



**HAL**  
open science

# Optimisation de la performance globale de systèmes intelligents dans un environnement multi-agents.

Santiago Montoya Zapata

► **To cite this version:**

Santiago Montoya Zapata. Optimisation de la performance globale de systèmes intelligents dans un environnement multi-agents.. Intelligence artificielle [cs.AI]. HESAM Université, 2024. Français. NNT : 2024HESAE025 . tel-04894760

**HAL Id: tel-04894760**

**<https://pastel.hal.science/tel-04894760v1>**

Submitted on 17 Jan 2025

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

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

**ECOLE DOCTORALE SCIENCES ET METIERS DE L'INGENIEUR**  
**Laboratoire d'Ingénierie des Systèmes Physiques et Numériques (LISPEN)**  
**Campus de Lille**

# THÈSE

*présentée par :* **Santiago MONTOYA ZAPATA**

*soutenue le :* **20 juin 2024**

*pour obtenir le grade de :* **Docteur d'HESAM Université**

*préparée à :* **École Nationale Supérieure d'Arts et Métiers**

*Spécialité :* **Génie industriel**

## **Optimisation de la performance globale de systèmes intelligents dans un environnement multi-agents**

**THÈSE dirigée par :**  
**M. Olivier GIBARU**

**et co-encadrée par :**  
**Mme. Nathalie KLEMENT**  
**M. Cristovão SILVA**

**Mme. Gülgün ALPAN**  
**M. Olivier CARDIN**  
**M. Gauthier PICARD**  
**M. Emmanuel ADAM**  
**M. Olivier GIBARU**  
**Mme. Nathalie KLEMENT**  
**M. Cristovão SILVA**  
**Mme. Meriem LAFOU**  
**M. Antoine LEBLANC**

PU, GSCOP, Grenoble INP  
MCF, HDR, LS2N, Université de Nantes  
DR, HDR, ONERA  
MCF, LAMIH, UPHF  
PU, LISPEN, ENSAM  
MCF, LISPEN, ENSAM  
PU, CEMMPRE, Université de Coimbra  
D, Référent IA, Groupe Renault  
Expert IA&Data, Groupe Renault

**Jury**  
Présidente  
Rapporteur  
Rapporteur  
Examinateur  
Examinateur  
Examinatrice  
Examinateur  
Examinatrice  
Invité

**T  
H  
È  
S  
E**



*A mi familia ...*





# Remerciements

Je tiens à exprimer ma gratitude aux personnes et aux institutions qui ont contribué à la réalisation de cette thèse.

Je remercie chaleureusement les membres de mon jury de thèse. Olivier Cardin, pour ses orientations et les pistes de recherche qu’il m’a proposées lors de nos discussions en conférence, et Gauthier Picard, pour la relecture attentive de mon travail et ses retours constructifs. Je suis également reconnaissant envers Gülgün Alpan d’avoir accepté de présider mon jury de thèse et pour son suivi tout au long de ma thèse à travers les comités réguliers et les journées de doctorants chez Renault. Enfin, merci à Emmanuel Adam pour sa participation et ses retours.

Je souhaite exprimer ma profonde reconnaissance à mes encadrants. Nathalie Klement, merci pour ton suivi rigoureux et ton investissement lors de la révision de tous les livrables de cette thèse. Cristovão Silva, ta vision scientifique, toujours très pragmatique, et ton expérience ont été des atouts précieux. Olivier Gibaru, merci d’avoir fait le lien avec le laboratoire, permettant ainsi le démarrage et le déroulement de cette thèse, ainsi que pour tes remarques constructives. Meriem Lafou, merci pour ton engagement, ta disponibilité, ton soutien “psychologique” et ta vision scientifique tout en gardant ta casquette industrielle. Lionel Roucoules, merci pour tes remarques et tes orientations tout au long de ce processus de thèse.

Je tiens également à remercier Antoine Leblanc, nos discussions existentielles informelles et ta passion pour la transformation de l’industrie, sans oublier tes analogies au foot, ont été des moments précieux. Ton accompagnement tout au long de mon développement professionnel, depuis mon stage de fin d’études jusqu’à ma défense de thèse, a été inestimable.

Au sein de Renault, je remercie chaleureusement Gabriel, Jules, Vanny, et Hachem pour leur aide, ainsi que tous mes autres collègues du département Montage, IFM et Métaverse Industriel pour leur soutien et leur collaboration. Gabriel, merci de m’avoir fait confiance et de m’avoir ouvert les portes de Renault. Jules, tes avis et le temps que tu as consacré à imaginer ce que l’on pourrait faire avec cette thèse ont beaucoup compté pour moi.

Je suis également reconnaissant envers Paulo Leitao et les autres membres du laboratoire Cedri

## REMERCIEMENTS

---

pour leur générosité et leur disposition pour aider et partager leurs connaissances lors de nos échanges. Un grand merci à Julia, Christophe, Kéno, Pietro et tous les autres membres du laboratoire IFL du KIT pour m'avoir accueilli et pour les échanges enrichissants pendant mon séjour. Merci également à Hugo, Marguerite, Lei, Guillaume, Arthur, Dorian, Shuhan, Marielle, Maxime, Eddy, Vincent, Anthony et aux reste de doctorants et ex-doctorants du LISPEN pour leur camaraderie et leur soutien.

Je remercie aussi mes amis et mes proches pour leur soutien indéfectible. Merci Dani et Jarit pour être toujours présentes de près ou de loin. Merci aux compiégnais Lisa, Camille et Samuel de m'avoir accueilli en tant que cinquième coloc.

Un énorme merci à ma famille en Europe. Bea, Daniel, Felipe, Naty, Juan Pablo, Silvis, Mary, Fernando, Nacho, Javi, Victoria, Marie Sara, Thaís, Ian, ces 8 années ont été incroyables et c'est en grande partie grâce à vous tous. Notre dernière retrouvaille ainsi que vos appels et messages avant la soutenance ont été la meilleure motivation ("Veenga, a disfrutar...") et la meilleure façon de boucler et de fêter cette thèse.

Merci ma chérie chérie pour ton accompagnement au quotidien et pour me permettre de continuer à vivre des aventures même pendant cette période de thèse. Merci pour ton soutien, ta patience et pour avoir passé des heures à critiquer mes introductions.

Enfin, un immense merci à ma famille. Muchas gracias a mis padres por su entrega absoluta hacia nosotros, por su amor y por todos los esfuerzos realizados para brindarnos lo mejor. Merci Diego pour ton aide inconditionnelle et pour avoir été mon encadrant de thèse additionnel.

Merci à tous pour votre aide précieuse et votre soutien constant.

# Table des matières

<b>Remerciements</b>	<b>5</b>
<b>Liste des tableaux</b>	<b>12</b>
<b>Table des figures</b>	<b>15</b>
<b>Acronymes</b>	<b>17</b>
<b>1 Introduction générale</b>	<b>19</b>
1.1 Contexte . . . . .	20
1.2 Cas d'étude industriel . . . . .	21
1.3 Description du processus logistique . . . . .	23
1.3.1 Types de flux logistiques . . . . .	23
1.3.2 Approvisionnement des zones kitting . . . . .	24
1.3.3 Zone kitting . . . . .	26
1.3.4 Livraison des kits à la ligne d'assemblage . . . . .	27
1.3.5 Systèmes d'information . . . . .	28
1.3.6 Besoins industriels : problèmes les plus récurrents . . . . .	33
1.4 Problématique . . . . .	34
1.5 Objectif de la thèse . . . . .	35
1.6 Périmètre de l'étude : zone kitting DOS . . . . .	36
1.6.1 Approvisionnement des pièces à la zone . . . . .	37
1.6.2 Préparations des kits . . . . .	38
1.7 Modèle à événements discrets du système d'étude . . . . .	39
1.7.1 Première approche de modélisation : modélisation détaillée . . . . .	41
1.7.2 Deuxième approche de modélisation : modélisation simplifiée . . . . .	42
1.7.3 Implémentation sur Simul8 . . . . .	44
1.7.4 Exemple d'une première application . . . . .	46
1.7.5 Conclusion . . . . .	49
1.8 Plan du manuscrit . . . . .	49

<b>2</b>	<b>Modélisation multi-agents : état de l'art et proposition</b>	<b>51</b>
2.1	Introduction . . . . .	52
2.2	Cadre théorique et revue de la littérature . . . . .	52
2.2.1	L'intelligence artificielle distribuée . . . . .	52
2.2.2	Agents . . . . .	54
2.2.3	Apprentissage des systèmes multi-agents . . . . .	56
2.2.4	Outils et plateformes de développement et de simulation de systèmes multi-agents	57
2.2.5	Applications des systèmes multi-agents . . . . .	58
2.3	méthode de modélisation multi-agents proposée . . . . .	62
2.3.1	Architecture du système multi-agents . . . . .	62
2.3.2	Protocole d'interaction global entre les agents opérateurs et les agents kits . . .	63
2.3.3	Modèle de comportement de l'agent kit . . . . .	64
2.3.4	Méthodologie d'estimation des coûts . . . . .	67
2.3.5	Méthodologie de collecte de données et d'identification de l'état du système . .	67
2.4	Conclusion . . . . .	71
<b>3</b>	<b>Système multi-agents pour réagir aux perturbations d'un processus de kitting d'une ligne d'assemblage automobile : application</b>	<b>73</b>
3.1	Introduction . . . . .	74
3.2	Illustration de la méthode proposée à travers une étude de cas ponctuel . . . . .	74
3.2.1	Description de l'étude de cas ponctuel . . . . .	74
3.2.2	Résultats expérimentaux . . . . .	77
3.3	Analyse de la distribution de temps entre deux alertes consécutives . . . . .	79
3.4	Méthodologie d'évaluation de la méthode multi-agents proposée sur une période de temps . . . . .	82
3.4.1	Identification de l'état du système en utilisant des informations directement issues du système réel . . . . .	82
3.4.2	Identification des kits candidats et calculs des critères de sélection du meilleur endroit de récupération de la pièce manquante . . . . .	84
3.4.3	Méthode de sélection du meilleur kit candidat . . . . .	86
3.5	Définition des KPIs globaux . . . . .	87
3.5.1	Alertes comme événements indépendants . . . . .	87
3.5.2	Alertes comme événements non indépendants . . . . .	88
3.6	Validation expérimentale . . . . .	91
3.6.1	Sélection de l'endroit de récupération de la pièce manquante . . . . .	92
3.6.2	KPIs résultants en traitant les alertes comme événements indépendants . . . .	95
3.6.3	KPIs résultants en traitant les alertes comme événements non indépendants . .	95

## TABLE DES MATIÈRES

---

3.7	Conclusion . . . . .	98
<b>4</b>	<b>Modèle de détection anticipée de la perturbation d'une pièce manquante dans un kit</b>	<b>101</b>
4.1	Introduction . . . . .	102
4.2	Objectif de l'agrégation des données . . . . .	102
4.3	Collecte des données . . . . .	104
4.3.1	Spotfire pour les données logistiques . . . . .	104
4.3.2	Le serveur Synersight des AGVs . . . . .	105
4.3.3	Les tableaux manufacturing sur GCP . . . . .	106
4.4	Stratégie d'agrégation . . . . .	107
4.4.1	Agrégation flux rang 0 et rang 1 . . . . .	107
4.4.2	Composition et passage des véhicules dans la zone de montage des portes . . .	109
4.4.3	Extraction des alertes à partir des appels DWS . . . . .	111
4.5	Création du <i>data set</i> . . . . .	112
4.5.1	Liaison code identifiant véhicule -support physique . . . . .	112
4.5.2	Liaison code véhicule - alerte . . . . .	113
4.5.3	Pré-traitement et transformation des données . . . . .	114
4.5.4	Limitations et difficultés . . . . .	115
4.6	Environnement et outils d'entraînement . . . . .	116
4.7	Analyse exploratoire des données . . . . .	117
4.7.1	Analyse du comportement des variables . . . . .	117
4.7.2	Distribution des classes . . . . .	119
4.8	Métriques d'évaluation des modèles . . . . .	123
4.9	Stratégie de modélisation . . . . .	124
4.9.1	Modélisation avant équilibrage des données . . . . .	124
4.9.2	Modélisation après équilibrage des données . . . . .	128
4.10	Conclusion . . . . .	136
<b>5</b>	<b>Application de la méthodologie proposée sur un nouveau cas d'étude</b>	<b>139</b>
5.1	Introduction . . . . .	140
5.2	Description du problème . . . . .	141
5.3	Méthodologie . . . . .	142
5.3.1	Architecture multi-agents du système d'étude . . . . .	142
5.3.2	Constitution du <i>data set</i> . . . . .	143
5.3.3	Modèle testé et métriques d'évaluation utilisées . . . . .	145
5.4	Résultats expérimentaux . . . . .	145
5.5	Conclusion . . . . .	148

## TABLE DES MATIÈRES

---

<b>Conclusion générale et perspectives</b>	<b>151</b>
<b>Bibliographie</b>	<b>157</b>
<b>Annexes</b>	<b>165</b>

# Liste des tableaux

1.1	Taille des emballages . . . . .	27
1.2	Nombre de références gérées par zone kitting . . . . .	27
1.3	Description des niveaux de décision et de planification [Klement et al., 2017] . . . . .	41
1.4	Données de production changement de cadence . . . . .	47
1.5	Durée des opérations d’une boucle des AGVs tracteurs . . . . .	47
2.1	Liste d’autres publications d’applications des systèmes multi-agents analysées . . . . .	61
2.2	Extrait des données fournies par le système d’information des alertes . . . . .	69
2.3	Extrait des données fournies par le système d’information des AGVs . . . . .	70
2.4	Extrait des données fournies par le MES . . . . .	70
3.1	Données fournies par le système d’information d’alerte . . . . .	75
3.2	AGVs en cours entre la zone de préparation des kits et la ligne d’assemblage . . . . .	76
3.3	Séquence et diversité des ordres de fabrication . . . . .	76
3.4	Résultat de la phase de négociation : identification des kits candidats . . . . .	77
3.5	Résultats : temps de déplacement de l’opérateur pour chaque kit candidat . . . . .	78
3.6	Nombre d’alertes par station de travail sur la période d’analyse . . . . .	81
3.7	Etat du système à l’instant t où le kit N°2220082 a été reconnu défectueux . . . . .	83
3.8	Liste des kits candidats pour le kit défectueux N°2220082 . . . . .	84
3.9	Coûts de récupération de la pièce manquante pour le kit défectueux N°2220082 . . . . .	85
3.10	Temps d’arrêt de production et impact sur les kits candidats pour le kit défectueux N°2220082 . . . . .	86
3.11	Extrait du DataFrame résultant suite à la prise de décision d’endroit de récupération de la pièce manquante (zone de kitting ou au meilleur kit candidat) . . . . .	93
3.12	KPIs résultants pour la méthode multi-agents proposée et la méthodologie actuelle en traitant les alertes comme des événements indépendants . . . . .	95
3.13	KPIs résultants pour la méthode multi-agents proposée avec un seul opérateur et la méthodologie actuelle en traitant les alertes comme des événements non indépendants . . . . .	96
3.14	KPIs résultants pour la méthode multi-agents proposée avec deux opérateurs et la méthodologie actuelle en traitant les alertes comme des événements non indépendants . . . . .	97



## LISTE DES TABLEAUX

---

4.1	Métriques d'évaluation pour les modèles de la première configuration . . . . .	126
4.2	Métriques d'évaluation pour les modèles de la deuxième configuration . . . . .	128
4.3	Métriques d'évaluation pour les modèles entraînés avec un <i>data set</i> équilibré en utilisant SMOTE . . . . .	131
4.4	Métriques d'évaluation pour les modèles entraînés suite à un sous-échantillonnage du <i>data set</i> (702 non alertes et 351 alertes) . . . . .	131
4.5	Métriques d'évaluation pour les modèles entraînés avec un <i>data set</i> équilibré en utilisant une combinaison de sous et sur-échantillonnage (1404 non alertes et 1404 alertes) . . . .	131
4.6	Métriques d'évaluation pour le modèle forêt aléatoire entraîné avec un <i>data set</i> composé de 351 données alertes et un nombre variable de non alertes . . . . .	132
4.7	Métriques d'évaluation pour le modèle forêt aléatoire entraîné avec un <i>data set</i> équilibré composé d'un nombre variable de données . . . . .	132
5.1	Exemple d'état du système à l'instant t . . . . .	144
5.2	Illustration du <i>data set</i> constitué pour le système d'étude . . . . .	145
5.3	Métriques d'évaluation pour les modèles entraînés avec une fenêtre d'analyse de 60 minutes, sous différentes valeurs de h . . . . .	147

# Table des figures

1.1	Schéma de production usine carrosserie-montage. . . . .	22
1.2	Schéma processus de montage. . . . .	22
1.3	Types de flux des pièces. . . . .	24
1.4	Types de solutions logistiques. . . . .	26
1.5	Cartographie atelier montage. . . . .	28
1.6	Systèmes d'information . . . . .	29
1.7	Cheminement d'une palette sur Reflex . . . . .	30
1.8	Historique d'événements Tag S18905 . . . . .	31
1.9	Liste de pièces pour le kit du véhicule 1330018 transmise par PSFV . . . . .	32
1.10	Écran disponible pour les opérateurs du système DWS . . . . .	32
1.11	Situation actuelle . . . . .	35
1.12	Fonctionnement souhaité du système d'approvisionnement de pièces à la ligne de montage de véhicules [Montoya Zapata et al., 2024b] . . . . .	36
1.13	Cible de la thèse . . . . .	37
1.14	Approvisionnement des pièces à la zone de kitting. . . . .	38
1.15	Préparation des kits. . . . .	39
1.16	Première approche de modélisation : modélisation détaillée . . . . .	43
1.17	Deuxième approche de modélisation : modélisation simplifiée . . . . .	43
1.18	Modèle à événements discrets. . . . .	45
1.19	Modèle à événements discrets modifié. . . . .	48
2.1	Cycle de contrôle d'un agent réactif [Amblard et al., 2021] . . . . .	55
2.2	Architecture du Procedural Reasoning System [Georgeff et Lansky, 1987], base des architectures BDI [Amblard et al., 2021] . . . . .	56
2.3	Utilisation de plateformes pour a) l'ensemble des applications et b) pour les applications à haute maturité [Müller et Fischer, 2014] . . . . .	58
2.4	Secteurs d'activité applications de systèmes multi-agents pour a) l'ensemble des applications et b) pour les applications à haute maturité [Müller et Fischer, 2014] . . . . .	59

