans (Yu et al., 2017) testent la performance des heuristiques FIFO et STT *Shortest-Sorting-Time*. Les résultats des expérimentations montrent l'inefficacité de ces méthodes dans le cas d'un système surchargé. Une nouvelle méthode, appelée règle du deuxième temps de tri le plus court (SSST : *Second-Shortest-Sorting-Time*), est proposée. Cette règle, basée sur la règle SST, est plus efficace dans le cas des systèmes de surcharge. En la comparant aux algorithmes de tri traditionnels (la règle FIFO et la règle SST), la règle SSST fonctionne de manière optimale pour les différentes instances testées. En outre, la nouvelle méthode est plus adaptée aux systèmes dynamiques et surchargés, ce qui correspond exactement aux caractéristiques du système étudié. Pour ce système à deux-robots de PAP, le taux de prélèvement d'objets est le plus élevé lorsque les deux robots sont disposés sur des côtés différents de la bande transporteuse. Cette position, ensemble avec la combinaison de règles SSST pour les deux robots, permet de prélever le plus possible d'objets sur le convoyeur.

#### **2.2.3.2 La théorie des jeux non-coopératifs pour un PAP multi-robots**

La charge de travail est aussi un point crucial dans des applications PAP multi-robots. En effet, sans la définition d'une logique de prise et dépose, la charge de travail devient



déséquilibrée entraînant par exemple la saturation de certains robots et une baisse d'activités pour d'autres.

Dans ce contexte, les auteurs dans (Huang et al., 2015) abordent le problème de la réalisation d'une coordination multi-robots. Ils comparent une méthode basée sur la théorie des jeux non-coopératifs avec une méthode dénommée GRASP *Greedy Randomized Adaptive Search Procedure*. Le but est de trouver la méthode la plus robuste contre les variations des flux de pièces à prélever, tout en réduisant le temps de calcul. GRASP permet de chercher la combinaison optimale de règles de répartition parmi plusieurs candidats afin de prélever le maximum d'objets sur le convoyeur. Chaque itération du GRASP comporte deux phases : la construction et la recherche locale. La phase de construction construit une solution initiale réalisable, dont le voisinage est exploré dans la recherche locale où la solution actuelle sera remplacée si la meilleure solution de son voisinage est recherchée. La recherche locale sera terminée lorsqu'il n'y aura pas de meilleure solution dans le voisinage. La solution finale est la meilleure pour toutes les itérations du GRASP. Les expérimentations réalisées dans (Huang et al., 2012) montrent que la méthode GRASP peut converger vers la solution optimale avec un faible temps de calcul et peut-être mise en œuvre facilement. La combinaison des règles de répartition des pièces donne de meilleurs résultats que l'utilisation d'une seule règle de répartition identique pour chaque robot. En second lieu les auteurs comparent la stratégie MCS *Monte-Carlo-Strategy* avec un algorithme génétique. MCS est utilisé pour estimer le flux de pièces minimum-maximal (un flux de pièces minimum-maximal est le flux de pièces maximal pour une combinaison de règles de répartition de pièces dans le cas le plus défavorable). Les expérimentations montrent que la MCS est efficace pour estimer le flux de pièces minimum-maximal.

Dans (Bozma and Kalalıoğlu, 2012) une nouvelle approche basée sur la théorie des jeux non-coopératifs a été utilisée pour la coordination entre plusieurs robots de PAP où chaque robot se sert des observations locales du convoyeur transportant les objets et des informations sur les actions de ses voisins afin de décider sur ses actions. Le but est de maximiser le nombre d'objets pris et déposés.

### **2.2.3.3 Résolution par algorithme d'intelligence artificielle : algorithmes basés sur la logique floue**

Dans (Mendelson et al., 2002) les auteurs ont développé un système décentralisé de palettisation multi-robots en utilisant un algorithme flou basé sur le comportement. Le rôle du système est d'effectuer une tâche de palettisation à partir des pièces arrivant sur le convoyeur dans le but de maximiser le nombre de pièces prises, le nombre des palettes, l'utilisation des robots et équilibrer la charge de travail entre les robots. Pour tester sa performance, l'algorithme basé sur la logique floue a été comparé avec un algorithme simple « Prends ce que tu peux ». Les algorithmes prennent en compte la fréquence d'arrivée des pièces et la vitesse du convoyeur. Les résultats montrent que l'algorithme simple donne de meilleures performances dans des conditions expérimentales. Cependant, l'algorithme basé sur la logique floue permet une meilleure répartition de la charge de travail entre les différents robots.

### **2.2.3.4 Programmation linéaire**

Quelques auteurs ont étudié la version hors ligne du problème en supposant que le flux de produits est connu. (Briand and Parlouar, 2019) par exemple ont proposé une formulation compacte de programmation linéaire en nombres entiers (PLNE) pour résoudre un problème d'ordonnancement des tâches dans un système de PAP multi-robot dénommé *Multi-robot Pick-and-Place Scheduling Problem* MPPSP. La résolution se fait hors ligne sous l'hypothèse de prévisibilité des flux de production. Le but est de maximiser le pourcentage de produits collectés. Cette approche est relativement efficace pour les petites instances. Cependant, pour les instances de grande taille, le nombre de variables et de contraintes explose. Les temps de calcul augmentent exponentiellement, rendant l'approche inutilisable en pratique.

(BOUCHRIT, 2016) a développé deux solutions en ligne et hors ligne pour l'ordonnancement d'opérations dans un système de PAP dans le but de maximiser le nombre de produits collectés. La solution en ligne est formalisée en PLNE. Les produits sont considérés comme des nœuds dans un réseau. L'objectif est de trouver le meilleur chemin qui maximise le nombre de produits. Plusieurs contraintes, telles que la vitesse des convoyeurs, l'équilibrage de la charge et le flux d'arrivée, sont prises en compte. Cependant, même pour de petites instances, le calcul pour la recherche de la solution optimale prend beaucoup de temps, ce qui empêche d'utiliser la solution en ligne. La résolution en ligne est une technique d'optimisation basée sur la simulation, où un modèle de simulation du système est couplé à un module d'optimisation pour minimiser le nombre de produits perdus.

### 2.3 Conclusion

Au travers des différents travaux de la littérature, on s'aperçoit que les différentes méthodes ont été évoquées pour la résolution et l'optimisation des systèmes de PAP. D'après le tableau récapitulatif (**Table 2**), nous pouvons classer les méthodes de résolution en 5 différentes catégories :

- Simulation
- Apprentissage kinesthésique (AK)
- Intelligence Artificielle (IA)
- Recherche opérationnelle (RO)
- Système multi-agents (SMA)

**Table 1** Table de synthèse des domaines de résolution des problématiques de PAP

Domaine de résolution Problématique	Simulation	AK	IA	RO	SMA
Réduction du temps de calcul	---	+++	---	+++	---
Maximiser le nombre de produits pris et déposés	---	---	+	+++	---
Calculer le nombre de robots requis	++	---	---	++	---
Equilibrer la charge de travail entre les robots	---	---	+++	+++	---
Ordonnancer les opérations et/ou planification	---	---	+	+++	---
Validation et visualisation	+++	---	---	---	---
Analyse du système - représentation de ses opérations	+++	---	+	---	---
Travail en milieu incertain	+++	+++	+++	+++	+++

À notre connaissance, les applications industrielles concrètes des systèmes de PAP sont très rares. (Hou et al., 2009) est la seule étude portant sur un problème réel. En plus, la littérature existant dans ce domaine souffre d'un manque de formalisation et d'études théoriques. À notre connaissance, seuls les travaux de (BOUCHRIT, 2016; Briand et al., 2018) ont proposé une description formelle du problème. Enfin, beaucoup de stratégies de contrôle proposées dans la littérature sont basées sur des règles très basiques telles que FIFO ou SPT. Ces règles sont trop simples pour résoudre efficacement le problème PAP dans un contexte industriel.

Le problème traité dans ce document concerne un cas industriel complexe issu de l'implémentation d'un système de PAP dans un centre de tri du groupe La Poste. Le système étudié est caractérisé par une variation très dynamique des données. Le but est d'ordonnancer les opérations de PAP afin de maximiser le nombre de conteneurs pris par le robot. Des contraintes concernant la prise de décision sont imposées par le système de fonctionnement actuel du centre de tri. La variabilité des différents paramètres est aussi à prendre en compte. La validation des résultats issus de l'application de la méthode choisie sera une étape importante avant l'implémentation réelle sur la cellule robotisée dans le centre de tri.

D'après le tableau de synthèse (**Table 1**) on constate que pour maximiser le nombre de produits collectés et pour ordonnancer les tâches de PAP, la RO, l'IA sont les plus utilisées. Le

SMA est la méthode la plus adéquate quand nous travaillons dans un milieu incertain. Pour la validation des résultats et la visualisation de système et de ses opérations, la simulation est la méthode la plus adaptée.

Les deux approches qui nous ont semblé les mieux adaptées pour la résolution de notre système de PAP dans son cadre industriel sont la RO et le système expert. La simulation sera utilisée pour la visualisation, le teste, la validation et la comparaison des différents modèles. Le SMA sera utilisé pour valider l'applicabilité du système de pilotage sur le système réel.

Le chapitre prochain a pour objectif de présenter les différentes approches choisies, et notamment les comparer en se basant sur les indicateurs de performance que nous avons définis.

**Table 2** Tableau comparatif des travaux antérieurs.

	<b>Problématique</b> <b>Méthodes de résolution</b>	<b>Réduire le temps de calcul</b>	<b>Maximiser le nombre de produits pris et déposés</b>	<b>Calculer le nombre de robots requis</b>	<b>Equilibrer la charge de travail entre les robots</b>	<b>Ordonnancer les opérations et/ou planification</b>	<b>Validation et visualisation</b>	<b>Analyse du système - représentation de ses opérations</b>	<b>Travail en milieu incertain</b>
<b>Simulation</b>	<b>Simulation de flux</b>			(Castelli et al., 2019; Stumm et al., 2014)			(B. Wan Muhamad Raziff et al., 2018; Castelli et al., 2019; Comba et al., 2013; Humbert et al., 2017, 2016, 2015; Johari et al., 2007, p. 5; Sam et al., 2012; Stumm et al., 2014)		
	<b>Réseaux de Petri</b>				(Grotzinger and Sciomachen, 1988)			(Grotzinger and Sciomachen, 1988; Hanna et al., 1994; Sciomachen	

								et al., 1990; Yasuda, 2012, 1999; Zhou and Leu, 1991)	
AK	<b>Programmation par démonstration</b>	(De Rengervé et al., 2011; Dimeas et al., 2019; Lin et al., 2013; Maeda et al., 2002; Skoglund et al., 2007)							
IA	<b>Logique floue</b>		(Mendelson et al., 2002)		(Edan et al., 2004; Mendelson et al., 2002)				
	<b>Système expert</b>				(Hou et al., 2009)	(Hou et al., 2009)			
RO	<b>Théorie des jeux non-coopératifs</b>	(Huang et al., 2015)	(Bozma and Kalalioğlu, 2012; Huang et al., 2015)						
	<b>Stratégies de gestion de stock</b>	(Mattone et al., 2000, 1998)	(Mattone et al., 2000, 1998; Yu et al., 2017)	(Humbert et al., 2016, 2015)	(Humbert et al., 2017, 2016, 2015)	(Mattone et al., 2000, 1998; Yu et al., 2017)			
	<b>Les algorithmes des colonies de fourmis</b>	(Daoud et al., 2014a, 2012; Wong et al., 2019)	(Daoud et al., 2014a)		(Daoud et al., 2012)	(Ridhi et al., 2019)			

	<b>Algorithme génétique</b>	(Bonert et al., 2000; Daoud et al., 2014a, 2012; Garcia-Najera and Brizuela, 2005; Huang et al., 2015; Peng and Zeng, 2013; Premachandra et al., 2020; Zhu and Chen, 2009)	(Huang et al., 2015)		(Zhu and Chen, 2009)	(Bonert et al., 2000; Garcia-Najera and Brizuela, 2005; Peng and Zeng, 2013; Premachandra et al., 2020)			
<b>SMA</b>	<b>Système multi-agent</b>								(Laengle et al., 1997; Laengle and Wörn, 2001; Lueth and Laengle, 1994; Švaco et al., 2012, 2011c, 2011a, 2011b)

## Chapitre 3

### 3 Evaluations préliminaires : Système expert

Après avoir présenté un état de l'art global sur les stratégies de résolution des problématiques liées aux systèmes de PAP, nous allons maintenant présenter la première méthode choisie pour réaliser l'étude préliminaire, à savoir le modèle expert à base de connaissance.

Pour rappel, le but du système de pilotage est de déterminer l'affectation optimale pour la commande de l'unité de chargement automatique. Les destinations assignées aux robots (brin robotique) et aux humains (brin manuel) doivent être soigneusement choisies (**Figure 28**). D'une part, les robots sont chers, et pour être rentables le robot doit gérer le maximum de flux (c'est pourquoi les destinations les plus fréquentées doivent être chargées par le robot). D'autre part, pour choisir et placer les conteneurs, le robot est plus lent que les humains et sa capacité limitée l'empêche de traiter les destinations urgentes. En outre, même si les destinations affectées à une ressource peuvent être modifiées au cours de la journée, ces changements de destination ne sont pas souhaitables. Ceci pourrait entraîner l'évacuation de chariots non pleins, la perte de temps (temps de préparation nécessaire pour changer un chariot plein en un autre vide), et l'augmentation de l'effort physique fourni par les opérateurs afin de changer le chariot à chaque changement de destination (puisque chaque chariot doit contenir des conteneurs ayant la même destination).

Ce chapitre est divisé en deux sections. Dans la première, nous allons proposer l'utilisation d'un système expert basé sur la connaissance et présenter les différentes parties qui le composent. Ce choix est cohérent avec la politique classique du partenaire industriel. Dans la deuxième section, une simulation des différents modèles ainsi que les résultats obtenus, sont présentés. Comme indiqué précédemment, ce type d'évaluation implique de nombreuses hypothèses. Les résultats obtenus seront utilisés comme borne minimale de la performance du système de pilotage.



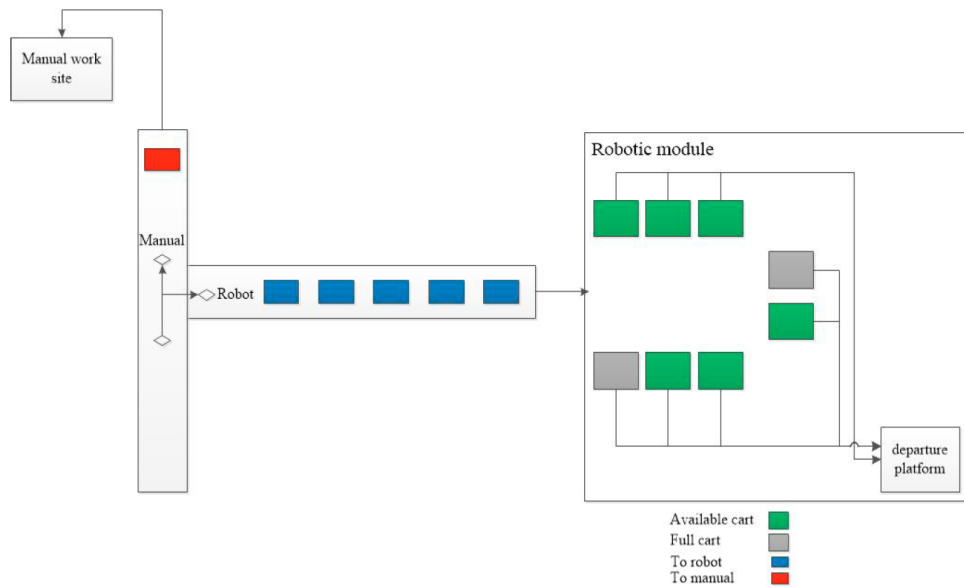


Figure 28 Schéma simplifié du système étudié.

### 3.1 Modèle à base de connaissance

L'expertise, qu'elle soit acquise par un spécialiste ou un expert après de nombreuses années de travail dans un domaine particulier, a toujours été une ressource précieuse pour les entreprises. Ces dernières s'en servent comme étant une aide à la décision pour éviter la défaillance des systèmes et améliorer leurs performances. Le développement des nouvelles technologies de l'information et de communication a rendu possible le stockage et l'exploitation de ces expertises en utilisant des systèmes experts basés sur la connaissance (Graham-Jones and Mellor, 1995). C'est en particulier le cas de la Poste.

#### 3.1.1 Définition d'un système expert basé sur la connaissance

Le système expert basé sur la connaissance (SEBC), également appelé système expert basé sur les règles (Xiaogang Wang et al., 2010), est une branche de l'intelligence artificielle, créée par la société de l'Intelligence Artificielle au cours des années 1960 (Liao, 2005). Ces systèmes sont largement utilisés dans le cas des problèmes complexes d'ingénierie. Leurs flexibilités et leurs capacités à fournir des solutions pour ce genre de problème sont remarquables lorsque les méthodes traditionnelles sont moins efficaces. C'est le cas de l'utilisation des solveurs génériques qui montrent rapidement leurs limites face à ce genre de problème. Aujourd'hui, les SEBC sont utilisés dans plusieurs domaines tels que la défense aérienne, le contrôle, l'énergie, le management, la neurochirurgie, l'assurance, etc.

Le SEBC est principalement utilisé pour remplacer le besoin de l'expertise humaine pour l'aide à la décision, la planification et le contrôle des systèmes industriels (Kumar, 2019). Il consiste à créer des systèmes numériques capables d'imiter le comportement intelligent des experts humains. En effet, il reproduit le processus utilisé pour la résolution des problèmes en combinant différents éléments de connaissance et s'auto-améliore de manière continue.

Avec l'implantation des systèmes SEBC, il est possible de remplacer le besoin de la présence d'un expert, qui peut coûter cher aux entreprises de budget limité, par l'expertise et la connaissance fournie par le SEBC. Ce dernier pourrait garantir aux entreprises de continuer à bénéficier des conseils experts tout au long de leurs projets (Ortiz, 1995).

L'idée est de transférer les connaissances et l'expertise d'un humain à un ordinateur. En cas de besoin d'un conseil pour résoudre un problème spécifique, les utilisateurs font appel à ce ordinateur. Ce dernier utilise les déductions et les conclusions qu'il a tiré des connaissances des experts reçues, pour donner conseil en jouant le rôle d'un consultant humain qui peut aussi expliquer, si nécessaire, la logique derrière ses conseils (Turban and Aronson, 2001).

Les connaissances d'experts reçues par le SEBC sont stockées sous la forme d'un ensemble de faits, de règles, d'heuristiques et de procédures facilement interprétables. La règle « si-alors » est l'une des règles les plus utilisées (Xiaogang Wang et al., 2010). Elle exécute la tâche donnée une fois la condition est vérifiée :

SI (Conditions 1 = Vrai)

(Condition 2 = Vrai)

ALORS (Exécution de l'action 1)

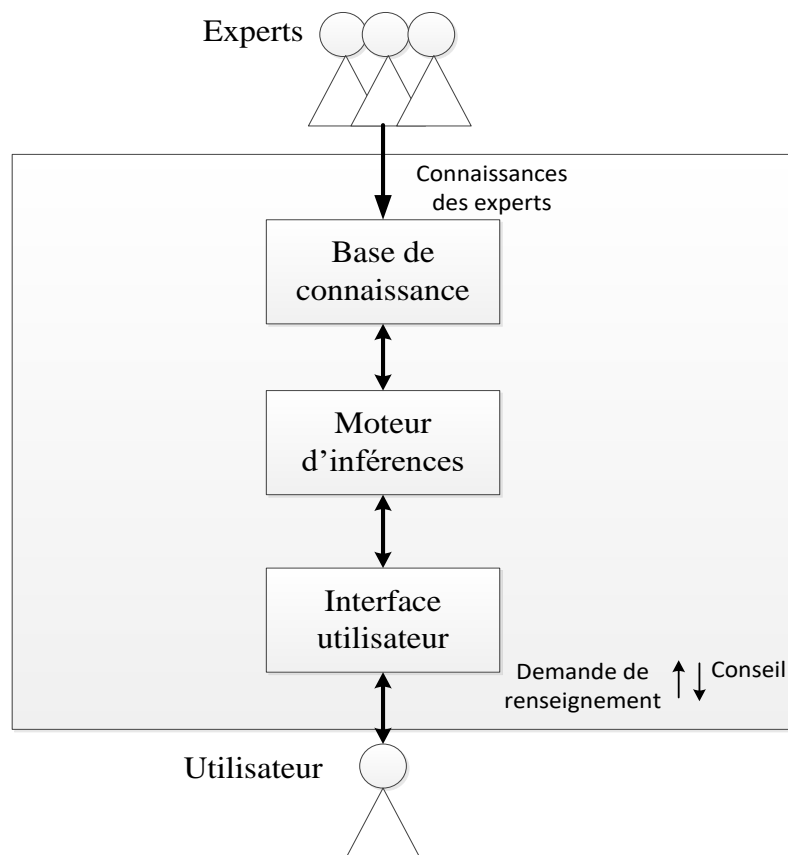
(Exécution de l'action 2)

Selon (Cakir and Cavdar, 2006) et (Ortiz, 1995), l'utilisation des SEBC présente plusieurs avantages comme :

- **La diffusion de la connaissance d'expert** qui permet de gagner du temps en évitant la répétition de la procédure d'apprentissage d'une opération plusieurs fois aux opérateurs. Un SEBC joue aussi le rôle d'un outil de formation grâce à sa capacité d'expliquer le cheminement suivi dans le raisonnement. Ainsi, en l'utilisant les opérateurs peuvent apprendre et améliorer leurs performances.
- **La normalisation des opérations** qui sont réalisées de manière diverse par différents opérateurs. L'accomplissement d'une tâche de la même façon à chaque fois permettra de mieux former et de mieux conseiller les opérateurs. Ceci, permettra une réalisation plus rapide des tâches par les opérateurs et facilitera le contrôle en cas de défaillance.
- **Une meilleure exploitation des capacités des experts** en leur évitant de réaliser des tâches pénibles et répétitives permettant d'optimiser leurs temps dans la réalisation des tâches nécessitant leurs créativité et leurs performances.
- **La standardisation des solutions** en utilisant les différentes connaissances, données et méthodes d'exécution pour créer une solution standard pour une problématique spécifique. Ceci permettra d'éviter toute forme de conflit ou manque d'efficacité.
- **Une reproduction facile** en les déployant dans les différents sièges de l'entreprise dès qu'ils sont fonctionnels. Ceci pourra augmenter la productivité et présenter une aide permanente aux opérateurs.

- **Un gain pour l'entreprise** puisqu'un SEBC est disponible 24/24h contrairement à un expert humain. De plus, lorsqu'un expert quitte son poste, une partie ou a totalité de ses connaissances sont perdues. Dans ce cas, une entreprise qui investit dans un SEBC, peut rendre l'expertise un avoir capital et une ressource permanente.

Le SEBC comprend généralement trois composants principaux : une base de connaissances, un moteur d'inférence appelé aussi mécanisme d'exploitation et une interface utilisateur (Dhaliwal and Benbasat, 1996). L'architecture générale d'un SEBC est présentée dans la **Figure 29**.



**Figure 29** Architecture générale d'un système expert

**Base de connaissances** : c'est le module où sont stockés, sous forme codée, nécessairement facile à lire et à comprendre, l'ensemble de données, de faits, de règles, des résultats partiels, des hypothèses, des solutions alternatives et finales, dérivé d'un ou de plusieurs experts humains. L'ensemble de ces éléments décrit toutes les connaissances sur la problématique et peut être utilisé par le moteur d'inférences pour la résoudre.

**Moteur d'inférences** : peut-être représenté comme un environnement de contrôle qui interagit avec l'utilisateur via une interface pour lui apporter de l'aide dont il a besoin. En cas d'un problème, ce module reçoit toutes les informations complémentaires de l'utilisateur et les

utilise, en plus des informations issues de la base de connaissances, pour inférer des conclusions et fournir des conseils d'experts.

**Interface opérateur** : c'est une implémentation logicielle qui permet d'assurer le dialogue entre l'utilisateur et le système expert. Cette interface peut contenir des fenêtres de dialogue interactif, menus, boutons ou autres, qui permettent à l'utilisateur de poser des questions au sujet d'un problème rencontré et de recevoir la réponse ainsi que le cheminement suivi par le moteur d'inférences, en cas de besoin, pour arriver à une conclusion.

### 3.1.2 Développement des modèles à base de connaissance :

Les centres de tri de La Poste ont l'habitude de travailler avec des systèmes experts depuis des décennies. Traditionnellement, des plannings figés, appelés plans de tri, sont développés par les experts, et actualisés au fur et à mesure des exécutions par les remontées du terrain. Dans le fonctionnement actuel, l'expert, qui est l'organisateur du centre de tri, établit chaque jour de la semaine un planning journalier pour organiser les tâches et gérer le trafic entrant/sortant du centre de tri. Ce planning peut facilement être interprété sous la règle « Si-Alors ». Ces règles ont été définies pour une journée postale, en prenant en compte quelques contraintes métiers bien connus comme les heures de départs camions ou les heures de vidages des machines de tri.

L'idée pour cette phase est de développer des SEBC définissant les règles d'affectation des bacs vers l'unité de chargement automatique. Ces règles doivent respecter les habitudes de travail des experts et des opérateurs sans changer le fonctionnement actuel du système informatique du centre de tri de Cestas à Bordeaux. Ainsi, le SEBC pourrait potentiellement être appliqué lors des premiers essais sur le système réel.

Deux systèmes sont envisagés ici. Le premier se base sur une observation d'une journée type et son apparente décomposition en trois phases distinctes. Le deuxième SEBC découpe la journée en 24 phases d'une heure. Le but est de mesurer l'évolution de ces deux modèles et de définir une performance que l'on pourra utiliser comme la borne inférieure pour l'évaluation des futurs modèles.

#### 3.1.2.1 SEBC 1 : 3 plans de tri

Ce plan nous a été proposé par un expert travaillant dans le centre de tri de Cestas. Le plan proposé devise la journée postale en 3 phases en se basant sur l'expérience et le savoir de l'expert. Un plan de tri est affecté à chaque phase. Nous présentons dans cette partie les 3 composants du SEBC 1.

##### 3.1.2.1.1 Développement de la base de connaissances

Ce module intègre les différentes données comme les plans de tri de différentes machines, les heures de coupures des machines (heure de vidage des machines de tri pour pouvoir préparer les chariots et les envoyer dans les temps aux camions) et les plans de départ des camions. Il intègre aussi les faits qui peuvent être transmis par les opérateurs (comme dans le cas d'un retard de camions ou d'un pic du flux de courriers) et le savoir de l'expert.

Pour le développement de la base de connaissances, nous avons récolté les données à partir du logiciel de pilotage MES (Manufacturing Execution System) développé localement (appelé SATURN) sur lequel sont enregistrés tous les plans de tri durant une journée postale, de 6 h 00 un jour donné à 6 h 00 le lendemain. Dans SATURN, on trouve des informations diverses sur les courriers récupérés depuis les différentes machines de tri comme le nombre de courriers qui sont passés sur la machine, le nombre de courriers par réceptacle (sortie machine), la destination et le centre émetteur et destinataire de chaque conteneur. Les différentes heures de coupure, partielle ou globale propre à chaque plan de tri, en correspondance avec les plans de tri des départs camions pour la même journée, figurent aussi sur SATURN.

L'expert choisi a examiné sur SATURN les données disponibles. Il a défini une journée qui lui semblait représentative d'une journée type. En parallèle, deux autres journées ont été extraites, afin de tester la robustesse du SEBC défini. Pour les trois journées types (15, 17 et 19 Octobre 2018), un fichier Excel pour chaque plan de tri et pour chaque machine a été récupéré (Figure 30).

date heure	Date	Etab d origine	Etab de destination	Type Conteneur	libelle traitement	Sortie	CAB	CE30	NOM CE30
6	17/10/2018 06:58	BORDEAUX PIC	REPRISE GF 171 issue ECO	Bak	Tri meca TG3	REBUT	22233733099008403016358500	0 03	Lot 171
6	17/10/2018 06:58	BORDEAUX PIC	REPRISE GF 170 issue ECO	Bak	Tri meca TG3	REBUT	22233733099008203016375290	0 02	Lot 170
6	17/10/2018 06:58	BORDEAUX PIC	REPRISE GF 170 issue ECO	Bak	Tri meca TG3	REBUT	22233733099008203016381290	0 02	Lot 170
6	17/10/2018 06:59	BORDEAUX PIC	REPRISE GF 171 issue ECO	Bak	Tri meca TG3	REBUT	22233733099008403016369500	0 03	Lot 171
6	17/10/2018 06:59	BORDEAUX PIC	BORDEAUX PIC	Ké7	Tri par lots de QL pour TCD	LV25	20133733033733008414806210	Y	TPD-MTI PF PIC
6	17/10/2018 06:59	BORDEAUX PIC	MONTIGNAC CCT1	Bak	Tri code postal	LV05	22233733024047003019948370	D 10	MARSAC PFDIC V1
6	17/10/2018 06:59	BORDEAUX PIC	POITIERS PIC	Bak	Tri meca TG3	LV07	22233733086339003016380510	F 01	TOURS HUB-Eco
6	17/10/2018 06:59	BORDEAUX PIC	BORDEAUX PIC	Ké7	Tri par lots de QL pour TCD	CM01	22233733033733008745627210	BRIN B	TPD- TTF PIC BRIN 8 MANU
7	17/10/2018 07:00	TOULOUSE PIC	BORDEAUX PIC	Bak	Tri meca TG3	REBUT	22231317033733003026038510	0 05	Lot 272
7	17/10/2018 07:01	BORDEAUX PIC	BORDEAUX CDIS	Bak	Hétérogène tous produits	LV17	20133733033231003188580000	Q 01	BX CDIS TCP Cedex
7	17/10/2018 07:01	BORDEAUX PIC	LANGUEDOC PIC	Bak	Tri meca TG3	LV15	22233733034470003016387320	N 02	LANGUEDOC PIC-Eco-LVerte
7	17/10/2018 07:02	BORDEAUX PIC	BORDEAUX PIC	Ké7	Tri par lots de QL pour TCD	LV26	22233733033733008751978210	Z	TPD- TTF PIC
7	17/10/2018 07:02	BORDEAUX PIC	REPRISE GF 171 issue ECO	Bak	Tri meca TG3	REBUT	22233733099008403016398500	0 03	Lot 171
7	17/10/2018 07:02	BORDEAUX PIC	STRASBOURG PIC	Bak	Tri meca TG3	LV08	22233733067192003016390500	0 01	PAGNY HUB-Eco
7	17/10/2018 07:02	BORDEAUX PIC	REPRISE GF 271	Bak	Tri meca TG3	REBUT	22233733099004703016400500	0 06	Lot 271
7	17/10/2018 07:02	BORDEAUX PIC	LORRAINE PIC	Bak	Tri meca TG3	LV08	22233733057204003016408290	G 01	PAGNY HUB-Eco
7	17/10/2018 07:02	BORDEAUX PIC	BASSE NORMANDIE PIC	Bak	Tri meca TG3	LV05	22233733014075003016405290	D 02	RENNES PFR-Eco
7	17/10/2018 07:02	BORDEAUX PIC	AUVERGNE PIC	Bak	Tri meca TG3	LV05	22233733063017003016407500	D 01	ST PRIEST HUB-Eco-Lverte bak
7	17/10/2018 07:02	BORDEAUX PIC	A RETRIER MECA GF 5	Bak	A trier mécanisable	LV01	22233733099005903016391060	0 08	DISTR PILOTEE 273
7	17/10/2018 07:02	BORDEAUX PIC	LORRAINE PIC	Bak	Tri meca TG3	LV08	22233733057204003016404320	G 01	PAGNY HUB-Eco
7	17/10/2018 07:02	BORDEAUX PIC	TOULOUSE PIC	Bak	Tri meca TG3	LV13	22233733031317003016414320	L 01	MIDI PYRENEES PIC PFR-Eco
7	17/10/2018 07:03	BORDEAUX PIC	AUVERGNE PIC	Bak	Tri meca TG3	LV05	22233733063017003016403290	D 01	ST PRIEST HUB-Eco-Lverte bak
7	17/10/2018 07:03	BORDEAUX PIC	AIN RHONE PIC	Bak	Tri meca TG3	LV05	22233733069541003016420320	D 01	ST PRIEST HUB-Eco-Lverte bak
7	17/10/2018 07:03	BORDEAUX PIC	TOULOUSE PIC	Bak	Tri meca TG3	LV13	22233733031317003016413500	L 01	MIDI PYRENEES PIC PFR-Eco
7	17/10/2018 07:03	BORDEAUX PIC	REPRISE GF 171 issue ECO	Bak	Tri meca TG3	REBUT	22233733099008403016418500	0 03	Lot 171
7	17/10/2018 07:03	BORDEAUX PIC	DISTR PILOTEE	Bak	A trier mécanisable	LV01	22233733099005703026357060	0 11	DISTR PILOTEE 270

Figure 30 Fichier Excel qui regroupe les données pour la journée du 17/10/2018

### 3.1.2.1.2 Développement du moteur d'inférences

En s'appuyant sur ses expériences passées et sur le trafic actuel dans le centre de tri, l'expert a pu distinguer 3 grandes phases pour une journée postale qui compte 24 heures (de 6 h à 6 h le lendemain) :

- Phase 1 : de 6 h à 18 h
- Phase 2 : de 18 h à minuit
- Phase 3 : de minuit à 6 h

Ce choix a été établi en se basant sur les heures de grandes coupures des machines de tri. Ces heures de coupures se basent, quant à eux, sur le plan de départ des camions.