TABLE DES FIGURES

---

2.5	Architecture proposée du système multi-agents du processus d’approvisionnement de pièces dans une ligne d’assemblage automobile [Montoya Zapata et al., 2023]. . . . .	63
2.6	Protocole d’interaction global entre les opérateurs et les agents kits [Montoya Zapata et al., 2024a] . . . . .	65
2.7	Modèle de comportement des agents kit [Montoya Zapata et al., 2024a] . . . . .	66
2.8	Méthodologie d’estimation des coûts a) de $K_D$ à chaque kit candidat et b) de $K_D$ à $K_E$	68
2.9	Disposition physique d’un scénario ponctuel . . . . .	71
3.1	Disposition physique du cas d’étude . . . . .	75
3.2	Meilleurs kits candidats proposés à l’opérateur de la ligne d’assemblage . . . . .	78
3.3	Ecart de temps entre deux alertes consécutives pour chaque poste de travail . . . . .	80
3.4	Ecart de temps entre deux alertes consécutives pour l’atelier d’étude . . . . .	82
3.5	Disposition physique de l’étude de cas . . . . .	92
3.6	Distribution des choix de l’endroit de récupération de la pièce manquante sur l’ensemble des scénarios . . . . .	94
3.7	Impact individuel et cumulé d’une alerte sur l’alerte suivante après la sélection du meilleur kit candidat ou de la récupération de la pièce de la zone de kitting . . . . .	96
4.1	Agrégation des données d’approvisionnement des kits . . . . .	103
4.2	Mise en relation des données d’approvisionnement et des données après la livraison des kits . . . . .	104
4.3	Distribution des tags AGV dans la zone de montage des portes . . . . .	107
4.4	Architecture de l’agrégation flux rang 0 et flux rang 1 . . . . .	108
4.5	Architecture de l’agrégation composition et passage des véhicules dans la zone de montage des portes . . . . .	110
4.6	Architecture d’extraction des alertes DWS . . . . .	111
4.7	Jeu de données final . . . . .	116
4.8	Nombre d’occurrences d’alertes en fonction des diversités des kits et des heures de passage	118
4.9	Courbe de densité de distribution des pourcentage PE dans les états d’alerte . . . . .	119
4.10	Distribution des classes dans le <i>data set</i> initial . . . . .	120
4.11	Démarche suivie pour équilibrer les proportions des classes. . . . .	122
4.12	Distribution des classes dans le <i>data set</i> généré . . . . .	122
4.13	Matrice de confusion pour un modèle de classification binaire . . . . .	123
4.14	Conception des modèles avec BigQuery . . . . .	125
4.15	Matrices de confusion sur les données de validation pour les modèles de la première configuration . . . . .	127

TABLE DES FIGURES

---

4.16	Matrices de confusion sur les données de validation pour les modèles de la deuxième configuration . . . . .	128
4.17	Architecture d'entraînement. . . . .	129
4.18	Importance des caractéristiques pour les modèles K plus proches voisins et forêt aléatoire du Tableau 4.3 . . . . .	134
4.19	Importance des caractéristiques pour les modèles K plus proches voisins et forêt aléatoire du Tableau 4.4 . . . . .	135
5.1	Disposition du périmètre d'étude dans le processus de tôlerie automobile . . . . .	142
5.2	Architecture multi-agents pour le périmètre d'étude du processus de tôlerie . . . . .	143
5.3	Evolution dans le temps du nombre de pièces dans le meuble de stockage intermédiaire S1 . . . . .	144
5.4	<i>Accuracy</i> des modèles entraînés en fonction de la taille de la fenêtre d'analyse . . . . .	146
5.5	Matrices de confusion pour les modèles entraînés avec une fenêtre d'analyse de 60 minutes, sous différentes valeurs de h . . . . .	147
A.1	Prétraitement des données AMH . . . . .	165
A.2	Agrégation flux rang 0 . . . . .	165
A.3	Agrégation flux rang 1 . . . . .	166
A.4	Agrégation flux rang 0 et flux rang 1 . . . . .	166
A.5	Agrégation du composition et passage des véhicules . . . . .	166
A.6	Agrégation des appels d'alerte avec les PJI . . . . .	167
A.7	Mouvements de GEI pour un seul article : on constate que les supports physiques changent à chaque fois . . . . .	167
A.8	Agrégation des compositions avec les flux rang 0 et rang 1 . . . . .	168

## TABLE DES FIGURES

---

# Acronymes

AGV	Automated Guided Vehicle
AMH	Automated Material Handling
CFP	Call For Proposals
CPS	Cyber-Physical Systems
DAI	Distributed Artificial Intelligence
DF	Directory Facilitator
DOS	Door Sub
DWS	Digital Workstation
FIPA	Foundation for Intelligent Physical Agents
GCP	Google Cloud Platform
GE	Grands Emballages
HMI	Human Machine Interface
IA	Intelligence Artificielle
IoT	Internet of Things
IoS	Internet of Services
$K_C$	Candidate kit
$K_D$	Defective kit
$K_E$	Closest kit agent to the kitting area exit
KPI	Key Performance Indicator
MAS	Multi-Agent System
MES	Manufacturing Execution System
ML	Machine Learning
PA	Product Agent
PE	Petits Emballages
PL1	Logistique Programmée 1
PL1'	Logistique Programmée 1'
PL2	Logistique Programmée 2
PSFV	Pilotage et Suivi des Flux Véhicule
RA	Resource Agent
$S_{AGV}$	AGV speed
$S_{PL}$	Production line speed
$S_O$	Operator speed

## ACRONYMES

---

UC	Unité de Conditionnement
UCM	Unité de Conditionnement et de Manutention
UM	Unité de Manutention
WMS	Warehouse Management System

# Chapitre 1

## Introduction générale

### Contenu

---

1.1	Contexte . . . . .	<b>20</b>
1.2	Cas d'étude industriel . . . . .	<b>21</b>
1.3	Description du processus logistique . . . . .	<b>23</b>
1.3.1	Types de flux logistiques . . . . .	23
1.3.2	Approvisionnement des zones kitting . . . . .	24
1.3.3	Zone kitting . . . . .	26
1.3.4	Livraison des kits à la ligne d'assemblage . . . . .	27
1.3.5	Systèmes d'information . . . . .	28
1.3.6	Besoins industriels : problèmes les plus récurrents . . . . .	33
1.4	Problématique . . . . .	<b>34</b>
1.5	Objectif de la thèse . . . . .	<b>35</b>
1.6	Périmètre de l'étude : zone kitting DOS . . . . .	<b>36</b>
1.6.1	Approvisionnement des pièces à la zone . . . . .	37
1.6.2	Préparations des kits . . . . .	38
1.7	Modèle à événements discrets du système d'étude . . . . .	<b>39</b>
1.7.1	Première approche de modélisation : modélisation détaillée . . . . .	41
1.7.2	Deuxième approche de modélisation : modélisation simplifiée . . . . .	42
1.7.3	Implémentation sur Simul8 . . . . .	44
1.7.4	Exemple d'une première application . . . . .	46
1.7.5	Conclusion . . . . .	49
1.8	Plan du manuscrit . . . . .	<b>49</b>

---



### 1.1 Contexte

Les environnements de production et les processus de fabrication deviennent de plus en plus complexes, dynamiques et incertains. Ils font face à des changements constants, avec une tendance actuelle marquée vers la personnalisation de masse et une variété croissante de produits. Ainsi, la durabilité d'un site de fabrication se manifeste par la flexibilité et l'adaptabilité de ses outils de production pour répondre aux demandes du marché [Lafou et al., 2016], [Beauville dit Eynaud et al., 2019].

La numérisation des usines traditionnelles permet le développement et l'application de concepts et de technologies tels que les systèmes Cyber-Physiques (CPS : Cyber-Physical Systems), l'Internet des Objets (IoT : Internet of Things) et des services (IoS : Internet of Services), le Big Data et l'Intelligence Artificielle (IA) sur les systèmes de production pour répondre à ces besoins de flexibilité et d'adaptabilité [Kagermann et al., 2013]. En particulier, les systèmes de contrôle traditionnels pour les processus de fabrication, généralement monolithiques et centralisés, évoluent vers des systèmes plus modulaires, intelligents et distribués. Les systèmes multi-agents (MAS) [Wooldridge, 2009] constituent une approche adaptée pour mettre en œuvre cette nouvelle alternative de systèmes de contrôle, opérant au niveau de l'usine [Rodrigues et al., 2013].

Dans le contexte du secteur automobile, les processus de production se caractérisent par une grande variété de produits, un volume de production élevé et fluctuant, ainsi que de nombreux composants de véhicules. La conséquence de la personnalisation de masse et de la variété croissante des produits est que presque chaque véhicule d'une ligne d'assemblage est unique par rapport aux autres. Ces caractéristiques rendent la logistique interne extrêmement complexe et sujette aux perturbations. Les constructeurs doivent donc être capables de simplifier, de synchroniser les flux et de proposer de nouveaux processus logistiques flexibles et intelligents. Des outils d'aide à la décision sont donc incontournables pour aider l'industriel dans sa prise de décision [Klement et Silva, 2020].

La logistique interne au sein d'un site de fabrication automobile désigne l'organisation de tous les flux de matières qui ont lieu à l'intérieur du site, depuis la réception des pièces en usine jusqu'à leur approvisionnement sur la ligne d'assemblage des véhicules. Elle a pour mission de garantir que pour chaque véhicule, la bonne pièce sera disponible au bon endroit et au bon moment afin d'éviter des problèmes qualité [de Lima et al., 2020].

L'approvisionnement des pièces au processus d'assemblage ou de montage automobile peut être classifiée en deux types : l'approvisionnement en bord de ligne et le kitting. L'approvisionnement en bord de ligne consiste à stocker toutes les diversités de pièces à assembler à côté de la ligne d'assemblage et l'opérateur doit choisir les pièces correspondantes à chaque véhicule parmi toutes celles qui sont disponibles. Cette méthode nécessite de grandes surfaces à côté de la ligne d'assemblage, les erreurs de qualité peuvent être fréquentes lorsqu'il y a une grande variété de produits et elle manque de flexibilité

face aux changements de production. C'est pourquoi de nombreux grands constructeurs automobiles mettent en œuvre le kitting [Brynzér et Johansson, 1995]. Cela signifie que le processus d'assemblage est fourni avec des collections de composants et/ou de sous-ensembles à différents endroits de la chaîne, afin de supporter une ou plusieurs opérations d'assemblage pour un produit donné. Cette méthode permet d'avoir plus de flexibilité dans les lignes de montage, de paralléliser différentes tâches d'assemblage et de construire des structures de produits avec de nombreuses pièces tout en garantissant une qualité d'assemblage élevée.

Le processus de kitting doit respecter une structure de produit et un planning de production prédéfinis. En outre, la préparation et la livraison d'un kit implique plusieurs briques technologiques telles que des robots de prélèvement de bacs, des véhicules à guidage automatique (AGV : Automated Guided Vehicle), des systèmes de gestion d'entrepôt (WMS : Warehouse Management System) et des systèmes de contrôle de la qualité. La gestion des interactions entre ces éléments technologiques eux-mêmes et les opérateurs est une question importante pour garantir une bonne performance globale du processus et trouver des solutions efficaces en cas de perturbations.

Au sein du Groupe Renault, le département Flux et Architecture Industriels est en charge de la conception et validation des processus logistiques internes de toutes les usines du groupe. Pour répondre aux besoins de flexibilité de plus en plus importants, plusieurs briques technologiques faisant appel à l'intelligence artificielle ont été et sont développées. Ces briques s'intègrent ou complètent les briques existantes dans les processus actuels et elles ont donc besoin de s'interfacer aussi entre elles. L'un des enjeux majeurs est de gérer les interactions et de faire cohabiter l'ensemble de ces systèmes intelligents tout en améliorant la performance globale du système du processus logistique. Les travaux de cette thèse se déroulent au sein de ce département et cherchent à optimiser la performance globale de ce processus logistique en s'appuyant sur des techniques de l'intelligence collective et des systèmes multi-agents.

## 1.2 Cas d'étude industriel

L'ingénierie dans le Groupe Renault se divise en deux branches : l'Ingénierie Produit et l'Ingénierie de Production. La première s'occupe de définir et concevoir les véhicules et la deuxième se charge de la conception des usines et sites industriels permettant la fabrication des dits véhicules.

La Figure 1.1 présente le schéma de production d'un véhicule dans une usine de type carrosserie-montage. Le processus comprend quatre grandes étapes : l'emboutissage, la tôlerie, la peinture et le montage. Les organes mécaniques tels que le moteur, la boîte de vitesse et le châssis sont produits dans des usines dédiées et ensuite livrés à la chaîne de montage.

Le montage est le processus qui compte le plus grand nombre d'opérateurs de la chaîne de pro-

## 1.2. CAS D'ÉTUDE INDUSTRIEL

duction. Au cours de cette phase, un véhicule peut être équipé jusqu'à 2000 composants. Comme présenté par la Figure 1.2, l'assemblage suit une séquence fixe dans différents ateliers dédiés à une zone spécifique du véhicule, comme le sous caisse (Underfloor), l'intérieur du véhicule (Trim) et le châssis (Chassis trim). Cette ligne principale est complétée par des sous ateliers dédiés à la préparation de sous ensembles qui rejoignent la chaîne principale à différents endroits. C'est le cas des zones de préparation du sous caisse (Underfloor SUB), des portes et des roues (Module SUB).

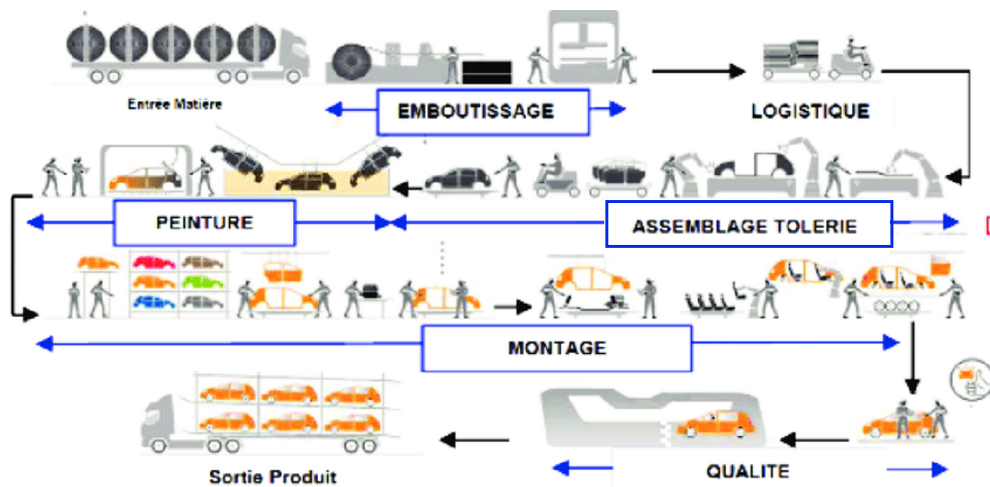


FIGURE 1.1 – Schéma de production usine carrosserie-montage.

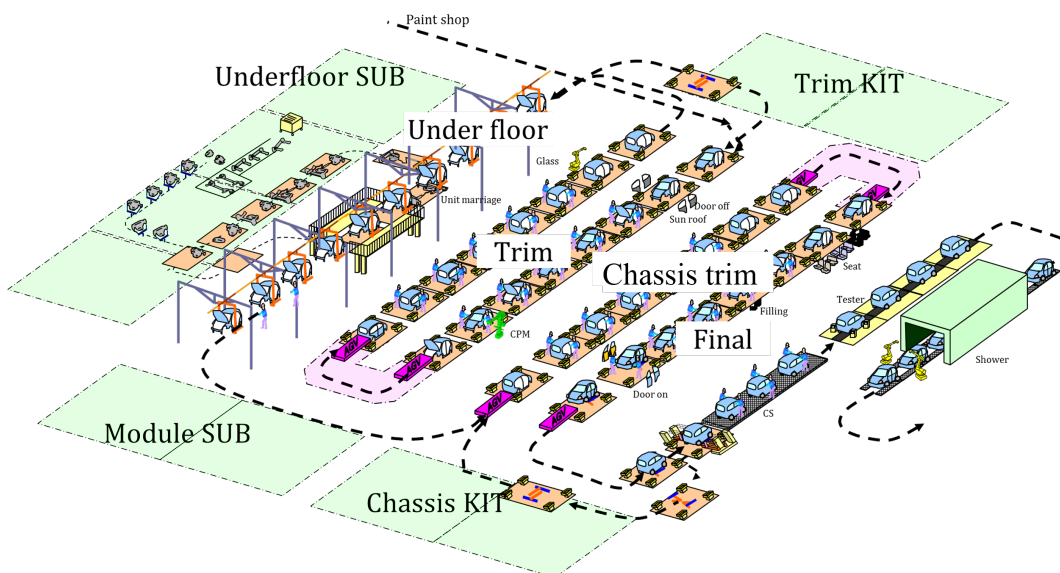


FIGURE 1.2 – Schéma processus de montage.

La ligne principale ainsi que les zones de préparation sont approvisionnées en pièces par la logis-

## 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE

---

tique interne de l'usine. La plupart de ces pièces sont livrées au site de production par des fournisseurs internes ou externes. Les fournisseurs internes sont d'autres sites de production du même groupe automobile qui se consacrent à la fabrication de pièces de véhicules pour les sites d'assemblage (comme les usines fabricant des organes mécaniques) et les fournisseurs externes sont des entreprises qui n'appartiennent pas au groupe automobile.

Afin de réduire la taille des zones de livraison en bord de ligne, d'alléger la charge cognitive des opérateurs et d'assurer la qualité des produits, des zones de préparation de kits sont installées autour de la ligne principale et des zones de préparation (Chassis KIT et Trim KIT sur la Figure 1.2). Chaque atelier de la ligne d'assemblage dispose d'une zone de kitting dédiée, afin de gérer la diversité des références.

Les zones de kitting sont alimentées par des opérateurs avec des emballages de pièces mono-référence, puis d'autres opérateurs préparent des kits pour les véhicules présents sur la ligne d'assemblage et enfin ces kits sont livrés sur la ligne par des AGVs. Ce processus logistique sera détaillé dans la Section 1.3.

Les travaux de cette thèse ont comme cas d'usage industriel une ligne d'assemblage alimentée par la méthode du kitting dans un site de production automobile. L'objectif étant d'optimiser la performance globale de l'ensemble du processus logistique d'approvisionnement de pièces depuis une zone de kitting jusqu'à la ligne d'assemblage.

## 1.3 Description du processus logistique

### 1.3.1 Types de flux logistiques

Depuis la réception à la zone de stockage ou à la zone de déchargement jusqu'au point de consommation sur la ligne de production, les flux des pièces sont classés en trois types : rang 0, rang 1 et rang 2 (voir Figure 1.3).

- Les flux de rang 0 consistent à décharger le camion de livraison et à stocker les produits dans l'entrepôt ou sur des aires temporaires. Ces flux sont opérés par des opérateurs équipés de cars à fourches.
- Les flux de rang 1 consistent à déplacer les pièces de l'entrepôt ou des zones temporaires vers les zones de kitting. Ces flux sont réalisés généralement à l'aide de cars à fourches ou à l'aide de bases roulantes conduits par des opérateurs. Sur certains cas, ils sont effectués par des tracteurs automatiques ou des Automated Material Handling (AMH).
- Les flux de rang 2 consistent à déplacer des kits de pièces des zones de kitting vers la ligne de production. Ces flux sont principalement assurés par des AGVs.

### 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE

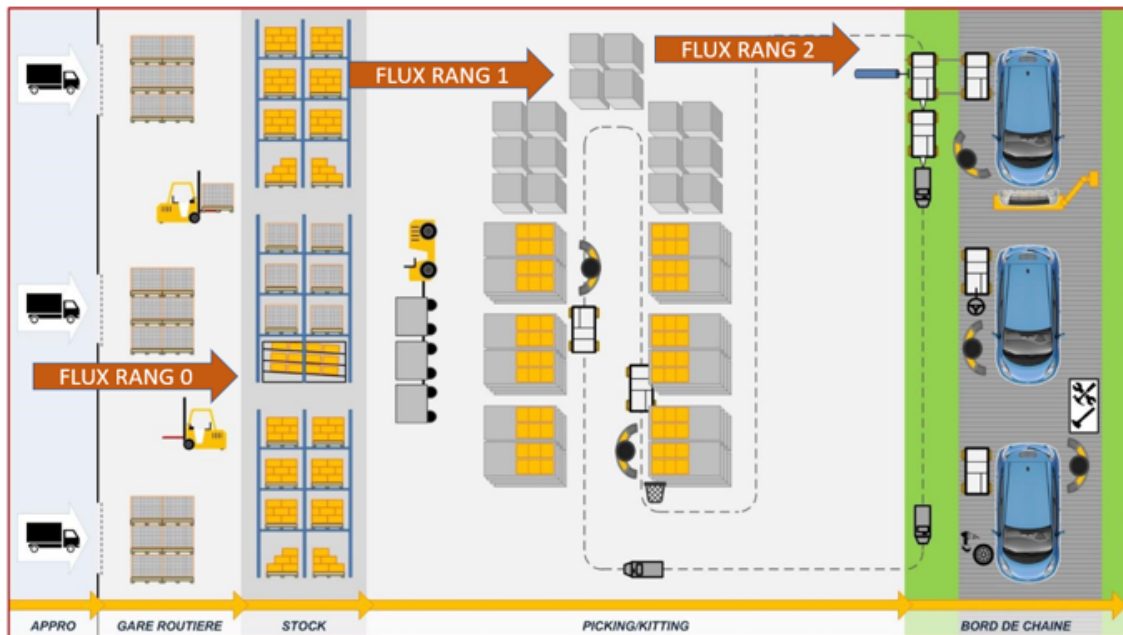


FIGURE 1.3 – Types de flux des pièces.

#### 1.3.2 Approvisionnement des zones kitting

En fonction de plusieurs facteurs tels que le nombre de diversités d'un type de pièce, ses dimensions et son poids, ainsi que son taux de consommation, une pièce peut être livrée à la zone kitting et ensuite à la ligne d'assemblage par différentes solutions logistiques. La Figure 1.4 met en évidence les 6 types des solutions existantes. Elles sont détaillées ci-après.

**Logistique programmée 1 (PL1).** L'objectif de cette solution est de transporter les marchandises de la zone de stockage vers les zones de kitting. Les palettes de ces marchandises doivent alors avoir déjà été étiquetées avec l'étiquette de la zone de kitting correspondante. Les opérateurs seront affectés à une zone de kitting particulière. Ils sont chargés de préparer et de fournir les pièces nécessaires à cette zone. La fréquence des distributions est déterminée en fonction de l'autonomie de la zone de kitting et d'une distribution à l'autre, seules les quantités à livrer changent. Les quantités sont déterminées par l'observation des niveaux des stocks dans la zone.

**Logistique programmée 1' (PL1').** L'objectif de la solution PL1' est le même que pour la PL1, il s'agit de transporter les marchandises de la zone de stockage vers les zones de kitting. Cependant, avec cette méthode, la planification des livraisons est effectuée en fonction de la séquence de production, ce qui n'existe pas dans la PL1. Les besoins de la ligne de montage pour le jour J+2 sont déterminés le jour J, en fonction de la séquence de production. Ainsi, les livraisons de la zone de stockage vers les zones de kitting sont planifiées par périodes de 2 heures ou plus, selon l'autonomie de la zone de

### 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE

---

kitting. Les palettes avec les pièces nécessaires sont préparées et placées dans des zones dédiées pour leur distribution au bon moment.

**Logistique programmée 2 (PL2).** La solution logistique PL2 peut être appliquée uniquement aux petits emballages. Comme pour PL1', cette solution planifie les livraisons par périodes de 2 heures en général, en fonction des besoins sur la ligne de montage. Les besoins sont également déterminés grâce à la séquence de production finale. La principale différence entre PL1' et PL2 est que pour les solutions PL2, les palettes de pièces sont préparées et étiquetées par le fournisseur et non par les opérateurs du site de production. Dans ce cas, les opérateurs sont uniquement chargés de la distribution des pièces. Cela permet d'éviter le stockage des pièces, car les palettes livrées seront consommées le jour même de la livraison ou le jour suivant.

**Solutions à flux direct.** Ces solutions sont basées sur le principe du juste-à-temps. Les pièces ne sont pas stockées dans l'usine, mais elles sont placées brièvement en zone d'attente avant d'être envoyées vers les zones de kitting. Ces solutions ne nécessitent pas la manipulation des marchandises par des caristes, ni de grandes surfaces de stockage. Elle est généralement utilisée pour les pièces assemblées à 100% de véhicules.

**L3PS.** L'objectif de cette solution est de livrer les pièces dans l'ordre de la séquence de production sur la ligne de montage. Les pièces sont produites et livrées séquentiellement par le fournisseur, en fonction de l'ordre des véhicules sur la ligne de montage. Comme on peut le voir sur la Figure 1.4, les emballages contenant les pièces ordonnées ne sont stockés que pendant quelques heures et sont ensuite envoyés vers la ligne d'assemblage ou vers une zone de kitting, en fonction de l'alternative de déploiement choisie. Cette solution est adaptée aux pièces présentant une grande variété ou un grand nombre de diversités et aux pièces de grandes dimensions qui peuvent être manipulées manuellement par un opérateur.

**Juste à temps (synchrone).** Les pièces qui sont livrées en utilisant cette solution sont parfaitement ordonnées avec la séquence de production. L'ordre de livraison est envoyé au fournisseur lorsque le véhicule associé entre dans la ligne de montage. Ainsi, la séquence des véhicules ne peut pas être modifiée. Le fournisseur n'a que quelques heures pour produire les pièces en séquence et les livrer directement sur la ligne de montage. Les pièces sont assemblées presque immédiatement et n'ont pas besoin d'être stockées dans l'usine. Le fournisseur peut également recevoir les prévisions de production et la séquence initiale de production afin de pouvoir planifier sa propre production et éviter toute pénurie. Pour mettre en œuvre cette solution, le fournisseur doit être situé très près de l'usine afin de pouvoir suivre la cadence de production. Cette solution est appliquée aux pièces de grande variété et de dimensions massives car elles sont difficiles à manipuler manuellement par un opérateur. Étant donné que cette solution livre les pièces directement à la ligne d'assemblage sans passer par une zone kitting, elle ne fera pas partie du périmètre d'étude de cette thèse.

### 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE

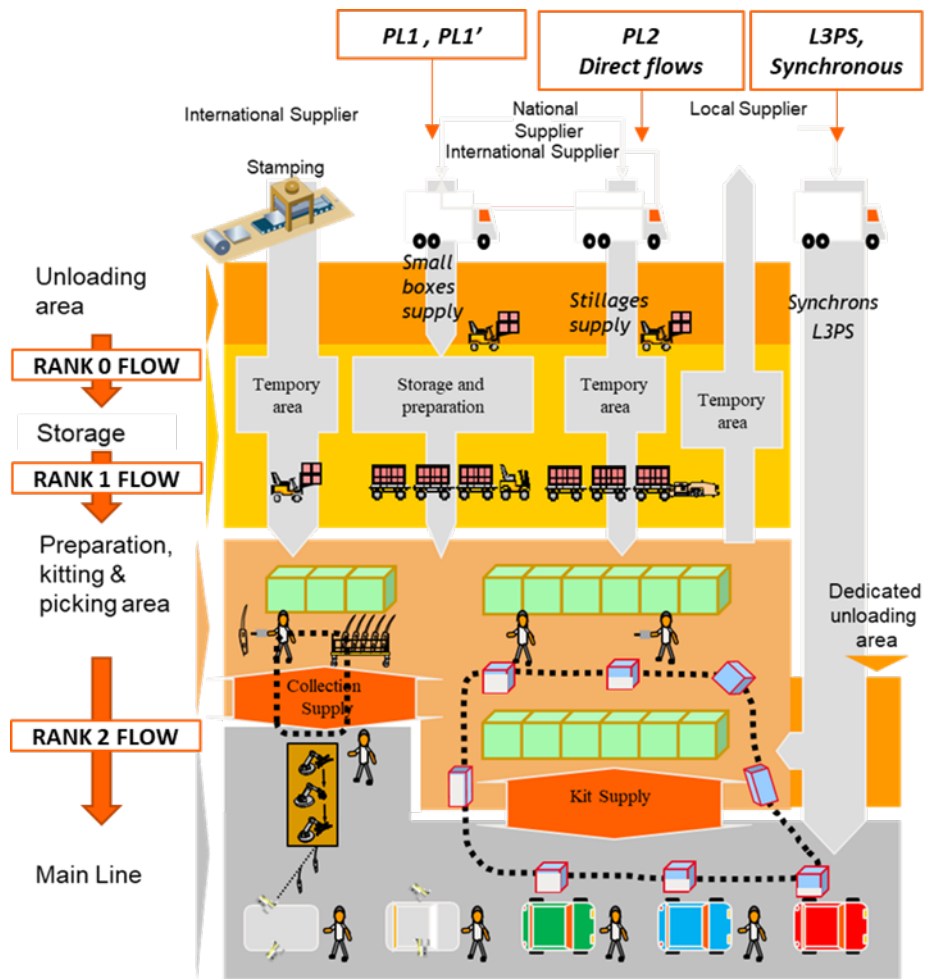


FIGURE 1.4 – Types de solutions logistiques.

#### 1.3.3 Zone kitting

Le processus de kitting consiste à regrouper des pièces de familles différentes qui seront assemblées sur un même véhicule. Pour créer un kit, les opérateurs collectent des pièces mises à disposition sur des meubles placés autour de leur zone de travail. Ils sont guidés par un système d'information connecté au système de production qui indique les pièces à prendre et à mettre dans le kit. Cela se fait grâce à un système d'éclairage installé sur les meubles contenant les emballages des pièces. Lorsqu'une pièce doit être prélevée, la lumière associée s'allume pour informer les opérateurs.

Une zone kitting peut être caractérisée en ayant trois informations :

- Le nombre de références gérées dans la zone.
- La taille des emballages contenant les dites références.
- Le taux des consommation de chacune des références.

### 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE

Les emballages sont classifiés en deux types : les petits emballages (PE) et les grands emballages (GE). Le Tableau 1.1, présente des exemples d'emballages ainsi que leurs dimensions.




Taille emballage	Type d'emballage	Exemple	Dimensions (mm x mm)
Petit emballage (PE)	Unité de conditionnement (UC)		400 x 600
Grand emballage (GE)	Unité de manutention (UM)		1200 x 1000
Grand emballage (GE)	Unité de conditionnement et de manutention (UCM)		1200 x 1000

TABLEAU 1.1 – Taille des emballages

Quant au nombre de références gérées dans la zone et à leurs taux de consommation, ils peuvent être très variables en fonction de la référence, de la zone et de l'usine. Par exemple, le Tableau 1.2 montre un extrait du nombre de références gérées par quelques zones d'une usine de carrosserie-montage du Groupe Renault. Il est possible de voir que certaines zones gèrent environ 100 références alors que d'autres en gèrent environ 1000. De la part des taux de consommation, ils peuvent aller d'1 à 650 pièces par jour.

Zone kitting	Nombre de PE	Nombre de GE	Total
Chassis Trim	603	397	1000
Door sub	423	119	542
Trim Line	304	121	425
Cockpit sub	211	39	250
Chassis Under floor sub	111	129	240
Trim Line	122	86	208
Engine sub	125	60	185
Finale	105	4	109
Under Floor	42	49	91
Mirror parts	46	41	87

TABLEAU 1.2 – Nombre de références gérées par zone kitting

#### 1.3.4 Livraison des kits à la ligne d'assemblage

Comme dit à la Section 1.3.1, lorsque les kits sont préparés, leur livraison à la ligne d'assemblage est assuré par des AGVs. La Figure 1.5 met en évidence la cartographie globale d'une ligne d'assem-



### 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE

---

blage. Sur cette image, les zones colorées par des couleurs claires représentent les zones de kitting qui approvisionnent les zones de la ligne principale, colorées en couleurs foncées. Ici, il est possible d'apprécier que les zones kitting sont proches des zones de la ligne de montage correspondantes. Cette configuration est faite afin de simplifier le plus possible les flux de Rang 2.

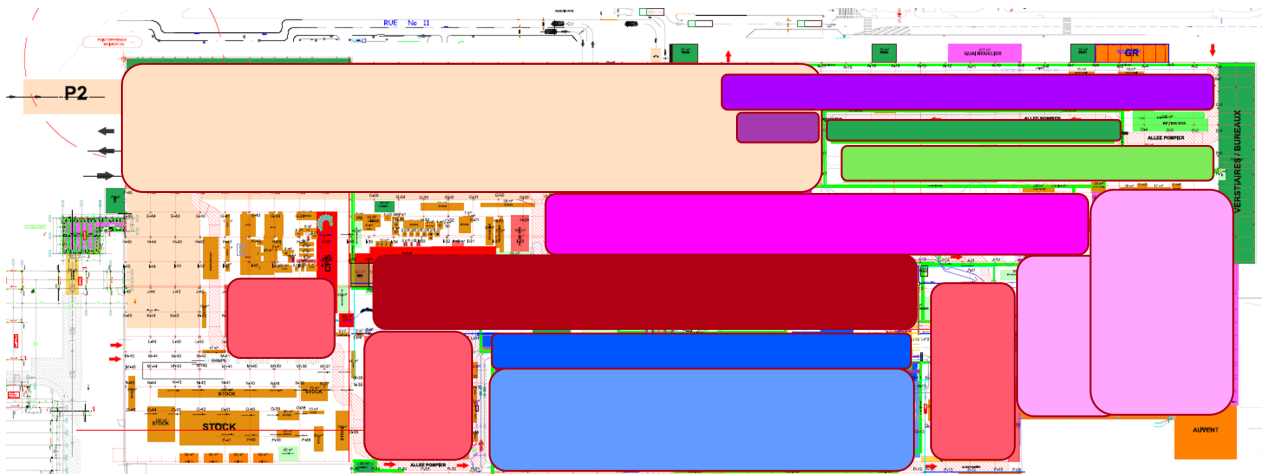


FIGURE 1.5 – Cartographie atelier montage.

#### 1.3.5 Systèmes d'information

En plus des systèmes physiques présents sur le terrain industriel, le processus de livraison des pièces à la ligne de montage est appuyé et suivi par plusieurs systèmes d'informations. Ces systèmes sont des sources d'informations importantes pour nos propos. En divisant ce processus global en trois phases : l'approvisionnement de pièces à la zone de kitting, la constitution des kits dans la zone de kitting et le processus de montage des portes, les systèmes d'information qui y sont présents sont listés sur la Figure 1.6 et sont décrits dans les sections suivantes.

##### 1.3.5.1 Reflex

Reflex est un logiciel de gestion d'entrepôts qui a pour but d'aider les industriels, les distributeurs et les prestataires logistiques à mettre en place une exécution logistique fiable, performante et agile dans leurs entrepôts. Plus spécifiquement, il peut être utilisé pour :

- Optimiser la gestion des réceptions, des stocks, des préparations, des inventaires et des retours.
- Livrer les produits dans des délais de plus en plus courts.
- Maîtriser les coûts logistiques : gains de productivité, optimisation des stocks et des transports.
- Gagner en agilité et être réactif aux changements : fortes rotations, saisonnalité, etc.
- Améliorer la collaboration en amont et en aval de l'entrepôt, avec les fournisseurs et transporteurs.

### 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE

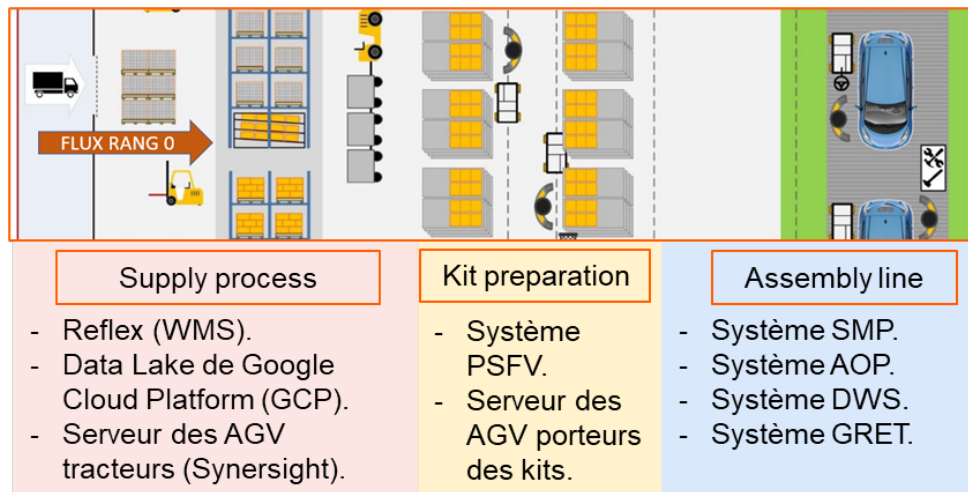


FIGURE 1.6 – Systèmes d’information

- Faire face aux enjeux de la logistique e-commerce.
- Répondre aux exigences réglementaires et qualité, en termes de traçabilité des flux et des marchandises.
- Orchestrer en temps réel la mécanisation et la robotisation.

Pour le cas d’étude traité, grâce à ce logiciel nous pouvons suivre les mouvements des emballages au sein de l’usine jusqu’à leur arrivée à la zone de kitting. Chaque emballage a une référence unique dans le logiciel et avec l’aide de cette référence, nous pouvons connaître la date et l’heure à laquelle un cariste a pris l’emballage de la gare routière et l’heure à laquelle il a été mis à disposition dans la zone de stock ou dans la zone de kitting. De plus, lors de la livraison à la zone de kitting, le cariste doit renseigner dans ce système le nombre de pièces (nombre d’unités de manutention) qu’il a déposé dans la zone. De cette façon, le stock et la commande des nouveaux emballages de ce type peuvent être gérés. A titre d’exemple, la Figure 1.7 présente l’une des fenêtres de l’interface de ce logiciel. Sur cette figure, il est possible de voir que le support avec référence 052030000000166865 a été réceptionné dans l’usine le 27 juillet 2021 à 7h43, qu’il a été mis en stock à 8h27 et finalement qu’il a été déposé dans la zone kitting à 12h03 le 7 septembre 2021.