Pour établir le plan de tri de la journée du 17-10-2018, un tri a été effectué pour déterminer les destinations qui représentent le trafic le plus important pour chaque phase. Ceci a permis d'identifier les 8 destinations les plus fréquentes par phase. Pour les 8 stations du module de

chargement automatique, une destination est affectée, par station par phase, au robot (**Figure 31**).

Le fait d'avoir des destinations fixes par phase, permettra de :

- **Diminuer le nombre de changements de destinations** puisqu'on affecte une seule destination par phase pour chaque station du robot. Le changement de destination n'est permis qu'au passage d'une phase à une autre.
- **Maximiser le nombre de chariots plein** puisque le fait d'avoir une seule destination affectée à une station par phase permettra de ne changer le chariot qu'une fois remplie. Ceci permettra de n'avoir que des chariots remplis durant toute la phase. Toutefois, il peut y avoir des chariots non remplis à la fin d'une phase (à l'heure de changement vers une deuxième phase).
- **Diminuer l'effort fait par les opérateurs** en diminuant le nombre de changements de destinations et en augmentant le nombre de chariots pleins, les opérateurs auront moins de chariots à remplacer.

		Tranche horaire 06h - 18h																											
Séparation	CE30	NOM CE30	6h	7h	8h	9h	10h	11h	12h	13h	14h	15h	16h	17h	18h	19h	20h	21h	22h	23h	0h	1h	2h	3h	4h	5h	06h - 18h	Total général	
1	0 03	Lot 171	30	19	28	105	61	76	22	6	14	9	22	6	1						6	1			3	9	398	418	
2	0 02	Lot 170	29	21	25	22	38	26	34	53	48	18	24	2	13	61	38				1	1				5	340	459	
3	0 06	Lot 271	31	17	24	95	72	55	7	2		1								1						3	304	308	
4	0 04	Lot 273	5	10	23	66	44	50	34	6	17		4													2	259	261	
5	F 01	TOURS HUB-Eco	8	158	101	3	1	3		7	1		1						1				1			2	283	287	
6	C 01	BONNEUIL HUB-Eco	12	71	43	10	6	6	4	22	1										1					1	175	177	
7	D 02	RENNES PFR-Eco	11	77	36	1			2	6	4	9	9	17	10	9				2						1	172	193	
8	L 01	MIDI PYRENEES PIC PFR-Eco	14	42	18	7	32	2	6		14	12	6	7	5	1					1					1	160	168	
		<b>Total</b>	140	415	298	309	254	218	109	102	99	49	66	32	29	71	38	0	3	9	2	1	1	0	3	23	2091	2271	
		Tranche horaire 18h - 24h																											
Séparation	CE30	NOM CE30	6h	7h	8h	9h	10h	11h	12h	13h	14h	15h	16h	17h	18h	19h	20h	21h	22h	23h	0h	1h	2h	3h	4h	5h	18h - 24h	Total général	
1	B 02	BONNEUIL HUB-L.Verte											2	1	9	18	58	95	49	106	4						335	342	
2	K 01	MIDI PYRENEES PIC			1										15	42	57	24	5	10							153	154	
3	D 01	ST PRIEST HUB-Eco-Lverte bak		29	26		1						3	2	1	6	24	21	17	53							122	183	
4	F 02	PAGNY HUB-L.Verte								3					5	13	28	18	10	31							105	108	
5	A 04 TD	POITIERS PIC													8	34	32	18	3	9							104	104	
6	F 03	RENNES PFR-L.Verte										2			7	7	28	11	15	30							98	100	
7	A 08 TD	RENNES PIC													11	18	31	14	3	7							84	84	
8	E 01	TOURS HUB-L.Verte									1				5	10	28	6	6	27							82	83	
		<b>Total</b>	0	29	27	0	1	0	0	3	0	3	5	3	61	148	286	207	108	273	4	0	0	0	0	0	1083	1158	
		Tranche horaire 24h - 06h																											
Séparation	CE30	NOM CE30	6h	7h	8h	9h	10h	11h	12h	13h	14h	15h	16h	17h	18h	19h	20h	21h	22h	23h	0h	1h	2h	3h	4h	5h	24h - 06h	Total général	
1	Q 01	BX CDIS TCP Cedex		1					2	2					2	12	9	17	4	3	5	6					96	154	
2	T 01	MERIGNAC PDC TPD		1	1	1			1		1	7	12	9	5	1	4	2	2	1	6	3	25	13	11	26	84	132	
3	T 03	MERIGNAC PDC - TCP CEDEX			1				1	2			3						1	2	2	1	2	19	2	26	17	84	
4	W 08	PESSAC QL							1		10	7	11	21								10	10	8	11	16	67	105	
5	S 01	MOUGUERRE PPDC V2								1	6	17	12	34	3					2	4	1	2	46	1		50	129	
6	W 04	NARROSSE PPDC VENT 1						20	7	9	2		39	1				1					46		3		49	133	
7	T 02	MERIGNAC PDC TCP					1	1	1							5	3	1	3		2	5		10	4	24	5	48	65
8	R 02	BANQUE POSTALE															1	2	1	1		3		14	12	15	2	46	51
		<b>Total</b>	0	2	2	1	21	8	15	7	17	71	39	66	25	19	25	11	10	19	22	17	200	48	129	79	495	853	

**Figure 31** Le calcul des destinations les plus importantes pour chaque phase.

### 3.1.2.1.3 Développement de l'interface utilisateur

L'interface utilisateur représente l'intermédiaire entre la base de connaissances et le moteur d'inférences. Le plan de tri du module de chargement automatique pour la journée du

17-10-2018 (**Table 3**) a été calculé en s'appuyant sur les différentes déductions, données et connaissances.

**Table 3** Règles d'affectation de l'expert

Plage horaire Station	6h – 18h	18h – 24h	24h – 6h
Station 1	O 03	B 02	Q 01
Station 2	O 02	K 01	T 01
Station 3	O 06	D 01	T 03
Station 4	O 04	F 02	W 08
Station 5	F 01	A 04 TD	S 01
Station 6	C 01	F 03	W 04
Station 7	D 02	A 08 TD	T 02
Station 8	L 01	E 01	R 02

### 3.1.2.2 SEBC 2 : plan de tri de 24 heures

Nous présentons dans cette sous-section le SEBC 2 basé sur la division de la journée postale en 24 heures permettant ainsi d'avoir 24 plans de tri d'une heure par journée. Cette configuration permet de mieux se rapprocher vers un système dynamique avec la mise en œuvre d'un plan de tri qui change toutes les heures.

#### 3.1.2.2.1 Développement de la base de connaissances

Pour s'assurer de l'efficacité de la comparaison avec le premier SEBC, les mêmes journées-types (le 15, 17 et 19 octobre 2018) seront utilisées pour l'évaluation de ce modèle.

#### 3.1.2.2.2 Développement du moteur d'inférences

Pour une journée postale de 24 heures, l'idée est de développer un plan de tri de 24 créneaux. Chaque heure correspond donc à une phase. Pour chaque phase, on calculera les premières destinations les plus fréquentes. Une seule destination par station par phase sera affectée au robot. Le changement des destinations ne sera permis qu'à la fin de chaque phase. Si une même destination est présente durant 2 phases successives, elle doit être affectée au même emplacement sur le robot pour réduire les changements de destinations.

#### 3.1.2.2.3 Développement de l'interface utilisateur

La **Table 4** représente les destinations affectées au robot par station robotisée pour les 4 premières heures de la journée postale du 17-10-2018.

**Table 4** Règles d'affectation - SEBC 2 - plan de tri de 24 heures

Plage horaire Station	6h – 7h	7h – 8h	8h – 9h	9h – 10h
Station 1	O 02	O 02	O 02	O 02
Station 2	O 03	O 03	O 03	O 03
Station 3	D 02	D 02	D 02	Z
Station 4	O 06	O 06	O 06	O 06
Station 5	O 07	O 11	O 09	O 11
Station 6	C 01	C 01	C 01	I 03/I 04/I 05/I 06/I 07/I 08
Station 7	O 05	D 01	D 01	N 11
Station 8	F 01	F 01	F 01	X 01/X 02
Station 9	N 03	Y	Y	Y
Station 10	L 01	L 01	O 04	O 04

### 3.2 Expérimentations

Face à la complexité des systèmes de production, les entreprises optent souvent pour la simulation comme une étape préalable à la fabrication. S'agissant d'un outil d'analyse sûr, rapide et peu coûteux, la simulation a gagné en importance ces dernières années permettant aux industriels de développer, tester et valider plusieurs configurations sans devoir faire aux contraintes économiques, mécaniques ou de sécurité.

L'objectif de cette phase est de simuler le système étudié, tester les différents scénarios et les analyser afin de mesurer leurs performances.

#### 3.2.1.1 Outils de simulation

Plusieurs logiciels de simulation existent sur le marché. La **Table 5** propose une comparaison des outils de simulation les plus utilisés.

Le travail présenté ici a été développé en utilisant le logiciel FlexSim. Bien que d'autres logiciels aient des fonctionnalités comparables à ce dernier, FlexSim a été choisi pour sa facilité d'utilisation, ses capacités d'interfaçage avec d'autres logiciels (et notamment les fonctionnalités d'émulation en natif) et le réalisme des graphismes 3D.

Le projet est réalisé dans un environnement industriel. Il s'agit d'un avantage d'une importance capitale pour bien expliquer la mise en œuvre et le fonctionnement des différents modèles au partenaire industriel.



**Table 5** Logiciels de simulation. Source (Sun et al., 2012)

Logiciel de Simulation	Interface graphique	Langage de programmation
Anylogic	2D + 3D	Java
Arena	2D + 3D	Visual Basic
Automod	2D + 3D	Automod
Plant simulation	2D + 3D	Simtalk
ExtendSim	2D + 3D	Modl
<b>FlexSim</b>	2D + 3D	C++, FlexScript
ProModel	2D + 3D	ProModel
Witness	2D + 3D	Witness

FlexSim est un puissant outil de modélisation et de simulations à événements discrets. Il permet la représentation, la visualisation et l'analyse des différents type de processus, de la fabrication à la chaîne logistique. Le logiciel possède plusieurs bibliothèques d'objets standards contenant différentes logiques et blocs d'activités prédéfinis pour l'optimisation des flux. Il est possible de personnaliser ses objets et de créer sa propre logique en utilisant les langages de programmation FlexScript et C++. Grâce à la modélisation orientée objet, le processus de modélisation est plus simple en glissant les objets dont nous avons besoin et en les déposant dans l'environnement 3D. De plus, grâce à son puissant moteur graphique, l'outil permet de représenter en 3D et de façon réaliste l'ensemble du système en fournissant des moyens visuels efficaces pour la conception et la modification du système. En outre, les outils d'analyse statistique du logiciel permettent une meilleure exploitation des modèles.

FlexSim offre la possibilité d'utiliser trois modes de modélisation :

- Modélisation en 3D : elle permet de modéliser un système physique et de visualiser le flux en 3D permettant de se projeter et d'analyser facilement le système.
- Process Flow : il est possible de construire des logiques complexes en utilisant des blocs d'activités sous forme de diagrammes de flux. La centralisation de la logique facilite la modification du modèle et le suivi de son évolution.
- Programmation de logiques, en Flexscript ou en C++.

### 3.2.1.2 Implémentation

La Simulation sur FlexSim sera utilisée pour évaluer et comparer les différents scénarios possibles. Les paramètres utilisés pour la modélisation des différents éléments du système sont représentés en **Table 6**.

Les données d'entrée utilisées pour alimenter les modèles de simulations sont :

- Les données issues du fichier Excel comportant pour chaque bac : la date d'arrivée sur l'anneau de tri, son type, sa destination, son identifiant (sous la forme d'un numéro) et le code du destinataire du chariot. La **Figure 32** présente un extrait de ces données d'entrée, chaque ligne correspond à un bac.

- Le plan de tri journalier. La **Figure 33** représente le plan de tri pour la journée du 17-10-2018.

	ArrivalTime	ItemName	Quantity	nature	sequence	traitement	CE30	numero	Destinaire
	981	Bak	1	1	1	1	2 0 03	1	337330
	990	Bak	1	1	1	1	2 0 03	2	337330
	1000	Bak	1	1	1	1	2 0 03	3	337330
	1010	Bak	1	1	1	1	2 0 03	4	337330
	1014	Bak	1	1	1	1	2 0 03	5	337330
	1030	Bak	1	1	1	1	2 0 03	6	337330
	1042	Bak	1	1	1	1	2 0 03	7	337330
	1071	Bak	1	1	1	1	2 0 03	8	337330
	1079	Bak	1	1	1	1	1 0 05	9	337330
	1091	Bak	1	1	1	1	2 0 03	10	337330
	1152	Bak	1	1	1	1	1 0 05	11	337330
	1168	Bak	1	1	1	1	2 0 03	12	337330
	1173	Bak	1	1	1	1	2 0 03	13	337330
	1580	Bak	1	1	1	1	2 0 02	14	990082
	1583	Bak	1	1	1	1	2 0 06	15	990047
	1696	Bak	1	1	1	1	2 0 04	16	990087
	1751	Bak	1	1	1	1	2 0 03	17	337330
	1781	Bak	1	1	1	1	2 0 03	18	337330
	1833	Bak	1	1	1	1	1 0 05	19	337330

**Figure 32** Table comportant les données d'entrée sur FlexSim

**Table 6** Paramétrage du système modélisé

Débit du robot /h	450 bac/h
Vitesse convoyeur	500mm/s
Capacité convoyeur Flexibac	21 bacs
Cadence ascenseur	4 s
Temps de changement du chariot	60 s

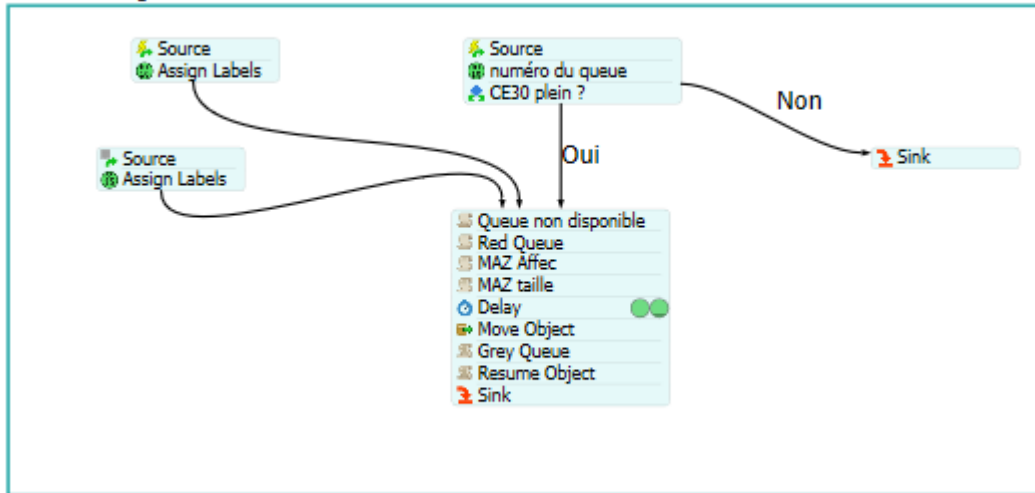
Col1	6	7	8	9	10	11
	1 0 02	0 02	0 02	0 02	0 02	0 02
	2 0 03	0 03	0 03	0 03	0 03	0 03
	3 D 02	D 02	D 02	Z	Z	Z
	4 0 06	0 06	0 06	0 06	0 06	0 06
	5 0 07	0 11	0 09	0 11	0 11	0 11
	6 C 01	C 01	C 01	I 03/I 04/I 05	I 03/I 04/I 05	I 03/I 04/I 0
	7 0 05	D 01	D 01	N 11	0 09	M 12
	8 F 01	F 01	F 01	X 01/X 02	L 01	X 01/X 02
	9 N 03	Y	Y	Y	Y	Y
	10 L 01	L 01	0 04	0 04	0 04	0 04

**Figure 33** Table représentant le plan de tri journalier sur FlexSim

Le module « Process Flow » de FlexSim a été utilisé pour développer deux parties du processus logique :

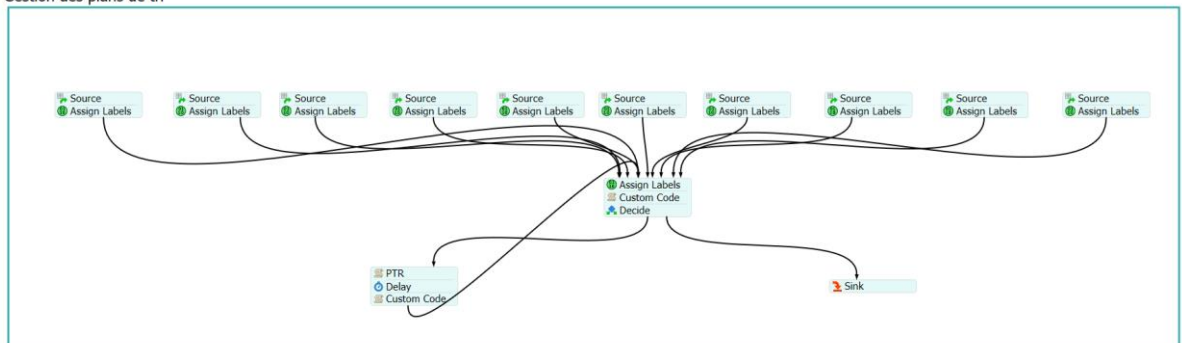
- Partie 1 : création et gestion des chariots (**Figure 34**).
- Partie 2 : gestion des plans de tri (**Figure 35**).

Création et gestion des lots



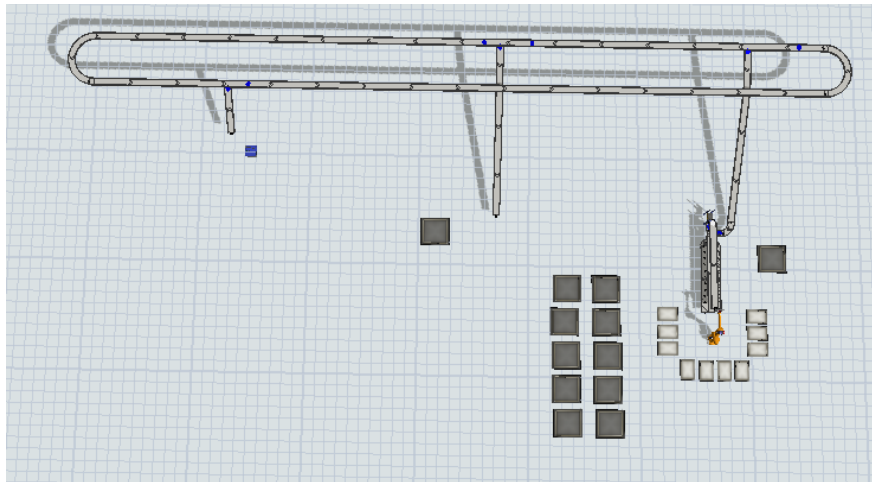
**Figure 34** Processus logique de création et gestion des lots sur le logiciel FlexSim

Gestion des plans de tri

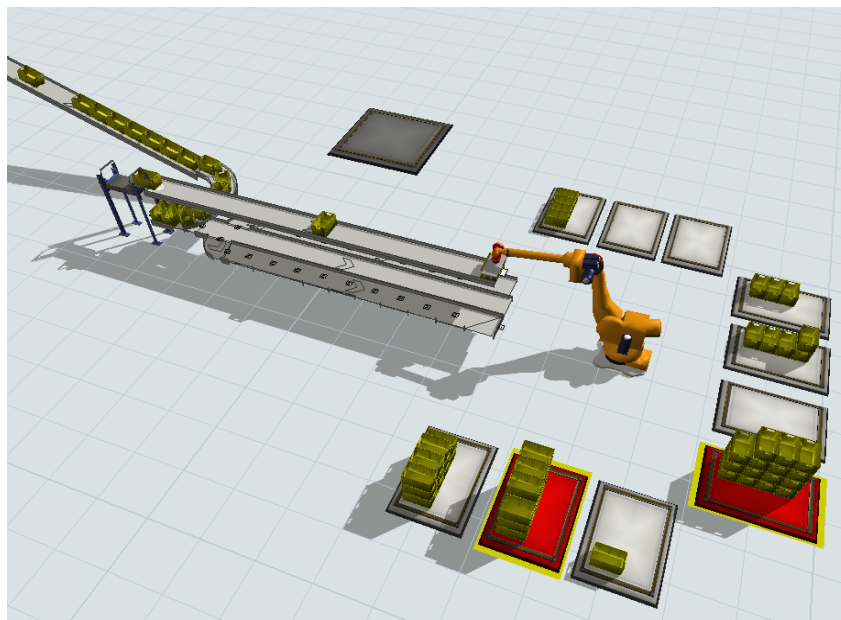


**Figure 35** Processus logique de gestion des plans de tri sur le logiciel FlexSim

Le modèle de simulation réalisé contient l'anneau de tri, l'injecteur des bacs, le brin manuel et le brin avec robot (**Figure 36**). Après leurs injections sur l'anneau de tri, les bacs continuent à tourner tant qu'ils ne sont pas encore affectés. Suivant la règle d'affectation choisie, les bacs sont envoyés au brin manuel ou au robot. Une fois les bacs affectés à Flexibac, ils sont chargés automatiquement par le robot dans le chariot correspondant à leurs destinations (**Figure 37**). Une fois rempli, le chariot est évacué (sa couleur devient rouge sur la simulation pendant un temps correspondant au temps moyen de changement du chariot plein par un autre vide, en considérant être à capacité infinie au niveau des opérateurs.).



**Figure 36** Capture d'écran du modèle de simulation sur FlexSim



**Figure 37** Capture d'écran du module de chargement automatique sur FlexSim

### 3.2.1.3 Validation du modèle de simulation

Pour s'assurer que les différents modèles de simulation sont correctement mis en œuvre, et que les hypothèses implantées dans le modèle sont raisonnables, la vérification est un processus essentiel (Sun et al., 2012).

Nous avons choisi le SEBC 2 pour la validation. Deux analyses ont été lancées en parallèle. La première est une analyse statique (en utilisant Excel) et la deuxième est faite par FlexSim. Une comparaison des résultats a été réalisée pour la validation du modèle de simulation.

L'analyse Excel, portant uniquement sur les données d'entrée, a permis d'estimer le nombre de bacs pris en charge par le robot. Cette analyse ne prenait évidemment pas en compte

les temps de transport ou la capacité limitée des ressources. Dès le début de la journée postale et pour chaque heure, on a calculé le nombre de conteneurs pour les 10 destinations les plus fréquentes. La courbe de l'évolution du nombre de conteneurs pour chaque heure est ensuite tracée.

En parallèle, une simulation FlexSim a été lancée pour chaque journée. Toutes les valeurs du vecteur de paramétrage ont été modifiées de façon à annuler tous les délais (**Table 7**). En effet, en supprimant toutes les contraintes en rapport avec le temps (en augmentant le débit du robot et la vitesse des convoyeurs par exemple), le nombre de bac par heure sur FlexSim devrait avoir une valeur très proche de celle trouvée sur Excel.

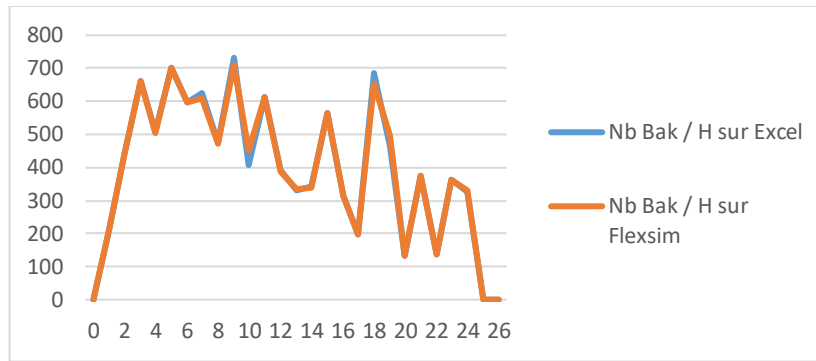
**Table 7** Vecteur de paramétrage pour la validation – Journée du 15/10

Débit du robot/h	900 bac/h
Vitesse convoyeur	100000mm/s
Capacité convoyeur Flexibac	Infinie
Cadence ascenseur	0 s
Temps de changement du chariot	0 s

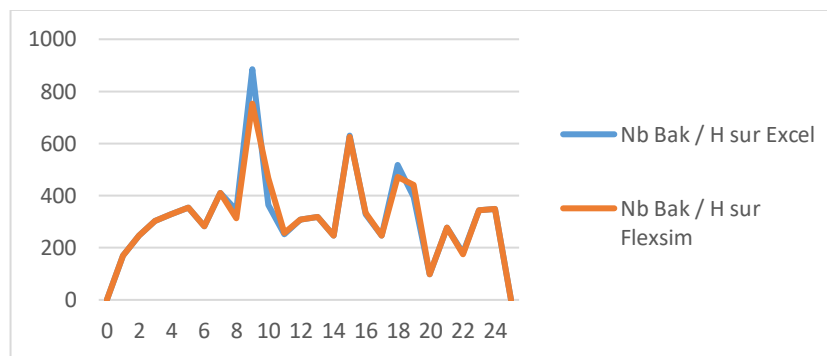
Les **Figure 38** et **Figure 39** montrent qu'en augmentant significativement tous les paramètres dans les modèles de simulations, les deux courbes représentant l'évolution du nombre de bacs par rapport au temps sur FlexSim et sur Excel ont des allures semblables. La **Table 8** indique que les pourcentages de bacs pris en charge par le robot sont similaires sur Excel et sur FlexSim pour chacune des 3 journées simulées. Ceci nous montre la cohérence des résultats obtenus par simulation. Par conséquent, on valide les modèles de simulation.

**Table 8** Pourcentage de bas vers le robot sur FlexSim et sur Excel pour le SEBC 2

	Pourcentage de bacs sur FlexSim	Pourcentage de bacs sur Excel
Journée 15/10	43 %	43 %
Journée 17/10	38 %	38 %
Journée 19/10	43 %	43 %



**Figure 38** Nombre de bacs sur Excel vs nombre de bacs sur FlexSim par rapport au temps – Journée du 19/10



**Figure 39** Nombre de bacs sur Excel vs nombre de bacs sur FlexSim par rapport au temps – Journée du 15/10

### 3.2.1.4 Résultats

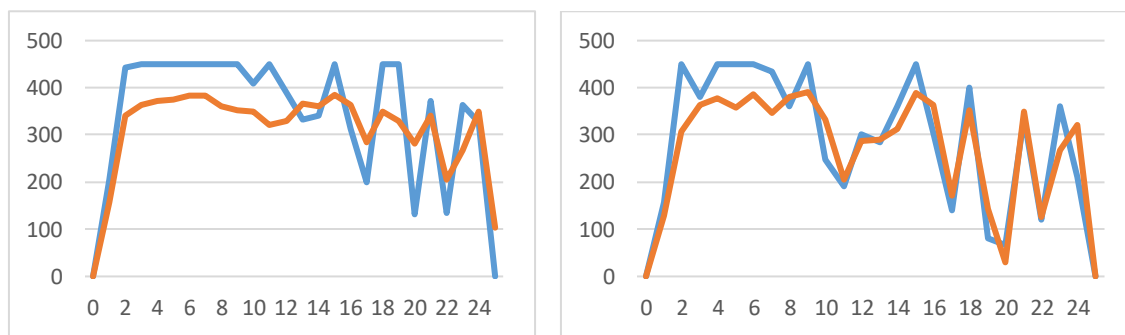
Douze simulations ont ensuite été réalisées dans 3 configurations différentes :

- Configuration 1 – SEBC 1 : 3 simulations ont été lancées avec les données des 3 journées en appliquant le plan-de-tri expert de journée du 17/10 aux journées : 15, 17 et 19. Ceci nous a permis d'évaluer la pertinence de ce plan d'affectation sur les trois journées-type.
- Configuration 2 – SEBC 2 : 3 simulations ont été lancées avec les données des 3 journées en appliquant pour chaque journée le plan de tri de 24 heures correspondant à cette même journée. Cette configuration a pour objectif d'évaluer la pertinence du SEBC 2 a posteriori, c'est-à-dire de l'appliquer sur la journée permettant de le calculer.
- Configuration 2Bis – SEBC 2 : 6 simulations ont été lancées. Pour chaque plan-de-tri de 24 heures, 2 simulations ont été réalisées en utilisant les données des 2 autres journées (exemple : plan du tri du 15/10 avec la journée du 17 puis avec la journée du 19). Cette configuration a pour objectif d'étudier la sensibilité des SEBC 2, en les appliquant sur les deux autres journées postales.

Pour chaque simulation, on calcule le nombre et le pourcentage de bacs pris en charge par le robot. Ce dernier est utilisé comme étant un indicateur de performance. Les résultats indiquant le pourcentage de bac pris en charge par le robot pour les 3 modèles sont représentés respectivement dans **Table 9**, **Table 10** et **Table 11**.

En utilisant les résultats issus des simulations FlexSim et des calculs Excel simples, on a pu tracer l'évolution du nombre de bacs pris en charge le robot par rapport au temps sur FlexSim et sur Excel (**Figure 40**).

À l'aide de l'outil de statistiques implanté dans FlexSim, nous avons pu également tracer l'évolution de chaque station robot (chargement et déchargement par destination). La **Figure 41** montre le chargement et le déchargement des chariots par rapport au temps pour les stations numéro 4 et 8 pour le SEBC 1 avec les données de la journée du 17. Cette figure montre que seulement 3 destinations par station sont prises en charge par le robot durant une journée postale. Elle montre également qu'il peut y avoir des chariots non remplis qu'à la fin de chaque phase.