#### 1.3.5.2 Data Lake de Google Cloud Platform (GCP)

Le data lake de Google est un dépôt centralisé ayant pour objectif de stocker, traiter et sécuriser de grands volumes de données structurées ou non structurées. Il peut traiter tout type de données et les stocker dans leur format natif. Les lacs de données sont utilisés non seulement pour stocker la donnée mais aussi pour préparer ces données à une analyse ou exploitation ultérieure. Pour ce faire, plusieurs langages de programmation tels que SQL, Python ou R peuvent être utilisés. Chez Renault,

### 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE

---

Dépôt	Emplacement				Date de début	Date de fin			
041	F001	1	141	30	C	07/09/21	0:00:00	07/09/21	12:03:41
041	F001	1	120	03	E	07/09/21	12:03:09		
041	F001	1	141	30	C	27/07/21	8:27:43	07/09/21	12:03:09
041						27/07/21	8:06:06	27/07/21	8:27:43
041	191G	D	A40			27/07/21	7:43:39	27/07/21	8:06:06
302	191G	Y	DA3			26/07/21	11:48:31	27/07/21	6:39:23
302	ANO		REC			26/07/21	11:15:27	26/07/21	11:48:31

FIGURE 1.7 – Cheminement d’une palette sur Reflex

des nombreux types de données y sont stockées. Parmi ces données, nous y retrouvons des données sauvegardées et générées par Reflex. Ainsi, nous utiliserons ce lac de données pour récupérer certaines des informations des mouvements des palettes ou des emballages d’une façon plus directe qu’en utilisant Reflex. Par exemple, sur GCP nous récupérerons le numéro de référence pièce correspondant à chaque référence de support, de cette manière, nous pourrons mettre en relation les informations obtenues de Reflex avec une référence de pièce spécifique.

#### 1.3.5.3 Serveur des AGVs Synersight

Synersight est une entreprise espagnole spécialisée dans la conception et la fabrication d’AGVs. Les AGV tracteurs et les AGV porteurs des chariots de kits qui sont présents dans la zone de kitting de portes de l’usine d’étude ont été fournis par Synersight. Lors de leur fonctionnement, les AGVs doivent suivre des bandes magnétiques mises en place au sol afin d’indiquer le chemin à prendre. A différents endroits de ces bandes magnétiques, il y a des étiquettes (tags RFID) qui permettent d’établir une communication entre les AGVs et leur serveur. Le serveur a pour mission de suivre et de coordonner tous les AGVs de l’usine. Ainsi, il utilise les étiquettes en question pour échanger des informations avec chaque AGV.

Ces étiquettes sont aussi appelées tags. Grâce à ces tags, nous pouvons connaître la position des AGVs d’une manière discrète mais presque en temps réel. La Figure 1.8 présente une image des informations brutes que nous pouvons récupérer du Tag numéro S18905. Il est possible d’y voir la date et l’heure de passage d’un AGV par l’étiquette, le temps qu’il y est resté, l’AGV qui y avait été précédemment ainsi que l’étiquette lue précédemment par l’AGV.

### 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE

Tag S18905					
AGV	Fecha lectura	Anterior AGV en el TAG	Tiempo	Anterior tag leído	Tiempo
3244	26/04/2022 14:04:04	AGV 3243	31m 0s	Tag S18917	1m 24s
3243	26/04/2022 13:33:04	AGV 3244	49m 58s	Tag S18917	1m 12s
3244	26/04/2022 12:43:06	AGV 3243	31m 21s	Tag S18917	1m 13s
3243	26/04/2022 12:11:45	AGV 3244	20m 58s	Tag S18917	1m 13s
3244	26/04/2022 11:50:47	AGV 3243	66m 35s	Tag S18917	5m 2s
3243	26/04/2022 10:44:12	AGV 3244	60m 20s	Tag S18917	14m 8s
3244	26/04/2022 09:43:52	AGV 3243	41m 20s	Tag S18907	12m 3s
3243	26/04/2022 09:02:32	AGV 3244	58m 37s	Tag S18907	10m 17s
3244	26/04/2022 08:03:55	AGV 3243	37m 22s	Tag S18907	12m 20s
3243	26/04/2022 07:26:33	AGV 3244	1032m 15s	Tag S18907	8m 5s
3244	25/04/2022 14:14:18	AGV 3243	30m 21s	Tag S18907	12m 11s
3243	25/04/2022 13:43:57	AGV 3244	54m 56s	Tag S18907	16m 20s
3244	25/04/2022 12:49:01	AGV 3243	57m 57s	Tag S18907	34m 37s
3243	25/04/2022 11:51:04	AGV 3244	67m 9s	Tag S18907	21m 36s
3244	25/04/2022 10:43:55	AGV 3243	46m 21s	Tag S18907	12m 28s
3243	25/04/2022 09:57:34	AGV 3244	63m 28s	Tag S18907	29m 45s
3244	25/04/2022 08:54:06	AGV 3243	88m 3s	Tag S18907	49m 10s
3243	25/04/2022 07:26:03	AGV 3244	3994m 37s	Tag S18907	5m 40s

FIGURE 1.8 – Historique d'événements Tag S18905

#### 1.3.5.4 Système PSFV (Pilotage et Suivi des Flux Véhicule)

PSFV est le MES (Manufacturing Execution System) standard Renault pour le pilotage des lignes de carrosseries montage des usines du groupe. Ses objectifs principaux sont, d'une part, transmettre en temps utile, aux opérateurs de la chaîne de production et aux systèmes utilisateurs, toutes les informations dont ils ont besoin pour fabriquer les véhicules. D'autre part, recueillir et transmettre en continu les informations sur le stade de fabrication des véhicules, de telle manière que les opérateurs de production puissent régler et adapter leurs tâches.

Dans le périmètre de notre cas d'application, ce système informe les opérateurs de la zone de kitting de la liste de pièces qu'ils doivent mettre sur chaque chariot de kit. La Figure 1.9 montre l'un des écrans d'information pour les opérateurs de la zone de kitting. Elle met en évidence la liste des pièces nécessaires pour le kit correspondant à la voiture numéro 1330018.

#### 1.3.5.5 Digital Work Station (DWS)

Ce système permet aux opérateurs qui travaillent sur la ligne de montage de remonter toute anomalie à leur supérieur, l'opérateur senior.

Le système DWS est composé d'un écran qui est disposé à côté de chaque opérateur de la ligne de montage. La Figure 1.10 illustre l'écran disponible pour les opérateurs. Comme on peut voir sur l'image, à travers ce système, l'opérateur nécessitant de l'aide peut directement informer son opérateur senior si son besoin est de type sécurité, requête, manque-pièce ou des moyens.

### 1.3. DESCRIPTION DU PROCESSUS LOGISTIQUE



FIGURE 1.9 – Liste de pièces pour le kit du véhicule 1330018 transmise par PSFV



FIGURE 1.10 – Écran disponible pour les opérateurs du système DWS

Dans le cadre de notre étude, grâce à ce système nous pouvons connaître l'heure à laquelle une alerte a été déclenchée pour le kit d'un véhicule donné.

### 1.3.6 Besoins industriels : problèmes les plus récurrents

Dans le cadre de la thèse, nous voulons nous concentrer sur les problèmes et les perturbations qui peuvent survenir pendant le processus de livraison des pièces depuis les zones de préparation de kits jusqu'à la ligne d'assemblage. Dans ce contexte, nous nous focaliserons sur les flux de Rang 1 et de Rang 2, et sur le processus interne des zones de préparation de commandes. Puisque la méthode logistique "Juste à temps" n'approvisionne pas les zones de kitting, cette méthode ne fera pas l'objet d'étude.

Les principales perturbations qui ont été identifiées peuvent être classées en trois catégories.

#### La conformité des kits sur la ligne de montage :

- L'opérateur de la ligne de montage n'est pas en mesure d'assembler la pièce sur le véhicule parce que la pièce est manquante dans le kit ou que ce n'est pas la bonne pièce.

#### Conformité de l'approvisionnement de la zone de kitting :

- Les pièces fournies à la zone de kitting sont erronées. Cela peut être dû à la livraison d'un mauvais emballage de pièces par les opérateurs à la zone de kitting ou parce que les pièces à l'intérieur du bon emballage ne correspondent pas à l'étiquette de celui-ci.
- Le fournisseur n'est pas en mesure de livrer les pièces au bon moment.

#### Disponibilité des AGVs :

- Les AGVs sont arrêtés par des obstacles sur leur chemin ou ils doivent s'arrêter pour recharger leurs batteries. Si ce n'est pas prévu, cela peut retarder le processus de livraison et arrêter la chaîne de montage.
- Lors de la constitution des kits sur les AGV, la préparation du kit de certaines diversités prennent plus de temps que d'autres et, selon le niveau d'expertise de l'opérateur de la zone de kitting, l'opération peut dépasser le temps de cycle. Étant donné que tous les AGVs de la zone suivent le même chemin, les AGV se trouvant en amont peuvent se retrouver bloqués. Cela génère des retards dans le processus de livraison des kits, et ensuite l'arrêt de la ligne d'assemblage.

Selon une étude menée en interne, l'anomalie la plus récurrente pendant le processus d'approvisionnement de pièces par kitting, est l'anomalie de pièce manquante dans un kit. Ce problème est généralement détecté par les opérateurs lorsque les kits sont déjà sur la ligne d'assemblage. Actuellement, lorsqu'un opérateur de ligne d'assemblage déclenche ce type d'alerte, un autre opérateur doit identifier le type de modèle du véhicule et la référence de la pièce manquante, puis se rendre dans

les zones de préparation de kits pour récupérer la pièce. Ce processus peut être long et entraîner des arrêts de la ligne de production. Dans ces travaux de thèse, nous nous concentrerons principalement sur cette problématique.

### 1.4 Problématique

Dans un contexte où un site de production automobile doit être capable de fabriquer différents modèles sur la même ligne d'assemblage avec une forte cadence de production, où le nombre de composants des véhicules est élevé, où le volume de production est évolutif et où le nombre des moyens et systèmes impliqués dans le processus d'approvisionnement de pièces à la ligne de montage est important, la logistique interne des usines automobiles est rendue complexe et est sujette à l'apparition de perturbations fréquentes.

Ainsi, la problématique à laquelle cette thèse tente d'apporter des réponses est : **comment optimiser la performance globale des processus logistiques internes en gérant les interactions des différents agents concernés dans le processus ?**

En outre, la cohabitation de différents systèmes intelligents dans un même environnement peut à la fois être conflictuelle ou coopérative. Le développement et l'arrivée de plus en plus de systèmes intelligents dans ce processus logistique doivent donc être pris en compte dans la réflexion.

Pour une compréhension de la suite de l'étude, il est nécessaire de définir trois termes clés de ces travaux : agent intelligent, système multi-agents et intelligence collective.

Par **agents intelligents** sont désignées toutes les entités ou systèmes capables de percevoir et interagir avec leur environnement, traiter les informations captées, prendre des décisions en fonction de leurs objectifs et de réaliser des actions pour les atteindre [Wooldridge, 2009].

Les **systèmes multi-agents** sont donc des systèmes distribués composés par sous-systèmes ou agents intelligents indépendants, qui coopèrent ou sont en compétition pour atteindre un certain objectif [Tuyls et Weiss, 2012]. Ces systèmes sont utilisés pour traiter des problèmes de nature distribuée, complexes et dont la taille est de plus en plus importante. Ces caractéristiques rendent laborieux ou impossible leur modélisation d'une manière centralisée [Xie et Liu, 2017]. Ainsi, afin de résoudre ce type de problèmes et de prendre en compte des environnements dynamiques et changeants, les agents intelligents doivent apprendre de manière autonome à partir de l'expérience [Weiss, 2013]. Des exemples d'applications de cette méthode sont le contrôle des feux de circulation, la conduite autonome et la simulation et surveillance de systèmes d'énergie [McArthur et al., 2007b], [McArthur et al., 2007a], [Catterson et al., 2012].

L'**intelligence collective** est définie comme la capacité d'un groupe d'agents à travailler et à coopérer ensemble, tout en faisant des contributions indépendantes, afin de résoudre des problèmes plus com-

## 1.5. OBJECTIF DE LA THÈSE

---

plexes que ceux qui auraient pu être résolus par chaque agent de manière autonome [Singh et Gupta, 2009]. Ce type de comportement est flagrant dans la nature dans le cas des colonies de fourmis [Russell, 1999] ou des groupes d'oiseaux migrateurs [Gunasekaran et al., 2017]. L'étude de ce concept a inspiré le développement de différents algorithmes d'optimisation [Bonabeau et al., 1999] tels que l'algorithme des colonies de fourmis [Dorigo et al., 1999], l'algorithme de la colonie d'abeilles artificielles [Karaboga et Basturk, 2008] et l'algorithme d'optimisation par essais particuliers [Banks et al., 2007].

### 1.5 Objectif de la thèse

Les systèmes actuellement impliqués dans le processus logistique d'approvisionnement de pièces à la ligne de montage ont été développés pour travailler d'une manière indépendante en limitant les interactions entre eux. La Figure 1.11 présente cette situation. Pour chacun des systèmes, ses objectifs, ses données d'entrée, son mode de fonctionnement et sa performance sont définis et calculés de manière individuelle. L'objectif de cette thèse est de repenser les interactions pouvant avoir lieu entre ces systèmes et les systèmes arrivants afin de transformer cette complexité en une opportunité pour améliorer la performance globale du processus logistique et non la performance individuelle de chaque système. Par exemple, dans les cas où une alerte par pièce manquante dans un kit est déclenchée, uniquement, les opérateurs de la ligne de montage ou des zones de kitting sont informés et s'occupent d'y répondre. En ayant des interactions plus riches entre les agents qui sont impliqués dans le processus de logistique interne, les alertes et les perturbations ne seraient pas communiquées qu'aux opérateurs mais aussi aux autres agents impliqués dans le système (Figure 1.12). De cette façon, ils pourraient proposer des solutions pour y faire face.

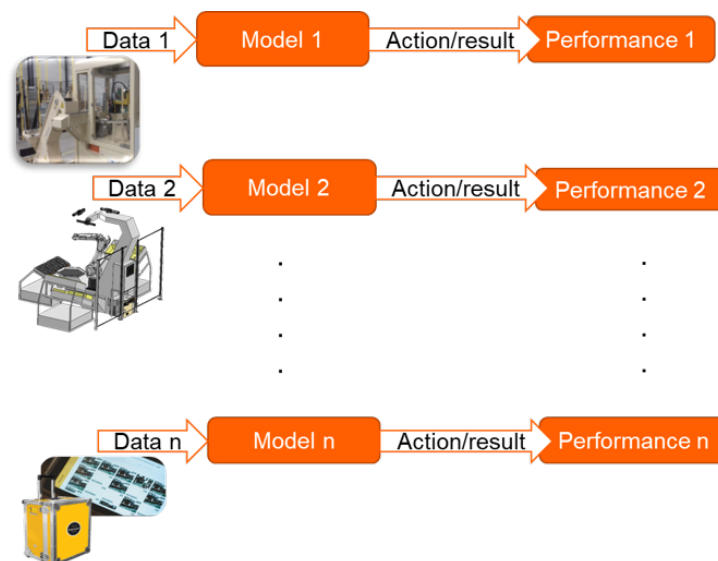


FIGURE 1.11 – Situation actuelle



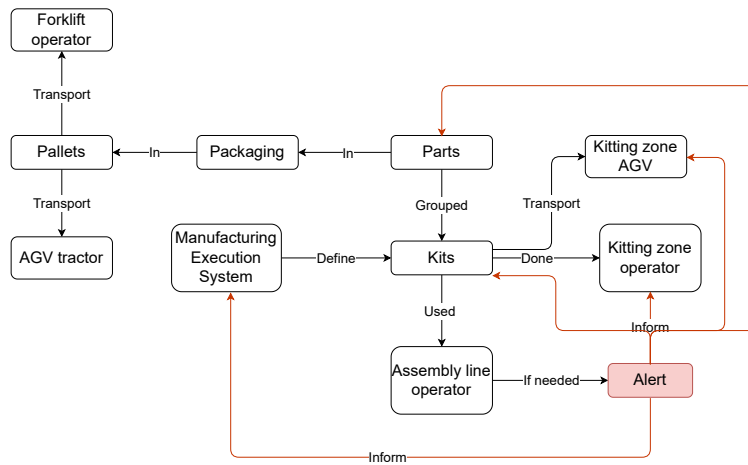


FIGURE 1.12 – Fonctionnement souhaité du système d’approvisionnement de pièces à la ligne de montage de véhicules [Montoya Zapata et al., 2024b]

Dans ce but, les travaux de cette thèse consistent à :

- Proposer une architecture permettant de modéliser ce processus logistique de livraison des pièces à une ligne de montage de véhicules comme un système multi-agents. Ainsi, des techniques d’apprentissage multi-agents et d’intelligence collective pourront être appliquées par la suite.
- Définir un protocole d’interaction entre les agents virtuels composants l’architecture proposée. De cette façon, une fois que l’anomalie ou perturbation est détectée, ces agents pourront communiquer afin de proposer des solutions potentielles pour l’éviter ou pour y faire face.
- Étudier le modèle de comportement du système global à travers les informations remontées par chacun des agents. La cible étant de détecter ou anticiper des perturbations pouvant arriver au système.

La cible idéale de cette thèse serait de créer un système cyber-physique, en couplant les briques citées précédemment avec le système réel. Ainsi, nous pourrions suivre son comportement en permanence et réagir au cas où une perturbation soit anticipée ou détectée. La Figure 1.13 schématise ce processus.

## 1.6 Périmètre de l’étude : zone kitting DOS

Les travaux de cette thèse ont été menés principalement sur la zone qui s’occupe de la préparation des kits des portes (zone kitting Door sub ou zone kitting DOS) d’une usine de carrosserie-montage du Groupe Renault. Une autre étude a également été menée dans la zone tôlerie, ce qui sera détaillé dans le Chapitre 5.

Dans le cadre du kitting, nous avons étudié les flux d’approvisionnement de pièces à la zone kitting

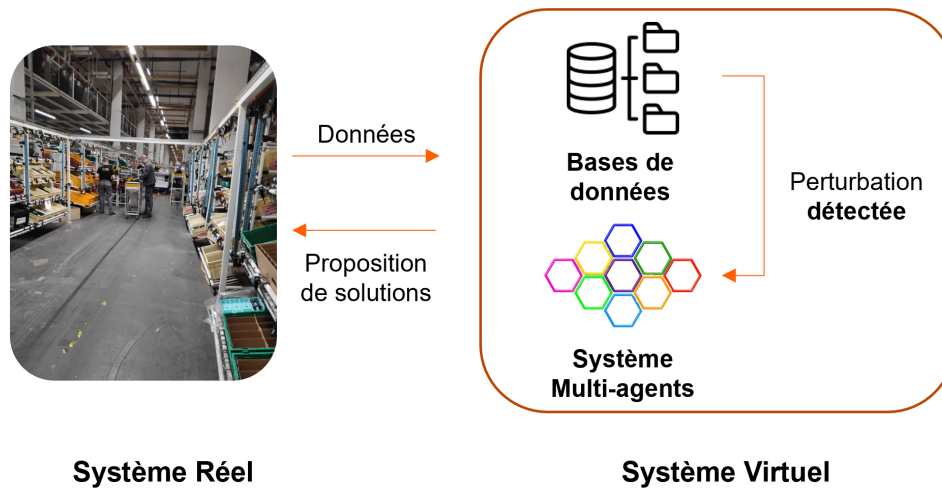


FIGURE 1.13 – Cible de la thèse

en question, la constitution des kits dans la zone et la livraison de ces kits depuis la zone kitting à la ligne de montage des portes. Actuellement, le site de production considéré fabrique plusieurs modèles de véhicules différents avec une cadence de production de l'ordre de 30 à 60 véhicules par heure. Chaque modèle comporte plusieurs versions et peut être personnalisé en fonction de la finition du véhicule et de l'équipement choisi par le client. En outre, ces modèles changent constamment.