**Figure 40** Courbe représentant l'évolution du nombre de bacs par rapport au temps pour les journées : 19 et 17 – SEBC 2 (Bleu : sur Excel, Orange : sur FlexSim).

**Table 9** Pourcentage de bacs vers le robot – SEBC 1

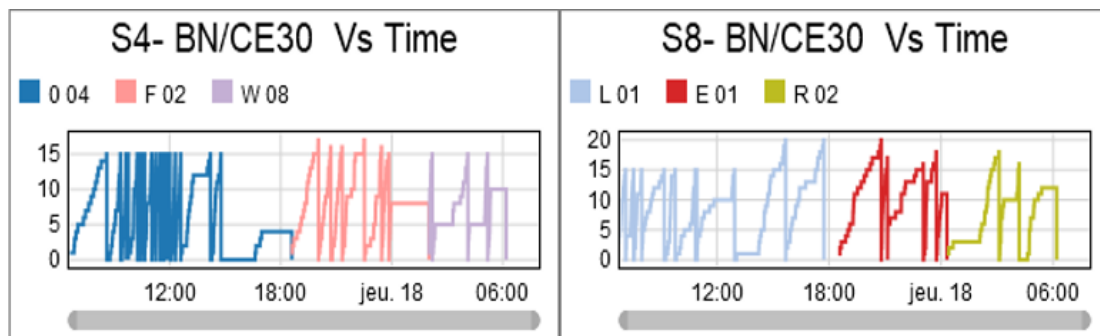
	Pourcentage de bacs vers le robot pour le SEBC 1
Journée 15/10	15 %
Journée 17/10	18 %
Journée 19/10	17 %

**Table 10** Pourcentage de bacs vers le robot - SEBC 2

	Pourcentage de bacs vers le robot pour le SEBC 2
Journée 15/10	38 %
Journée 17/10	34 %
Journée 19/10	32 %

**Table 11** Pourcentage de bacs vers le robot - configuration 2Bis

	Plan du 15	Plan du 17	Plan du 19
Journée 15/10		28 %	34 %
Journée 17/10	29 %		32 %
Journée 19/10	30 %	28 %	



**Figure 41** Chargement et déchargement des chariots par destination par rapport au temps pour les stations 4 et 8 sur le robot – SEBC 1 – journée du 17.

Les **Table 9**, **Table 10** et **Table 11** montrent que les résultats obtenus, en terme de pourcentage de bacs pris en charge par le robot, pour les configurations 2 et 2Bis sont systématiquement plus importants que ceux de la configuration 1. La première conclusion apportée est une amélioration de la performance entre SEBC 1 et SEBC 2. Cette conclusion a eu un impact direct pour le partenaire, qui a utilisé ce principe pour ses premiers essais in situ.

La **Table 12** présente la sensibilité du SEBC 2 à la journée utilisée. Pour cela, la perte en pourcentage de bac est calculée pour chaque journée en utilisant à chaque fois un plan de tri différent. La perte est considérée nulle quand on applique à une journée le plan de tri correspondant à la même journée (Exemple : plan du 15 pour la journée du 15). La perte pour les autres journées est calculée en prenant en compte la différence entre le nombre de bacs vers le robot quand le plan de tri de la même journée est appliqué et le nombre de bac vers le robot quand le plan de tri de la journée à étudier est appliqué (Exemple : plan du 15 pour la journée du 17).

### 3.3 Conclusion

L'étude présentée dans ce chapitre a permis de détailler les différents et d'évaluer leurs performances par simulation pour la résolution de notre problématique. L'analyse a permis de confirmer que le SEBC 2 (configuration 2) est plus performant que les deux autres. En effet, comme le montre la **Table 12**, avec la configuration 2Bis la perte varie globalement entre 6 %



et 26 % pour les 3 journées étudiées, i.e. on perd entre 550 (perte acceptable) et 2 200 (perte importante) bacs par jour. Le SEBC 1 est le moins performant.

Le SEBC 1 sera donc défini comme la borne basse qui servira comme base de comparaison pour les modèles suivants.

**Table 12** Sensibilité du SEBC 2

	Plan du 15	Plan du 17	Plan du 19
Journée 15/10	0 %	-26 %	-11 %
Journée 17/10	-18 %	0 %	-6 %
Journée 19/10	-6 %	-12 %	0 %

Cette étude nous a été bénéfique dans la détermination de la borne inférieure de comparaison. Elle nous permettra aussi de mieux nous positionner par rapport aux travaux postérieurs. Dans le chapitre suivant, nous présenterons une évaluation basée sur des techniques de recherche opérationnelle. Ce type d'évaluation impliquera plusieurs hypothèses qui simplifient les diverses contraintes du système industriel et permettra de définir la borne supérieure de la performance.

## Chapitre 4

### 4 Modèle basé sur les méthodes de recherche opérationnelle

Comme vu dans le chapitre 1, les techniques de recherche opérationnelle sont de plus en plus utilisées pour résoudre les problèmes d'ordonnancement de type PAP. Dans le cas d'un problème industriel, le choix d'une méthode efficace capable de produire une solution optimale en un temps de calcul raisonnable, est une difficulté à laquelle tout décideur est confronté. Dans ce chapitre, nous nous intéressons à employer les méthodes de recherche opérationnelle pour ordonnancer dynamiquement l'affectation des tâches au robot dans le but de maximiser son utilisation puisqu'un robot inactif est un gaspillage de ressources.

Nous présentons en premier lieu, une brève étude, qui nous a permis, dans la phase préliminaire de l'étude du projet, de confirmer qu'un seul robot serait suffisant pour atteindre de nos objectifs. Dans un second lieu, une formulation mathématique de la problématique ainsi qu'une résolution exacte avec un solveur seront proposées. La taille importante du problème industriel, le caractère dynamique et incertain de l'arrivée des conteneurs, sont les raisons pour lesquelles nous proposons un ensemble d'heuristiques en ligne permettant d'affecter dynamiquement les tâches au robot. Une description détaillée des méthodes proposées sera donnée.

#### 4.1 Nombre de robots

Dans son appel d'offres initial La Poste prévoyait d'implémenter jusqu'à 5 cellules robotisées avec objectif de maximiser le volume de courrier journalier pris en charge par l'ensemble des robots. Les robots sont coûteux. Pour être rentables, ils doivent être utilisés à haute capacité et doivent gérer le maximum de flux. Une étude préliminaire réalisée par le responsable du projet, a estimé le seuil de rentabilité d'un robot à 6 000 bacs par jour.

Une évaluation du nombre optimal des robots à mettre en place est alors une étape primordiale pour l'évaluation de la rentabilité. Ainsi, dans le cadre d'un stage que j'ai encadré, nous avons étudié la rentabilité des 5 robots. Un groupe d'étudiants de l'IMT Atlantique spécialisé en Recherche Opérationnelle a participé à cette étude.

Nous commençons par une modélisation statique simple où l'idée est de découper la journée postale en  $T$  créneaux de temps égaux. Pour chaque créneau le programme déterminera le planning optimal d'affectation des destinations vers les robots.

Le problème considéré peut être formulé sous la forme d'un programme linéaire mixte en nombres entiers (Selma et al., 2019). Avec :

- $D$  et  $R$  désignent les ensembles de destinations et de robots, respectivement.
- $x_{dt}^r$  est égal à 1 si la destination  $d$  est assignée au robot  $r$  à l'instant  $t$ , et à 0 sinon.
- $y_{dt}^r$  est égal à 1 s'il y a un changement de destination  $d$  au cours de la période  $t$  pour le robot  $r$ , et 0 sinon.
- $z_t^r$  est le nombre de changements de destination dans la période  $t$  pour le robot  $r$ .

La journée est décomposée en  $T$  périodes. Le problème est d'assigner les  $M$  destinations les plus chargées par période à chaque robot. En outre, la charge de travail totale d'un robot pendant chaque période ne peut pas dépasser sa capacité  $C$ .

La fonction objectif a deux composantes avec des poids respectifs  $\gamma_1$  et  $\gamma_2$ . La première est de maximiser le nombre de conteneurs pris en charge par les robots et la seconde est de minimiser le nombre de changements de destination entre deux périodes consécutives (c.-à-d. pour minimiser l'effort des employés) en pénalisant le robot pour chaque changement d'affectation.

La fonction objectif est donnée par l'équation (1). Les contraintes (2) stipulent que chaque destination est assignée à un robot au maximum dans chaque période, et les contraintes (3) empêchent d'assigner plus de  $M$  destinations à un robot dans chaque période. Les contraintes (4) stipulent que la charge de chaque robot ne peut pas dépasser sa capacité. Les contraintes (5) fixent  $y_{dt}^r$  à 1 s'il y a un changement vers la destination  $d$  dans la période  $t$  pour le robot  $r$ . Les contraintes (6) calculent le nombre de changements de destination dans la période  $t$ .

Enfin, les domaines des variables sont donnés dans les équations (7) et (8).

$$\max \gamma_1 \left( \sum_{d \in D} \sum_{t \in T} \sum_{r \in R} x_{dt}^r w_{dt} \right) - \gamma_2 \left( \sum_{t \in T} \sum_{r \in R} z_t^r \right) \quad (1)$$

s.t.

$$\sum_{r \in R} x_{dt}^r \leq 1 \quad \forall d \in D, t \in T \quad (2)$$

$$\sum_{d \in D} x_{dt}^r \leq M \quad \forall t \in T, r \in R \quad (3)$$

$$\sum_{d \in D} w_{dt} x_{dt}^r \leq C \quad \forall t \in T, r \in R \quad (4)$$

$$y_{dt}^r \geq x_{dt}^r - x_{dt-1}^r \quad \forall d \in D, t \in T, r \in R \quad (5)$$

$$z_t^r = \sum_{d \in D} y_{dt}^r \quad \forall t \in T, r \in R \quad (6)$$

$$z_t^r \geq 0 \quad \forall t \in T, r \in R \quad (7)$$

$$x_{dt}^r, y_{dt}^r \in \{0, 1\} \quad \forall d \in D, t \in T, r \in R \quad (8)$$

Le programme a été résolu en utilisant le serveur CPLEX 12.9 en utilisant les données d'une journée postale avec des définitions de créneaux de deux heures.

**Table 13** Taux d'utilisation des robots

Nombre de robots	Taux d'utilisation (bacs)	Nombre de bacs total pris	Objectif : 6000 bacs par robot par jour
1	18,76%	6015	6000
2	29,79%	9552	12000
3	38,10%	12220	18000
4	44,51%	14274	24000
5	49,75%	15976	30000

La **Table 13** montre que seulement le premier robot est rentable puisqu'il dépasse le seuil de 6 000 bacs par jour. Pour atteindre son objectif, La Poste se limitera donc par l'installation d'un seul robot de chargement.

Pour la réalisation de cette étude préliminaire, nous n'avons pas pris en compte toutes les contraintes afin de permettre une résolution du problème dans un temps de calcul raisonnable. Une modélisation du problème complet avec toutes les contraintes industrielles sera présentée dans la section suivante.

## 4.2 Résolution du problème

Pour notre objectif, qui est de maximiser l'utilisation des robots, un comportement intéressant dans lequel un robot est utilisé à 100 % serait le suivant : le robot charge continuellement des conteneurs dans un premier chariot, puis lorsque ce chariot atteint sa capacité maximale, le robot charge également continuellement des bacs dans le chariot suivant, pendant ce temps le premier chariot est remplacé et une nouvelle destination lui est affectée. Une fois le second chariot est plein, le robot charge le troisième chariot, et ainsi de suite. Cela repose aussi sur le fait que le temps de chargement d'un chariot est plus long que le temps de remplacement de ce même chariot. Compte-rendu les diverses contraintes industrielles, ce planning est loin d'être réalisable dans le cas réel. Pour se rapprocher de ce comportement, les questions suivantes se posent :

- **Règle d'affectations destinations** : quelle destination sera sélectionnée pour être chargée automatiquement par le robot et à quel instant ? Dans la cellule robotisée, soit il y a un chariot correspondant à la destination du conteneur dont la destination est sélectionnée, soit une station de sortie vide est disponible et la prochaine destination doit être soigneusement sélectionnée pour maximiser la charge de travail du robot.
- **Règle de remplacement des chariots** : parmi plusieurs chariots, dépassant le seuil de remplissage minimum, présents sur la cellule robotisée, quel chariot doit être remplacé ? Les chariots partiellement remplis dans la station de sortie empêchent le robot de manipuler des conteneurs de destination supplémentaire. Cependant, un chariot peut être retiré avant d'être plein (lorsqu'il atteint un seuil de remplissage minimal). Quand plusieurs conteneurs appartenant à la même destination sont présents sur l'anneau de tri mais que cette destination ne peut pas être prise en charge par le robot (parce que toutes les stations sont affectées) un opérateur peut remplacer un chariot qui atteint le seuil minimal de chargement.

Lorsque plusieurs chariots atteignent le seuil de remplissage minimal, le choix de la destination à remplacer, a un impact critique sur les performances du système.

- **Règle de rejet des conteneurs :** lorsque la capacité de stockage de l'anneau de tri est atteinte, quels conteneurs doivent être rejetés ? Lorsque l'anneau de tri est plein, certains conteneurs peuvent être retirés de l'anneau. Ces conteneurs seront par la suite manipulés manuellement par des opérateurs humains. Comme La Poste peut affecter suffisamment d'employés pour manipuler les conteneurs dans n'importe quelle situation, nous nous concentrons uniquement sur le planning du robot, et ces conteneurs chargés manuellement sont considérés comme rejetés. L'objectif est de minimiser le nombre de conteneurs rejetés.

Nous avons essayé de répondre à toutes ces questions, avec l'aide d'une étudiante en master M2 « Optimisation et Recherche Opérationnelle ».

Dans un premier temps, nous avons essayé de résoudre le problème hors-ligne en utilisant le solveur CPLEX 12.9. Une formulation mathématique ainsi que les résultats des expérimentations sont présentés dans la sous-section suivante.

#### 4.2.1 Résolution hors-ligne

Pour maximiser la charge de travail des robots, le robot doit prendre et placer autant de conteneurs que possible. Comme les conteneurs arrivent continuellement dans un ordre aléatoire sur l'anneau de tri, La Poste a besoin d'un algorithme pour affecter dynamiquement les conteneurs au robot afin de maximiser la charge de travail de ce dernier. L'algorithme doit respecter la contrainte de compatibilité entre les conteneurs et les chariots (les conteneurs doivent être chargés en fonction de leur destination). Il doit aussi respecter le taux de remplissage minimal des chariots : un chariot doit rester sur la station jusqu'à ce qu'il atteigne le taux de remplissage minimal  $\bar{A}$ . Tous les conteneurs placés sur l'anneau de tri doivent être expédiés dans la journée. Un conteneur peut faire plusieurs tours sur l'anneau avant d'être affecté au robot.

##### 4.2.1.1 Présentation du modèle

Pour la résolution hors ligne, la date de sortie  $r_j$  et les destinations  $c_j$  les dates d'arrivée des conteneurs sur l'anneau de tri sont connues à l'avance pendant tout l'horizon de planification.

Pour manipuler un conteneur, le robot a besoin de la présence d'un chariot avec la destination correspondante dans l'unité de chargement automatique, et  $o$  stations de sortie sont disponibles. Chaque chariot doit contenir entre  $\bar{A}$  et  $A$  conteneurs, et le remplacement d'un chariot par un nouveau prend  $\rho$  unité de temps. Pendant le remplacement d'un chariot, le robot peut continuer à charger d'autres destinations.

Les bacs arrivent sur l'anneau. Ils peuvent faire plusieurs tours en attente de prise en charge. L'anneau fait office de stock tampon, sa capacité est limitée et varie en fonction des centres de tri (entre 300 et 500). Au maximum  $I_{max}$  conteneurs peuvent être en attente de traitement. Lorsque la capacité de stockage est atteinte, il est possible de rejeter certains

conteneurs. Si un conteneur est rejeté, il est envoyé vers la zone de chargement manuel (elle n'est donc pas prise en compte dans le planning du robot). L'objectif est de maximiser le nombre de conteneurs pris en charge par le robot. En d'autres termes, l'objectif est de minimiser le nombre de conteneurs rejetés.

Le problème considéré peut-être formulé en programme linéaire mixte en nombres entiers avec les variables de décision suivantes :

- $h_i^t$  est égale à 1 si le conteneur  $i$  est prélevé et placé à l'instant  $t$  par le robot, et 0 sinon
- $u_i$  est égal à 1 si le conteneur  $i$  est rejeté par le robot, et à 0 sinon, avec  $i \in I$
- $\bar{s}_l$  heure de début de la manipulation du  $l^{ième}$  lot
- $\bar{c}_l$  heure d'achèvement du traitement du  $l^{ième}$  lot
- $b_{il}$  égale à 1 si le conteneur  $i$  est dans le  $l^{ième}$  lot, et 0 sinon
- $a_l$  est égal à 1 si le  $l^{ième}$  lot manipulé est utilisé (c'est-à-dire non vide)
- $s_i$  heure de début de la manipulation du  $i^{ième}$  conteneur
- $c_i$  heure d'achèvement de la manipulation du  $i^{ième}$  conteneur

$$\min \sum_{i=0}^{|B|} u_i \quad (1)$$

$$s_i = \sum_{t=1}^{|T|} t \cdot h_i^t \quad i \in B \quad (2)$$

$$\sum_{i \in B} h_i^t \leq 1 \quad t \in T \quad (3)$$

$$\sum_{t=1}^T h_i^t = 1 - u_i \quad i \in B \quad (4)$$

$$\sum_{l \in L} b_{il} = 1 - u_i \quad i \in B \quad (5)$$

$$s_i \geq (1 - u_i)r_i \quad i \in B \quad (6)$$

$$c_i = (1 - u_i)p \quad i \in B \quad (7)$$

$$|\{i | r_i \leq t\}| - \sum_{i \in B} \sum_{t=1}^T h_i^t - \sum_{i \in B} u_i \leq I_{max} \quad t \in T \quad (8)$$

$$\bar{s}_l \leq s_i + M(1 - b_{il}^k) \quad i \in B, l \in L T \quad (9)$$

$$h_i^t \leq \sum_{l \in L} b_{il} \quad i \in B, t \in T \quad (10)$$

$$\bar{c}_l \geq c_i + \rho - M(1 - b_{il}) \quad i \in B, l \in L \quad (11)$$

$$\bar{s}_l \leq \bar{c}_l \quad l \in L \quad (12)$$

$$\bar{c}_{l-o} \leq \bar{s}_l \quad l \in L, l \geq o \quad (13)$$

$$a_l * \bar{A} \leq \sum_{i \in B} b_{il} \leq a_l^k * A \quad l \in L T \quad (14)$$

$$b_{il} + b_{i'l} \leq 1 \quad (i, i') \in I, l \in L T \quad (15)$$

$$a_l \geq a_{l+o} \quad l \in L T \quad (16)$$

$$h_i^t = 0 \quad i \in B, t \in T \quad (17)$$

$$h_i^t \in \{0,1\} \quad i \in B, t \in T \quad (18)$$

$$u_i \in \{0,1\} \quad i \in B \quad (19)$$

$$b_{il} \in \{0,1\} \quad i \in B, l \in L \quad (20)$$

$$a_l \in \{0,1\} \quad l \in L \quad (21)$$

$$s_i, c_i \in N \quad i \in B \quad (22)$$

$$\bar{s}_l, \bar{c}_l \in N \quad l \in L \quad (23)$$

La fonction objectif (1) consiste à minimiser le nombre de conteneurs rejetés. Les contraintes (2) concernent l'heure de début de la manipulation de chaque conteneur. Les contraintes (3) stipulent que chaque conteneur peut se mettre dans un chariot au maximum. Les contraintes (4) (resp. (5)) stipulent que chaque conteneur peut soit être pris en charge par le robot soit rejeté. Les contraintes (6) stipulent qu'un conteneur ne peut pas être manipulé avant sa date de sortie. Les contraintes (7) calculent le temps d'achèvement de la manipulation d'un conteneur. Les contraintes de limite de stock sont données dans (8), où le niveau de stock (capacité de l'anneau de tri) dans chaque période est calculé comme étant le nombre de conteneurs sur l'anneau moins le nombre de conteneurs rejetés ou affectés au robot. Les contraintes (9) empêchent de manipuler un conteneur dont le chariot qui lui est associé ne se trouve pas sur une station. Les contraintes (10) stipulent que chaque conteneur pris en charge par le robot se voit attribuer un chariot. Les contraintes (11) stipulent que chaque chariot est présent sur la station de sortie jusqu'à l'arrivée du dernier conteneur appartenant à ce chariot plus le temps de changement de chariot, avec  $M = \max_{i \in B} \bar{d}_i$ . Les contraintes (12) stipulent que le traitement d'un chariot ne peut pas être fini avant la date de début de traitement de ce dernier. Les contraintes (13) stipulent qu'il y a au maximum un chariot par station de sortie à instant  $t$ . Les contraintes (14) imposent que le nombre de conteneurs dans chaque chariot soit compris dans l'intervalle  $[\bar{A}, A]$ . Les contraintes (15) empêchent deux conteneurs ayant deux destinations différentes d'être dans le même chariot. Enfin, les contraintes (16) brisent la symétrie.

#### 4.2.1.2 Analyses et résultats de la résolution hors-ligne

Les solutions optimales hors-ligne sont calculées en résolvant le modèle avec CPLEX 12.9. Les expérimentations ont été réalisées sur des instances avec moins de 200 conteneurs. Ces instances correspondent à des données réelles définies à partir d'un centre de tri de La Poste. Le nombre de conteneurs étant faible, la capacité de rétention maximale de l'anneau de tri (stock) est réduite à la moitié du nombre de conteneurs.

CPLEX n'a pas été en mesure de résoudre les instances avec plus de 200 conteneurs. Comme le montre la **Table 14** pour  $\alpha = 10$ , lorsque la valeur de  $\bar{A}$  est importante et que le

nombre de conteneurs est de 50 et 100, la solution optimale est donnée en moins de 0,2 secondes par CPLEX. Cependant, lorsque  $|B| = 200$  et  $\bar{A} = 4$ , le temps de calcul augmente.

CPLEX n'a pas été en mesure de résoudre les instances avec 200 conteneurs et  $\bar{A} = 1$  car le nombre de contraintes était trop important pour construire le modèle CPLEX.

**Table 14** Résultats de la résolution hors-ligne

$ B $	$\bar{A}$	Nombre de rejets	Temps (s)
50	1	14	12.6
	4	22	0.2
	8	22	0.1
	12	22	0.1
	16	31	0.1
100	1	36	295.9
	4	52	0.9
	8	56	0.3
	12	56	0.2
	16	68	0.6
200	1	-	-
	4	88	137.9
	8	108	7.4
	12	118	1.1
	16	120	2.3

Devant les limites rencontrées avec le solveur CPLEX pour la résolution du problème pour les données réelles, nous avons décidé d'investiguer l'utilisation d'heuristiques en ligne pour planifier dynamiquement l'affectation des conteneurs au robot. Nous présenterons ces heuristiques dans la section suivante.

#### 4.2.2 Résolution en ligne : Heuristiques

Comme les conteneurs arrivent de manière aléatoire, dans un ordre inconnu, nous étudions plusieurs heuristiques pour répondre à chacune des questions, posées dans la section 4.2.1, à savoir : règle d'affectation des destinations, règle de remplacement des chariots, règle de rejet des conteneurs.

##### 4.2.2.1 Présentations des heuristiques

###### Règle d'affectation des destinations

Dès qu'une station sur le robot est disponible et doit être affectée à une destination, on choisit parmi les règles suivantes :



- Most Frequent Destination (**MFD**) : les destinations les plus fréquentes ayant au moins  $\alpha$  conteneurs sur l'anneau de tri sont affectées au robot. Avec :  $\alpha$  = nombre minimum de bacs nécessaire avant d'affecter la destination vers le robot. Cela gardera le robot occupé, et évitera qu'un chariot non rempli ne bloque une des stations de la cellule robotisée.
- Earliest Release Date (**ERD**) : les conteneurs qui arrivent les premiers sont affectés en premier au robot.
- Less Loaded Package (**LLP**) : on choisit les conteneurs dont la destination appartient au chariot le moins rempli afin de remplir les chariots plus rapidement.
- Randomly (**R**) : on choisit aléatoirement les conteneurs à envoyer vers le robot.

#### Règle de rejet des conteneurs

Quand le nombre de bacs sur l'anneau de tri atteint un seuil  $I_{max}$  prédéfini et pour ne pas bloquer l'anneau on choisit une parmi les règles suivantes :

- Least Frequent Destination (**LFD**) : les conteneurs, dont la destination est la moins fréquente, doivent être envoyés vers le chantier manuel. LFD facilite la gestion du stock des conteneurs tout en gardant les conteneurs appartenant aux destinations les plus populaires sur l'anneau.
- Randomly (**R**) : On envoie aléatoirement les conteneurs vers le manuel pour ne pas saturer l'anneau de tri

#### Règle de remplacement de chariot

Quand une destination importante (dont le nombre de bacs permet de remplir un chariot) est présente dans la liste d'attente du robot, le choix de la destination à remplacer parmi ceux en cours de remplissage sur l'unité robotique doit être effectué. On choisit parmi les règles suivantes :

- Least Recently Used (**LRU**) : parmi les chariots remplis au-dessus du seuil minimal  $\bar{A}$ , les chariots inactifs depuis le plus longtemps sont remplacés. La règle LRU est couramment utilisée dans le problème de la pagination (Räcke et al., 2002). Elle est pertinente dans le contexte de La Poste puisque le flux de courriers n'est pas homogène au cours de la journée.
- Most Loaded (**ML**) : pour cette règle, on choisit parmi les destinations présentes dans la cellule robotisée, celle qui est la plus chargée pour la remplacer afin de minimiser le nombre de changements.

#### 4.2.2.2 Comparaison des heuristiques

Les heuristiques ont été implémentées dans Julia. Les résultats ont été évalués sur les données de trois journées types. Le seuil maximal, le temps de traitement et le temps de changement de chariot sont fixés à  $A = 16$ ,  $p = 1$ , et  $\rho = 7$ , respectivement. Le nombre de stations varie entre 8 et 10, la capacité de l'anneau  $I_{max} \in \{100, 200, 400\}$  et le seuil de remplissage minimal  $\bar{A} \in \{1, 4, 8, 12, 16\}$ .

En se basant sur ces valeurs, les expérimentations ont été réalisées sur 90 instances.

Tout d'abord, les performances de chaque combinaison de règles heuristique ont été évaluées. Les résultats montrent que la meilleure combinaison consiste à sélectionner le prochain conteneur selon la règle MFD, à rejeter les conteneurs avec LFD, et à remplacer le chariot avec LRU. En effet, cette heuristique conduit aux meilleurs résultats en moyenne sur toutes les instances.

La **Table 15** montre l'impact des différentes règles. Plus précisément, il donne l'écart en pourcentage entre le nombre moyen de rejets obtenus par l'heuristique (sur toutes les instances et toutes les heuristiques) en utilisant chaque règle et le nombre moyen de rejets obtenus par les méthodes utilisant le meilleur ensemble de règles (sur toutes les instances et toutes les heuristiques). La règle de sélection des conteneurs a un impact significatif. Par exemple, MFD conduit à 4.63 % moins de rejets que la règle ERD. De même, la règle de rejet des conteneurs a un impact important, LFD conduit à 23 % de moins de rejets que R. Au contraire, la règle de remplacement des chariots a un faible impact avec un écart de 0.91 % seulement entre LRU et ML.

Toutes les heuristiques testées donnent de meilleurs résultats que les règles basées sur le choix aléatoire R. Les meilleurs résultats correspondent aux règles qui accélèrent le remplissage et le remplacement des chariots. En effet, la règle MFD affecte au robot la destination la plus populaire avec le plus grand nombre de conteneurs sur l'anneau au moment de la décision. Cela ne garantit pas que le chariot correspondant soit rempli rapidement, mais dans une approche dynamique, c'est la meilleure stratégie pour garantir que le chariot soit rempli. La règle LRU élimine les chariots dont le taux de remplissage est potentiellement plus faible que les autres.

**Table 15** Comparaison des règles heuristiques

Règle d'affection des destinations	ERD	MFD	LLP	R
Ecart (%)	4.63	0.00	2.67	9.72
Règle de rejet des conteneurs	R	LFD		
Ecart (%)	23.15	0.00		
Règle de remplacement de chariot	LRU	ML		
Ecart (%)	0.00	0.91		

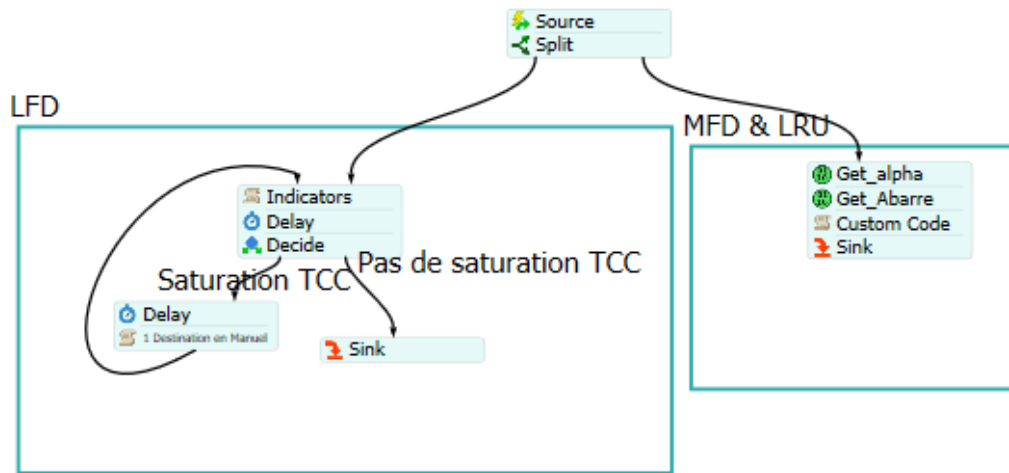
### 4.3 Expérimentations

Dans cette section, nous présentons les résultats des expérimentations. Les règles heuristiques choisies dans la section précédente, à savoir : MFD, LFD, LRU seront évaluées par simulation en utilisant le simulateur FlexSim. Afin de s'assurer de l'efficacité de la comparaison avec les résultats du SEBC, les mêmes journées-types (le 15, 17 et 19 octobre 2018) seront utilisées pour l'évaluation des heuristiques.

#### 4.3.1 Présentation du modèle de simulation

Les heuristiques MFD, LFD et LRU ont été implémentées dans le modèle en utilisant le module logique « Process flow » (**Figure 42**). Nous avons choisi différents scénarios pour l'évaluation du modèle. Le vecteur paramètre change pour chaque scénario (**Figure 43**). Les mêmes

ressources présentées dans le chapitre précédent seront utilisées pour la conception de ce modèle.



**Figure 42** Processus logique présentant les règles heuristiques

Parameters						
	R1	R2	R3	alpha	Ibarre	Abarre
Row 1	MFD	LFD	LRU	12	200	12

**Figure 43** Vecteur de paramétrage

Dès l'entrée du premier conteneur sur l'anneau de tri, la liste contenant le nombre des conteneurs est mise à jour pour être comparée à la valeur prédéfinie de  $I_{max}$ . Cela permettra de ne pas saturer l'anneau de tri et de ne pas bloquer ses entrées et sorties. Si le seuil de  $I_{max}$  conteneurs est dépassé, les conteneurs appartenant aux destinations les moins populaires sont envoyés vers le manuel.