Pendant les travaux de thèse, l'usine a été en pleine transition, passant de la fabrication de modèles de voitures thermiques à la fabrication de voitures électriques. Nous avons choisi d'étudier la zone de préparation des kits de portes, car elle n'a pas été très impactée par cette transition. Cette zone n'a pas subi de changements importants liés au type de motorisation des véhicules. De plus, sa configuration permet l'interaction de plusieurs types d'agents, notamment son flux de Rang 1 qui est effectué par un tracteur automatique.

### 1.6.1 Approvisionnement des pièces à la zone

La Figure 1.14, présente le cheminement des pièces qui sont kittées dans la zone kitting DOS. D'abord les emballages arrivent à la gare routière DA (numéro 1 sur la cartographie), ensuite un opérateur (cariste) dépose les emballages sur les plateformes des tracteurs automatiques au point 2. Le tracteur transporte les pièces au point 3 pour qu'un autre cariste amène les pièces au point 4 (zone de stock tampon). A partir de cette zone, un troisième cariste s'occupe de mettre les pièces en stock sur des palettiers (étagères) et d'approvisionner les meubles de la zone kitting, dont certains se trouvent sous les palettiers.

## 1.6. PÉRIMÈTRE DE L'ÉTUDE : ZONE KITTING DOS

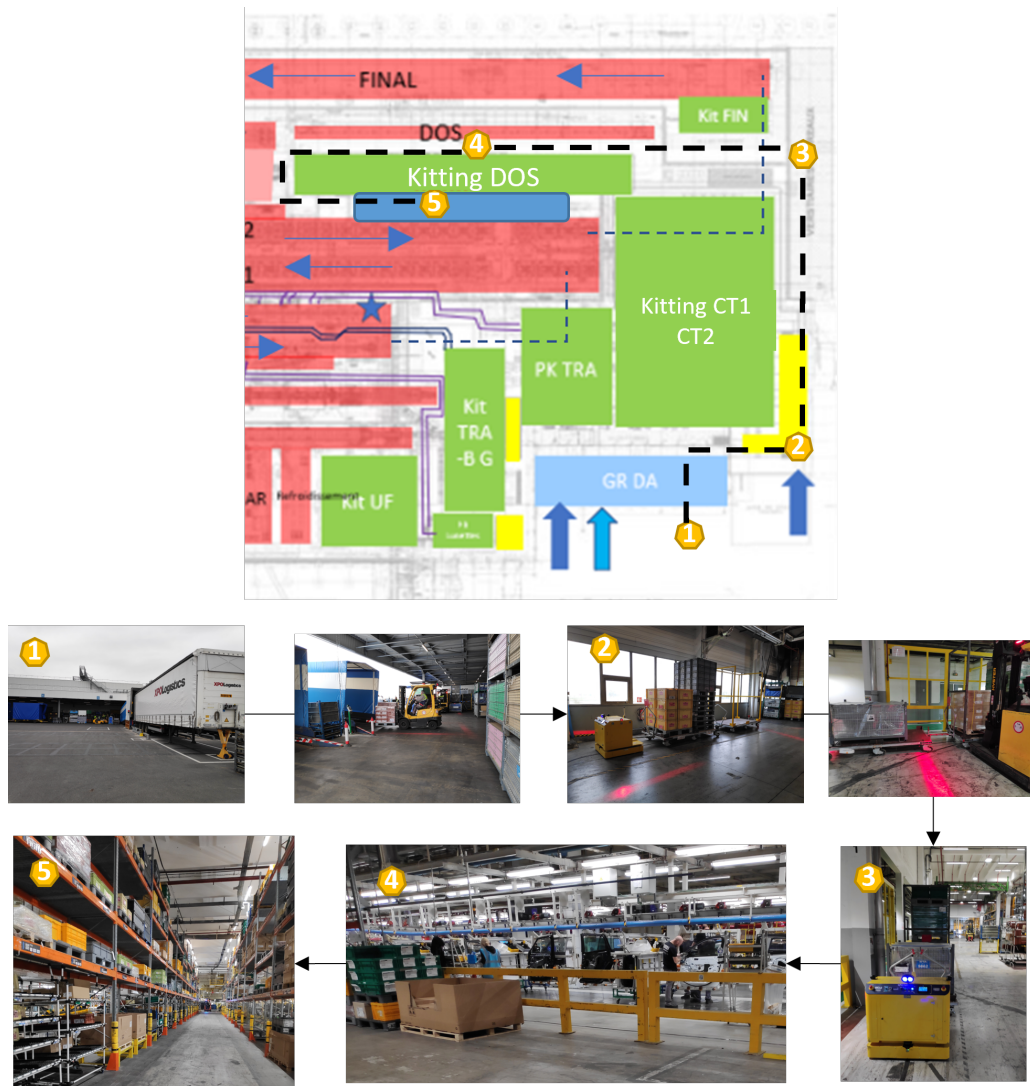


FIGURE 1.14 – Approvisionnement des pièces à la zone de kitting.

### 1.6.2 Préparations des kits

Comme expliqué précédemment, un kit est un ensemble de pièces de familles différentes destinées à être assemblées sur un même véhicule. Ces pièces sont transportées et regroupées dans un chariot de kit. Les pièces de petite taille sont regroupées dans des boîtes appelées "kit boxes", tandis que les pièces de grande taille sont placées directement sur le chariot de kit. Ainsi, un kit est composé de plusieurs "kit boxes" et de plusieurs pièces de grande taille. La Figure 1.15 montre les étapes principales du processus de constitution des kits pour les portes des véhicules.

1. Les chariots de kit rentrent dans la zone avec leurs kit boxes vides.
2. Des opérateurs préparent des kit boxes avec des pièces de petite taille.

## 1.7. MODÈLE À ÉVÉNEMENTS DISCRETS DU SYSTÈME D'ÉTUDE

---

3. Les boîtes sont ordonnées en fonction de la séquence de production. Un opérateur enlève les kit boxes vides et met les kit boxes pleines.
4. Les chariots continuent à avancer. Ils prennent une direction à droite ou à gauche en fonction du côté de la voiture qu'ils doivent approvisionner.
5. Les pièces de grande taille sont mises dans les chariots.
6. Les chariots partent vers la ligne de montage.



FIGURE 1.15 – Préparation des kits.

Afin de mieux appréhender le système réel, nous avons commencé par le représenter, au global, via un modèle de simulation à événements discrets.

### 1.7 Modèle à événements discrets du système d'étude

La simulation de phénomènes selon [Shannon, 1975], est le processus de conception d'un modèle d'un système réel et de réalisation d'expériences avec ce modèle dans le but soit de comprendre le

comportement du système, soit d'évaluer diverses stratégies (dans les limites imposées par un ensemble de critères) pour l'exploitation du système.

Les modèles de simulation sont une approche largement utilisée pour la conception et la planification des entrepôts logistiques [Fauadi et al., 2018]. Les systèmes de file d'attente sont l'un des aspects les plus importants à considérer lors de la simulation d'un entrepôt logistique [Bychkov et al., 2017]. Il est essentiel que le fonctionnement du système logistique permette de garantir la satisfaction des clients à coût minimal. Dans le cas du système logistique que nous étudions, le client que nous devons satisfaire est la ligne de montage véhicule.

Pour entamer le travail sur la composante virtuelle (ou cyber) de notre système, nous optons pour la simulation à événements discrets comme première stratégie de modélisation [Montoya Zapata et al., 2024c]. Chaque événement se produit à un instant particulier dans le temps et marque un changement d'état dans le système [Robinson, 2004]. Entre des événements consécutifs, aucun changement dans le système n'est supposé se produire. Cette approche a une grande capacité d'imiter la dynamique du système réel [Ingalls, 2008] et elle est donc capable de fournir un bon aperçu de la façon dont les modifications proposées pourraient affecter le système [Negahban et Smith, 2014].

Les suites les plus utilisées dans l'industrie sont Siemens Plant Simulation, Arena, Simio, Witness, AnyLogic, et Simul8 [Dias et al., 2016]. C'est ce dernier qui a été retenu pour la construction des premiers modèles pour des raisons de simplicité de prise en main, de convivialité et de coût. Un transfert des modèles vers Witness serait envisageable dans le futur afin qu'ils soient exploitables plus facilement par l'industriel.

La modélisation d'un système sous Simul8 se fait par utilisation de blocs qui correspondent à des classes d'objets : point d'entrée des produits, stock, activité, point de sortie des produits, ressources. Les paramètres de ces objets sont réglables, soit en les modifiant directement, soit en les programmant au sein de Simul8 avec son langage de programmation : Visual Logic. Celui-ci permet à l'utilisateur de mettre en place une logique plus avancée et plus précise du modèle de simulation.

Les stations de travail sont représentées par un ensemble d'activités correspondant aux tâches effectuées sur le poste. A chaque tâche peut être associée une ressource, comme un bras robotisé ou un opérateur. Il est possible d'entrer des paramètres de panne sur les ressources ou sur le bloc de l'activité lui-même. Également, le temps requis pour la réalisation d'une activité peut être contrôlée par des lois de distribution statistiques ou de conditions logiques. L'arrivée des produits peut être fixe, importée depuis un tableau externe à la simulation ou contrôlée par des lois de distribution statistiques. Les produits qui rentrent dans le système peuvent être appelés aussi des entités. Ils ont des attributs ou des caractéristiques propres à chacun comme la taille, le type, le coût, la valeur ou d'autres.

Dans le domaine du génie industriel, de nombreux problèmes peuvent être résolus en fonction du niveau de décision utilisé [Klement et al., 2017]. Certains de ces problèmes sont résumés dans le

## 1.7. MODÈLE À ÉVÉNEMENTS DISCRETS DU SYSTÈME D'ÉTUDE

---

Tableau 1.3. Dans le cadre de ces travaux de thèse, la simulation à événements discrets nous a permis de travailler sur un premier problème de niveau tactique. Néanmoins, par la suite, nous nous sommes davantage intéressés aux problèmes de niveau opérationnel (comme la réaction au problème de pièce manquante dans un kit). Ainsi, nous nous sommes plus orientés vers les systèmes multi-agents que vers la simulation à événements discrets. Le couplage entre les systèmes multi-agents et la simulation à événements discrets reste une perspective intéressante.

Les sous-sections suivantes présentent une première modélisation du système d'étude sur Simul8 et le problème étudié, qui concerne la cadence de livraisons des kits à la ligne de montage.

Niveau de décision	Horizon temporel	Problème étudié
Stratégique	Années	Dimensionnement du problème
Tactique	Mois	Planification des activités et affectation des ressources
Opérationnel	Semaines, Jours	Programmation des activités et réponse aux risques

TABLEAU 1.3 – Description des niveaux de décision et de planification [Klement et al., 2017]

### 1.7.1 Première approche de modélisation : modélisation détaillée

Comme il est décrit dans la Section 1.6.1, l'approvisionnement de pièces à la zone kitting DOS comporte cinq étapes (l'arrivée des pièces à la gare routière, la dépose des pièces sur les AGV tracteurs, le transport des pièces par l'AGV, la mise en stock tampon et la mise en stock palettier et mise dans les meubles de la zone kitting) et elle est assurée par une boucle de deux AGV et trois caristes. Ensuite, au sein de la zone kitting, il y a une deuxième boucle d'AGV qui transporte les kits qui sont constitués au fur et à mesure que l'AGV avance dans la zone. Puis, ces AGV transportent les kits jusqu'à la ligne de montage des portes.

La Figure 1.16, présente la première modélisation que nous voulions faire du système. Ici, l'intention était de modéliser le système en suivant d'une manière très détaillée le circuit des AGVs. Nous pouvons voir sur l'image en orange claire les principales zones de travail et de stationnement des pièces : la gare routière, la zone de stock tampon, la zone de stock palettié, la zone de kitting et la zone de montage des portes (zone process DOS). Ces zones sont liées par les circuits des AGVs. À droite se trouve le circuit des AGVs tracteurs, qui transportent les pièces depuis une gare routière vers la zone de stockage, et au milieu se trouve le circuit des petits AGVs, qui transportent les kits vers la zone de montage des portes. Chacun des rectangles blancs représente des tags des AGVs. Ces tags sont des étiquettes collées par terre qui transmettent des consignes aux AGVs. Les consignes peuvent être de vitesse, de priorité des AGVs en cas de croisement, d'arrêt ou de direction. Ces tags, et les différentes zones unies par des flèches noires, représentent les parcours des AGVs et les flèches bleues représentent les chemins des caristes.

L'idée avec cette modélisation était de la compléter avec des informations telles que la distance entre les tags des AGVs, la vitesse de déplacement des AGVs entre chaque tag, la distance moyenne parcourue par chaque cariste, les vitesses de déplacement des caristes, la position de stockage de chaque référence de pièce, la position des pièces au sein de la zone de kitting ainsi que le temps de prise et de dépose de chaque pièce. De cette façon, nous pourrions représenter en détail le parcours de chaque pièce et ainsi modéliser des perturbations ou des problèmes sur notre système numérique exactement au même endroit qu'ils ont lieu sur notre système réel.

Cette approche nous apportait une clarté importante en termes de la compréhension du modèle, surtout concernant les flux et les cheminements de pièces. Aspect remarquable au moment d'expliquer le modèle à des personnes externes au projet. Néanmoins, malgré celui-là et d'autres avantages, cette modélisation était trop lourde et consommatrice de temps. Également, avec la quantité d'information que nous voulions représenter, des erreurs de modélisation pourraient passer inaperçues très facilement. Pour cette raison, pour suivre la cadence de livraison des kits à la ligne de montage, nous partons sur une hypothèse de modélisation beaucoup plus simplifiée.

### 1.7.2 Deuxième approche de modélisation : modélisation simplifiée

Pour cette approche, nous divisons le processus d'approvisionnement de kits à la zone de montage des portes en trois grandes étapes (voir la Figure 1.17) : l'approvisionnement de pièces à la zone de kitting (en rouge sur l'image), la préparation et constitutions des kits (en jaune sur l'image) et la livraison des kits à la zone de montage des portes (en bleu sur l'image).

Sur cette modélisation, nous avons gardé les principaux points d'arrivée et de départ des pièces : la gare routière, le point de départ et d'arrivée des AGVs tracteurs, la zone de stock, les meubles de la zone de kitting, les AGV de la zone de kitting (comme récepteurs des pièces lors de la constitution des kits) et la zone de montage des portes (zone process). En revanche, le circuit des AGVs n'est pas si détaillé que sur la modélisation présentée avant. Ces trajets seront représentés comme des activités de transport nécessitant des ressources et une durée donnée.

L'une des caractéristiques importantes concernant la séparation de ces trois étapes c'est le type d'élément qui est transmis entre chaque unité de travail. Dans la partie d'approvisionnement de pièces à la zone kitting, ce qui est transféré ce sont des palettes de pièces. Chaque bloc blanc et chaque flèche a une capacité  $C_i$  correspondant à la capacité de pièces capable de transporter ou de conserver pour une courte période. Par exemple, la capacité des AGVs tracteurs est égal à cinq puisqu'ils sont capables de transporter cinq emballages ou palettes durant un seul trajet, tandis que la capacité d'un opérateur cariste est d'un car ils peuvent porter uniquement une seule palette par trajet.



1.7. MODÈLE À ÉVÉNEMENTS DISCRETS DU SYSTÈME D'ÉTUDE

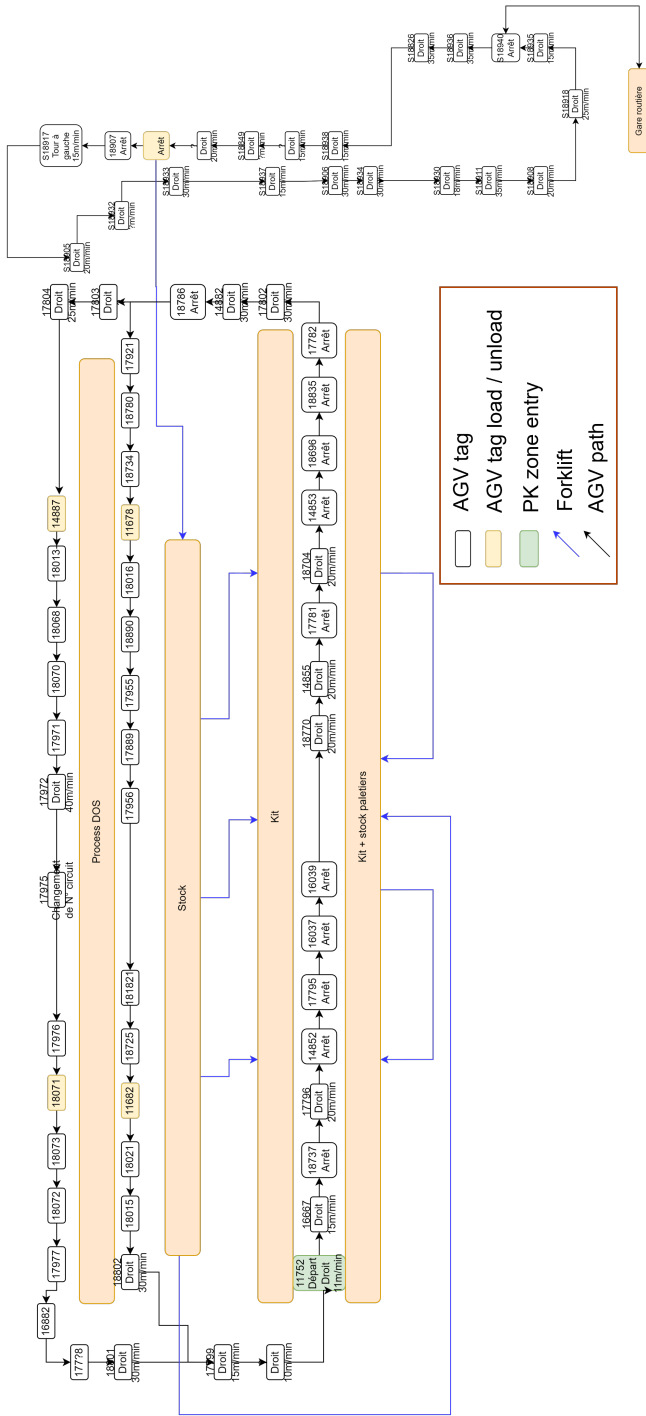


FIGURE 1.16 – Première approche de modélisation : modélisation détaillée

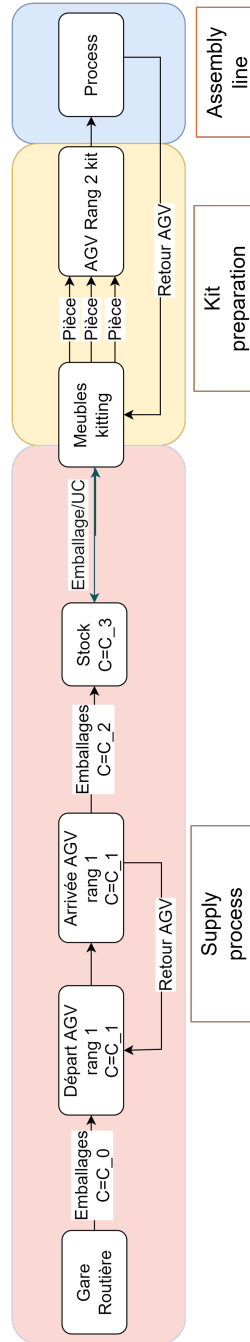


FIGURE 1.17 – Deuxième approche de modélisation : modélisation simplifiée



De la part de la zone de préparation des kits, d'abord les meubles peuvent être approvisionnés en forme de palettes notamment pour les pièces de grande taille ou en forme d'UC. Ensuite, l'élément transféré entre les meubles et les chariots de kits ce sont des pièces unitaires. Finalement, une fois le kit constitué, c'est lui qui est transféré à la zone de montage des portes par les AGV.

### 1.7.3 Implémentation sur Simul8

Le modèle que nous voulons implémenter sur Simul8 est le modèle simplifié. La Figure 1.18 montre une première modélisation du système faite sur ce logiciel. Pour celle-ci, nous avons gardé la même structure du modèle simplifié, où le processus global est divisé en trois parties : l'approvisionnement de pièces à la zone de kitting, la constitution et préparation des kits et la livraison à la zone de montage de portes.

Pour la réalisation de ce modèle nous avons utilisé différents types des blocs structurels tels que des entités, des blocs d'activités, des ressources et des variables globales. D'un point de vue global, le système compte avec trois types d'entités, deux entités physiques : les pièces et les AGV, et une entité d'information qui est créée pour donner le top de retour pour les AGV tracteurs. En revenant aux entités physiques, les pièces sont des entités qui rentrent et sortent dans le modèle, elles peuvent être appelées des entités transitoires, tandis que les AGV sont des entités internes au modèle puisqu'elles circulent dans le dit modèle sans sortir, rentrer ou augmenter leur nombre.

Ainsi, les pièces rentrent par les points d'entrée *Ref A* et *Ref B*, elles sont stockées et puis, elles passent au bloc d'activité *Transport\_Camion\_AGV\_Tracteur* qui représente le transfert des pièces de la gare routière au point de départ des AGVs tracteurs. Ensuite, un deuxième bloc d'activité représente le chargement de l'AGV tracteur qui attend le chargement de cinq palettes et la disponibilité d'un AGV pour envoyer ces éléments vers le bloc d'activité *Aller\_AGV*. Ce dernier, représente le temps de transport entre le point de départ et le point d'arrivée de l'AGV tracteur. Au point d'arrivée, les palettes sont transportées une par une par un cariste (bloc d'activité *Transport\_AGV\_Palettiers*) à la zone de stock palettiers. Une fois que le cariste a terminé de décharger l'AGV tracteur, c'est-à-dire, qu'il a transporté cinq palettes, un élément d'information est généré est envoyé à la queue d'attente du bloc *Top\_Retour* afin que l'AGV tracteur puisse effectuer son trajet de retour et arriver au point de départ (ou point de chargement). Les pièces qui sont stockées aux palettiers sont ensuite transportées par le bloc *Transport\_AGV\_Palettiers\_2* aux blocs d'attente *Meuble\_Ref\_A* et *Meuble\_Ref\_B*, qui simulent les meubles de la zone de kitting. Cette activité clôt la phase d'approvisionnement de pièces à la zone de kitting.

L'étape de préparation des kits est constituée par cinq blocs d'activité disposés en série qui représentent les opérateurs de la zone de kitting qui prennent des pièces des meubles et les transfèrent dans les chariots des kits. Le dernier bloc d'activité (*Trajet\_sortie KIT\_entrée\_process*) simule le transport

## 1.7. MODÈLE À ÉVÉNEMENTS DISCRETS DU SYSTÈME D'ÉTUDE

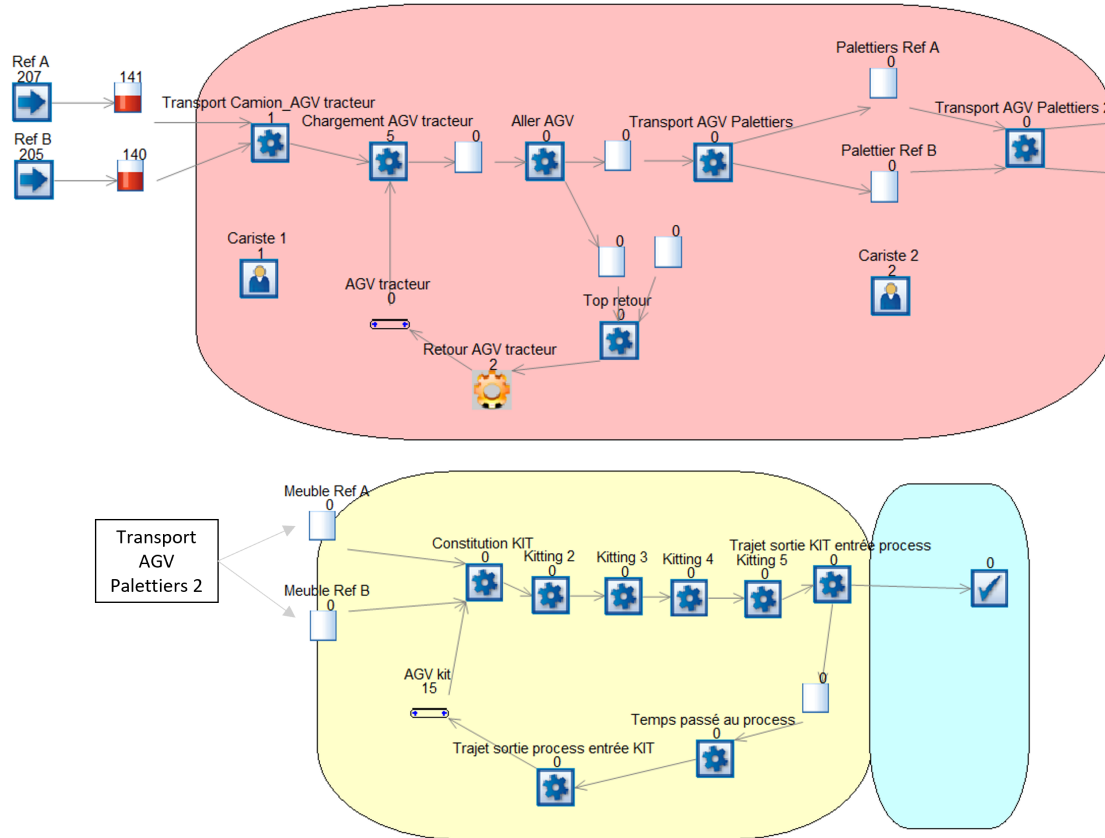


FIGURE 1.18 – Modèle à événements discrets.

des chariots des kits par les AGV à la zone de montage de portes. Ici, les pièces sortent du système et vont vers un point de sortie alors que les AGV font leur trajet de retour.

En résumé les éléments structurels et que nous avons dans cette première implémentation et leurs fonctions sont :

- Entités :
  - Les pièces (entités transitoires).
  - Les AGV séparés en deux circuits (entités internes).
  - Le top de retour des AGVs du processus d'approvisionnement de la zone de kitting.
- Points d'entrée :
  - Le point pour une référence générique A : Ref A.
  - Le point pour une référence générique B : Ref B.
- Activités :
  - Le transport des pièces vers le point de chargement des AGVs : Transport Camion AGV

Tracteur.

- Le chargement des AGVs tracteurs : Chargement AGV tracteur.
- Le transport des pièces effectué par l'AGV tracteur : Aller AGV.
- Le transport des pièces vers le stock palettiers : Transport AGV Palettiers.
- Le bloc qui attend l'information pour autoriser le retour des AGVs tracteurs : Top retour.
- Le retour des AGVs tracteurs : Retour AGV tracteur.
- Transport des pièces du stock palettiers vers les meubles de la zone de kitting : Transport AGV Palettiers 2.
- Les activités de constitution des kits : Constitution KIT, Kitting 2, Kitting 3, Kitting 4 et Kitting 5.
- Le transport des chariots de kit vers la zone de montage des portes : Trajet sortie KIT entrée process.
- L'activité simulant le temps passé par les AGV dans la zone de montage des portes : Temps passé au process.
- Le retour des AGVs de la zone de montage des portes jusqu'à la zone de kitting : Trajet sortie process entrée KIT.
- Ressources :
  - Le cariste en charge de faire le transport des palettes de la gare routière au point de chargement des AGVs tracteurs : Cariste 1.
  - Les deux caristes en charge d'amener les pièces du point d'arrivée des AGVs tracteurs jusqu'aux zones de stock et jusqu'aux meubles de la zone de kitting : Cariste 2.
- Variable globale :
  - Variable qui compte le nombre de palettes déchargées par le Cariste 2 des AGVs tracteurs pour autoriser l'AGV à partir.

### 1.7.4 Exemple d'une première application

Afin de tester et réaliser une première validation de la première partie du modèle, l'approvisionnement de pièces à la zone de kitting, nous simulons un changement de cadence de production de l'usine de 30 véhicules par heure à une cadence de 60 véhicules par heure. Ainsi la question à laquelle nous voulons répondre c'est : combien d'AGV tracteurs sont nécessaires pour couvrir le besoin de consommation de pièces dans la zone de kitting ?

Actuellement, avec une cadence de production d'environ 30 véhicules par heure, entre 190 et 200 véhicules sont produits par équipe de travail. Cela représente un besoin d'approvisionnement de 150 palettes par équipe. Avec une cadence de 60 véhicules par heure, 420 véhicules sont produits par équipe. Le besoin d'approvisionnement pour ce volume de production serait donc de 340 palettes par équipe, le Tableau 1.4 récapitule ces informations.

## 1.7. MODÈLE À ÉVÉNEMENTS DISCRETS DU SYSTÈME D'ÉTUDE

	<b>Situation actuelle</b>	<b>Situation future</b>
Cadence de production	30 véh/h	60 véh/h
Nombre de véhicules produits	190 – 200 véh	420 véh
Besoin d'approvisionnement	150 palettes	340 palettes
Nombre de palettes par AGV	5,5 palettes	5,5 palettes
Nombre d'AGV tracteurs	2 AGV	A définir

TABLEAU 1.4 – Données de production changement de cadence

<b>Opération</b>	<b>Bloc modèle Simul8</b>	<b>Début (min)</b>	<b>Durée (min)</b>	<b>Fin (min)</b>
Chargement AGV tracteur	Transport Camion AGV tracteur	0	2	2
Transfert vers zone de déchargement	Aller AGV (queue)	2	4	6
Aleas transfert	Aller AGV	6	2	8
Attente zone de déchargement	Transport AGV Palettiers	8	3	11
Déchargement AGV tracteur	Transport AGV Palettiers	11	2	13
Rechargement vide	Top retour	13	2	15
Retour AGV	Retour AGV	15	8	23
Aléas retour	Retour AGV	23	1	24
Attente déchargement	Retour AGV	24	2	26
Déchargement vide	Retour AGV	26	2	28

TABLEAU 1.5 – Durée des opérations d'une boucle des AGVs tracteurs

Pour répondre à la question, nous allons entreprendre deux méthodes. D'abord, nous allons lister les temps nécessaires pour effectuer les opérations de ce processus d'approvisionnement de pièces à la zone de kitting afin de connaître le temps total requis pour effectuer une boucle complète dans le circuit des AGVs tracteurs. Le Tableau 1.5 présente les temps en question. Ensuite, avec cette valeur de temps et une capacité de transport de 5,5 palettes par AGV nous calculerons la capacité d'approvisionnement totale par équipe de travail en nombre de palettes. Pour la deuxième méthode, nous allons utiliser le modèle implémenté sur Simul8 en conservant les mêmes valeurs de temps pour les opérations de travail et de capacité de transport des AGVs.

En utilisant la première méthode, nous pouvons voir que la somme de temps résultante pour la réalisation d'une boucle est de 28 minutes. Nous pouvons ainsi calculer facilement qu'avec 5 AGV nous avons une capacité d'approvisionnement de 413 palettes pour 7 heures de travail. Par conséquent, le besoin de 340 palettes à approvisionner est satisfait avec un engagement de 82%.

Quant au modèle sur Simul8, avec 5 AGVs dans la boucle nous avons un résultat final de 412.5 palettes livrées à la fin d'une journée de travail. Ce fait nous permet de corroborer que le modèle

## 1.7. MODÈLE À ÉVÉNEMENTS DISCRETS DU SYSTÈME D'ÉTUDE

Simul8 est cohérent avec notre calcul manuel. Néanmoins, avec ce modèle nous pouvons aller au-delà de ce calcul.

Comme il est possible de voir sur la partie droite de la Figure 1.16, un tronçon du circuit des AGVs tracteurs est très étroit (sur la partie supérieure de la figure). Sur ce tronçon il ne peut pas y avoir de croisement d'AGV, c'est-à-dire, si jamais il y a déjà un AGV engagé sur l'allée, celui qui vient en sens contraire doit attendre qu'il finisse ce tronçon du circuit. Pour modéliser ce comportement, nous avons ajouté un bloc d'activité appelé *Retour partagé* sur le circuit de retour des AGVs (voir Figure 1.19) et nous avons partagé le temps total de retour en les deux blocs de retour. Également, nous avons ajouté aussi une ressource appelée *Allée usine* qui est partagée entre le bloc *Retour partagé* et le bloc *Aller AGV*. De cette manière, nous garantissons qu'il n'y a pas deux AGV qui font l'aller et le retour au même temps sur ce tronçon du circuit.

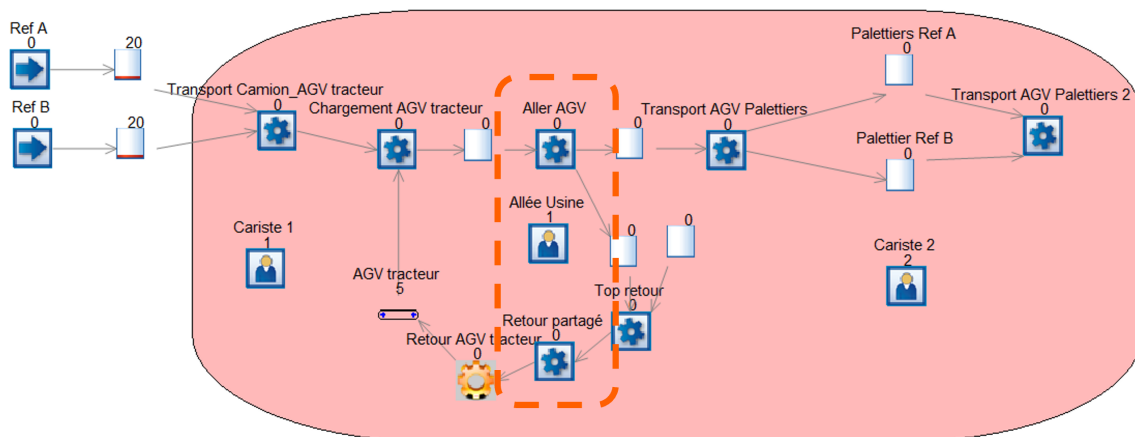


FIGURE 1.19 – Modèle à événements discrets modifié.

En lançant la simulation après avoir effectué ces modifications, dans le cas où il n'y a que deux AGV dans le circuit, le résultat est le même pour les deux modèles Simul8 (Figure 1.18 et Figure 1.19) : 181.5 palettes approvisionnées à la fin de la journée. Néanmoins, quand il y a 5 AGV dans le circuit, le résultat avec le modèle tenant compte le partage de l'allée est de 335.5 palettes livrées à la fin de la journée. Ce résultat est 19% inférieur au résultat obtenu et montre qu'avec 5 AGV les besoins de livraison de pièces ne sont pas satisfaits. Pour livrer 340 palettes dans une journée de travail, il faut au moins 6 AGV dans le circuit. D'où l'importance d'ajouter cette contrainte à la simulation quand le nombre d'AGV dans le circuit augmente.

### 1.7.5 Conclusion

La démarche que nous venons d’amener dans cette section nous permet de conclure plusieurs choses. D’abord, un système peut être modélisé de différentes façons qui prendront plus ou moins de temps. Il faut donc bien choisir le niveau de complexité voulu pour le modèle en fonction des objectifs visés et des conclusions que l’on souhaite tirer.

Deuxièmement, nous pouvons constater que ces modèles ne trouveront pas le scénario optimal pour nous. Ils permettront uniquement d’imiter le comportement dynamique du système lors de l’évaluation de différents scénarios alternatifs. Ainsi, c’est notre mission de proposer plusieurs scénarios possibles afin de découvrir le plus juste pour l’application donnée.

Troisièmement, le cas du partage d’une allée d’un site de production montre l’importance qui a l’intégration de ce type de contraintes dans la modélisation d’un système. Également, cela affirme la pertinence de Simul8 pour modéliser ce genre d’interactions que nous ne pouvons pas faire manuellement.

## 1.8 Plan du manuscrit

Le reste de ce manuscrit est divisé en quatre chapitres, selon l’organisation suivante :

Le Chapitre 2 présente un état de l’art en se focalisant sur les deux concepts clés de cette thèse : l’intelligence collective et les systèmes multi-agents, et sur leurs applications. De plus, ce chapitre décrit l’architecture que nous proposons afin de modéliser un système logistique interne comme un système multi-agents. Les trois niveaux de l’architecture sont détaillés ainsi que son intégration sur le terrain industriel.

Le Chapitre 3 propose une démarche pour évaluer la méthode multi-agents proposée, non seulement sur un scénario ponctuel mais sur l’ensemble des scénarios qui ont lieu au cours d’une journée de production type au sein d’une usine d’assemblage automobile.

Le Chapitre 4 est dédié à l’explication de la stratégie d’agrégation employée pour constituer un data set à partir des données collectées, grâce aux systèmes d’informations suivant le processus de logistique interne. Ce chapitre traite également l’entraînement et l’évaluation des modèles de machine learning pour la détection, de manière anticipée, de l’anomalie de pièce manquante dans un kit.

Le Chapitre 5 présente l’application des méthodes proposées dans les chapitres précédents sur un nouveau cas d’étude, ayant lieu au sein du processus de la tôlerie. Dans ce chapitre, une méthode multi-agents pour prédire les arrêts des stations de travail faisant partie de ce processus est proposée.

Finalement, les conclusions générales des travaux entrepris ainsi que la présentation de travaux futurs concluent ce mémoire.

## 1.8. PLAN DU MANUSCRIT

---

## Chapitre 2

# Modélisation multi-agents : état de l'art et proposition

### Contenu

---

2.1	Introduction . . . . .	<b>52</b>
2.2	Cadre théorique et revue de la littérature . . . . .	<b>52</b>
2.2.1	L'intelligence artificielle distribuée . . . . .	52
2.2.2	Agents . . . . .	54
2.2.3	Apprentissage des systèmes multi-agents . . . . .	56
2.2.4	Outils et plateformes de développement et de simulation de systèmes multi-agents . . . . .	57
2.2.5	Applications des systèmes multi-agents . . . . .	58
2.3	méthode de modélisation multi-agents proposée . . . . .	<b>62</b>
2.3.1	Architecture du système multi-agents . . . . .	62
2.3.2	Protocole d'interaction global entre les agents opérateurs et les agents kits . . . . .	63
2.3.3	Modèle de comportement de l'agent kit . . . . .	64
2.3.4	Méthodologie d'estimation des coûts . . . . .	67
2.3.5	Méthodologie de collecte de données et d'identification de l'état du système . . . . .	67
2.4	Conclusion . . . . .	<b>71</b>

---



### 2.1 Introduction

La première partie de ce chapitre fournit une vue synthétique de l'état de l'art sur un des concepts clés de cette thèse : les systèmes multi-agents. Dans un premier temps, il présente les systèmes multi-agents comme une discipline issue de l'intelligence artificielle distribuée. Ensuite, le concept d'agent et les types d'agents sont définis. Puis, deux revues systématiques de la littérature sont discutées autour des applications et des environnements de développement de ces systèmes. Enfin, la dernière partie présente les conclusions tirées de cette étude. Cet état de l'art apporte un référentiel académique pour le positionnement de ces travaux de recherche et permet au lecteur d'avoir une meilleure compréhension de la vaste discipline des systèmes multi-agents.