Une fois sur l'anneau, les conteneurs peuvent faire plusieurs tournées avant d'être affectés au robot ou au manuel. Le nombre de conteneurs ayant une destination commune est calculé continuellement. La destination qui atteint le seuil de  $\alpha$  conteneurs est affectée au robot, si au moins une station de sortie est disponible sur la cellule robotisée. Un chariot est considéré plein si sa hauteur atteint 4 074 mm (entre 15 et 21 conteneurs selon le type, s'il contient des courriers grands ou petits formats). Une fois plein, le chariot est remplacé par un autre vide, manuellement par un opérateur, et une autre destination pourra être affectée au robot en faisant appel à la règle MFD.

Si une destination populaire apparaît sur l'anneau de tri et qu'aucune station de sortie ne soit disponible, la destination qui a au moins  $\bar{A}$  conteneurs et qui a reçu son dernier conteneur depuis le plus longtemps est affectée au robot.

### 4.3.2 Analyse de sensibilité

Les analyses de sensibilités sont effectuées sur des données réelles où on considère un robot avec 10 stations de sorties.

La zone de stockage qui est l'anneau de tri, peut actuellement stocker jusqu'à 500 conteneurs, selon les dimensions du centre de tri. Comme La Poste souhaite conserver un stock de sécurité d'une centaine de conteneurs, nous avons testé 100, 200 et 300 pour  $I_{\max}$ .

$I_{\max}$  ne concerne que les conteneurs en attente de traitement (manuel ou robotique). Pour étudier l'impact de la capacité du stock sur le nombre de conteneurs prélevés et placés par le robot, nous avons fait trois simulations sur les trois journées types avec les paramètres :  $\bar{A} = 12$  et  $\alpha=8$ .

La **Table 16** montre l'impact non négligeable de la capacité de rétention de l'anneau de tri sur le nombre de conteneurs collectés par le robot. En effet, pour la journée du 15/10 par exemple, une capacité de 300 conteneurs entraîne 19 % de taux de prise en charge en plus par rapport à une capacité de 100 pour la. La variation du pourcentage de conteneurs affectés au robot n'est pas linéaire par rapport à  $I_{\max}$ . Une explication possible est que le nombre de destinations et le nombre de conteneurs par destinations n'est pas stable et fluctuent entre les journées.

Le meilleur résultat est obtenu pour une capacité de rétention maximale entre 200 et 300 conteneurs.

**Table 16** Variation du pourcentage de conteneurs par rapport à  $I_{\max}$

	100	200	300
15/10	23 %	30 %	42 %
17/10	30 %	35 %	24 %
19/10	34 %	35 %	32 %
Moyenne	29 %	33 %	32 %

Les paramètres  $\alpha$  qui représente le seuil déclenchant le traitement des conteneurs ayant une destination commune et  $\bar{A}$  le seuil de remplissage minimal, doivent être soigneusement définis. La **Table 17** montre que, pour  $\bar{A} = 12$  et  $I_{\max} = 200$ , en fixant  $\alpha$  entre 4 et 8, on obtient un taux de conteneurs pris en charge par le robot plus important que pour une valeur plus grande de  $\alpha$ . En effet, pour une valeur importante de  $\alpha$ , on augmente la chance de rejeter les conteneurs appartenant à des destinations fréquentes vers le manuel.

Comme le montre La **Table 18**, pour  $\alpha = 4$  et  $I_{\max} = 200$ , fixer  $\bar{A}$  entre 12 et 16 semble un bon compromis pour équilibrer le nombre de conteneurs rejetés et le nombre de chariots créés. Le meilleur taux moyen est obtenu lorsque  $\bar{A}$  est égale à 16. Lorsque  $\bar{A}=8$ , le taux moyen de conteneurs pris en charge par le robot sur les trois journées est 8 % inférieur à la meilleure valeur obtenue avec  $\bar{A}=16$ .

Comme la règle MFD est efficace, il serait intéressant de choisir des valeurs spécifiques pour  $\bar{A}$  et  $\alpha$  pour chaque destination. Cependant, pour une destination donnée, le flux de lettres varie d'un jour à l'autre, ou même entre les périodes de la journée, donc ce paramètre devrait être modifié dynamiquement pour chaque destination. La sélection dynamique des paramètres  $\bar{A}$  et  $\alpha$  est laissée pour un travail futur.

**Table 17** Variation du pourcentage de conteneurs par rapport à  $\alpha$

	4	8	12
15/10	30.56 %	31.28 %	30.75 %
17/10	36.15 %	35.25 %	28.62 %
19/10	30.36 %	34.95 %	31.55 %
Moyenne	32.35 %	33.82 %	30.30 %

**Table 18** Variation du pourcentage de conteneurs par rapport à  $\bar{A}$

	8	12	16
15/10	27.48 %	30.56 %	30.93 %
17/10	33.59 %	36.15 %	35.40 %
19/10	15.48 %	30.36 %	34.02 %
Moyenne	25.51 %	32.35 %	33.45 %

#### 4.4 Résultats et comparaisons

La **Table 19** donne le nombre de chariots plein, le nombre moyen de conteneurs par chariot ainsi que le pourcentage de conteneurs pour les trois journées types étudiées pour  $\alpha = 8$ ,  $\bar{A} = 16$  et  $I_{\max} = 200$ . Les résultats sont meilleurs avec les stratégies de décisions basées sur les heuristiques qu'avec celles basées sur le système expert défini comme borne basse de comparaison.

**Table 19** Pourcentage de conteneurs avec : ( $\alpha = 8$ ,  $\bar{A} = 16$  et  $I_{\max} = 200$ )

	Pourcentage de conteneurs	Nombre de chariots pleins	Nombre moyen de conteneurs par chariot
15/10	34 %	368	16
17/10	32 %	405	15
19/10	36 %	520	17

#### 4.5 Conclusion

Dans ce chapitre, nous avons présenté deux méthodes basées sur les techniques de recherche opérationnelle pour l'ordonnancement des tâches affectées au robot. Une formulation mathématique a été proposée ainsi que la résolution exacte à l'aide du solveur CPLEX. Cette

méthode a montré ses limites devant la taille importante de notre problème industriel. En effet, le temps de calcul évolue de manière exponentielle avec la taille du problème. Un ensemble d'heuristiques intégrant des stratégies de décision a été proposé. Ces heuristiques ont été testées et évaluées par simulation sur des données réelles.

Les résultats obtenus montrent l'efficacité de ces heuristiques pour répondre au caractère dynamique et incertain du problème.

## Chapitre 5

### 5 Implémentation par SMA

Dans la littérature (Boissier et al., 2004; MANO et al., 2005), trois types d'applications des systèmes multi-agent (SMA) sont distingués : la simulation, le contrôle et l'intégration :

- Les SMA sont de plus en plus utilisés pour modéliser et simuler des systèmes réels complexes dont l'environnement est souvent dynamique. Cela permettra de mieux comprendre leurs comportements et de prédire leurs performances. Les simulations reposent souvent sur des techniques hors-ligne de testabilité. Elles permettent d'expérimenter des scénarios dont la réalisation sur des systèmes réels n'est pas possible.
- Récemment, les SMA sont devenus des solutions prometteuses pour le contrôle en temps réel des problèmes dans différents domaines. La résolution par SMA permet de reformuler le problème en mettant en lumière les caractéristiques cognitives et comportementales des agents, nécessaires à la réalisation d'une mission collective. Les fonctions de contrôle sont ainsi distribuées sur tous les agents. Un protocole de négociation permet de résoudre les problèmes de conflits entre les agents.
- Les SMA représentent aujourd'hui une technologie de choix pour la conception d'applications nécessitant une coopération entre humains, logiciels et systèmes mécaniques. Ces applications cherchent à répondre à des exigences de plus en plus complexes du fait de la multiplicité des services, des acteurs et des ressources.

La notion d'agent industriel est liée au deuxième type d'applications.

Nous avons opté pour l'utilisation de la technologie multi-agent pour résoudre notre problématique de PAP industriel. Comme nous l'avons déjà montré, les méthodes classiques ont montré leurs limites face à la taille de notre problème industriel. L'avantage de l'utilisation d'un tel système réside principalement dans sa scalabilité et sa stabilité face aux variations dans les environnements incertains.

Dans ce chapitre nous allons présenter notre système de contrôle basé sur la technologie multi-agent. Ce système est conçu à partir des règles d'affectation établies dans le chapitre précédent. Nous présenterons dans un premier temps quelques éléments sur la notion d'agent industriel, puis nous préciserons la méthodologie d'analyse et de conception adoptée pour

l'implémentation de notre système multi-agent. Cette méthodologie a été inspirée de différentes approches de la littérature. Nous présenterons également la plateforme choisie pour le développement de notre système.

Une comparaison des expérimentations avec les résultats des travaux présentés dans le chapitre précédent permettra de juger la pertinence du SMA.

### 5.1 Agents industriels

Au cours des décennies, l'utilisation des SMA a suscité un grand intérêt de la part des chercheurs dans le domaine du contrôle des systèmes de production. Cette forte utilisation est justifiée par le besoin de plus en plus de systèmes flexibles, agiles, scalables et capables de gérer la variabilité et les perturbations au sein d'un marché économique qui ne cesse d'évoluer.

Dans cette section, nous présentons une brève définition de la notion de l'agent ainsi que quelques travaux de la littérature sur l'emploi des SMA pour gérer les systèmes dynamiques dans des environnements à fortes perturbations.

#### 5.1.1 Présentation des SMA et de la notion d'agent

Les mécanismes traditionnels de contrôle, généralement basés sur des architectures centralisées et hiérarchiques, sont de moins en moins aptes à suivre l'évolution des systèmes de production et à assurer leurs stabilités face à la variation très dynamique et rapide des exigences de la production (Trentesaux, 2002). Étant donné l'énorme coût et les longs délais de leurs implémentations et leurs rigidités, de tels systèmes hiérarchiques centralisés qui ont des capacités limitées d'adaptation conduisent à des situations où l'ensemble du système de production est mis à l'arrêt suite à la défaillance d'une seule machine. Le défi est donc de développer un système de contrôle flexible, collaboratif et robuste (Colombo et al., 2006).

Les SMA répondent parfaitement à ces exigences en offrant une meilleure robustesse, une meilleure adaptation aux changements et une plus grande efficacité pour systèmes de production. Grâce aux entités autonomes qui les constituent, appelée *Agents*, les SMA exécutent de manière décentralisée et autonome les différentes opérations afin d'atteindre un objectif final. En se basant sur les définitions de (Ferber, 1995; Wooldridge, 2002; Wooldridge and Jennings, 1995), il est possible de définir un agent comme *une entité autonome qui représente un objet physique (comme un robot ou une machine) ou un objet logique virtuel (comme une commande) capable de percevoir son environnement, sociable (capable de communiquer) et proactive (peut prendre l'initiative de décider)*.

À partir des définitions existantes, les caractéristiques les plus importantes d'un agent sont (**Figure 44**) :

**Autonomie** : un agent est capable d'agir en lui-même sans l'intervention d'une entité extérieure pour atteindre ses buts. Il possède le contrôle sur son état et son comportement.

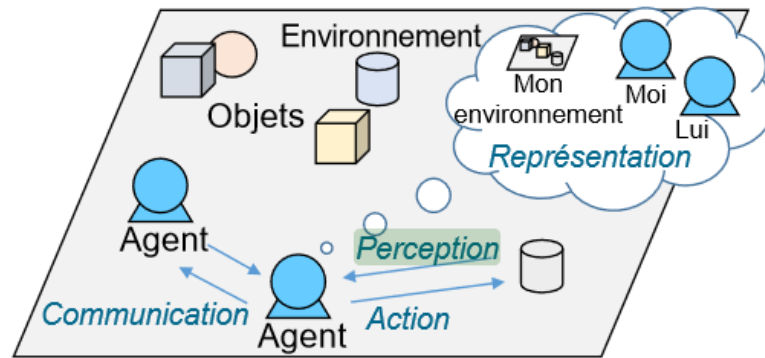
**Perception** : c'est la représentation (souvent partielle) de l'agent de son environnement.

**Réactivité** : la capacité d'un agent à percevoir son environnement et à réagir pour répondre aux changements qui peuvent avoir lieu dans un temps adéquat.



**Proactivité** : un agent doit être capable de prendre l’initiative pour atteindre son objectif et ne pas se limiter à répondre à son environnement.

**Sociabilité** : Un agent peut interagir et coopérer avec d’autres agents afin de réaliser sa mission.



**Figure 44** Schéma représentatif des interactions d’un agent dans son environnement (Guériaux, 2016)

Les études de la littérature font la distinction entre deux classes d’agents : Agents cognitifs et agents réactifs (la **Figure 45** résume la différence). La première classe est la plus répandue dans le domaine de l’intelligence artificielle distribuée. Dans ce cadre, selon (Ferber, 1995), le SMA est composé d’un nombre réduit d’agents “intelligents”. Ces agents ont des comportements complexes et une base de connaissances assez riche leur permettant d’avoir une représentation explicite de leur environnement. L’autre classe d’agents réactifs affirme qu’il n’est pas nécessaire d’avoir des agents intelligents individuellement pour avoir un SMA intelligent. En effet, les agents ont un comportement “stimulus-réponse”, ils réagissent à des stimuli simples issus de leur environnement sans avoir une représentation de ce dernier ni de vrais mécanismes de raisonnement. Cependant, du fait de leur nombre élevé, les agents réactifs peuvent résoudre des problèmes qualifiés de complexes.

AGENTS COGNITIFS	AGENTS REACTIFS
Représentation explicite de l’environnement	Pas de représentation explicite
Peut tenir compte de son passé	Pas de mémoire locale
Agents complexes	Fonctionnement stimulus/action
Nombre d’agents réduit	Nombre d’agents élevé

**Figure 45** Les agents cognitifs vs réactifs (Reichgelt, 1990)

Dans un SMA, les agents sont capables d’agir en toute autonomie et de décider librement de leurs propres actions, pour accomplir leurs objectifs individuels. Toutefois, quand leurs visions partielles du système et de leurs environnements ne leur permettent pas d’avoir suffisamment de connaissances pour le faire, ils collaborent et communiquent ensemble pour

le réaliser (Leitão, 2009) en utilisant des langages, ontologies et protocoles de communication appropriés.

### 5.1.2 Les SMA pour gérer la variabilité dans un environnement incertain

Pour qu'un système opérant retrouve un état d'équilibre stable après une perturbation (variabilité des données d'entrée du système, annulation de commande, panne, etc.), son système de pilotage, basé sur la technologie multi-agent, fait appel à l'ensemble des agents pour exécuter leurs comportements programmés. Chaque agent, en répondant aux principes d'autonomie et de localité, accomplit une action. Cette action est la conséquence de sa perception de son environnement et de ses relations avec les autres agents de l'organisation. Les agents considèrent une perturbation comme une donnée normale dans l'évolution du système et la traite donc sans aucune difficulté (Ferber, 1995).

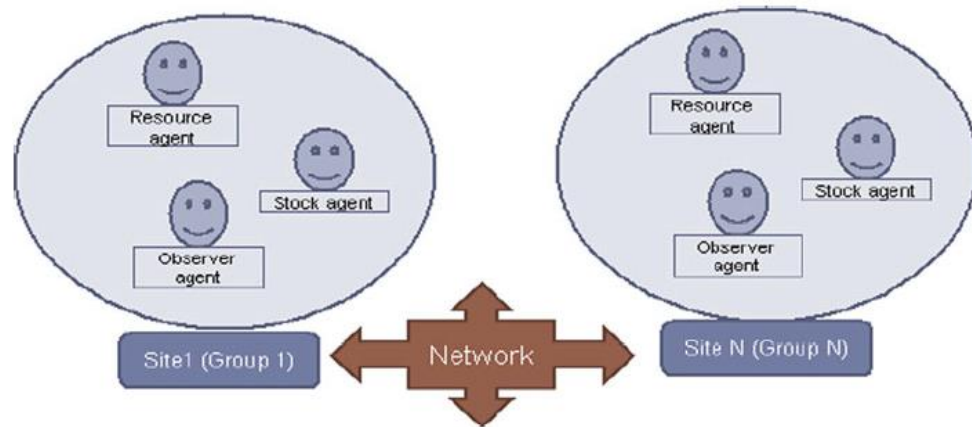
À l'analyse de la littérature autour de l'usage des SMA, pour faire face à la variabilité et aux différentes perturbations dans les systèmes de production, nous avons constaté que les systèmes étudiés se comportent souvent de la même manière. Dans un état stable, chaque agent du système cherche à atteindre son objectif individuel. Ils coopèrent, si besoin, afin de parvenir à réaliser un objectif commun. Dès qu'une perturbation est détectée (retard, événement inattendu, panne, etc.), le système rentre dans un état d'instabilité, les agents changent alors leurs comportements afin de répondre à cette perturbation en cédant le contrôle à un seul type d'agent (souvent appelé superviseur ou observateur). Le raisonnement doit se faire rapidement pour maîtriser les conséquences le plus rapidement possible. Dans ce cas, l'agent superviseur peut choisir entre trois comportements : (1) contacter directement l'agent concerné pour lui demander des solutions ; (2) lancer un appel d'offres et comparer les solutions proposées afin de choisir la solution optimale ; (3) chercher lui-même la solution alternative qui permet de corriger la perturbation et de l'envoyer aux agents concernés pour application. La convergence vers un état stable est détectée quand les agents exécutent la solution proposée par l'agent superviseur. Nous allons nous inspirer de cette approche pour la conception de notre système multi-agent. Dans la suite de cette sous-section, nous présentons quelques exemples de la littérature.

Les auteurs dans (Mguis et al., 2012) présentent un SMA pour la résolution d'un problème de tournées de véhicules dans une situation d'urgence dans un environnement incertain et dynamique. Le système conçu permet la planification des tournées de manière à éviter les retards de services qui peuvent avoir lieu à cause d'une perturbation (panne d'un véhicule, changement de route, etc.). Dès la réception d'une alerte signalant une perturbation, un agent « interface » informe l'agent « superviseur » par cette dernière. L'agent superviseur lance un appel d'offres aux différents agents du système. Les agents capables de proposer une réponse satisfaisante à cette demande envoient un message contenant l'offre proposée à l'agent superviseur qui compare les offres et choisit la meilleure. L'agent superviseur envoie la nouvelle planification à l'agent interface qui se charge d'envoyer les instructions nécessaires pour la correction aux agents concernés par la perturbation. Les auteurs présentent en détail les différentes étapes de la méthodologie de conception du système.

Dans (Mezgebe et al., 2019), les auteurs proposent l'application d'un algorithme de consensus pour un SMA de production afin de contrôler les événements inattendus comme dans le cas d'une commande urgente. L'algorithme de consensus permet aux agents localement autonomes de communiquer ensemble, de collaborer et de négocier entre eux afin de parvenir à un accord vers un objectif commun. Pour valider l'applicabilité de l'algorithme proposé, un problème de contrôle industriel, tiré de l'industrie du bois et transposé à une plateforme d'application académique, a été utilisé comme cas d'étude. Le service de planification de l'entreprise propose un planning prévisionnel hebdomadaire pour chaque commande client. L'objectif est de maintenir la même performance du planning prédictif après qu'une commande urgente de dernière minute ait eu lieu. Pour ce faire, les agents « produits » utilisent l'approche Leader-Follower variant dans le temps pour mettre en œuvre le consensus. Cette approche consiste en un groupe d'agents leaders disposant d'un certain pouvoir sur un grand groupe d'agents suiveurs. Tous les agents leaders négocient entre eux et partagent l'offre retenue avec leurs suiveurs pour la mettre en œuvre. Le consensus sera atteint si la variabilité des intérêts de tous les agents produits se rapproche de zéro. Les résultats des simulations ont montré l'efficacité de l'algorithme de consensus pour la minimisation de l'impact des perturbations sur la durée totale des travaux (makespan). Des travaux similaires peuvent être trouvés dans (Mezgebe et al., 2020).

(Aissani et al., 2012) proposent un modèle multi-agent réactif pour un système de contrôle de production qui génère des solutions d'ordonnancement en ligne pour des entreprises multi-sites tout en garantissant l'atteinte d'un objectif global. Chaque site a été modélisé comme un ensemble composé d'un agent « observateur » et d'un sous-ensemble de plusieurs agents « stock » et agents « ressources » (**Figure 46**). Le module de prise de décision de l'agent ressource est basé sur un algorithme d'apprentissage réactif basé sur l'apprentissage par renforcement SARSA. Cet algorithme permet d'une part à l'agent Ressource d'être flexible et de fournir une réponse optimale dans un environnement dynamique et d'autre part au système d'être réactif et de répondre en temps-réel aux perturbations qui peuvent se manifester.

Dans la société Mère, quand l'agent observateur reçoit une commande, il crée une demande pour connaître l'état du stock et l'envoie aux agents stock des différents sites. L'agent stock est ensuite consulté pour savoir si le type de produit demandé est disponible en stock. Dans tous les cas, l'agent stock demande aux agents ressources de proposer une solution. Grâce à son module de prise de décision, l'agent ressource génère la solution optimale, ainsi que sa récompense. Toutes ces solutions seront renvoyées à l'agent observateur, qui choisira la meilleure solution en terme de récompense. Ce processus d'interaction aide les agents à prendre la décision la plus efficace pour le planning de production. Une comparaison du modèle multi-agent développé avec un algorithme génétique et un Programme Linéaire en Nombres Entiers montrent son efficacité en termes de réactivité et son aptitude à trouver la solution optimale.



**Figure 46** Modèle multi-agent pour le système multi-sites

Un SMA a été développé dans (Chiddarwar and Babu, 2011) pour la planification coordonnée des mouvements de quatre robots. En cas de conflit entre les robots qui partagent le même espace de travail, les auteurs proposent une séquence de modification de trajectoire qui utilise un algorithme incrémental pour modifier la trajectoire de chaque robot. En effet, chaque robot est considéré comme un obstacle dynamique par les robots voisins pendant la planification de la tâche. Parmi les différents agents nécessaires à la réussite de mise en œuvre du système proposé, l'agent « C », qui joue le rôle de l'agent « superviseur ». Cet agent utilise une approche à deux phases pour arriver à son objectif. Il génère le chemin optimal sans collision pour chaque robot dans la première phase de planification du chemin. Dans le cas d'un conflit entre les robots, l'agent C passe à la deuxième phase dénommée phase de coordination, où il modifie le chemin de chaque robot, sélectionne le chemin le plus court et calcule le chemin de coordination entre les robots. L'approche proposée a permis de réduire considérablement le temps nécessaire à l'exécution des tâches, d'optimiser la consommation d'énergie et d'augmenter la flexibilité du système grâce à la nature plug-and-play de ses modules.

## 5.2 Méthodologies de conception du SMA

Une méthodologie de conception est l'ensemble de modèles, de modélisation, d'approches et de concepts permettant de formaliser les phases de développement d'un système et de donner un guide général à suivre pour mieux répondre aux besoins initiaux. Une brève analyse des méthodologies de développement des SMA existantes sera présentée dans cette section, suivie d'une description et de l'application de la méthodologie utilisée sur notre cas d'étude.

### 5.2.1 Analyse et comparaison des méthodologies

Il existe dans la littérature de nombreuses méthodologies qui ont pour but la formalisation et la structuration des étapes de conception des SMA, allant de la phase d'analyse des besoins jusqu'à la phase d'implémentation. Chacune d'entre elles a sa propre base conceptuelle propre et recouvre un certain nombre d'aspects dans leurs domaines d'application. Cela explique la raison pour laquelle peu de méthodologies sont arrivées à maturité ou sont décrites de manière suffisamment détaillée pour être considérées comme une norme pour étudier et concevoir les différents SMA. Parmi ces méthodes, on peut ainsi citer : Tropos, Gaia, PASSI, MaSE,

Prometheus, AAI, Alaadin, DESIRE, Voyelles, ASPECS, INGENIA, MESSAGE, MMTS, MASSIVE, Agent UML, MAS-ML, MASK, etc.

Plusieurs auteurs ont travaillé sur des études comparatives des différentes méthodologies afin de comprendre les points forts, les points faibles et les domaines d'application de chaque méthodologie. Le but étant d'aider le concepteur à faire son choix de la méthode la plus appropriée pour résoudre un problème donné. (Bernon et al., 2002) ont comparé quelques méthodologies orientées objets suivant huit critères principaux : des plus généraux comme la prise en compte des différentes phases de l'identification des besoins jusqu'au déploiement, aux plus précises, comme la proposition de guide pour l'identification des agents pour les systèmes les plus complexes. (Dam and Winikoff, 2003) ont, quant à eux, comparé les trois méthodes MaSE, Prometheus et Tropos en se basant sur quatre aspects majeurs, à savoir : les concepts, le langage de modélisation, le processus et la pragmatique. (Sabas et al., 2001, 2002) ont comparé les principales méthodologies de SMA (GAIA, MaSE, MMTS, etc.) suivant six axes de comparaison. Le but est de proposer une piste d'unification des méthodologies étudiées. D'autres études comparatives peuvent être trouvées dans (Henderson-Sellers and Giorgini, 2005; O'Malley and DeLoach, 2002; Shehory and Sturm, 2001).

Une mise à jour d'une étude comparative, qui avait été faite précédemment, nous a permis d'établir le tableau suivant (**Table 20**) basé sur l'évolution à ce jour de quelques méthodologies en fonction des étapes couvertes par le processus de développement.

**Table 20** Table de comparaison des méthodologies de conception de SMA

Méthodologie Phase	MaSE	GAIA	DESIRE	Alaadin	Tropos	Voyelles	PASSI	AAII
Définition des besoins	--	-	-	-	++	-	+	-
Analyse du système	++	++	±	++	++	++	++	++
Conception	++	++	++	+	+	++	++	+
Implémentation	+	--	±	++	+	+	+	--
Test	±	-	++	-	±	+	±	--
Gestion de la qualité	--	--	--	--	--	--	--	--
Gestion du projet	--	--	--	--	--	---	--	--

Notations :

- ++ : phase couverte par la méthodologie
- + : phase couverte par la méthodologie de manière indirecte
- -- : phase non couverte par la méthodologie
- - : phase explicitement non couverte par la méthodologie
- ± : phase potentiellement couverte par la méthodologie

Nous précisons que l'objectif de cette section n'est pas de faire l'état de l'art sur les méthodologies existantes de conception des SMA mais d'en donner un bref aperçu afin de présenter la démarche que nous avons suivie pour le déploiement de notre SMA. Cette démarche a été déduite en analysant les différentes méthodologies qui existent dans la littérature et en exploitant les avantages que certaines approches offrent par rapport à d'autres.

Une analyse de la littérature a permis d'identifier trois phases principales communes entre la majorité de méthodologies employées pour la conception des SMA :

- **Phase d'analyse** : elle permet de décrire le contexte et d'identifier les besoins du système compte-tenu des objectifs. Cela facilitera la compréhension du problème et l'identification des différents rôles nécessaires pour l'accomplissement de la mission de chaque entité du système.
- **Phase de conception** : cette étape permet de bien définir les fonctionnalités du système et de son environnement en identifiant les différents groupes, classes et organisations. Les protocoles d'interactions entre les différents acteurs (les agents) ainsi que les paramètres nécessaires à l'implémentation sont spécifiés lors de cette étape.
- **Phase d'implémentation** : la structure interne de chaque agent du système ainsi que ses comportements interne et externe sont définis lors de cette étape. Cette dernière étape permet d'analyser les résultats (test et validation).

Selon le besoin et le niveau de complexité du système étudié, il est possible de modéliser ou non les différentes phases en évaluant leur pertinence et leur apport en terme de précision pour le développement du système. La **Figure 47** présente les différents modèles que l'on peut utiliser pour les différentes phases de mise en œuvre du SMA.

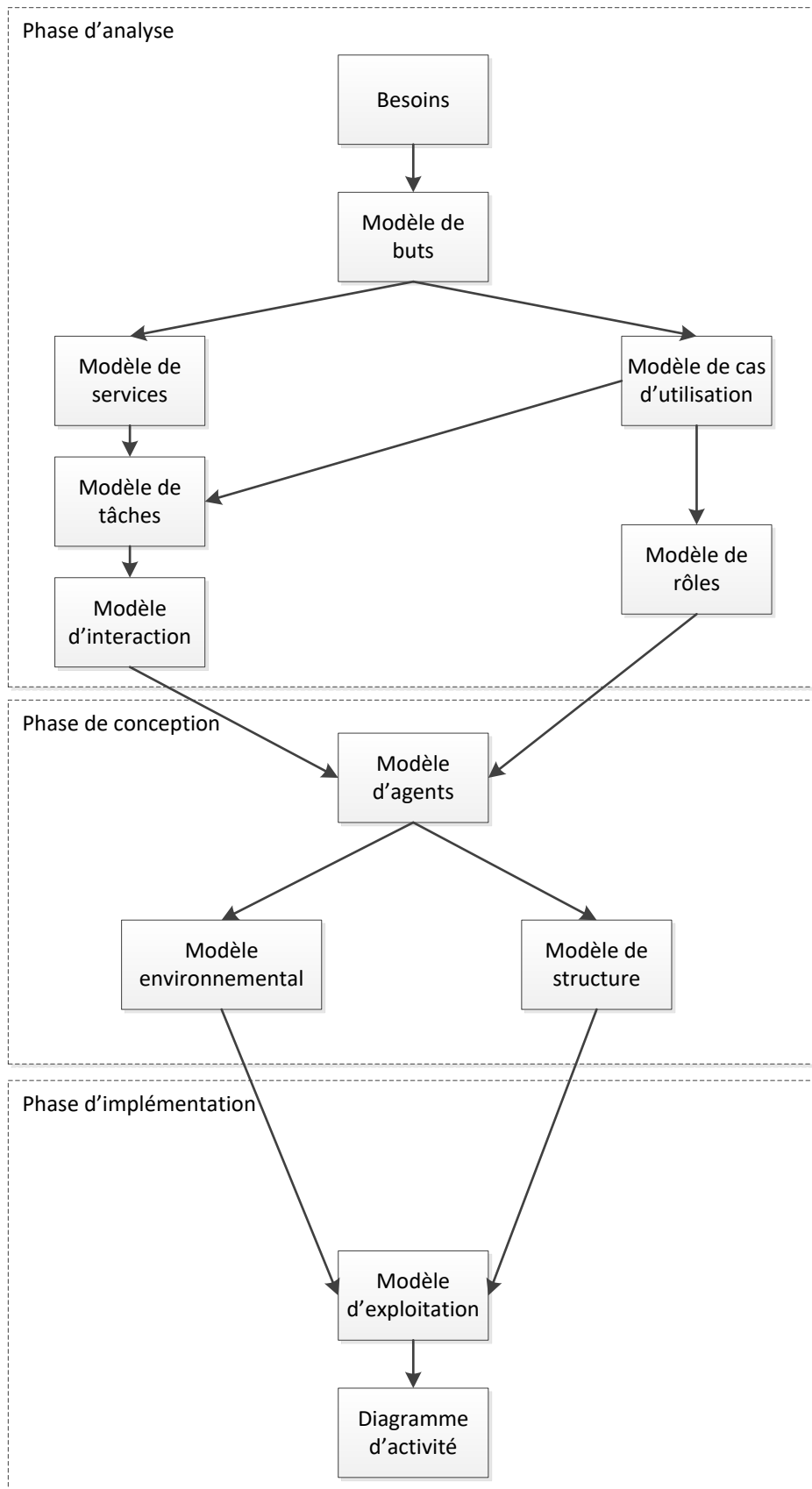
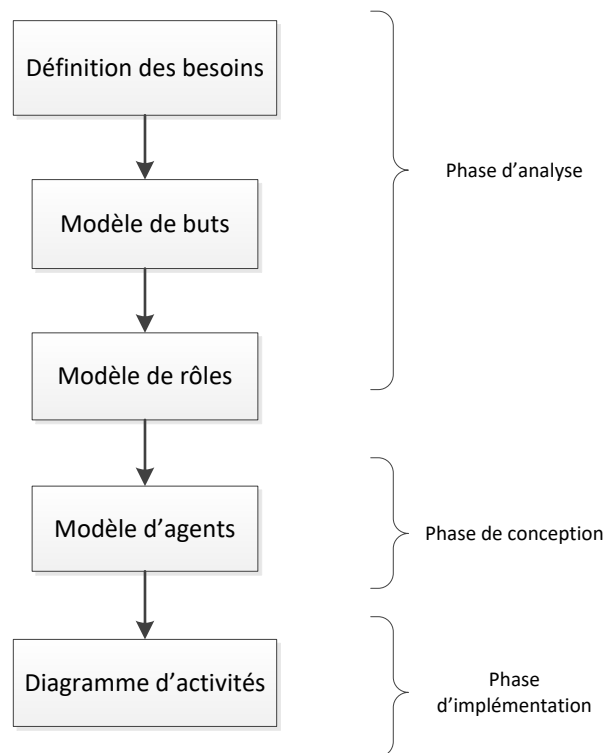


Figure 47 Méthodologie de développement d'un SMA

### 5.2.2 Application de la méthodologie sur notre SMA

Pour la mise en œuvre de notre SMA, nous adoptons l'approche méthodologique présentée dans la section précédente. Notre approche ne comporte pas la totalité des étapes méthodologiques mentionnées dans la **Figure 47**. En effet, les étapes que nous avons choisi de modéliser ont permis de concrétiser la réalisation tout en garantissant la cohérence entre les objectifs de départ et les résultats attendus du système. La **Figure 48** donne une présentation synthétisée des diagrammes que nous utiliserons pour chaque phase de conception. Dans cette section, nous détaillerons les trois phases de la méthodologie et nous présentons les principaux diagrammes.