Dans la deuxième partie de ce chapitre, nous proposons une méthode basée sur les systèmes multi-agents, afin de réduire le temps de réaction à une alerte de *pièce manquante* déclenchée par un opérateur sur une ligne d'assemblage, utilisant le processus de kitting. La méthode proposée exploite les propriétés de la modélisation basée sur les agents, pour fournir des solutions dynamiques dans une approche décentralisée. Les travaux antérieurs sur les systèmes multi-agents de fabrication ont fortement inspiré la méthode proposée, en particulier les architectures multi-agents GRACE [Leitão et Rodrigues, 2011], PROSA [Van Brussel et al., 1998] et ADACOR [Leitão et Restivo, 2006].

### 2.2 Cadre théorique et revue de la littérature

#### 2.2.1 L'intelligence artificielle distribuée

La recherche dans le domaine de l'Intelligence Artificielle (IA) a débuté dans les années 1950. Depuis lors, le domaine de l'IA a connu plusieurs périodes d'intérêt et de déception. Cependant, l'enthousiasme et le financement ont récemment repris. La plupart des premières applications de l'IA concernent ce que l'on appelle l'IA symbolique. Elle se concentrait principalement sur des domaines tels que la résolution de problèmes, la reconnaissance de formes et le raisonnement [Garnelo et Shanahan, 2019]. Toutefois, cette approche a ensuite été complétée par une approche de modélisation cognitive. Elle s'est intéressée aux théories de l'intelligence liées aux réseaux de neurones. Elle a eu du succès dans les problèmes de reconnaissance de formes et de tâches d'apprentissage [Singh et Gupta, 2009].

Plus tard, dans les années 1970, l'Intelligence Artificielle Distribuée (IAD) a émergé avec l'objectif de résoudre des problèmes dans des domaines où plusieurs entités ou individus sont physiquement ou logiquement distribués en termes d'information, d'expertise ou de ressources [Alonso, 2002]. La résolution des problèmes de cette nature implique le partage et l'échange de plusieurs avis. L'IAD cherche à proposer une conception modulaire de solutions avec une exécution distribuée et coopérative [Amblard et al., 2021].

L'IAD a principalement trois axes de travail : l'IA parallèle, la résolution coopérative de problèmes et les systèmes multi-agents.

### 2.2.1.1 L'IA parallèle

Cette branche vise les problèmes qui peuvent être décomposés en sous-problèmes indépendants, dont le résultat final est simplement la composition de tous les sous-résultats. Cette approche permet de gagner en performance au niveau du temps de calcul ou de la taille des problèmes à résoudre, grâce à l'avancement de l'informatique et l'usage de machines multiprocesseurs [Amblard et al., 2021].

### 2.2.1.2 La résolution coopérative de problèmes

Tout comme l'IA parallèle, cette discipline cherche à résoudre des problèmes décomposables en sous problèmes indépendants. Néanmoins, la particularité est que les agents affectés à la recherche des solutions pour les sous problèmes ont été conçus pour être coopératifs afin d'atteindre un même objectif global. Ici, des techniques de répartition de tâches et d'échange d'informations sont utilisées afin d'atteindre une convergence robuste vers le but global [Durfee et Rosenschein, 1994].

### 2.2.1.3 Les systèmes multi-agents

Les systèmes multi-agents sont des systèmes composés d'agents autonomes ayant des objectifs individuels qui sont en interaction dans un même environnement. Ces agents peuvent coopérer ou entrer en conflit pour atteindre leurs propres objectifs [Tuyls et Weiss, 2012]. A la différence des systèmes de résolution coopérative de problèmes, un système multi-agents peut être intégré par des agents réalisés par des concepteurs différents et qui peuvent préexister, indépendamment des autres, dans un système donné avant d'intégrer le système multi-agents. Une fois dans un système multi-agents, les agents partagent aussi un objectif global. Néanmoins, l'autonomie propre à chaque agent peut générer des difficultés mettant en péril l'atteinte de cet objectif. A cet effet, il est nécessaire de mettre en place des modes de régulation et de coordination venant de la sociologie ou de la théorie des organisations. En outre, certaines de ces techniques de coordination permettant la coopération de différents agents sont inspirées aussi de **l'intelligence collective** de certaines espèces dans la nature [Durfee et Rosenschein, 1994].

Les challenges actuels tels que la conduite automatisée, le contrôle distribué des feux de circulation et la coordination de grands essaims de robots, ainsi que les applications web comme Wikipedia, Amazon et YouTube ont une nature distribuée et impliquent des degrés élevés d'incertitude [Verbraeken et al., 2020]. Ainsi, les agents doivent prendre en compte des environnements dynamiques et changeants. Afin de résoudre ce type de problèmes, ils doivent apprendre de manière autonome à partir de

l'expérience [Weiss, 2013].

### 2.2.2 Agents

#### 2.2.2.1 Agent

Un agent est une entité capable de percevoir, de prendre des décisions et de réaliser des actions en vue d'atteindre ses objectifs [Wooldridge, 2009]. La perception permet à l'agent d'acquérir des informations de son environnement. Son environnement est composé et défini par les autres agents et objets qu'il peut sentir à travers ses capteurs, et avec qui il peut interagir et ou communiquer. Basé sur les informations acquises, l'agent est en mesure de mettre à jour sa perception de l'environnement et de prendre des décisions par rapport aux actions qu'il doit réaliser. C'est cette phase de prise de décision et d'adaptation des actions en fonction de ses objectifs et de l'état de l'environnement, qui conduit à cataloguer les agents en tant qu'entités intelligentes.

C'est également cette autonomie et cette pro-activité quant aux choix de ses actions qui différencie un agent d'un objet. Du point de vue de l'ingénierie logicielle, un objet est une entité qui modifie son comportement suite à des envois de messages tandis qu'un agent est une entité agissante qui est à l'origine de ses propres actions.

Grâce aux caractéristiques mentionnées précédemment, les agents d'un système multi-agents sont capables d'interagir pour s'auto-organiser, auto-adapater et de faire émerger des régularités macroscopiques à partir de leurs comportements individuels.

#### 2.2.2.2 Types d'agents

Conceptuellement, il est possible de classifier les agents en deux types, en fonction de leur capacité de raisonnement : les agents réactifs et les agents cognitifs. Cependant, les agents existants se situent sur un continuum entre ces deux concepts.

#### **Agents réactifs**

Les agents réactifs se caractérisent pour avoir un comportement réflexe suite à leurs perceptions, de type stimulus/action.

Les systèmes multi-agents composés d'agents réactifs sont généralement de grande taille. Ils peuvent avoir des milliers ou des millions d'agents et vu que leur degré d'intelligence est restreint, la cible avec ces agents est d'avoir l'émergence d'un comportement collectif intelligent. La Figure 2.1 représente le cycle de contrôle d'un agent de ce type.

```
/*cycle de contrôle d'un agent réactif*/  
Variables  
rules : règles de type condition-action  
percepts : ensemble de perceptions  
Traitement  
Répéter {  
  percepts := observer(); /*perception de l'environnement*/  
  state := interpréter_perceptions(percepts); /*mise à jour  
  de son état interne*/  
  rule := choix_règle(state,rules); /*choix d'une règle exécutable*/  
  exécuter(rule[action]); /*exécution de la partie action de la règle*/  
} Indéfiniment ;
```

FIGURE 2.1 – Cycle de contrôle d'un agent réactif [Amblard et al., 2021]

### Agents cognitifs

Les agents cognitifs sont lesdits agents intelligents. Ils ont des objectifs clairs. Grâce à leur interaction avec l'environnement et leur perception de cet environnement, ils sont capables de planifier leur comportement. Ils sont aussi capables d'apprendre des expériences passées.

Étant donné le degré de complexité important de ces agents, les systèmes multi-agents composés d'agents cognitifs ont une taille plus réduite que ceux composés d'agents réactifs. Cependant ils sont plus autonomes pour effectuer des tâches. Le nombre d'agents est de l'ordre de dizaines ou de centaines.

Les agents cognitifs peuvent être classifiés en deux groupes, les agents rationnels et les agents intentionnels. Les agents rationnels vont chercher en permanence à faire améliorer leur état, c'est-à-dire, à se rapprocher constamment de leurs buts. Leur objectif est de maximiser leur performance. Cette caractéristique engendre que les systèmes multi-agents composés par des agents rationnels sont souvent de nature compétitive. D'un autre côté, selon [Rao et Georgeff, 1991] les agents intentionnels sont dotés de trois propriétés mentales : croyance, but et intention (Belief, Desire et Intention). Ils sont aussi connus comme des agents BDI. Belief représente l'image que l'agent a de son environnement ; Desire, les buts des agents et Intention, le choix des actions à être effectuées par l'agent afin de mettre en place ses plans. Dans [Georgeff et Lansky, 1987] Georgeff et Lansky représentent la base du comportement de ces agents comme présenté par la Figure 2.2.

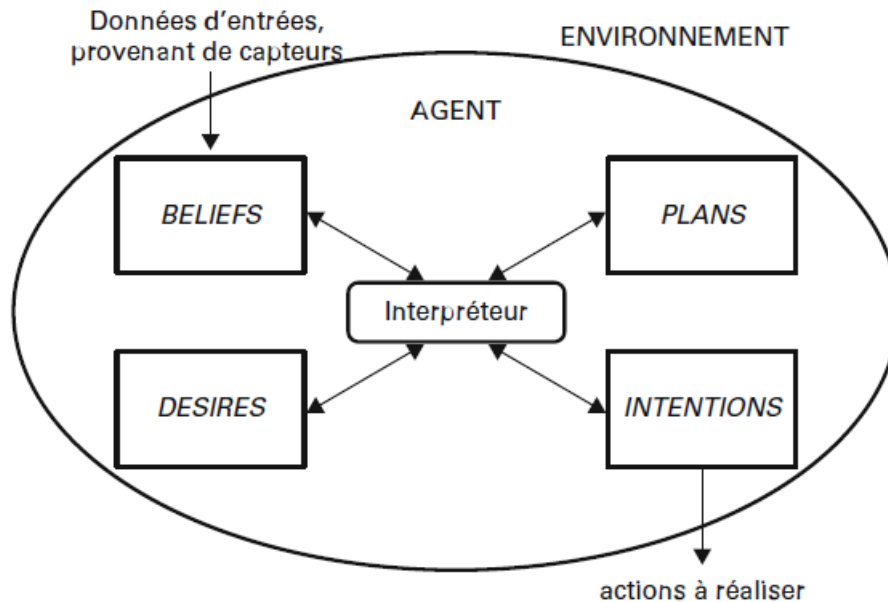


FIGURE 2.2 – Architecture du Procedural Reasoning System [Georgeff et Lansky, 1987], base des architectures BDI [Amblard et al., 2021]

### 2.2.3 Apprentissage des systèmes multi-agents

L'apprentissage multi-agents est le domaine qui intègre les techniques d'apprentissage automatique dans les systèmes multi-agents [Stone, 2007]. Comme évoqué précédemment, un agent doit être capable de percevoir son environnement, de s'y adapter et d'interagir efficacement avec les autres agents. L'apprentissage automatique permet d'entraîner les agents à réaliser cela d'une manière progressive et dynamique.

Dans la littérature, il y a trois grands types d'apprentissage : l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement. L'apprentissage supervisé consiste à entraîner un algorithme sur la base d'exemples annotés sous forme de couple : entrée, sortie désirée. Lors d'un apprentissage non supervisé, l'algorithme s'entraîne sur des données qui ne sont pas forcément annotées mais qu'il cherche à classer automatiquement selon les caractéristiques des entrées. Pour sa part, l'apprentissage par renforcement guide l'entraînement d'un agent à partir d'une récompense ou pénalité en fonctions des actions menées par l'agent.

Les algorithmes d'apprentissage par renforcement sont les plus étudiés dans le domaine des systèmes multi-agents. Leur théorie est basée principalement sur le cadre des jeux stochastiques ou jeux de Markov [Littman, 1994]. L'objectif est de créer des agents capables de s'adapter, étant donné que chaque agent est confronté à un problème de cible dynamique : "ce qui doit être appris par un agent

dépend de ce qui a été appris par les autres agents et change en fonction de cela” [Tuyls et Weiss, 2012]. Récemment, des publications comme [Nguyen et al., 2020] montrent l’intérêt de la communauté de recherche pour l’utilisation des techniques d’apprentissage profond pour compléter les techniques d’apprentissage par renforcement.

En fonction de l’objectif global du système multi-agents, au niveau collectif, l’apprentissage par renforcement peut être coopératif, compétitif ou mixte. Dans le cas coopératif tous les agents ont la même fonction de récompense, l’idée étant de maximiser la récompense en tant que groupe. Dans le cas compétitif, les agents ont pour but de maximiser leur récompense individuelle.

### 2.2.4 Outils et plateformes de développement et de simulation de systèmes multi-agents

Les développements théoriques des systèmes multi-agents ont été suivis par des développements en termes d’outils et de plateformes permettant d’accueillir et de concevoir des applications basées sur le paradigme multi-agents, notamment sur le paradigme des agents BDI. La différence entre outil et plateforme de développement se base sur les objectifs et les publics visés. Les outils de développement ont une flexibilité et une richesse plus importantes mais ils sont utilisés surtout par des informaticiens. Les plateformes, pour leur part, cherchent plutôt à proposer des alternatives pour la modélisation rapide des systèmes multi-agents par des experts d’autres domaines.

Les revues systématiques de la littérature [Müller et Fischer, 2014], [Falco et Robiolo, 2019] sur les systèmes multi-agents permettent d’identifier que JADE est la plateforme la plus utilisée pour la modélisation et simulation de systèmes multi-agents. La Figure 2.3, reproduite de [Müller et Fischer, 2014], met en évidence que la plupart des applications analysées dans l’étude n’utilisent pas de plateforme ou d’outil dédié aux systèmes multi-agents. Néanmoins, pour les applications qui en utilisent, il est possible d’observer que JADE et WADE sont les plateformes les plus utilisées, autant pour la totalité des applications comme pour celles à haute maturité selon l’échelle TRL de la NASA (ayant un TRL8 ou TRL9). En plus, il faut préciser que WADE est une extension de JADE. La supériorité de JADE est aussi confirmée par [Falco et Robiolo, 2019]. Ici JADE est utilisée sur 54% des applications parmi les 64 qui ont fourni des informations concernant leur environnement de travail.

Un autre aspect remarquable de la Figure 2.3 est la quantité élevée d’applications qui utilisent d’autres plateformes : 20 applications ont utilisé une plateforme différente chacune. Ce fait montre que même si JADE est la plateforme la plus utilisée, il y a une grande dispersion quant au nombre de plateformes utilisées et au nombre d’applications qui créent ou utilisent leurs propres environnements de développement.

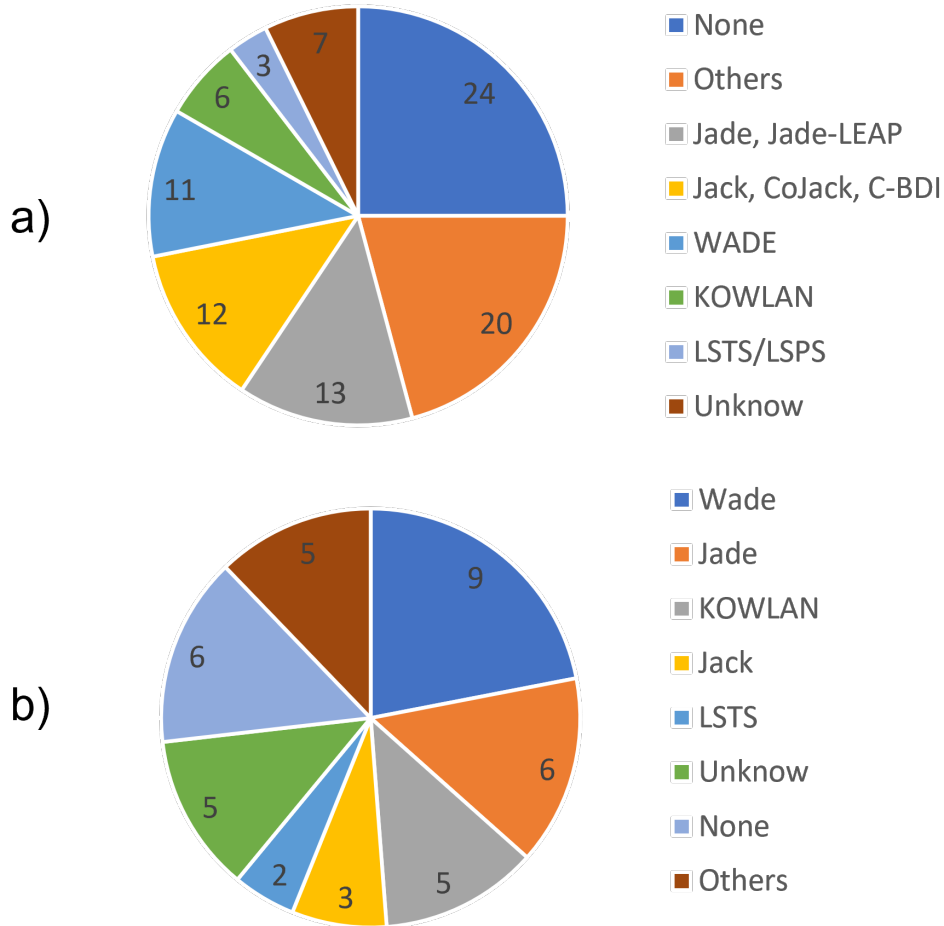


FIGURE 2.3 – Utilisation de plateformes pour a) l’ensemble des applications et b) pour les applications à haute maturité [Müller et Fischer, 2014]

### 2.2.5 Applications des systèmes multi-agents

Nombreuses sources de la littérature mentionnent que les systèmes multi-agents sont utilisés dans un grand nombre de domaines et de types d’applications. [Amblard et al., 2021] présente 6 grandes familles d’application de ces systèmes : la simulation de systèmes complexes, l’étude théorique de dilemmes sociaux, la modélisation à base d’agents, la résolution collective de problèmes, les agents logiciels et des problèmes intégrant l’humain dans la boucle.