**Figure 48** Etapes de la méthodologie de conception adoptée

#### 5.2.2.1 Phase 1 : analyse du système

L'objectif de cette étape est d'identifier le but global du système ainsi que les différents sous-buts et rôles permettant de l'atteindre. Dans notre cas, l'objectif principal est de définir les destinations à envoyer au robot. Le bon choix de la destination (la destination la plus fréquente) nous évitera d'avoir des chariots à moitié pleins, raccourcira le temps de chargement des chariots et permettra donc de maximiser le taux de conteneurs chargés automatiquement.

##### **Modèle des buts :**

Ce modèle décrit les événements auxquels le système doit fournir une réponse moyennant ses agents. Il permet également de raffiner le but global du système en un ensemble de sous-buts. Les agents peuvent envoyer des messages et planifier leurs actions afin d'atteindre leurs



buts élémentaires (sous-but). L'interaction entre les agents leur permet d'atteindre le but global. Dans le cas de notre système, le but global du système est de maximiser le nombre de produits collectés par le robot de chargement automatique. Pour les sous-buts, on peut citer : associer le container à une destination, calculer la liste des destinations les plus fréquentes, calculer le taux de chargement de l'anneau de tri et vérifier si le convoyeur est disponible. La **Figure 50** représente le diagramme de but de notre système.

### **Modèle des rôles :**

Ce modèle décrit les rôles devant être joués par les agents du système. Il décrit aussi les buts dont ces rôles sont responsables, les tâches que chaque rôle exécute pour atteindre ses objectifs. Chaque rôle est chargé d'atteindre ou contribue à atteindre des objectifs ou sous-objectifs spécifiques du système. Un rôle peut être composé de plusieurs tâches qui, prises ensemble, définissent le comportement requis de ce rôle. Le modèle contient également des relations statiques entre rôles-objectifs et rôles-capacités. Pour atteindre (« achieves ») son but, un rôle exige (« requires ») une ou plusieurs capacités (« capacity ») spécifiques. Les capacités se traduisent, généralement par des plans d'exécution.

**Figure 51** présente le modèle des rôles. Parmi les rôles qui permettent d'atteindre nos objectifs, on peut citer :

- Auto-identifier un container
- Déterminer la composition de chaque destination
- Vérifier si l'anneau de tri est plein
- Associer une destination au robot

### **5.2.2.2 Phase 2 : conception**

L'étape essentielle de cette phase liée aux SMA est l'identification des agents et de leurs interactions. Le rôle de chaque agent du système correspond à la tâche qu'il accomplit. L'élaboration du modèle d'agent aide à définir les classes d'agents et à établir la durée de vie et la multiplicité des instances de chaque classe.

### **Modèle d'agents :**

Ce modèle permet d'identifier les différents types d'agents, d'attribuer les rôles aux différents agents du système et de décrire les caractéristiques principales des agents y compris les capacités de raisonnement, les comportements, les actions et les interactions. Chaque type d'agent possède des capacités lui permettant de jouer son rôle et il peut être instancié plusieurs fois selon les exigences du système. Pour le fonctionnement de notre système, nous avons identifié cinq types d'agents :

- **Agent Container (AC) :** il existe autant d'AC que de conteneurs sur une journée postale (près de 30 000 conteneurs par jour en moyenne). Le cycle de vie d'un AC se termine dès que l'agent est envoyé au chantier manuel ou bien déposé sur un chariot.

- **Agent destination (AD) :** le nombre d'AD est égal au nombre de destinations dans le système (260 destinations). Communique principalement avec AML et AF. Ce type d'agents est chargé de gérer la popularité des destinations afin de déterminer les destinations à affecter au robot. Selon le nombre de conteneurs pour chaque destination, AD postule pour avoir une place sur le chantier manuel ou sur le robot.
- **Agent Main Loop (AML) :** c'est l'agent anneau de tri. Il aura pour rôle la gestion de la capacité de l'anneau de tri. Puisque ce dernier a une capacité limitée en nombre de containers, AML doit rejeter les conteneurs en les envoyant vers le convoyeur manuel dès que le seuil maximal est dépassé. AD communique à AML le nombre de conteneurs pour chaque destination présente sur l'anneau de tri. AML trie la liste de destinations selon le nombre de containers. Les conteneurs à rejeter sont choisis parmi les destinations les moins populaires présentes sur l'anneau de tri.
- **Agent Flexibac (AF) :** cet agent communique régulièrement avec AD. Quand AF reçoit un nouveau candidat de la part d'AD, il le met dans une liste d'attente. Dès qu'une station sur l'unité de chargement automatique (Flexibac) se libère, AF lui affecte, le candidat AD qui a le maximum de containers. AF gère aussi la capacité du convoyeur en termes de nombre de conteneurs permettant ainsi à AD d'envoyer ou non AC vers le robot selon la disponibilité du convoyeur.
- **Agent Robot (AR) :** il calcule le nombre de conteneurs sur le convoyeur robot et communique régulièrement l'état du convoyeur à AF. Cet agent est chargé aussi de mettre le container sur la bonne station sur Flexibac et de vider le chariot dès qu'il est plein.

Le modèle d'agents est présenté dans la **Figure 52**. Ce diagramme facilite la compréhension du système et donne une vision globale de la structure du SMA et de son interaction avec l'environnement.

### 5.2.2.3 Phase 3 d'implémentation

Cette étape permet de définir la façon dont le système doit être construit à partir des données issues des deux phases précédentes. Le diagramme d'activité élaboré au cours de cette phase a permis de définir ce que le système doit faire. Ce diagramme a été enrichi graduellement au cours de cette phase.

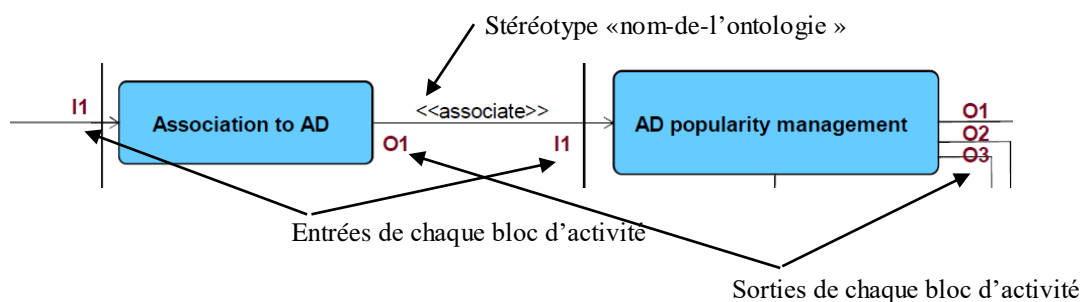
#### Diagramme d'activité :

Ce diagramme décrit la succession d'activités et d'actions qui permettront aux agents d'atteindre leurs buts. Il permet aussi de définir la séquence et les conditions d'exécution des activités.

Une activité est une spécification d'un comportement représenté par un flux d'exécution à travers le séquençage des actions. Une action est représentée par une étape unique au sein d'une activité et constitue donc l'unité fondamentale de la spécification du comportement.

Le diagramme d'activités est particulièrement approprié pour représenter le comportement du système (Niati et al., 2020). Notamment, des concepts comme « SwimLanes », défini par les couloirs introduisent des lignes parallèles verticales ou horizontales qui regroupent les étapes d'un processus par acteur. Ces concepts peuvent être utilisés pour regrouper les activités en fonction des entités, dans notre cas, les entités sont représentées par les agents du système. Les messages échangés entre les agents sont ensuite modélisés comme des liens entre les différentes activités à travers les SwimLanes. Un stéréotype « nom-de-l'ontologie » sera ajouté aux liens entre les activités afin d'organiser l'envoi de messages et de faciliter la compréhension du système. Enfin, les méthodes et les activités de chaque agent sont détaillées dans chaque SwimLane via des actions et des conditions correspondantes.

La **Figure 53** présente le diagramme d'activité de notre système. Les rectangles bleus représentent les activités. Un deuxième diagramme UML plus détaillé et qui comportera une représentation détaillée de chaque bloc a été réalisé. La taille importante du diagramme ne nous permet pas de le présenter en entier. Cependant, l'ensemble d'actions qui constituent chaque activité diagramme sera présenté dans la section 5.4.2. Afin de faciliter la compréhension du diagramme, la notation *I1..In* et *O1..On* a été ajoutée pour identifier les entrées et les sorties de chaque bloc d'activités. Cela permettra aussi de faire le lien entre les deux diagrammes (**Figure 49**).



**Figure 49** Représentation des blocs d'activités dans le diagramme d'activité

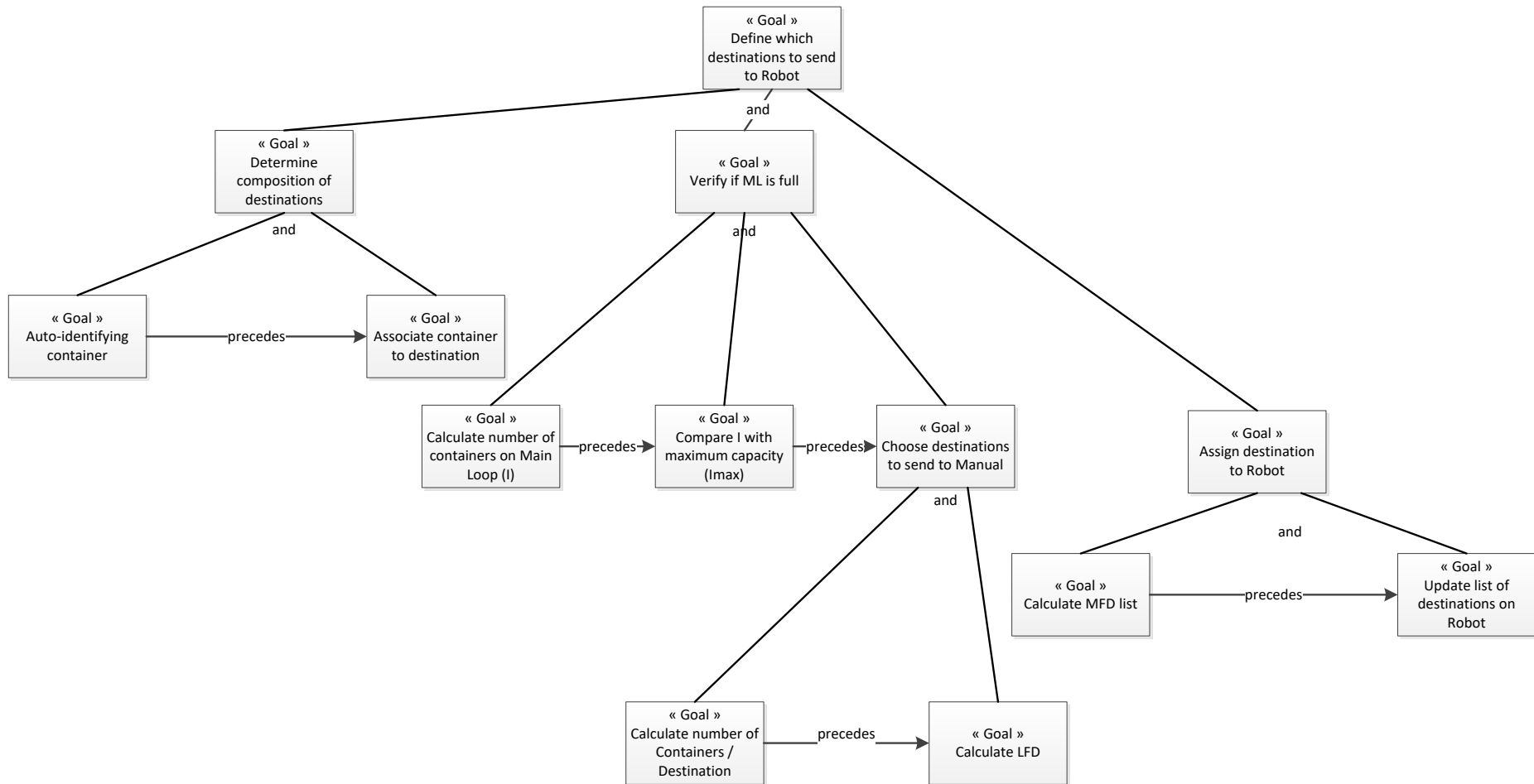
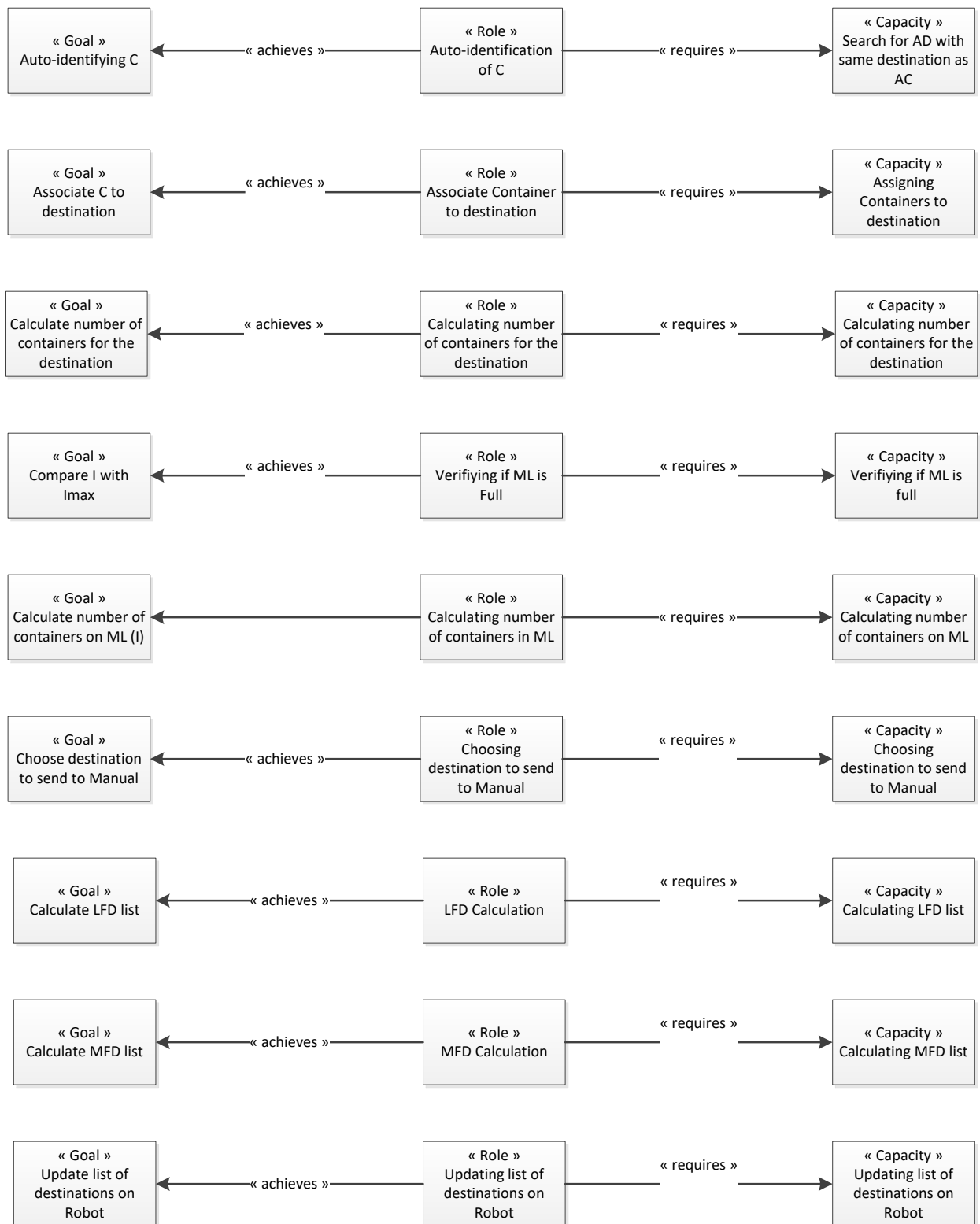


Figure 50 Modèle de Buts

## Implémentation par SMA



**Figure 51** Modèle de Rôles

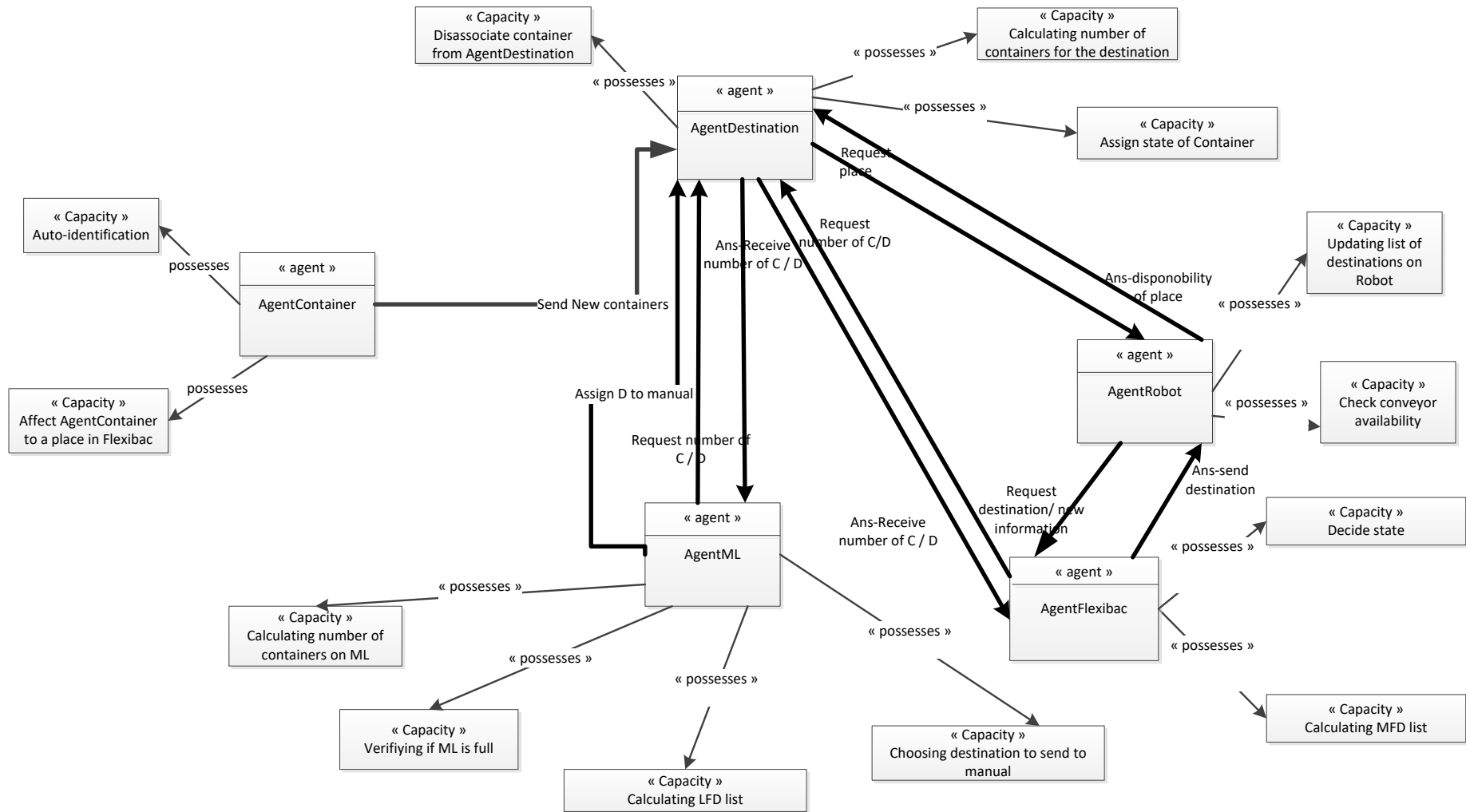


Figure 52 Modèle d'agents

# Implémentation par SMA

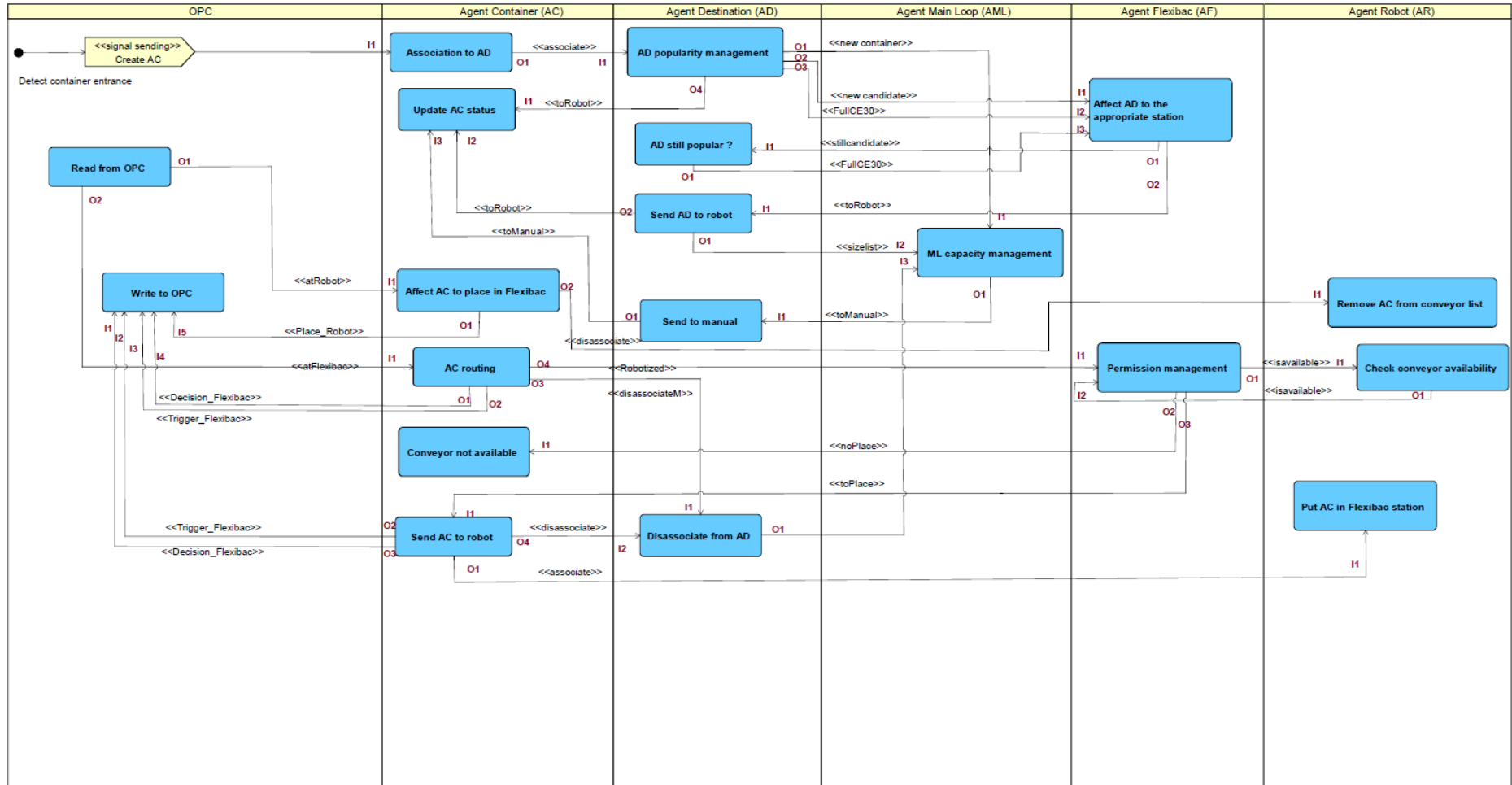


Figure 53 Diagramme d'activité

### 5.3 Outil de modélisation

L'émergence des nouveaux paradigmes de la programmation orientée objet a donné lieu à l'élaboration de plusieurs outils de développements et environnements informatiques d'exécution pour les SMA. Leur objectif principal est de guider au mieux l'utilisateur dans la phase d'implémentation pour l'aider à suivre sa méthodologie pour la programmation de ses agents étant donné qu'il est souvent difficile d'utiliser des langages standards de programmation comme C, C++, JAVA, etc. pour le faire. Ces plateformes incluent des bibliothèques et des fonctions qui simplifient la communication entre les agents, des structures organisationnelles adaptées et un outil graphique pour la visualisation des agents.

#### 5.3.1 Choix de l'outil

Plusieurs plateformes de différents types ont été développées pour la programmation multi-agents. Parmi ces plateformes on peut citer : JADE, Jack, DECAF, Zeus, AgentTool, MadKit, Brainstorm, RMIT, Jack, AgentBuilder, etc.

Au cours des dernières années, de nombreux chercheurs se sont intéressés à l'évaluation des performances des plateformes de développements des SMA en les classant selon divers critères basés principalement sur le type d'usage et la capacité de chaque plateforme. Parmi les critères les plus utilisés pour la comparaison on peut citer :

- La méthodologie de conception : certains outils sont basés sur des méthodologies bien définies, d'autres sont ouverts à plusieurs méthodologies. Le choix de l'outil peut se faire par rapport à la méthodologie utilisée.
- Documentation et facilité d'apprentissage : ce critère repose sur l'existence et la qualité de la documentation, le temps qu'il faut consacrer pour la maîtrise de l'outil et les connaissances préalables à l'utilisation.
- Souplesse de l'outil : c'est la flexibilité et la marge de liberté qu'offre l'outil pour l'utilisation de certaines fonctionnalités comme pour modifier un comportement ou étendre les informations.
- La communication : l'existence ou non de protocole de communication, des outils d'envoi de messages et d'échange d'informations et de connaissances entre les agents.
- Outil de « débogage » : il facilite la découverte et la correction des erreurs de programmation.

D'autres études comparatives des plateformes les plus utilisées sont présentées dans (Bordini et al., 2006; Garneau and Delisle, 2002; Kravari and Bassiliades, 2015; Rousset et al., 2014; Tobias and Hofmann, 2004). Ces études se focalisent sur des critères généraux comme le langage de programmation, la méthodologie supportée par la plateforme, le domaine d'utilisation et la disponibilité de la documentation ou sur des critères plus spécifiques comme l'utilisation parallèle des ressources (Rousset et al., 2014).

La **Figure 54** présente une évaluation, extraite d'une étude proposée par (Garneau and Delisle, 2002), pour la comparaison de 7 outils de programmation multi-agent parmi les plus populaires selon 6 critères que nous avons jugé essentielles pour l'implémentation de notre



SMA. Parmi ces critères : l'existence ou non d'un support pour la gestion du SMA qui permet la modification et la suppression des agents et facilite la vérification et la validation ; l'extensibilité du code qui représente une fonctionnalité très utile en permettant la modification et l'extension des codes, module ou agents prédéfinis.

Critère	Plateforme						
	Jade	DECAF	AgentBuilder	Zeus	Jack	AgentTool	Madkit
Souplesse de l'outil	3	0	1	1	3	0	3
Communication inter-agents	4	2	4	4	3	2	3
Outil de « débogage »	3	2	4	4	0	2	4
Gestion du SMA	4	0	3	3	0	1	4
Extensibilité du code	4	1	1	2	4	0	3
Documentation disponible	3	1	4	4	3	1	3
<b>Total (sur 24)</b>	<b>21</b>	<b>6</b>	<b>17</b>	<b>15</b>	<b>13</b>	<b>6</b>	<b>20</b>

**Figure 54** Évaluation de différents outils SMA (Garneau and Delisle, 2002)

Le barème d'évaluation est compris entre 0 et 4. Il peut être interprété comme suit : 0 si la plateforme ne répond pas au critère ; 1 si elle répond peu ; 2 si elle répond moyennement ; 3 si elle répond bien ; et 4 si elle répond très bien au critère.

Pour le choix de la plateforme sur laquelle notre SMA sera implémenté, nous tenons à citer Michael Wooldridge: *“Every programmer knows that what makes a “good” programming language is at least in part a kind of magic: there is an indefinable “rightness” to the best languages, that make them somehow easier, more fun, more natural, just better to use”*.

En se basant sur l'étude mentionnée ci-dessus et sur l'expérience de notre équipe, notre choix s'est porté sur JADE. Cette plateforme est parmi les plateformes les plus utilisées grâce à sa facilité d'utilisation, sa souplesse, son outil de débogage qui permet d'identifier et de corriger les erreurs de codage, le langage de communication FIPA-ACL (Agent Communication language) et finalement son ouverture aux différentes méthodologies de conception. Une présentation détaillée de la plateforme sera présentée dans la sous-section suivante.

### 5.3.2 Présentation de l'outil

JADE (**J**ava **A**gent **D**evelopment Framework) est une plateforme open-source, écrite en langage Java, développée par Télécom Italia (Bellifemine et al., 2001) pour le développement et l'exécution d'applications multi-agents distribués conformes aux spécifications de la FIPA (Foundation for Intelligent Physical Agents). Jade n'est basée sur aucune méthodologie en particulier. La plateforme comporte trois modules principaux (exigés par la norme FIPA) : l'AMS (Agent Management System) qui est l'agent qui supervise l'accès et l'utilisation de la plateforme, il est chargé de gérer l'enregistrement, l'authentification et le cycle de vie des agents. L'ACC (Agent Communication Channel) gère la communication entre les agents à

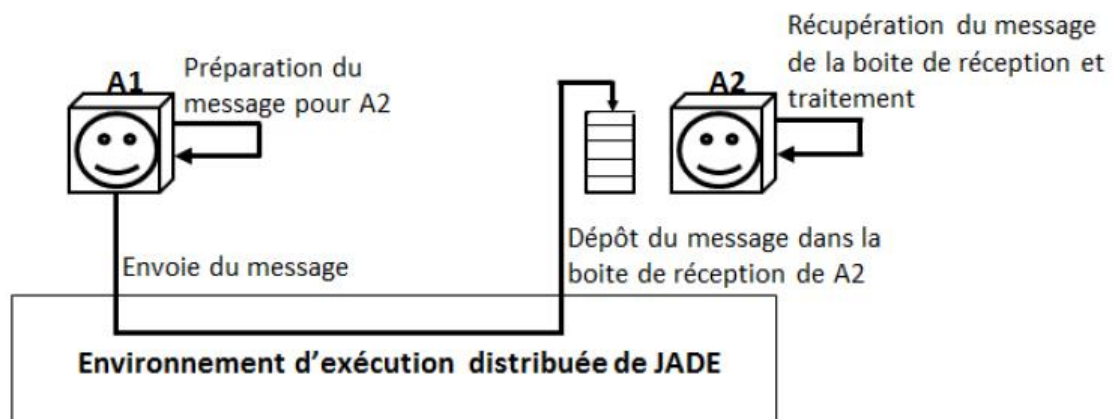
## Implémentation par SMA

l'intérieur et à l'extérieur de la plate-forme. L'ACC offre un service d'acheminement des messages fiable, ordonné et précis imposé par la FIPA97. Le DF (Directory Facilitator) fournit le service d'annuaire pages jaunes qui contiennent la description des services des agents qui y sont inscrits.

Toutes les communications entre agents s'effectuent par échange de messages, où FIPA ACL est le langage utilisé pour représenter les messages. Lorsqu'un agent JADE envoie un message, on distingue trois cas :

- L'agent récepteur se trouve dans le même conteneur que l'agent émetteur : l'objet Java représentant le message ACL est transmis au récepteur sans aucune traduction du message.
- L'agent récepteur vit sur la même plate-forme JADE, mais dans un conteneur différent : la communication se fait via la méthode RMI (Remote Method Invocation).
- Le récepteur vit sur une plate-forme d'agent différente : les communications se font par IIOP, HTTP, JMS, etc. selon la configuration de la plateforme.

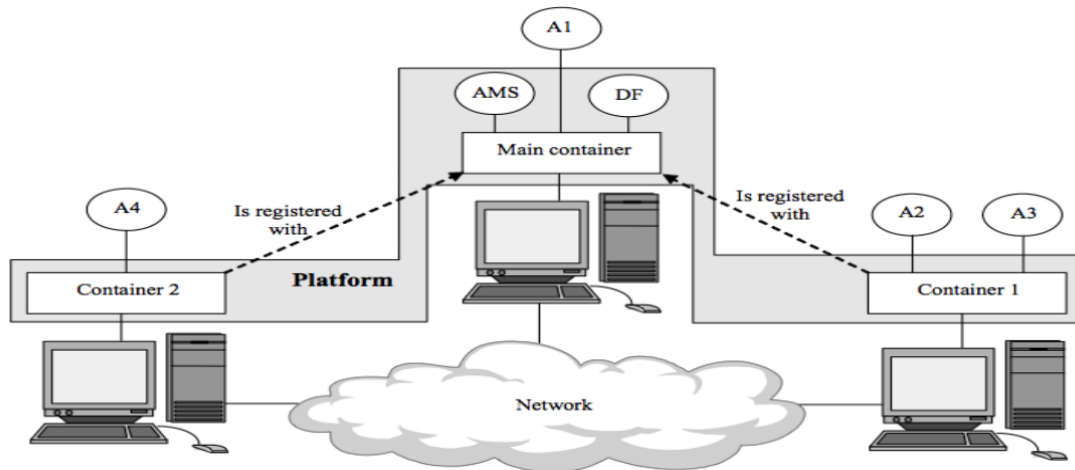
Le paradigme de communication entre agents est l'envoi de messages asynchrones où chaque agent possède d'une sorte de « boîte aux lettres » dans laquelle les messages sont stockés en attente d'exécution (**Figure 55**).



**Figure 55** Paradigme d'envoi de messages dans JADE

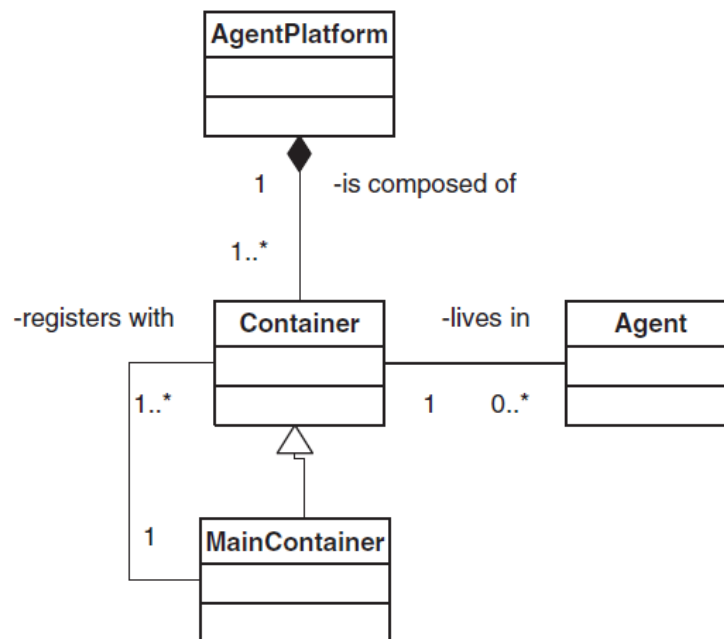
JADE comprend trois éléments essentiels à son fonctionnement :

- Un runtime Environment : c'est l'environnement où les agents peuvent « vivre ». L'activation de cet environnement est essentielle pour l'exécution des agents.
- Une bibliothèque de classes : facilite le développement des agents.
- Un ensemble d'outils graphiques : permettent la gestion, le contrôle de la plateforme et l'administration des agents actifs.



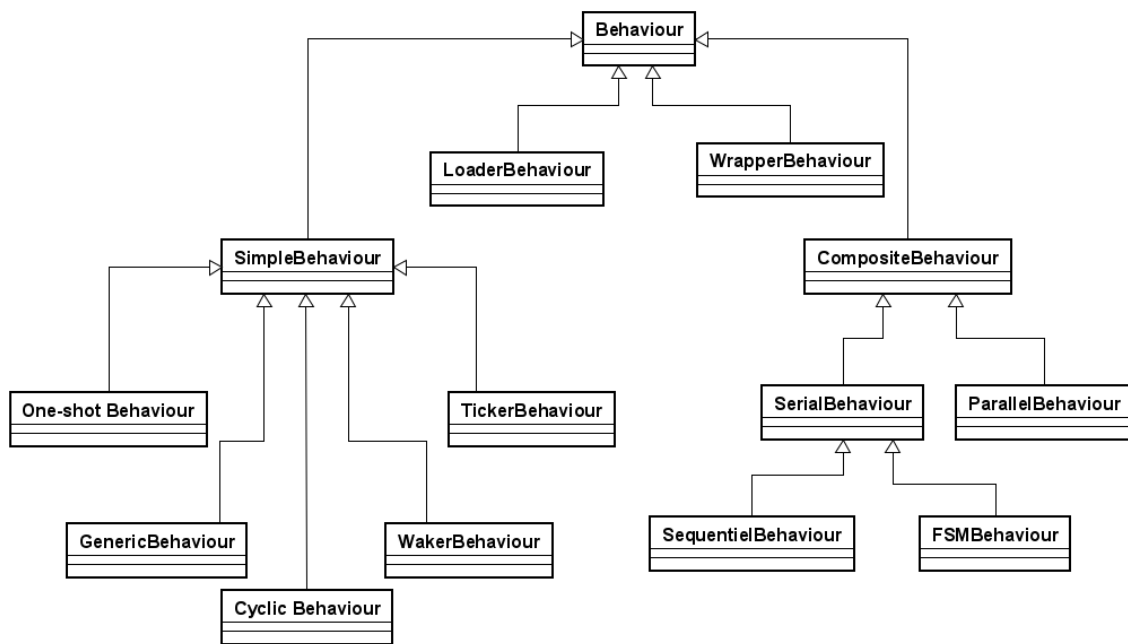
**Figure 56** Architecture multi-agents – JADE (Metawei et al., 2012)

Chaque instance de JADE est appelée « Container » (puisqu'elle contient les agents). Un unique conteneur spécial appelé « Main Container » existe pour chaque application. Les autres conteneurs, s'ils existent, sont lancés dès le démarrage de ce dernier. La **Figure 56** montre les éléments principaux d'une architecture multi-agents de la plateforme JADE. La **Figure 57** schématise la relation entre les principaux éléments dans la plateforme JADE.



**Figure 57** Éléments principaux de la plateforme multi-agents sur JADE et leurs relations. (Bellifemine et al., 2007)

Chaque agent possède un ensemble de comportements. Un comportement « behaviour » représente la tâche qu'un agent peut exécuter. JADE comprend des comportements prêts à l'emploi pour les tâches les plus courantes de la programmation d'agents. La **Figure 58** présente les hiérarchies de comportements dans JADE.



**Figure 58** Modélisation UML de la hiérarchie des comportements dans JADE

Les Behaviours simples : JADE offre cinq types de Behaviours simples :

- Cyclic Behaviour : il exécute la même opération à chaque fois qu’il est appelé.
- One-shot Behaviour : il exécute sa tâche une seule fois.
- Generic Behaviour : il exécute son opération selon l’état et la valeur d’un trigger prédéfini par le programmeur.
- Waker Behaviour : après un retard défini en argument par le programmeur, il exécute son opération et se termine juste après l’exécution.
- Ticker Behaviour : il exécute sa tâche périodiquement.

Les Behaviours composés : on distingue trois types. Ils servent à présenter des tâches complexes et sont généralement composés d’un ensemble de sous-comportements :

- Sequential Behaviour : commence par l’exécution de son premier sous-comportement. Il passe au prochain dès que le premier se termine.
- FSM Behaviour : il exécute les sous-comportements comme dans le cas d’un automate à états finis.
- Parallel Behaviour : il exécute plusieurs comportements en parallèle.

## 5.4 Implémentation du SMA

Pour notre modèle multi-agent, les comportements des agents seront régis par les règles heuristiques développées dans le chapitre précédent. Le but est de valider l’applicabilité du système de pilotage basé sur la technologie multi-agent sur le système réel.

Dans cette partie, nous allons présenter en détail les agents de notre système, leurs catégorisations, leurs interactions, leurs responsabilités et les comportements utilisés dans JADE pour les mettre en place.

### 5.4.1 Catégorisation des agents

Comme mentionné dans la section 5.1.2, pour la catégorisation des agents dans notre modèle, nous allons nous inspirer de l'approche basée sur la définition d'un agent superviseur (observateur) qui, en cas d'une perturbation dans le système, crée un plan d'action permettant d'assigner des tâches aux agents concernés pour revenir à un état d'équilibre stable.

Dans notre système, pour atteindre notre but, qui est de maximiser le nombre de conteneurs chargés par le robot, les agents AC doivent être bien orientés. Un état stable du système peut être défini comme un état dans lequel : l'anneau de tri n'est pas saturé ; le convoyeur Flexibac n'est pas saturé ; toutes les stations robot sont affectées à des destinations et le robot charge les conteneurs dans les chariots. Chaque agent cherche alors à atteindre son objectif individuel. Dans le cas d'une perturbation, un agent observateur est désigné, selon le type de perturbation, pour faciliter la convergence vers un état stable. Les agents dans notre système peuvent donc être répartis en deux catégories :

- **Agents superviseurs** : 1 agent AML, 1 agent AF, 1 agent AR et autant d'agents AD qu'il y aura de destinations dans la journée postale étudiée
- **Agents suiveurs** : autant d'agents AC qu'il y aura de destinations dans la journée postale étudiée

Dans la suite du paragraphe, nous présentons quatre cas de perturbations et précisons à chaque fois l'agent qui jouera le rôle du superviseur. Un récapitulatif sera présenté dans la **Table 21**.

**Situation 1** : l'anneau de tri est saturé. Il faut choisir les destinations (agents AD) à envoyer vers le chantier Manuel pour ne pas bloquer l'entrée de l'anneau. Dans ce cas, l'agent AML joue le rôle du superviseur. Pour revenir à un état stable, AML lance le calcul de la règle LFD dans le but de décider quelles destinations à envoyer vers le chantier manuel. Après avoir choisi les destinations, AML informe AD qui se chargera de l'exécution de l'opération et de la direction des AC vers le manuel.

**Situation 2** : le convoyeur Flexibac est saturé et bloque l'arrivée des destinations assignées au robot. Dans ce cas, la solution est de suspendre l'envoi des conteneurs vers le robot jusqu'à nouvel ordre. L'agent superviseur AF vérifie la disponibilité des places sur le convoyeur auprès d'AR avant chaque affectation d'un container vers le robot. Si le convoyeur est bloqué, AF informe AC qui se chargera de changer son statut de « robotisé » à « inconnu » afin de rester sur l'anneau de tri.

**Situation 3** : dans le cas où le traitement d'une destination sur le robot serait terminé, après avoir vidé le chariot, il faut désigner une nouvelle destination. Pour ce faire, l'agent superviseur AF doit choisir la destination qui accueille le maximum d'AC parmi les destinations présentes sur la liste d'attente. AF informe l'agent AD concerné qu'il doit être dirigé vers le

robot et change son statut, ainsi que celui des AD qui lui appartiennent vers « robotisé ». AF supprime finalement AD de sa liste d'attente.

**Situation 4 :** une destination présente sur le robot qui ne reçoit pas de conteneurs depuis longtemps. Ce cas se présente quand une destination AD, avec un nombre de conteneurs AC qui remplissent un chariot, est présente dans la liste d'attente et qu'aucune station sur le robot n'est disponible. Dans ce cas, l'agent superviseur AF doit décider quelle destination à remplacer par la nouvelle destination potentielle. Il envoie alors un message aux agents AD pour savoir s'ils sont toujours candidats. L'AD qui est resté inactif le plus longtemps est finalement remplacé par le nouveau candidat.

**Table 21** Tableau de catégorisation des agents selon le type de perturbation

Perturbation	Agent superviseur	Agent suiveur
Situation 1	AML	AD, AC
Situation 2	AF	AC
Situation 3	AF	AD, AC
Situation 4	AF	AD, AC

#### 5.4.2 Présentation des comportements des agents du système

La **Table 22** présente l'ensemble des tâches réalisées par chaque agent du système ainsi que les différents comportements (Behaviours) de JADE utilisés pour implémenter ces tâches.

Les **Figure 59** à **Figure 74** détaillent les actions à l'intérieur de chaque bloc d'activité du diagramme UML activité, précédemment mentionné dans la section 5.2.2.3.

**Table 22** Tâches réalisées par chaque agent et le type de comportement sur JADE

Agent	Tâches	Type de comportement sur JADE
AC	<ul style="list-style-type: none"> <li>• <b>Associer le container à une destination ( Figure 59)</b></li> <li>• <b>Modifier le statut du container (Figure 60)</b></li> <li>• <b>Routage du container (Figure 61)</b> <ul style="list-style-type: none"> <li>○ Orienter le container selon son statut</li> <li>○ Demander la permission à l'agent AF pour aller sur Flexibac si le statut est robotisé</li> <li>○ Se désassocier d'AD si le statut est manuel</li> </ul> </li> <li>• <b>Envoyer le container vers le robot (Figure 62)</b> <ul style="list-style-type: none"> <li>○ Détecter la place d'AC sur l'unité Flexibac</li> <li>○ Se désassocier d'AD</li> <li>○ S'associer à l'agent AR</li> </ul> </li> </ul>	CyclicBehaviour

	<ul style="list-style-type: none"> <li>• <b>Affecter le container à une place dans l'unité robotique Flexibac (Figure 63)</b> <ul style="list-style-type: none"> <li>○ Déposer AC dans la station</li> <li>○ Se désassocier d'AR</li> </ul> </li> </ul>	
AD	<ul style="list-style-type: none"> <li>• <b>Gestion de popularité d'AD (Figure 64)</b> <ul style="list-style-type: none"> <li>○ Détecter l'association d'AC</li> <li>○ Vérifier le statut d'AD</li> <li>○ Si le statut d'AD est robotisé, envoyer AC au robot</li> <li>○ Si le nombre d'AC envoyé au robot remplit un chariot, Changer le statut d'AD à Inconnu</li> <li>○ Si le statut d'AD est Inconnu, informer AML de la présence d'un nouveau container</li> <li>○ Si Le nombre d'AC avec statut Inconnu est supérieur ou égal au seuil prédéfini (alpha), informer AF d'un nouveau candidat</li> </ul> </li> <li>• <b>Vérifier si AD est toujours candidat (Figure 65)</b> <ul style="list-style-type: none"> <li>○ Vérifier le nombre d'AC dans AD</li> <li>○ Changer le statut d'AD à Inconnu si AD n'est plus candidat</li> </ul> </li> <li>• <b>Envoyer AD au robot (Figure 66)</b> <ul style="list-style-type: none"> <li>○ Changer le statut d'AD à Robotisé</li> <li>○ Changer le statut d'AC à Robotisé</li> </ul> </li> <li>• <b>Envoyer AD au chantier manuel (Figure 67)</b> <ul style="list-style-type: none"> <li>○ Modifier le statut d'AD à Manuel</li> <li>○ Modifier le statut d'AC à Manuel</li> </ul> </li> <li>• <b>Se désassocier d'AD (Figure 68)</b> <ul style="list-style-type: none"> <li>○ Supprimer AC de la liste de conteneurs d'AD</li> <li>○ Supprimer AC de la liste d'AML</li> </ul> </li> </ul>	CyclicBehaviour
		OneShotBehaviour
		CyclicBehaviour
AML	<ul style="list-style-type: none"> <li>• <b>Gérer la capacité de l'anneau de tri (Figure 69)</b> <ul style="list-style-type: none"> <li>○ Détecter l'arrivée des nouveaux containers</li> <li>○ Vérifier si l'anneau de tri est saturé</li> <li>○ Calculer la LFD</li> <li>○ Choisir l'agent AD à envoyer au manuel</li> </ul> </li> </ul>	CyclicBehaviour
		OneShotBehaviour

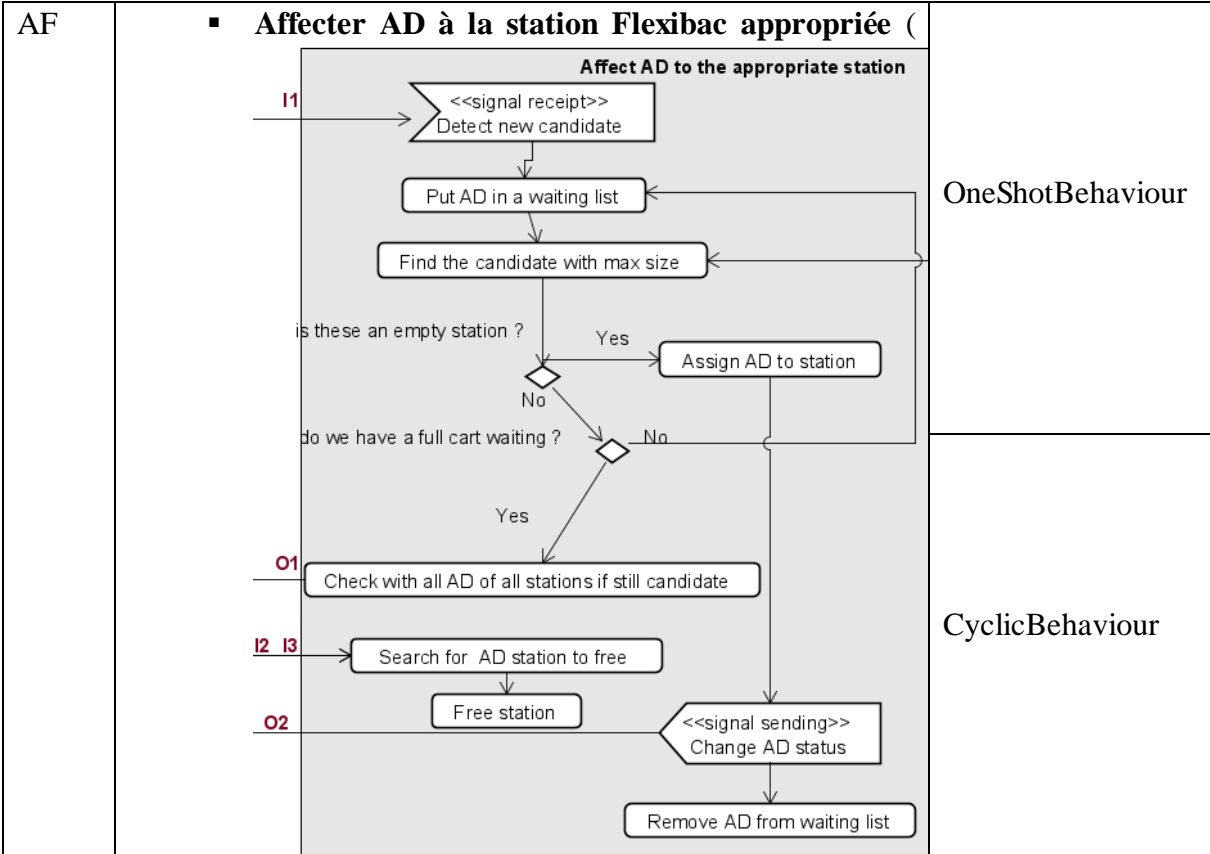
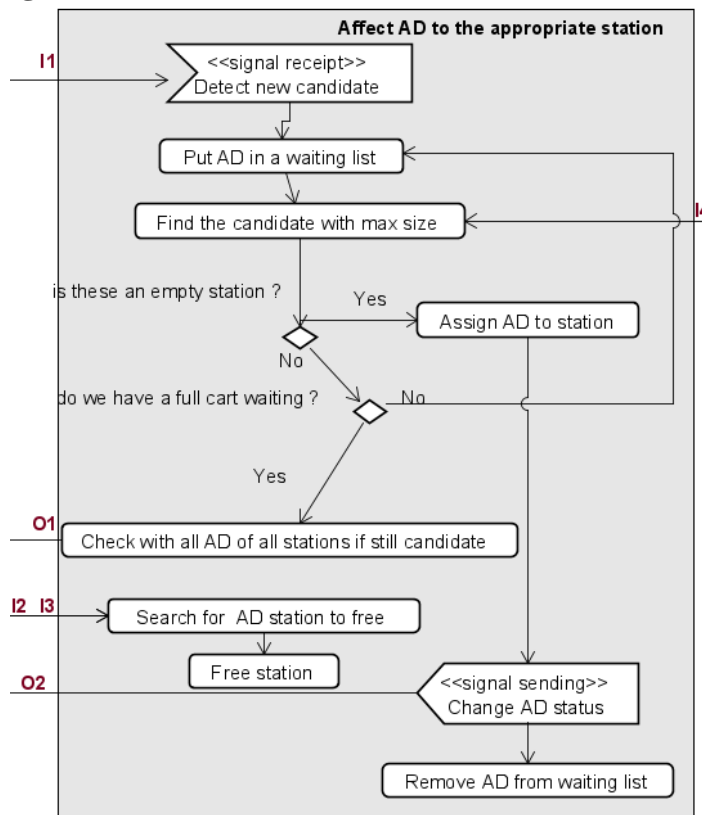


Figure 70)



- Détecter les nouveaux candidats
- Ajouter les nouveaux candidats à une liste d'attente



	<ul style="list-style-type: none"> <li>○ Trouver le meilleur candidat</li> <li>○ Affecter AD à la station Flexibac si elle est vide</li> <li>○ Vider la station si un candidat avec un chariot plein est présent dans la liste d'attente</li> <li>▪ <b>Gérer la permission (Figure 71)</b> <ul style="list-style-type: none"> <li>○ Informer AC de la disponibilité du convoyeur Flexibac</li> <li>○ Déposer AC dans la bonne station</li> <li>○ Vider la station si elle est pleine</li> <li>○ Affecter AD à la station vide</li> <li>○ Informer AC d'aller sur le robot</li> </ul> </li> </ul>	
AR	<ul style="list-style-type: none"> <li>▪ <b>Vérifier la disponibilité du convoyeur Flexibac (Figure 72) :</b> <ul style="list-style-type: none"> <li>○ Calculer le nombre de container sur le convoyeur et le comparé au seul prédéfini</li> <li>○ Informer l'agent AF de l'état du convoyeur</li> </ul> </li> <li>▪ <b>Mettre AC dans une station Flexibac (Figure 73)</b></li> <li>▪ <b>Retirer AC de la station Flexibac (Figure 74)</b></li> </ul>	CyclicBehaviour

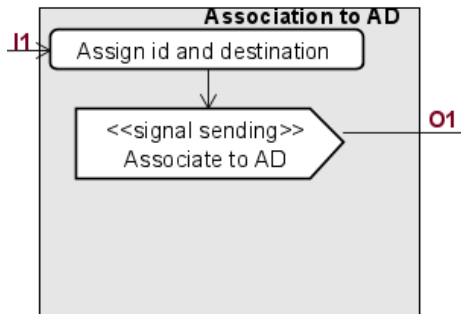


Figure 59 Association à AD

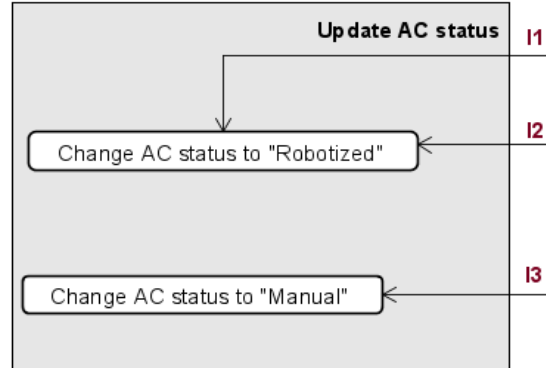


Figure 60 Modifier le statut d'AC

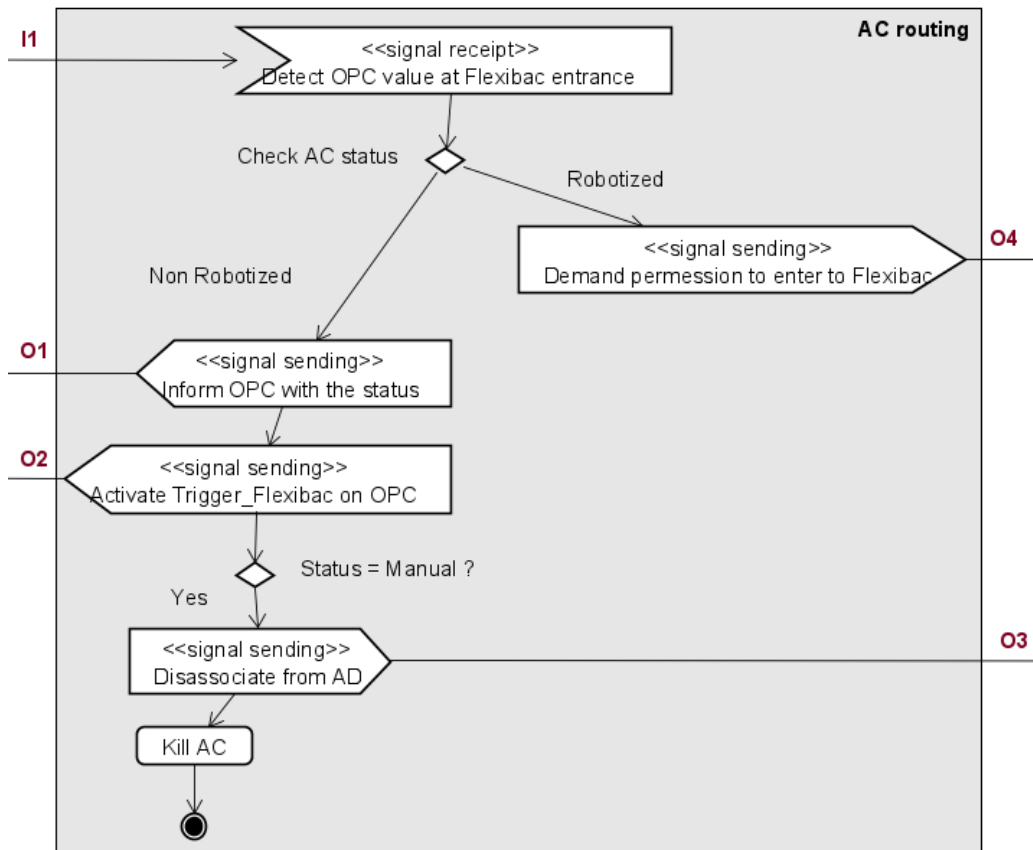


Figure 61 Routage d'AC

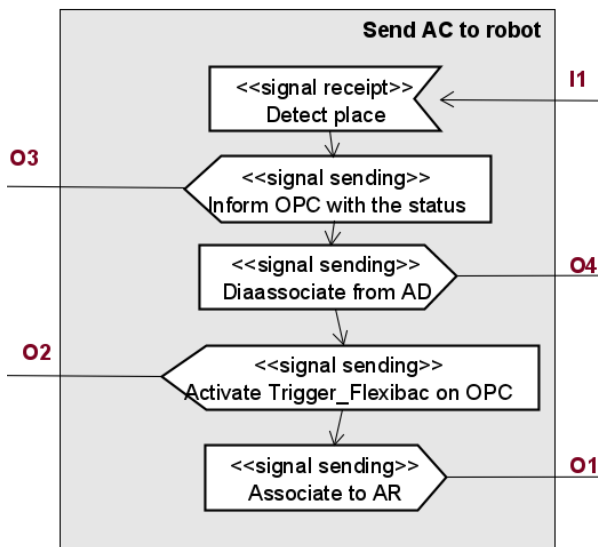


Figure 62 Envoyer AC vers le robot

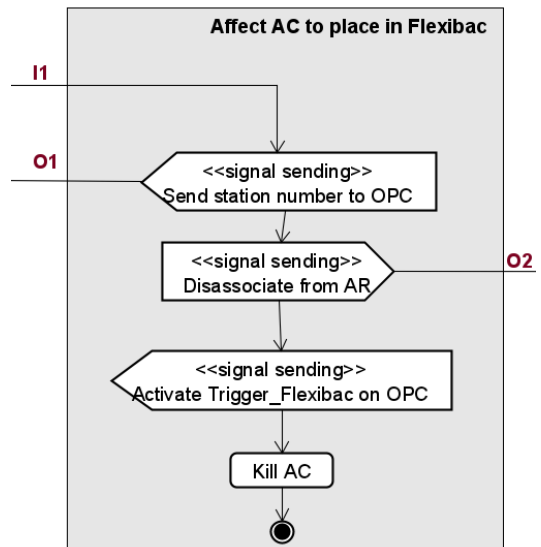


Figure 63 Affecter AC à une place sur le robot

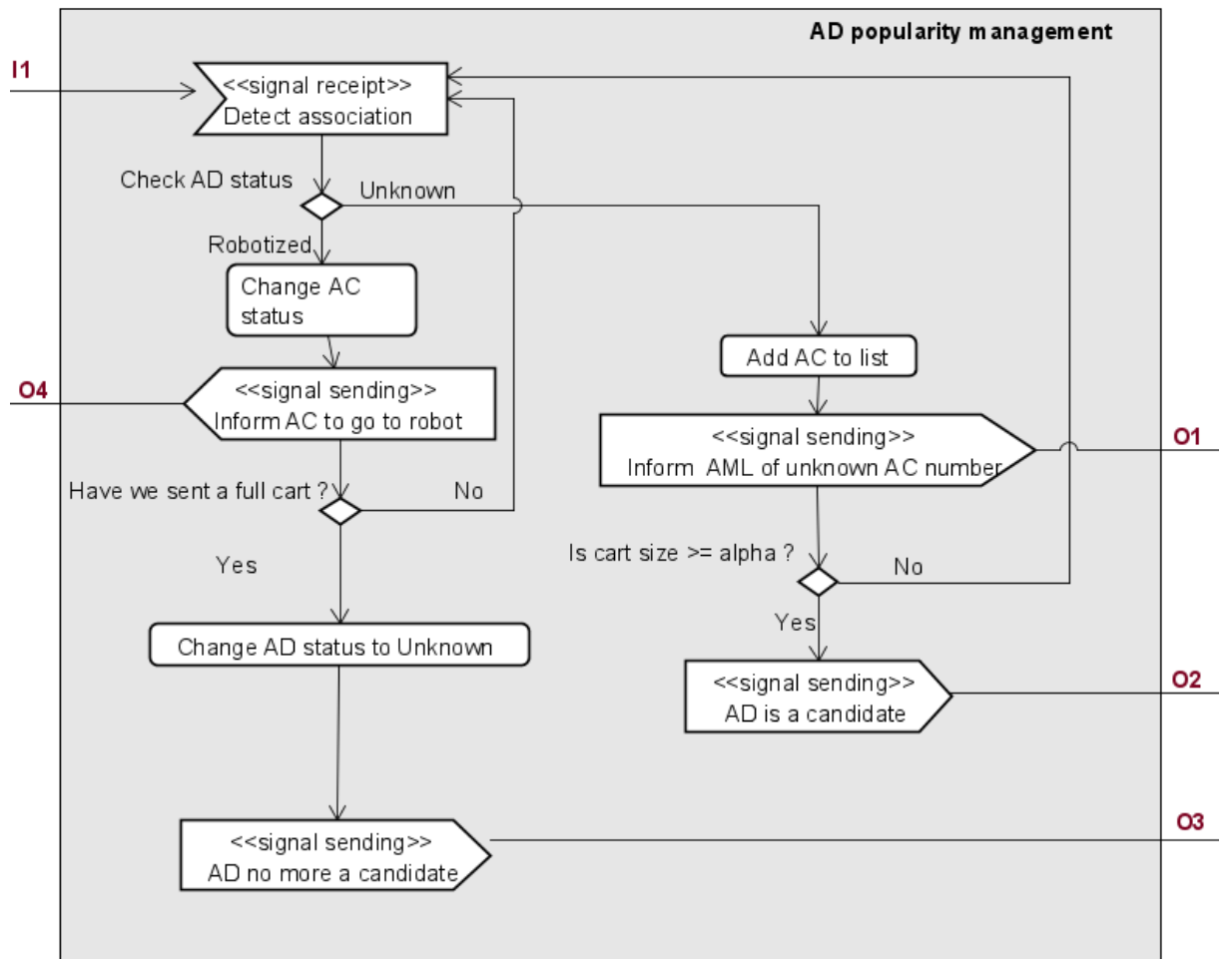


Figure 64 Gestion de popularité d'AD

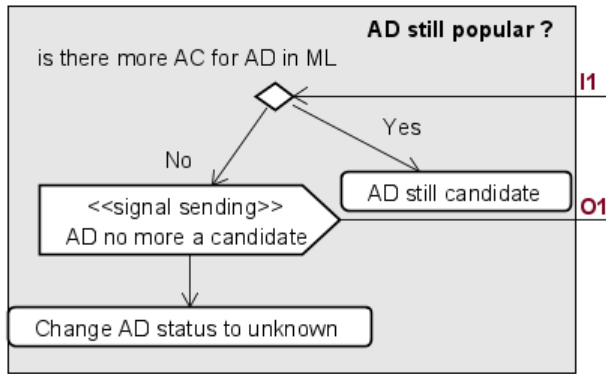


Figure 65 Vérifier si AD est toujours candidat

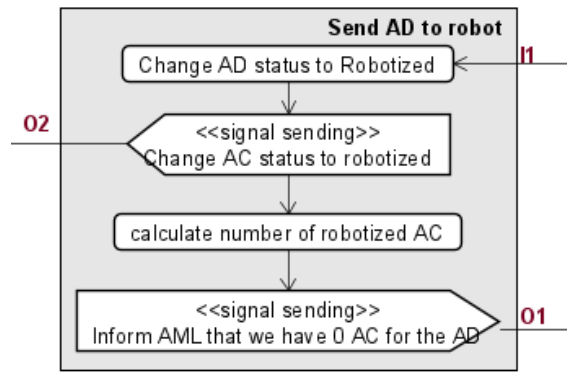


Figure 66 Envoyer AD au robot

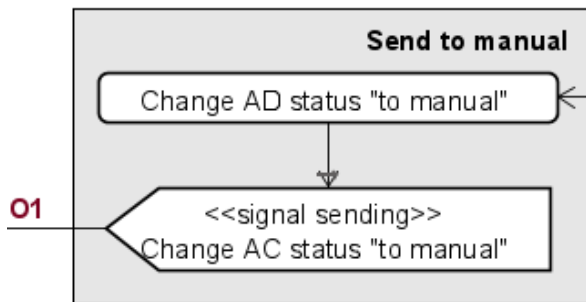


Figure 67 Envoyer AD au chantier manuel

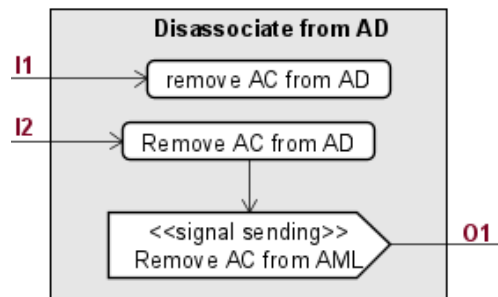


Figure 68 Se désassocier d'AD

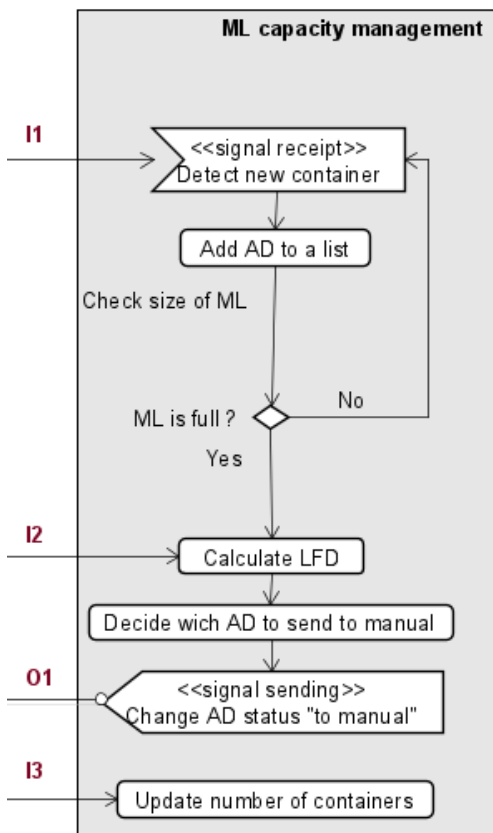


Figure 69 Gérer la capacité de l'anneau de tri

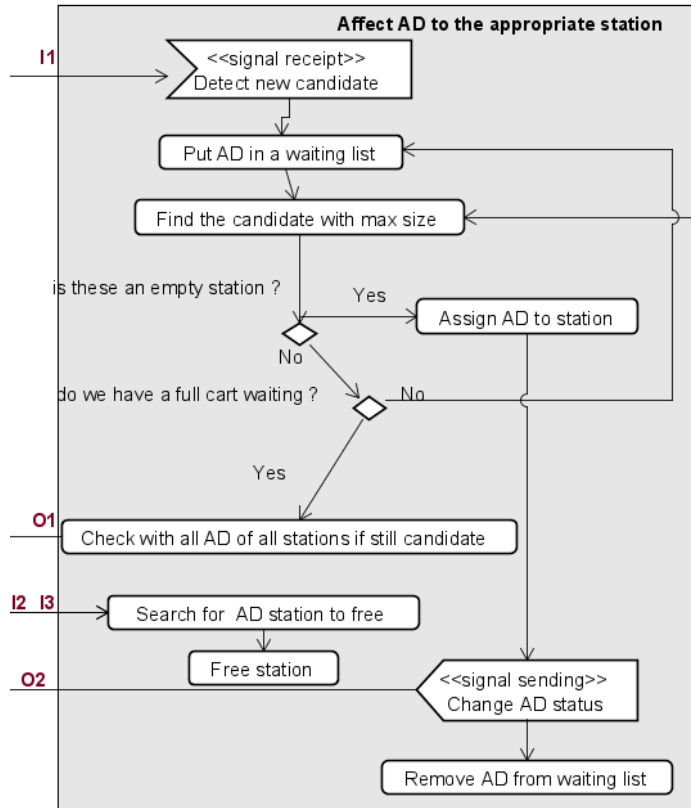
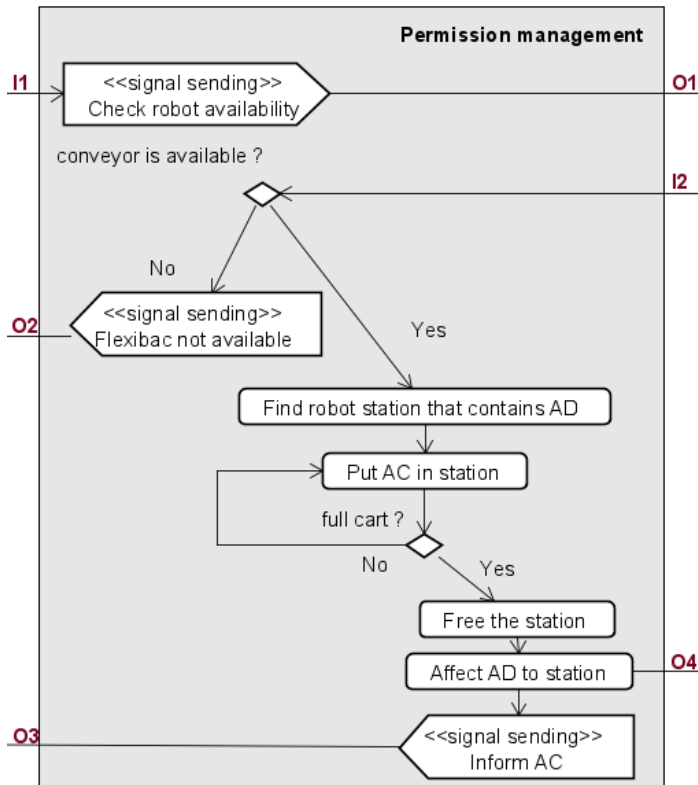
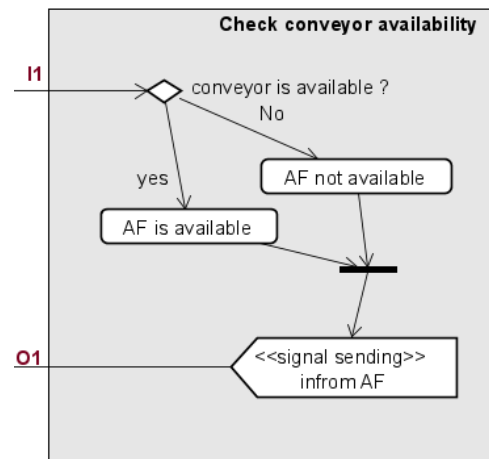


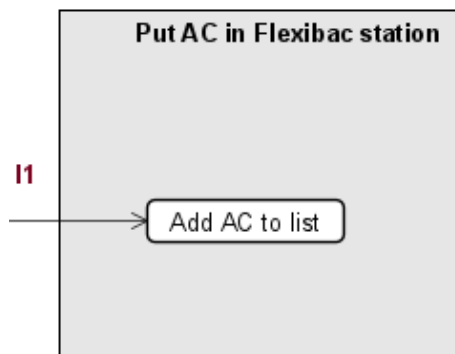
Figure 70 Affecter AD à la station Flexibac appropriée



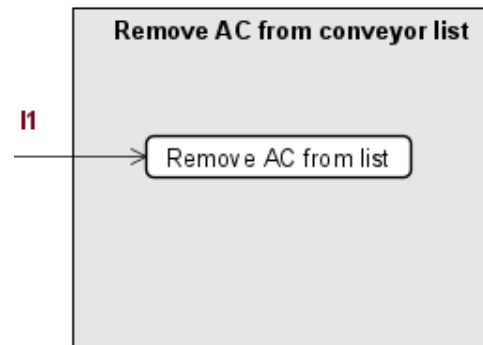
**Figure 71** Gérer la permission



**Figure 72** Vérifier la disponibilité du convoyeur Flexibac



**Figure 73** Ajouter AC à une station Flexibac



**Figure 74** Retirer AC de la station Flexibac

### 5.4.3 Présentation des interactions entre les agents

Dans notre système, les agents sont situés dans le même environnement multi-agent JADE. Ils communiquent entre eux par envoi et réception de messages. Un message est composé de :

- Un émetteur : l'agent même
- Un ou plusieurs récepteurs : le message est envoyé simultanément aux différents récepteurs
- Un contenu
- Type : informer, répondre à un appel d'offres, proposer un service, etc.
- Autres paramètres facultatifs : langue, ontologie, protocole, etc.

L'envoi et la réception des messages sont assurés par les méthodes `send ()` et `receive ()`.

Chaque agent possède sa propre boîte aux lettres dans laquelle les messages sont stockés dans l'attente d'être traités. Un agent peut recevoir plusieurs messages en même temps. Pour assurer une bonne exécution des tâches, et une compréhension correcte du contenu des messages, l'agent doit ouvrir le bon message au bon moment. Pour ce faire, une sémantique doit être mise en œuvre afin de différencier les messages reçus. Dans ce contexte, nous avons choisi de définir un ensemble d'ontologie pour l'envoi et la réception de messages. En plus des champs récepteur et contenu, le paramètre ontologie sera modifié pour chaque envoi de messages. La **Table 23** présente les interactions par envoi de messages entre tous les agents du système. Elle précise l'agent émetteur, l'agent récepteur, le type de message et l'ontologie.

**Table 23** Table représentatif des interactions entre les agents

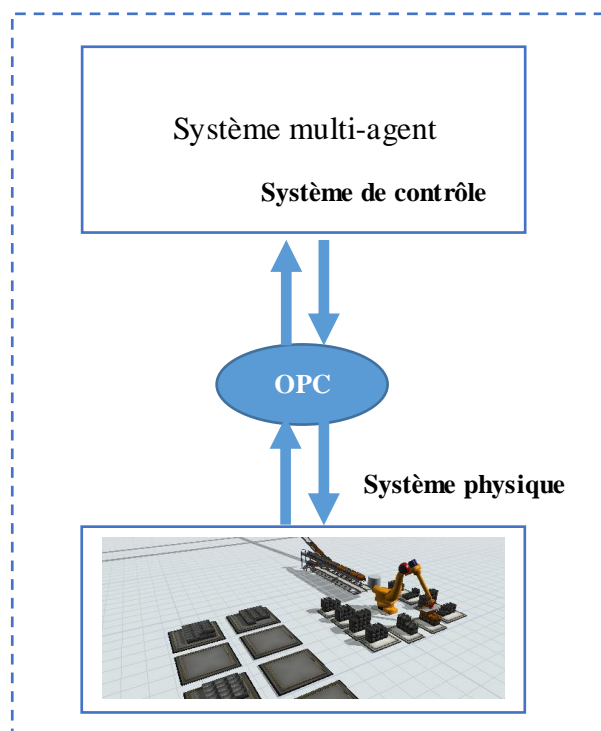
Agent émetteur	Agent récepteur	Type de message	Ontologie
AC	AD	INFORM	associate
AC	AF	INFORM	Robotized
AC	AD	INFORM	disassociateM
AC	AD	INFORM	disassociate
AC	AR	INFORM	disassociate
AC	AR	INFORM	associate
AD	AC	INFORM	toRobot
AD	AC	INFORM	toManual
AD	AML	INFORM	new container
AD	AF	INFORM	new candidate
AD	AF	INFORM	FullICE30
AD	AML	INFORM	sizelist
AD	AC	INFORM	toManual
AD	AML	INFORM	remove container
AML	AD	INFORM	toManual
AF	AD	INFORM	toRobot
AF	AD	INFORM	stillcandidate
AF	AR	INFORM	isavailable
AF	AC	INFORM	noPlace
AF	AC	INFORM	toPlace
AR	AF	INFORM	isavailable

## 5.5 Expérimentations et résultats

Dans cette section, nous allons présenter nos expérimentations réalisées pour l'évaluation de notre système de contrôle multi-agent. Nous allons utiliser les données réelles des trois journées-type déjà utilisées pour l'évaluation des modèles précédents (modèle basé sur la connaissance et modèle de règles heuristiques).

Le système multi-agent pour le contrôle de la cellule robotisée a été mis en œuvre en utilisant l'IDE Netbeans. La **Figure 77** montre une capture du main-container de JADE qui affiche les premiers agents Destination.

Nous avons choisi de construire un émulateur pour l'évaluation de notre système de contrôle multi-agent. Comme tous les modèles sont des approximations de systèmes réels (nécessité des simplifications et des réductions de modèles lors du processus de modélisation), il peut exister des différences remarquables entre les performances du système réel et celles du modèle de simulation. Ces différences conduisent souvent à un " écart de crédibilité " qu'un émulateur tente de minimiser en rapprochant le modèle de la réalité. Comblé le fossé de crédibilité n'est pas la seule raison qui nous a conduit à construire un émulateur. En effet, l'utilisation de modèle d'émulation dans des conditions réalistes et en temps réel reflète avec précision le système qui sera mis en œuvre. L'émulateur peut être utilisé pour effectuer une série illimitée de procédures de vérification, ce qui n'est pas toujours possible sur le système réel. Le but de ces tests est de garantir la performance ou la réaction du système de contrôle.

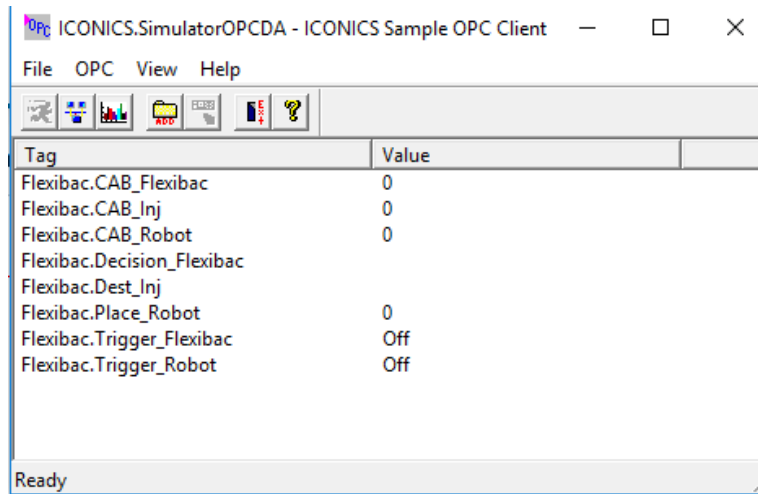


**Figure 75** Architecture d'implémentation de l'émulateur

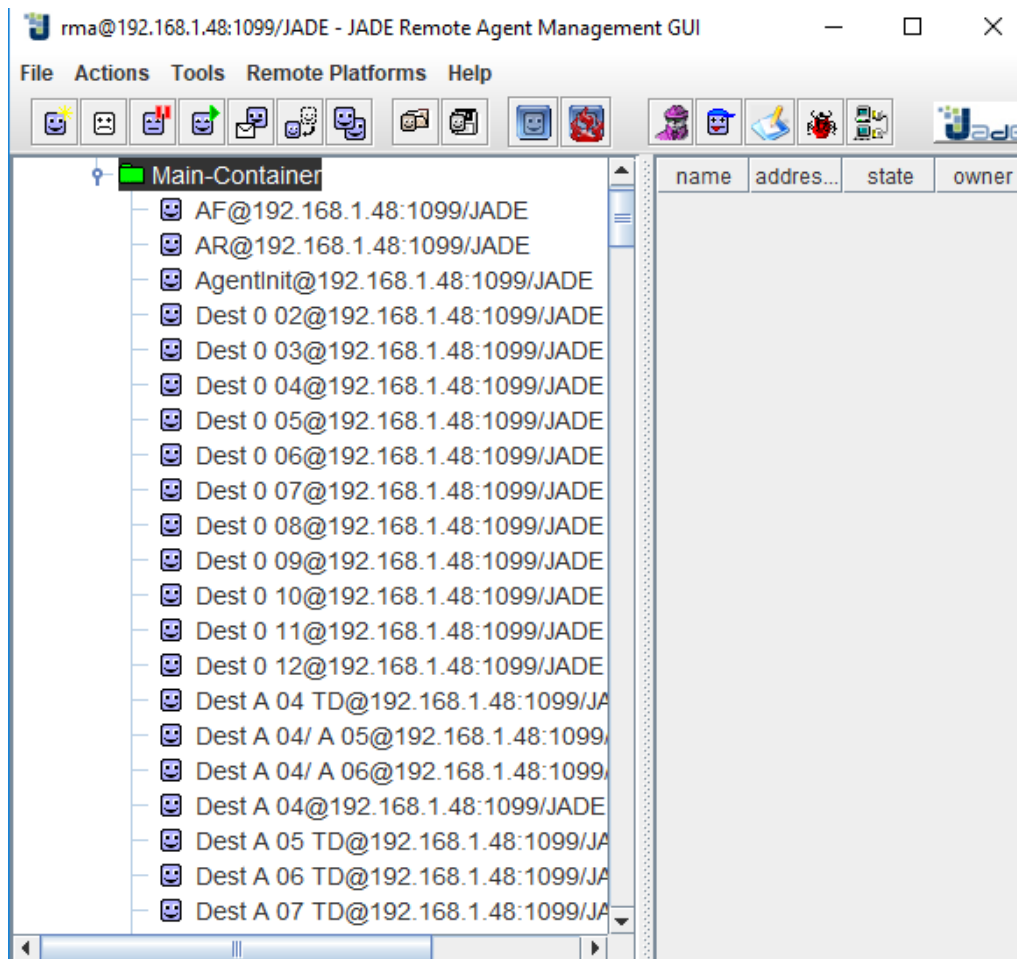
La **Figure 75** présente l'architecture de l'émulateur. Le système de contrôle multi-agent est séparé du système physique modélisé sur FlexSim. Le système physique modélisé sur FlexSim et le système de contrôle fonctionnent avec des horloges différentes et se synchronisent via une couche de communication réalisée à l'aide d'un serveur OPC DA (**Figure 76**).

## Implémentation par SMA

L'émulateur résultant a permis de tester le système de commande hors ligne et d'évaluer son efficacité pour une future implémentation sur le système réel.



**Figure 76** Serveur OPC – paramètres Flexibac



**Figure 77** Main-Container sur JADE



## Implémentation par SMA

Les tests sont lancés pour les 3 journées postales à temps réelles de l'émulation. Le taux de rafraîchissement des données du serveur OPC dans FlexSim est de 50 ms. Ce taux est le taux minimal accepté par le logiciel. Cela nous empêche d'accélérer le simulateur plus que 2 fois la vitesse réelle. En effet, en augmentant la vitesse du simulateur, ce dernier n'écrit pas toutes les données sur le serveur OPC. À la réception, le SMA n'arrivera pas à lire ces données. Vu la taille des données et la dynamique de notre système, une émulation de 24 h peut être réalisée au mieux en 12 h. À cause de l'importance du temps d'exécution, le temps de mise au point de l'algorithme a été très long (temps de débogage important). De ce fait, les résultats que nous présentons sont des résultats préliminaires. Nous n'avons pu tester qu'un seul vecteur paramètre, actuellement, pour chacune des trois journées. Le vecteur choisi : ( $\alpha = 8$ ,  $\bar{A} = 16$  et  $I_{\max} = 200$ ) est celui qui correspond au meilleur taux de conteneurs chargés automatiquement pour le modèle à base d'heuristiques.

Une comparaison des résultats des expérimentations pour les 3 méthodes : système expert, règles heuristiques et SMA est présentée dans la **Table 24**. Les résultats montrent que, sur les 3 journées étudiées, la moyenne du pourcentage de bacs pris en charge par le robot pour le SMA est 10 % meilleure que pour le SEBC considéré comme borne minimale de comparaison. Les meilleurs résultats sont obtenus pour les modèles heuristiques évalués par simulation.

**Table 24** Comparaison des résultats

	SEBC	Modèle heuristiques	SMA
15/10	15%	34%	27%
17/10	18%	32%	29%
19/10	17%	36%	25%
Moyenne	17%	34%	27%

## 5.6 Conclusion

Dans ce chapitre, nous avons présenté notre système de contrôle basé sur la technologie multi-agent. Nous avons détaillé les différentes étapes de la méthodologie de mise en place de notre SMA, de la phase d'analyse jusqu'à la phase d'implémentation. L'utilisation d'un SMA est justifiée par sa scalabilité et sa stabilité face aux variations dans des environnements non structurés. Ce système est bien adapté dans notre cas vu la taille importante du problème industriel traité, le caractère dynamique et incertain de l'arrivée des conteneurs ainsi que les contraintes complexes rencontrées.

Nous avons choisi la plateforme JADE pour la programmation du SMA. Cette plateforme est parmi les plateformes les plus utilisées grâce à sa souplesse et à son ouverture aux différentes méthodologies de conception.

Une présentation détaillée des agents est donnée. Elle a permis de mieux expliquer leurs comportements, leurs missions et leurs interactions au sein de leur environnement.

## Implémentation par SMA

Nous avons évalué notre SMA par émulation sur des données réelles. Cette étape est assez importante pour faciliter la transition entre les tests hors-ligne et l'implémentation du système de contrôle sur le système réel. Les essais ont été lancés à temps réel de l'émulation. Des résultats primaires sont obtenus pour un seul vecteur paramètre (durée d'exécution très longue dû à taille importante du problème industriel) ont montré l'efficacité du SMA à résoudre le problème d'ordonnancement PAP.

## Conclusion générale et perspectives

Malgré l'utilisation croissante des robots de PAP dans l'industrie et le besoin d'algorithmes d'ordonnancement appropriés pour optimiser leur efficacité, très peu de recherches ont été menées sur les problèmes industriels. La majorité des articles étudiés dans la littérature traitent des problèmes relativement simples provenant de systèmes expérimentaux. C'est bien dans ce contexte que s'inscrit notre thèse.

La résolution du problème industriel de type PAP d'un centre de tri postal par système multi-agent constitue la principale contribution des travaux consignés dans cette thèse.

Nous avons commencé par la présentation du fonctionnement actuel (avant la mise en place de la cellule robotisée) d'un centre de tri postal. Nous avons également donné une explication détaillée de la problématique et expliqué le besoin de La Poste d'automatiser une partie du processus du chargement de conteneurs.

Un état de l'art détaillé sur les problématiques de PAP présentes dans la littérature ainsi que les solutions proposées pour les résoudre a été effectué. Nous avons fait le choix de structurer cette étude bibliographique selon le type du problème étudié. Cette étude bibliographique a montré la complexité et la diversité des applications de PAP.

Notre intérêt s'est par la suite focalisé sur les SEBC. Trois configurations de SEBC ont été construites et ont été évaluées via le simulateur FlexSim. Cette étude nous a permis de définir la borne minimale de comparaison.

Une modélisation exacte via le solveur CPLEX a ensuite été présentée. Cette méthode a rapidement montré ses limites devant la taille importante de notre problème industriel et ses diverses contraintes. Cela, nous a conduit à investiguer l'utilisation d'heuristiques en ligne pour planifier dynamiquement l'affectation des conteneurs au robot. Une évaluation par simulation de règles heuristiques ainsi qu'une analyse de sensibilité ont été détaillées. L'analyse de sensibilité a montré l'impact de la variation des paramètres sur le rendement du système.

Finalement, nous avons proposé un système de contrôle basé sur le multi-agent pour l'affectation dynamique des destinations au robot. L'avantage de l'utilisation d'un tel système est justifié par le besoin d'un système évolutif et stable face aux variations dans des environnements incertains. Dans notre cas, il est relativement simple d'étendre le système en ajoutant d'autres agents dans le but de l'améliorer par exemple.

Les résultats de cette thèse sont en cours d'implémentation sur un centre de tri de La Poste. Les premiers essais sur le système industriel ont été réalisés en s'appuyant sur le SEBC.

Néanmoins, ces travaux nous laissent avec plusieurs perspectives à explorer :

Faire varier les paramètres  $\alpha$  et  $\bar{A}$  et  $I_{\max}$  pour trouver dynamiquement les valeurs optimales par période de la journée et par destination. En effet, l'analyse de sensibilité réalisée au cours de ces travaux nous a confirmé l'impact du choix de ces paramètres sur le taux de conteneurs chargés par le robot. Par exemple, choisir une grande valeur pour  $\alpha$  quand le trafic est important sur l'anneau de tri serait avantageux. Il permettra d'affecter plus rapidement les destinations au robot puisqu'il y a plus de chance d'avoir  $\alpha$  conteneurs appartenant à la même destination. Le choix d'une valeur plus petite pour  $\alpha$  quand le trafic est faible sur l'anneau diminuera le nombre de rejets et augmentera le taux d'utilisation du robot.

Au niveau de notre SMA, il serait intéressant d'ajouter un « agent camion ». En effet, comme les départs des camions ont une grande influence sur les heures de vidage des machines de tri et par la suite sur la date d'échéance de l'envoi des conteneurs vers le robot, l'agent camion, en communiquant avec les autres agents du système, permettra de déterminer les destinations à affecter en urgence vers le robot ou vers le manuel. L'agent camion permettra aussi d'éviter les retards des départs des camions en affectant les tâches les plus urgentes aux opérateurs humains. Le caractère flexible et scalable de notre SMA simplifie l'ajout d'agents supplémentaires.

L'utilisation des algorithmes de l'IA et de l'apprentissage automatique, en particulier l'algorithme Iterative Closest Point (ICP) serait une piste intéressante à explorer. En effet, le problème principal réside dans l'arrivée dynamique, incertaine et inconnue des conteneurs vers l'anneau de tri. Le fait de pouvoir prédire la distribution de chaque destination à tout moment de la journée et à toute période de l'année facilitera le choix de l'affectation optimale qui maximisera l'utilisation du robot. Pour l'emploi de l'algorithme ICP, on commence par calculer la distribution de chaque destination par journée postale. Ensuite, on trace un nuage de points correspondant à chaque destination. L'idée est de construire une base d'apprentissage dans laquelle sont stockés tous les nuages de points de toutes les destinations avec pour chaque destination son plan d'affectation au robot et son vecteur paramètre  $(\alpha, \bar{A}, I_{\max})$ . Pour la prédiction du comportement d'une destination, il faut commencer, dès l'arrivée des premiers conteneurs sur l'anneau, par le traçage de la distribution qui correspond à cette destination. L'ICP sera ensuite utilisée pour comparer cette distribution à celles déjà existantes dans la base de données. Plus la base de données est riche plus les résultats sont fiables. L'ICP générera itérativement des paires de points correspondants. Elle calculera ensuite l'ensemble de transformations optimales (rotations et translations) qui minimisent la distance entre chaque paire de points. L'ICP comparera par la suite le nuage de points de la destination qui nous intéresse avec tous les nuages de points des destinations stockées dans la base de données. Le but est de déterminer la destination qui ressemble le plus à cette destination. La convergence de l'algorithme ICP a été prouvée théoriquement. Après avoir déterminé la destination à qui notre destination ressemblera le plus, on appliquera le plan d'affectation ainsi que le vecteur paramètre de cette destination à la destination actuelle.

## Bibliographie

- Ahmadi, R.H., Mamer, J.W., 1999. Routing heuristics for automated pick and place machines. *European Journal of Operational Research* 117, 533–552.
- Aissani, N., Bekrar, A., Trentesaux, D., Beldjilali, B., 2012. Dynamic scheduling for multi-site companies: a decisional approach based on reinforcement multi-agent learning. *J Intell Manuf.* <https://doi.org/10.1007/s10845-011-0580-y>
- B. Wan Muhamad Raziff, W.M.A., Sam, R., Masrie, M., Janin, Z., 2018. Design and Simulation of Pick and Place System Using Solidworks Simulation, in: 2018 IEEE 5th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA). Presented at the 2018 IEEE 5th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA), pp. 1–6. <https://doi.org/10.1109/ICSIMA.2018.8688800>
- Bellifemine, F., Poggi, A., Rimassa, G., 2001. Developing multi-agent systems with a FIPA-compliant agent framework. *Softw. Pract. Exp.* 31, 103–128. [https://doi.org/10.1002/1097-024X\(200102\)31:2<103::AID-SPE358>3.0.CO;2-O](https://doi.org/10.1002/1097-024X(200102)31:2<103::AID-SPE358>3.0.CO;2-O)
- Bellifemine, F.L., Caire, G., Greenwood, D., 2007. *Developing Multi-Agent Systems with JADE*, 1 edition. ed. Wiley, Hoboken, NJ.
- Bernon, C., Glize, P., Picard, G., Glize, P., 2002. The ADELFE Methodology For an Intranet System Design, in: *Procs. Agent-Oriented Information Systems 2002*. pp. 27–28.
- Boissier, O., Gitton, S., Glize, P., 2004. *Caractéristiques des Systèmes et des Applications* pp 25.
- Bonert, M., Shu, L.H., Benhabib, B., 2000. Motion planning for multi-robot assembly systems. *Int J Comput. Integr. Manuf.* 13, 301–310. <https://doi.org/10.1080/095119200407660>
- Bordini, R.H., Braubach, L., Dastani, M., Seghrouchni, A.E.F., Gomez-Sanz, J.J., Leite, J., O'Hare, G., Pokahr, A., Ricci, A., 2006. A Survey of Programming Languages and Platforms for Multi-Agent Systems. *Informatica* 30.
- BOUCHRIT, A., 2016. *Optimal scheduling for robotized pick and place packaging systems*, Masters Thesis, Polytechnic of Milan.
- Bozma, H.I., Kalalioğlu, M.E., 2012. Multirobot coordination in pick-and-place tasks on a moving conveyor. *Robot. Comput.-Integr. Manuf.* 28, 530–538. <https://doi.org/10.1016/j.rcim.2011.12.001>
- Briand, C., Ndiaye, J., Parlouar, R., 2018. A MILP formulation for multi-robot pick-and-place scheduling. Presented at the 16th International Conference on Project Management and Scheduling, Rome, Italy.
- Briand, C., Parlouar, R., 2019. *Optimisation des opérations de pick-and-place de systèmes d'emballage robotisés*. Presented at the 16e Colloque national S-mart, Les Karelis - France.

## Bibliographie

---

- Cakir, M.C., Cavdar, K., 2006. Development of a knowledge-based expert system for solving metal cutting problems. *Mater. Des.* 27, 1027–1034. <https://doi.org/10.1016/j.matdes.2005.01.022>
- Castelli, K., Zaki, A.M.A., Giberti, H., 2019. Development of a Practical Tool for Designing Multi-Robot Systems in Pick-and-Place Applications. *Robotics* 8, 71. <https://doi.org/10.3390/robotics8030071>
- Chiddarwar, S.S., Babu, N.R., 2011. Multi-Agent System for Off-Line Coordinated Motion Planning of Multiple Industrial Robots. *Int. J. Adv. Robot. Syst.* 8, 11. <https://doi.org/10.5772/10533>
- Colombo, A.W., Schoop, R., Neubert, R., 2006. An agent-based intelligent control platform for industrial holonic manufacturing systems. *IEEE Trans. Ind. Electron.* 53, 322–337. <https://doi.org/10.1109/TIE.2005.862210>
- Coloni, A., Dorigo, M., Maniezzo, V., Varela, F., Bourguine, P., 1992. Distributed Optimization by Ant Colonies. *undefined*.
- Comba, L., Belforte, G., Gay, P., 2013. Plant layout and pick-and-place strategies for improving performances in secondary packaging plants of food products.
- Dam, K.H., Winikoff, M., 2003. Comparing Agent-Oriented Methodologies, in: Giorgini, P., Henderson-Sellers, B., Winikoff, M. (Eds.), *Fifth International Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS-2003)*, Lecture Notes in Computer Science. Springer, Melbourne, Australia, pp. 78–93. [https://doi.org/10.1007/978-3-540-25943-5\\_6](https://doi.org/10.1007/978-3-540-25943-5_6)
- Daoud, S., Chehade, H., Yalaoui, F., Amodeo, L., 2014a. Efficient metaheuristics for pick and place robotic systems optimization. *J. Intell. Manuf.* 25. <https://doi.org/10.1007/s10845-012-0668-z>
- Daoud, S., Chehade, H., Yalaoui, F., Amodeo, L., 2014b. Solving a robotic assembly line balancing problem using efficient hybrid methods. *J. Heuristics*. <https://doi.org/10.1007/s10732-014-9239-0>
- Daoud, S., Yalaoui, F., Amodeo, L., Chehade, H., Duperray, P., 2012. METHODES HYBRIDEES POUR L'EQUILIBRAGE DE CHARGES DE LIGNES D'ASSEMBLAGE ROBOTISEES.
- De Rengervé, A., Hirel, J., Quoy, M., Andry, P., Gaussier, P., 2011. On-Line Learning and Planning in a Pick-and-Place Task Demonstrated Through Body Manipulation, in: *IEEE International Conference on Development and Learning (ICDL) and Epigenetic Robotics (Epirob)*. Frankfurt am Main, Germany, pp. 1–6. <https://doi.org/10.1109/DEVLRN.2011.6037336>
- Dewi, T., Nurmaini, S., Risma, P., Oktarina, Y., Roriz, M., 2020. Inverse kinematic analysis of 4 DOF pick and place arm robot manipulator using fuzzy logic controller. *Int. J. Electr. Comput. Eng. IJECE* 10, 1376–1386. <https://doi.org/10.11591/ijece.v10i2.pp1376-1386>
- Dhaliwal, J.S., Benbasat, I., 1996. The Use and Effects of Knowledge-Based System Explanations: Theoretical Foundations and a Framework for Empirical Evaluation. *Inf. Syst. Res.* <https://doi.org/10.1287/isre.7.3.342>
- Dimeas, F., Fotiadis, F., Papageorgiou, D., Sidiropoulos, A., Doulgeri, Z., 2019. Towards Progressive Automation of Repetitive Tasks Through Physical Human-Robot Interaction, in: Ficuciello, F., Ruggiero, F., Finzi, A. (Eds.), *Human Friendly Robotics*, Springer Proceedings in Advanced Robotics. Springer International Publishing, Cham, pp. 151–163. [https://doi.org/10.1007/978-3-319-89327-3\\_12](https://doi.org/10.1007/978-3-319-89327-3_12)

- Drezner, Z., Nof, S.Y., 1984. On Optimizing Bin Picking and Insertion Plans for Assembly Robots. *IIE Trans.* 16, 262–270. <https://doi.org/10.1080/07408178408974693>
- Edan, Y., Berman, S., BOTEACH, E., MENDELSON, E., 2004. Distributed Multi-Robot Assembly/Packaging Algorithms. *Intell. Autom. Soft Comput.* 10. <https://doi.org/10.1080/10798587.2004.10642888>
- Ferber, J., 1995. *Les Systèmes Multi Agents: vers une intelligence collective*, I.I.A. Informatique intelli-gence artificielle. InterEditions. ed.
- Garcia-Najera, A., Brizuela, C.A., 2005. PCB assembly: an efficient genetic algorithm for slot assignment and component pick and place sequence problems, in: 2005 IEEE Congress on Evolutionary Computation. Presented at the 2005 IEEE Congress on Evolutionary Computation, p. 1485–1492 Vol. 2. <https://doi.org/10.1109/CEC.2005.1554865>
- Garneau, T., Delisle, S., 2002. Programmation orientée agent : évaluation comparative d’outils et environnements. Presented at the Systèmes multi-agents et systèmes complexes: Ingénierie, résolution de problèmes et simulation.
- Graham-Jones, P.J., Mellor, B.G., 1995. Expert and knowledge-based systems in failure analysis. *Eng. Fail. Anal.* 2, 137–149. [https://doi.org/10.1016/1350-6307\(95\)00009-F](https://doi.org/10.1016/1350-6307(95)00009-F)
- Grotzinger, S., Sciomachen, A., 1988. A Petri net characterization of a high speed placement machine, in: 38th Electronics Components Conference 1988., Proceedings. Presented at the 38th Electronics Components Conference 1988., Proceedings., pp. 64–68. <https://doi.org/10.1109/ECC.1988.12571>
- Guériaux, M., 2016. *Systèmes multi-agents, auto-organisation et contrôle par apprentissage constructiviste pour la modélisation et la régulation dans les systèmes coopératifs de trafic (These de doctorat)*. Lyon.
- Hanna, M., Buck, A., Smith, R., 1994. Fuzzy Petri nets to control vision system and robot behaviour under uncertain situations within an FMS cell, in: Proceedings of 1994 IEEE 3rd International Fuzzy Systems Conference. Presented at the Proceedings of 1994 IEEE 3rd International Fuzzy Systems Conference, pp. 1889–1894 vol.3. <https://doi.org/10.1109/FUZZY.1994.343571>
- Henderson-Sellers, B., Giorgini, P., 2005. *Agent-Oriented Methodologies*. Idea Group Inc (IGI).
- Hop, N., Tabucanon, M., 2001. Extended dynamic point specification approach to sequencing robot moves for PCB assembly. *Int. J. Prod. Res. - INT J PROD RES* 39, 1671–1687. <https://doi.org/10.1080/00207540010024113>
- Hou, J.-L., Wu, N., Wu, Y.-J., 2009. A job assignment model for conveyor-aided picking system. *Comput. Ind. Eng.* 56, 1254–1264. <https://doi.org/10.1016/j.cie.2008.07.017>
- Huang, Y., Chiba, R., Arai, T., Ueyama, T., Ota, J., 2015. Robust multi-robot coordination in pick-and-place tasks based on part-dispatching rules. *Robot. Auton. Syst.* 64, 70–83. <https://doi.org/10.1016/j.robot.2014.10.018>
- Huang, Y., Chiba, R., Arai, T., Ueyama, T., Ota, J., 2012. Part dispatching rule-based multi-robot coordination in pick-and-place task, in: 2012 IEEE International Conference on Robotics and Biomimetics (ROBIO). Presented at the 2012 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 1887–1892. <https://doi.org/10.1109/ROBIO.2012.6491243>
- Humbert, G., Brun, X., Pham, M.-T., Guillemot, M., Noterman, D., 2016. Development of a methodology to improve the performance of multi-robot pick place applications: From simulation to experimentation, in: 2016 IEEE International Conference on Industrial Technology (ICIT). Presented at the 2016 IEEE International Conference on Industrial Technology (ICIT), pp. 1960–1965. <https://doi.org/10.1109/ICIT.2016.7475067>

- Humbert, G., Pham, M.T., Brun, X., Guillemot, M., Noterman, D., 2015. Comparative analysis of pick place strategies for a multi-robot application, in: 2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA). Presented at the 2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA), pp. 1–8. <https://doi.org/10.1109/ETFA.2015.7301450>
- Humbert, G., Tu Pham, M., Brun, X., Guillemot, M., Noterman, D., 2017. Development of a methodology for performance analysis and synthesis of control strategies of multi-robot pick & place applications, in: Advances on Mechanics, Design Engineering and Manufacturing: Proceedings of the International Joint Conference on Mechanics, Design Engineering & Advanced Manufacturing (JCM 2016), 14-16 September, 2016, Catania, Italy, Lecture Notes in Mechanical Engineering. Springer International Publishing, Cham, pp. 639–646. [https://doi.org/10.1007/978-3-319-45781-9\\_64](https://doi.org/10.1007/978-3-319-45781-9_64)
- Johari, N.A.M., Haron, H., Jaya, A.S.M., 2007. Robotic modeling and simulation of palletizer robot using Workspace5, in: Computer Graphics, Imaging and Visualisation (CGIV 2007). Presented at the Computer Graphics, Imaging and Visualisation (CGIV 2007), pp. 217–222. <https://doi.org/10.1109/CGIV.2007.73>
- Kravari, K., Bassiliades, N., 2015. A Survey of Agent Platforms. *J. Artif. Soc. Soc. Simul.* 18, 11.
- Kumar, S.P.L., 2019. Knowledge-based expert system in manufacturing planning: state-of-the-art review. *Int. J. Prod. Res.* 57, 4766–4790. <https://doi.org/10.1080/00207543.2018.1424372>
- Laengle, T., Lueth, T.C., Rembold, U., Woern, H., 1997. Distributed control architecture for intelligent systems, in: Proceedings of 40th Midwest Symposium on Circuits and Systems. Dedicated to the Memory of Professor Mac Van Valkenburg. Presented at the Proceedings of 40th Midwest Symposium on Circuits and Systems. Dedicated to the Memory of Professor Mac Van Valkenburg, pp. 514–517 vol.1. <https://doi.org/10.1109/MWSCAS.1997.666186>
- Laengle, T., Wörn, H., 2001. Human–Robot Cooperation Using Multi-Agent-Systems. *J. Intell. Robot. Syst.* 32, 143–160. <https://doi.org/10.1023/A:1013901228979>
- Leitão, P., 2009. Agent-based distributed manufacturing control: A state-of-the-art survey. *Eng. Appl. Artif. Intell., Distributed Control of Production Systems* 22, 979–991. <https://doi.org/10.1016/j.engappai.2008.09.005>
- Liao, S.-H., 2005. Expert system methodologies and applications—a decade review from 1995 to 2004. *Expert Syst. Appl.* 28, 93–103.
- Lin, H.-I., Cheng, C.-H., Chen, W.-K., 2013. Learning a pick-and-place robot task from human demonstration, in: 2013 CACS International Automatic Control Conference (CACS). Presented at the 2013 CACS International Automatic Control Conference (CACS), pp. 312–317. <https://doi.org/10.1109/CACS.2013.6734152>
- Lueth, T.C., Laengle, T., 1994. Task description, decomposition, and allocation in a distributed autonomous multi-agent robot system, in: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94). Presented at the Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94), pp. 1516–1523 vol.3. <https://doi.org/10.1109/IROS.1994.407653>
- Maeda, Y., Ishido, N., Kikuchi, H., Arai, T., 2002. Teaching of grasp/graspless manipulation for industrial robots by human demonstration, in: IEEE/RSJ International Conference on Intelligent Robots and Systems. Presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1523–1528 vol.2. <https://doi.org/10.1109/IRDS.2002.1043971>



- MANO, J.P., GLEIZES, M.P., GLIZE, P., 2005. Collective and emergent problem solving based on multi-agent systems : principles and applications. Presented at the traitement du signal, pp. 375–391.
- Mattone, R., Adduci, L., Wolf, A., 1998. Online scheduling algorithms for improving performance of pick-and-place operations on a moving conveyor belt, in: Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146). Presented at the Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146), pp. 2099–2105 vol.3. <https://doi.org/10.1109/ROBOT.1998.680630>
- Mattone, R., Divona, M., Wolf, A., 2000. Sorting of items on a moving conveyor belt. Part 2: performance evaluation and optimization of pick-and-place operations. *Robot. Comput.-Integr. Manuf.* 16, 81–90. [https://doi.org/10.1016/S0736-5845\(99\)00041-1](https://doi.org/10.1016/S0736-5845(99)00041-1)
- Mendelson, E., Nayer, O., Berman, S., Edan, Y., 2002. Behavior-based control of multi-robot assembly/palletizing systems, in: Proceedings of the 5th Biannual World Automation Congress. Presented at the Proceedings of the 5th Biannual World Automation Congress, pp. 1–6. <https://doi.org/10.1109/WAC.2002.1049412>
- Metawei, M.A., Ghoneim, S.A., Haggag, S.M., Nassar, S.M., 2012. Load balancing in distributed multi-agent computing systems. *Ain Shams Eng. J.* 3, 237–249. <https://doi.org/10.1016/j.asej.2012.03.001>
- Mezgebe, T.T., Bril-El Haouzi, H., Demesure, G., Pannequin, R., Thomas, A., 2020. Multi-agent systems negotiation to deal with dynamic scheduling in disturbed industrial context. *J. Intell. Manuf.* 31, 1367–1382. <https://doi.org/10.1007/s10845-019-01515-7>
- Mezgebe, T.T., Demesure, G., Bril El Haouzi, H., Pannequin, R., Thomas, A., 2019. CoMM: A consensus algorithm for multi-agent based manufacturing system to deal with perturbation. *Int. J. Adv. Manuf. Technol.* 105, 3911–3926. <https://doi.org/10.1007/s00170-019-03820-0>
- Mguis, F., Zidi, K., Ghedira, K., Borne, P., 2012. Modélisation d'un Système Multi-Agent pour la résolution d'un Problème de Tournées de Véhicule dans une situation d'urgence, in: 9th International Conference on Modeling, Optimization & SIMulation. Bordeaux, France, p. 7 pages.
- Mizuuchi, I., Inaba, M., Inoue, H., 1999. Adaptive pick-and-place behaviors in a whole-body humanoid robot with an autonomous layer based on parallel sensor-motor modules. *Robot. Auton. Syst., Intelligent Autonomous Systems (IAS-5)* 28, 99–113. [https://doi.org/10.1016/S0921-8890\(99\)00009-3](https://doi.org/10.1016/S0921-8890(99)00009-3)
- Niati, A., Selma, C., Tamzalit, D., Bruneliere, H., Mebarki, N., Cardin, O., 2020. Towards a digital twin for cyber-physical production systems: a multi-paradigm modeling approach in the postal industry, in: Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings, MODELS '20. Association for Computing Machinery, New York, NY, USA, pp. 1–7. <https://doi.org/10.1145/3417990.3421438>
- O'Malley, S.A., DeLoach, S.A., 2002. Determining When to Use an Agent-Oriented Software Engineering Paradigm, in: Wooldridge, M.J., Weiß, G., Ciancarini, P. (Eds.), *Agent-Oriented Software Engineering II*, Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 188–205. [https://doi.org/10.1007/3-540-70657-7\\_13](https://doi.org/10.1007/3-540-70657-7_13)
- Ortiz, C., 1995. Développement d'un prototype de système expert en électrothermie.
- Paola Blanco Rendon, D., 2013. Modelling and simulation of a scheduling algorithm for a pick and place packaging system.

- Peng, G., Zeng, K., 2013. An Ad-Hoc Method with Genetic Algorithm for Printed Circuit Board Assembly Optimization on the Sequential Pick-and-Place Machine, in: 2013 International Conference on Parallel and Distributed Computing, Applications and Technologies. Presented at the 2013 International Conference on Parallel and Distributed Computing, Applications and Technologies, pp. 128–133. <https://doi.org/10.1109/PDCAT.2013.27>
- Premachandra, H.A.G.C., Herath, H.M.A.N., Suriyage, M.P., Thathsarana, K.M., Amarasinghe, Y.W.R., Gopura, R.A.R.C., Nanayakkara, S.A., 2020. Genetic Algorithm Based Pick and Place Sequence Optimization for a Color and Size Sorting Delta Robot, in: 2020 6th International Conference on Control, Automation and Robotics (ICCAR). Presented at the 2020 6th International Conference on Control, Automation and Robotics (ICCAR), pp. 209–213. <https://doi.org/10.1109/ICCAR49639.2020.9108045>
- Räcke, H., Sohler, C., Westermann, M., 2002. Online Scheduling for Sorting Buffers, in: Möhring, R., Raman, R. (Eds.), Algorithms — ESA 2002, Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 820–832. [https://doi.org/10.1007/3-540-45749-6\\_71](https://doi.org/10.1007/3-540-45749-6_71)
- Reichgelt, H., 1990. Different styles of agent architectures. Presented at the Proceedings of the 1st belief representation and agent architectures workshop, Galliers J .R. ( ed. ), Cambridge, pp. 29–39.
- Ridhi, B., Ahmadiéh, K., David, B., 2019. Ant Colony Optimization Algorithm for Industrial Robot Programming in a Digital Twin, in: 2019 25th International Conference on Automation and Computing (ICAC). pp. 1–5.
- Rousset, A., Herrmann, B., Lang, C., 2014. Étude comparative des plateformes parallèles pour systèmes multi-agents, in: Felber, P., Philippe, L., Riviere, E., Tisserand, A. (Eds.), ComPAS 2014 : Conférence En Parallélisme, Architecture et Systèmes. Neuchâtel, Switzerland.
- Sabas, A., Badri, M., Delisle, S., 2002. A Multidimensional Framework for the Evaluation of Multiagent System Methodologies.
- Sabas, A., Delisle, S., Badri, M., 2001. Vers une unification des méthodologies de développement des systèmes multiagents, in: Actes Des Journées Francophones Pour l'intelligence Artificielle Distribuée et Les Systèmes Multi-Agents. pp. 327–330.
- Sam, R., Arrifin, K., Buniyamin, N., 2012. Simulation of pick and place robotics system using Solidworks Softmotion, in: 2012 International Conference on System Engineering and Technology (ICSET). Presented at the 2012 International Conference on System Engineering and Technology (ICSET), pp. 1–6. <https://doi.org/10.1109/ICSEngT.2012.6339325>
- Schubert, R., 2000. Process and apparatus for introducing products into containers. US6122895A.
- Sciomachen, A., Grotzinger, S., Archetti, F., 1990. Petri net-based emulation for a highly concurrent pick-and-place machine. IEEE Trans. Robot. Autom. 6, 242–247. <https://doi.org/10.1109/70.54739>
- Selma, C., Tamzalit, D., Mebarki, N., Cardin, O., Bruggeman, L., Thiériot, D., 2018. Industry 4.0 and Service Companies: The Case of the French Postal Service, in: Borangiu, T., Trentesaux, D., Thomas, A., Cavalieri, S. (Eds.), Service Orientation in Holonic and Multi-Agent Manufacturing - Proceedings of SOHOMA 2018, Studies in Computational Intelligence. Springer, pp. 436–447. [https://doi.org/10.1007/978-3-030-03003-2\\_34](https://doi.org/10.1007/978-3-030-03003-2_34)

## Bibliographie

---

- Selma, C., Thevenin, S., Mebarki, N., Cardin, O., Tamzalit, D., Thiériot, D., Bruggeman, L., 2019. Heuristics for Robots-Humans Tasks Assignment in a Containers Loading Center, in: IFAC-PapersOnLine. Elsevier, pp. 13–18. <https://doi.org/10.1016/j.ifacol.2019.10.011>
- Shehory, O., Sturm, A., 2001. Evaluation of modeling techniques for agent-based systems, in: Proceedings of the Fifth International Conference on Autonomous Agents, AGENTS '01. Association for Computing Machinery, Montreal, Quebec, Canada, pp. 624–631. <https://doi.org/10.1145/375735.376473>
- Skoglund, A., Iliev, B., Kadmiry, B., Palm, R., 2007. Programming by Demonstration of Pick-and-Place Tasks for Industrial Manipulators using Task Primitives, in: 2007 International Symposium on Computational Intelligence in Robotics and Automation. Presented at the 2007 International Symposium on Computational Intelligence in Robotics and Automation, pp. 368–373. <https://doi.org/10.1109/CIRA.2007.382863>
- Smys, S., Ranganathan, G., 2019. Robot assisted sensing control and manufacture in automobile industry. *J ISMAC* 1, 180–187.
- Stumm, S., Pintar, D., Kuhlenkoetter, B., 2014. A novel concept for realistic simulation of industrial pick and place applications, in: *ISR/Robotik 2014; 41st International Symposium on Robotics*. Presented at the *ISR/Robotik 2014; 41st International Symposium on Robotics*, pp. 1–7.
- Su, C.-T., Fu, H.-P., 1998. A simulated annealing heuristic for robotics assembly using the dynamic pick-and-place model. *Prod. Plan. Control* 9. <https://doi.org/10.1080/095372898233560>
- Sun, Z., Lee, L.H., Chew, E.P., Tan, K.C., 2012. MicroPort: A general simulation platform for seaport container terminals. *Adv. Eng. Inform., Network and Supply Chain System Integration for Mass Customization and Sustainable Behavior* 26, 80–89. <https://doi.org/10.1016/j.aei.2011.08.010>
- Švaco, M., Šekoranja, B., Jerbic, B., 2012. Industrial Robotic System with Adaptive Control, in: *Complex Adaptive Systems*. <https://doi.org/10.1016/j.procs.2012.09.048>
- Švaco, M., Šekoranja, B., Jerbić, B., 2011a. Multiagent robotic collaborative framework, in: Katalinić, B. (Ed.), *Annals of DAAAM for 2011 & Proceedings of the 22nd International DAAAM Symposium*. Presented at the *Annals of DAAAM for 2011 and 22nd International DAAAM Symposium “Intelligent Manufacturing and Automation: Power of Knowledge and Creativity,”* DAAAM International, Vienna, pp. 0731–0732.
- Švaco, M., Šekoranja, B., Jerbić, B., 2011b. Autonomous Planning Framework for Distributed Multiagent Robotic Systems, in: Camarinha-Matos, L.M. (Ed.), *Technological Innovation for Sustainability, IFIP Advances in Information and Communication Technology*. Springer, Berlin, Heidelberg, pp. 147–154. [https://doi.org/10.1007/978-3-642-19170-1\\_16](https://doi.org/10.1007/978-3-642-19170-1_16)
- Švaco, M., Šekoranja, B., Jerbić, B., 2011c. A multiagent framework for industrial robotic applications. *Procedia Comput. Sci.* 6 291.
- Tobias, R., Hofmann, C., 2004. Evaluation of free Java-libraries for social-scientific agent based simulation. *J. Artif. Soc. Soc. Simul.* 7, 1–6.
- Trentesaux, D., 2002. *Pilotage hétérarchique des systèmes de production (HDR)*. Université de Valenciennes et du Hainaut-Cambresis.
- Turban, E., Aronson, J.E., 2001. *Decision support systems and intelligent systems, sixth Edition (6th ed)*. ed. Hong Kong: Prentice International Hall.

## Bibliographie

---

- Wilhelm, W.E., Choudhry, N.D., Damodaran, P., 2007. A model to optimize placement operations on dual-head placement machines. *Discrete Optim.* 4, 232–256. <https://doi.org/10.1016/j.disopt.2006.11.006>
- Wong, C.-C., Feng, H.-M., Lai, Y.-C., Yu, C.-J., 2019. Ant Colony Optimization and image model-based robot manipulator system for pick-and-place tasks. *J. Intell. Fuzzy Syst.* 36, 1083–1098.
- Wooldridge, M., 2002. Intelligent agents, in: *An Introduction To MultiAgent Systems*. Gerhard Weiss, Cambridge, Massachusetts London, England.
- Wooldridge, M.J., Jennings, N.R., 1995. Intelligent Agents: Theory and Practice. *Knowl. Eng. Rev.* 10, 115–152.
- Xiaogang Wang, Yuewei Bai, Chilan Cai, Xiaoguang Yan, 2010. A production rule-based knowledge system for software quality evaluation, in: *2010 2nd International Conference on Computer Engineering and Technology*. Presented at the 2010 2nd International Conference on Computer Engineering and Technology, pp. V6-208-V6-211. <https://doi.org/10.1109/ICCET.2010.5486303>
- Yasuda, G., 2012. Implementation of Distributed Control Architecture for Multiple Robot Systems Using Petri Nets, in: *Chapters*. IntechOpen.
- Yasuda, G., 1999. An object-oriented multitasking control environment for multirobot system programming and execution with 3D graphic simulation. *Int. J. Prod. Econ.* 60–61, 241–250. [https://doi.org/10.1016/S0925-5273\(98\)00181-9](https://doi.org/10.1016/S0925-5273(98)00181-9)
- Yu, C., Liu, X., Qiao, F., Xie, F., 2017. Multi-robot coordination for high-speed pick-and-place tasks, in: *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. Presented at the 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 1743–1750. <https://doi.org/10.1109/ROBIO.2017.8324670>
- Zhou, M., Leu, M.C., 1991. Petri net modeling of a flexible assembly station for printed circuit boards, in: *Proceedings. 1991 IEEE International Conference on Robotics and Automation*. Presented at the 1991 IEEE International Conference on Robotics and Automation, IEEE Comput. Soc. Press, Sacramento, CA, USA, pp. 2530–2535. <https://doi.org/10.1109/ROBOT.1991.132007>
- Zhu, G.-Y., Chen, Z.-J., 2009. A Differential Evolution Optimization Approach to Solve the Pick-and-Placing Problem, in: *2009 Fifth International Conference on Natural Computation*. Presented at the 2009 Fifth International Conference on Natural Computation, pp. 66–70. <https://doi.org/10.1109/ICNC.2009.153>

---

**Titre : Résolution d'un problème industriel de type pick & place d'un centre de tri postal**

**Mots clés :** Pick & Place, Modèle multi-agent, Robotique, Heuristique, Simulation, Recherche opérationnelle

**Résumé :** Les problèmes de type Pick & Place (PAP) sont très étudiés dans la littérature, mais, à notre connaissance, très peu de travaux étudient les systèmes de PAP dans un contexte industriel. L'objectif de cette thèse est la résolution d'un problème industriel de type PAP au sein d'un centre de tri postal, où des bacs remplis de courriers arrivent dynamiquement et dans un ordre inconnu, et où des opérateurs placent ces bacs dans des chariots en fonction de leur destination. Compte tenu de la diversité importante des destinations journalières, un équilibre doit être trouvé en temps réel entre les flux traités par les humains et par le robot.

Ce problème a été résolu en quatre phases. En premier lieu, des modèles à base de connaissance ont été proposés à partir de l'expérience de l'opérateur

logistique. Le résultat de l'application de ces modèles sur une simulation du système réel est considéré comme une borne inférieure de la performance du système. En second lieu, un modèle mathématique du système a été établi, le relâchement de plusieurs contraintes permettant de traiter le problème comme un problème d'ordonnancement classique. Les résultats de cet ordonnancement, inapplicables sur le terrain, nous ont conduits à investiguer l'utilisation d'heuristiques en ligne. Une troisième étape a été de proposer un modèle heuristique, à base de règles dynamiques, évaluée en simulation. Enfin, un modèle multi-agents, intégrant ces règles de décisions, a été développé afin de valider l'applicabilité d'un tel système de pilotage sur le système réel.

---

**Title : Solving an industrial pick & place problem of a postal sorting center**

**Keywords :** Pick & Place, Multi-agent model, Robotics, Heuristics, Simulation, Operational research

**Abstract :** Pick & Place (PAP) problems are widely studied in the literature, but, to the best of our knowledge, very few works study PAP systems in an industrial context. The objective of this thesis is to solve an industrial PAP problem in a postal sorting center, where containers filled with mail arrive dynamically and in an unknown order, and where operators place these containers in carts according to their destination. Given the large diversity of daily destinations, a balance must be found in real time between the flows processed by humans and by the robot.

This problem was solved in four phases. First, knowledge-based models were proposed based on the experience of the logistics operator.

The result of applying these models on a simulation of the real system is considered as a lower bound of the system performance. Second, a mathematical model of the system has been established, with the release of several constraints allowing to treat the problem as a classical scheduling problem. The results of this scheduling, which cannot be applied in the real system, led us to investigate the use of online heuristics. A third step was to propose a heuristic model, based on dynamic rules, evaluated in simulation. Finally, a multi-agent model, integrating these decision rules, was developed to validate the applicability of such a control system on the real system.