Pour avoir une vision plus précise des secteurs d’application dans l’industrie, nous allons nous baser à nouveau sur les deux revues systématiques de la littérature [Müller et Fischer, 2014], [Falco et Robiolo, 2019]. Dans la première, Müller et Fischer analysent 152 applications publiées entre l’année





## 2.2. CADRE THÉORIQUE ET REVUE DE LA LITTÉRATURE

---

D'autre part, l'étude [Falco et Robiolo, 2019] confirme l'importance du secteur de la logistique externe et de la production industrielle comme secteur d'application des systèmes multi-agents. Dans leur classification, ce secteur se situe à la deuxième place quant à la modélisation et simulation et aussi quant au nombre d'implémentations réelles existantes. Un autre élément pertinent que cette étude révèle, c'est qu'uniquement 22,56% des articles analysés présentent des cas d'études réels et que les 77,44% restants ne présentent que des simulations, évaluations ou des comparaisons à travers des cas d'études contrôlés. En outre, la plupart de ces études ont été menées par le secteur académique avec 82%, alors que des études coopératives entre l'industrie et l'académie se limitent à 12% et ceux sans présence de l'académie à 6%. Ainsi, la création de liens entre l'académie et l'industrie est nécessaire pour augmenter le nombre d'applications réelles dans le domaine des systèmes multi-agents.

Par ailleurs, le Tableau 2.1 rapporte une liste de publications, majoritairement plus récentes que les études précédentes, présentant des applications des systèmes multi-agents. Les publications sont classifiées par type d'industrie, ou secteur d'activité, et met en évidence s'il s'agit d'une application d'un cas réel ou s'il s'agit des cas d'études contrôlés ou des simulations. L'étude de ces articles nous permet de remarquer certaines tendances sur les domaines des jeux vidéo et de la fabrication ou de la production industrielle et du transport et la gestion du trafic, qui sont les deux domaines prédominants dans la liste. Également, des observations des études précédentes peuvent être confirmées.

Les algorithmes les plus avancés et les plus sophistiqués dans l'univers des systèmes multi-agents se trouvent, généralement, dans les jeux vidéo. Des jeux populaires comme cache-cache [Baker et al., 2020] et capturer le drapeau [Jaderberg et al., 2019], développés respectivement par Open AI et Deepmind, en sont des exemples. Dans ces deux jeux, il y a deux équipes composées de plusieurs agents qui doivent coopérer pour gagner la partie contre l'équipe adverse. Une application similaire peut être trouvée dans [Movahedi et Bastanfard, 2021] où les algorithmes d'apprentissage par renforcement sont implémentés dans un contexte de jeu de Polo.

En ce qui concerne les applications dans le domaine des systèmes production ou de fabrication industrielle, nous pouvons identifier deux branches différentes : les systèmes multi-agents homogènes et les systèmes multi-agents hétérogènes [Dorri et al., 2018]. La première concerne les systèmes composés de plusieurs agents ayant les mêmes caractéristiques et des objectifs similaires. Dans ce cas, nous pouvons trouver des applications comme le contrôle robotique mobile [Tigli et Thomas, 1994], [Sanogo et al., 2023], l'apprentissage des robots par démonstration [Papadopoulos et al., 2021] et la surveillance et le diagnostic d'une flotte de trains [Le Mortellec et al., 2013], [Fadil et al., 2019].

La seconde concerne les systèmes composés d'agents, où chacun a des caractéristiques différentes des autres. L'utilisation des systèmes multi-agents dans ce contexte permet de traiter des problèmes tels que l'ordonnancement en ligne dans les usines intelligentes [Trentesaux et al., 2013], [Tonelli et al., 2016], [Rodrigues et al., 2018], [Zhou et al., 2021], [Firme et al., 2023], l'organisation de systèmes

## 2.2. CADRE THÉORIQUE ET REVUE DE LA LITTÉRATURE

Référence	Industrie										Cas réel?	
	Aérospatiale	Energie	Production	Jeux	Robotique	Software	Transport	Oui	Non			
[Le Mortellec et al., 2013]							X	X				
[Rodrigues et al., 2013]			X					X				
[Trenteseaux et al., 2013]			X								X	
[Adam, 2015]			X								X	
[Barbosa et al., 2015]			X								X	
[Balbo et al., 2016]							X				X	
[Block et al., 2016]			X								X	
[Tonelli et al., 2016]			X								X	
[Güret et al., 2017]			X					X			X	
[Xie et Liu, 2017]		X										
[Rodrigues et al., 2018]			X					X				
[Fadil et al., 2019]												
[Jaderberg et al., 2019]				X				X				
[Baker et al., 2020]				X								
[Daoud et al., 2021]										X	X	
[Movahedi et Bastanfard, 2021]				X							X	
[Vogel-Heuser et al., 2021]			X								X	
[Papadopoulos et al., 2021]					X						X	
[Zhou et al., 2021]			X								X	
[Zhu et al., 2021]								X				
[Leitão et al., 2022]			X								X	
[Piardi et al., 2022]			X								X	
[Picard, 2022]										X	X	
[Tchappi et al., 2022]										X	X	
[Firne et al., 2023]			X								X	
[Karaduman et al., 2023]			X								X	
[Piardi et al., 2023]			X								X	
[Picard, 2023]	X										X	
[Sanogo et al., 2023]			X								X	
[Torballi et Alpan, 2023]			X								X	

TABLEAU 2.1 – Liste d'autres publications d'applications des systèmes multi-agents analysées

## 2.3. MÉTHODE DE MODÉLISATION MULTI-AGENTS PROPOSÉE

---

de production reconfigurables [Adam, 2015], [Barbosa et al., 2015], l'organisation et l'ordonnement dans le périmètre de la logistique externe [Torbalı et Alpan, 2023], ainsi qu'au sein d'entrepôts logistiques [Piardi et al., 2022], [Piardi et al., 2023]. De plus, des applications des MAS dans la modélisation des systèmes de fabrication pour améliorer les contrôles de qualité [Rodrigues et al., 2013], [Leitão et al., 2022] et dans le contrôle des processus de production [Giret et al., 2017], [Block et al., 2016], [Karaduman et al., 2023] peuvent être trouvées.

Dans le domaine du transport et de la gestion du trafic, nous pouvons identifier des applications telles que la coordination de trajectoires de véhicules aériens [Picard, 2022], l'allocation de ressources dans le transport à la demande avec des véhicules autonomes [Daoud et al., 2021] et la modélisation, et gestion, du trafic routier [Balbo et al., 2016], [Tchappi et al., 2022].

Finalement, en regardant le nombre de publications qui présentent des cas d'étude réels, il est possible de voir que la plupart des articles concernant l'industrie informatique ou les jeux vidéo rentrent dans cette catégorie. Tandis que, comme il a été déjà révélé par [Falco et Robiolo, 2019], le nombre d'applications sur des cas réels dans le domaine des systèmes production est réduit, par rapport aux cas d'étude menés dans des environnements contrôlés. Le même constat peut être appliqué au secteur du transport.

## 2.3 méthode de modélisation multi-agents proposée

Les efforts de ce travail se concentrent sur le développement d'un système, basé sur des agents, qui propose des solutions potentielles aux opérateurs lorsqu'ils sont confrontés à une perturbation due à une pièce manquante dans un kit sur une ligne d'assemblage automobile.

### 2.3.1 Architecture du système multi-agents

Nous modélisons le processus d'approvisionnement de pièces sous la forme d'un MAS, comme illustré à la Figure 2.5. Cette figure représente une architecture à trois niveaux composée d'agents physiques et virtuels. Le premier niveau, en bas de la figure, représente le niveau du processus de production, composé d'agents physiques du domaine industriel. Le deuxième niveau est la représentation numérique du premier niveau. Chaque agent du premier niveau a son agent numérique correspondant dans le deuxième niveau. Enfin, le troisième niveau est le niveau de la prise de décision. Ici, un agent est chargé de centraliser les informations envoyées par les agents du deuxième niveau et de communiquer des propositions aux opérateurs. Une description plus détaillée de cette architecture peut être trouvée dans [Montoya Zapata et al., 2023].

Cette architecture considère un réseau d'agents intelligents capables d'interagir selon une approche basée sur les services. Chacun des agents considérés fournit un ensemble de services. Ces services sont

### 2.3. MÉTHODE DE MODÉLISATION MULTI-AGENTS PROPOSÉE

partagés à travers un facilitateur de répertoire (DF : Directory Facilitator), qui fonctionne comme un “répertoire des pages jaunes” [Gamboa Quintanilla et al., 2016]. Ainsi, tous les agents peuvent avoir accès à l’ensemble des services disponibles dans le système global.

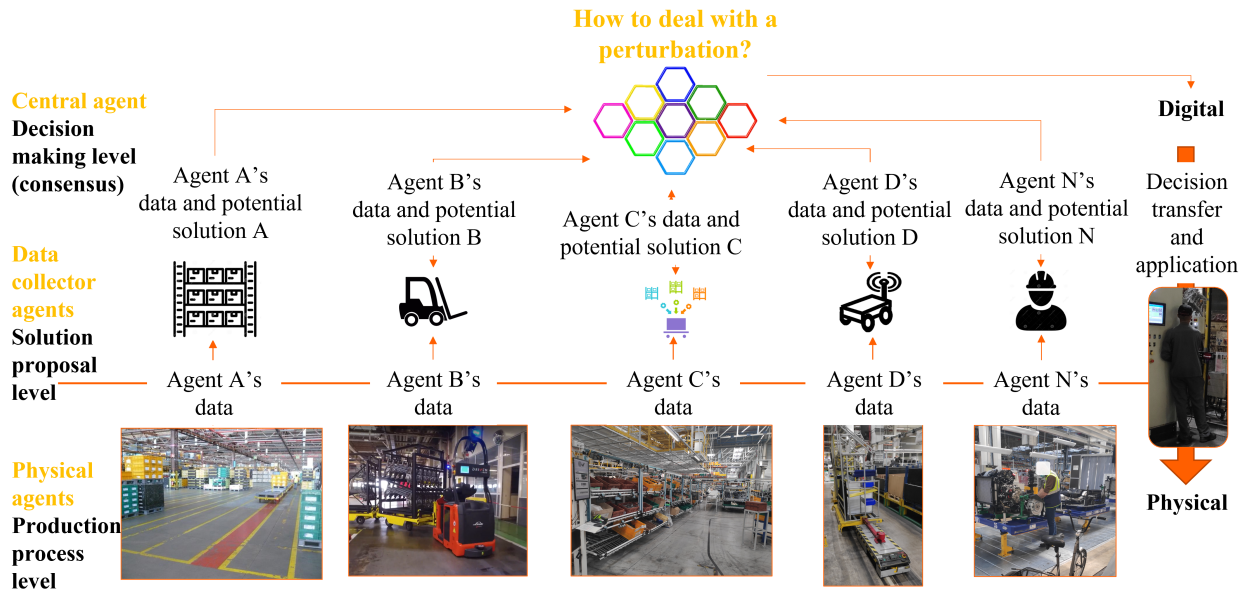


FIGURE 2.5 – Architecture proposée du système multi-agents du processus d’approvisionnement de pièces dans une ligne d’assemblage automobile [Montoya Zapata et al., 2023].

Dans ce chapitre, le système est composé des agents suivants : les opérateurs de la ligne d’assemblage et les opérateurs de la zone de préparation de kits en tant qu’agents de ressources (RA : Resource Agent), et les kits en tant qu’agents de produits (PA : Product Agent), en utilisant le formalisme de [Leitão et al., 2022]. Ici, nous faisons référence aux agents de kits comme la composition des chariots de kits et des AGV utilisés pour leur transport.

#### 2.3.2 Protocole d’interaction global entre les agents opérateurs et les agents kits

L’un des objectifs de ces travaux est d’intégrer cette méthode dans la procédure actuelle utilisée pour réagir au problème de pièce manquante sur le terrain industriel. Pour cette raison, nous voulons que les opérateurs jouent un rôle actif dans le processus et nous voulons changer le moins possible le processus actuel.

Actuellement, lorsqu’une pièce manque dans l’un des kits, le problème est souvent identifié par l’un des opérateurs sur la ligne d’assemblage. A ce stade, l’opérateur déclare le problème par le biais du système d’alerte DWS. Pour résoudre le problème, un opérateur doit identifier la référence de la pièce manquante et se rendre dans la zone de préparation des kits pour récupérer la bonne pièce. Étant

donné que plusieurs types de véhicules sont assemblés sur la même ligne de production, l'identification de la référence et la récupération de la pièce peut être une tâche longue et générer des arrêts de production.

L'approche proposée pour améliorer ce processus est illustrée par la Figure 2.6. Ici, le déclencheur pour activer le MAS est la déclaration d'alerte par l'opérateur de la ligne d'assemblage. Afin d'initialiser le MAS, il est nécessaire d'identifier l'état actuel du système étudié. Cet état peut être défini grâce aux systèmes d'information présents sur le terrain industriel. À partir du système d'alerte, le numéro d'identification et la position sur la ligne de production du kit défectueux ( $K_D$ ) sont identifiés. À partir du système MES, il est possible de déterminer la séquence de production et d'identifier le type ou la diversité des kits en production en cours. De plus, à partir du serveur AGV, les positions des AGV sont identifiées.

Grâce à ces informations, les fichiers d'entrée sont générés et le MAS est initialisé. Ainsi, un agent kit est créé pour chaque kit présent sur la ligne de production au moment actuel. Chaque agent kit connaît son type, sa position et son état de qualité (défectueux ou non défectueux). L'agent kit  $K_D$  prend le rôle d'initiateur et informe les autres agents kit du problème. Les autres agents kit prennent le rôle de participants. La phase de négociation commence.

Pendant la phase de négociation, l'agent kit initiateur envoie un appel à propositions (CPF : Call For Proposals) à tous les agents kit participants, afin de vérifier s'ils ont la pièce manquante. Chaque agent participant répond par une réponse positive ou négative. Si un agent kit participant possède la pièce manquante, il fournit également le coût associé à la récupération de cette pièce. Le détail de ce coût sera donné en Section 2.3.4.

Une fois que l'agent kit initiateur a reçu toutes les propositions des agents kit participants, il les évalue et présente les  $n$  solutions les plus adaptées à l'opérateur de la ligne d'assemblage. Chaque solution proposée spécifie la position de l'agent kit où l'opérateur peut récupérer la pièce manquante ainsi que son coût associé. Ensuite, l'opérateur de la ligne d'assemblage, en fonction du contexte actuel du système, décide d'accepter ou de rejeter les solutions données.

Parallèlement, l'agent kit le plus proche de la sortie de la zone de préparation des kits ( $K_E$ ) est identifié. En cas d'acceptation par l'opérateur de la ligne d'assemblage pour l'une des propositions présentées, l'opérateur de la zone de préparation des kits est invité à mettre la pièce manquante dans le kit  $K_E$ . Ainsi, l'opérateur de la ligne d'assemblage peut compléter le kit  $K_D$  en prenant la pièce manquante du kit candidat choisi ( $K_C$ ), puis compléter le kit  $K_C$  avec la pièce fournie par le kit  $K_E$ .

### 2.3.3 Modèle de comportement de l'agent kit

Dans la méthode proposée, les agents kit jouent un rôle actif dans la résolution de la perturbation de la ligne d'assemblage. L'interaction entre les agents kit suit une variation du protocole Contract-

### 2.3. MÉTHODE DE MODÉLISATION MULTI-AGENTS PROPOSÉE

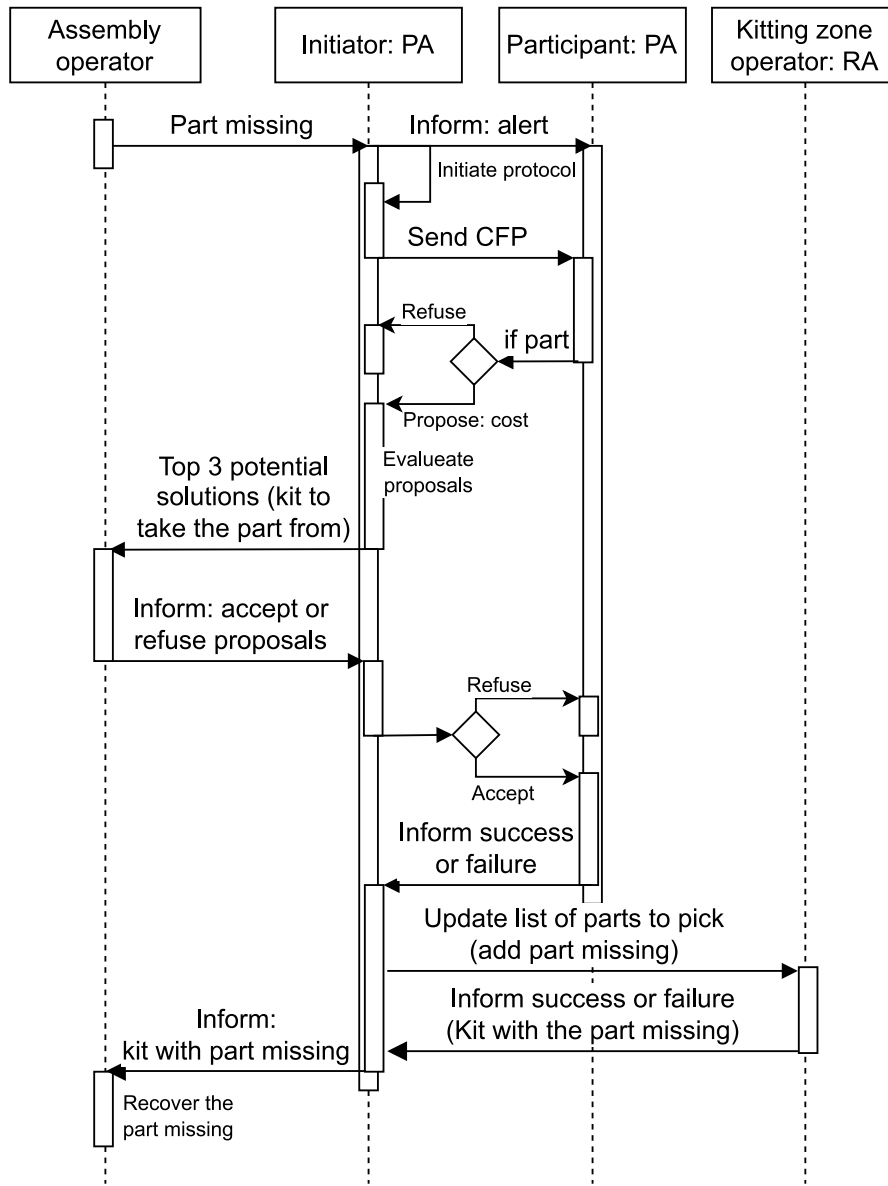


FIGURE 2.6 – Protocole d’interaction global entre les opérateurs et les agents kits [Montoya Zapata et al., 2024a]

Net. Il s’agit d’un protocole de communication utilisé dans les MAS pour gérer l’allocation des tâches et la négociation entre les agents. Dans ce protocole, un agent prend le rôle d’initiateur. Cet agent propose une tâche à un ou plusieurs autres agents. Ces derniers ont le rôle de participants. Ils font une proposition à l’agent initiateur afin d’effectuer la tâche. Ainsi, l’agent initiateur peut attribuer la tâche en cherchant à optimiser une fonction qui définit la tâche. Cette caractéristique est généralement représentée comme le coût [Smith, 1980].

### 2.3. MÉTHODE DE MODÉLISATION MULTI-AGENTS PROPOSÉE

Dans notre modèle, les agents kit peuvent être soit des initiateurs, soit des participants. La figure 2.7 présente le modèle global de comportement des agents kit sous forme de réseau de Petri (côté gauche de l'image) et met l'accent sur l'état du participant du réseau de Petri (côté droit).

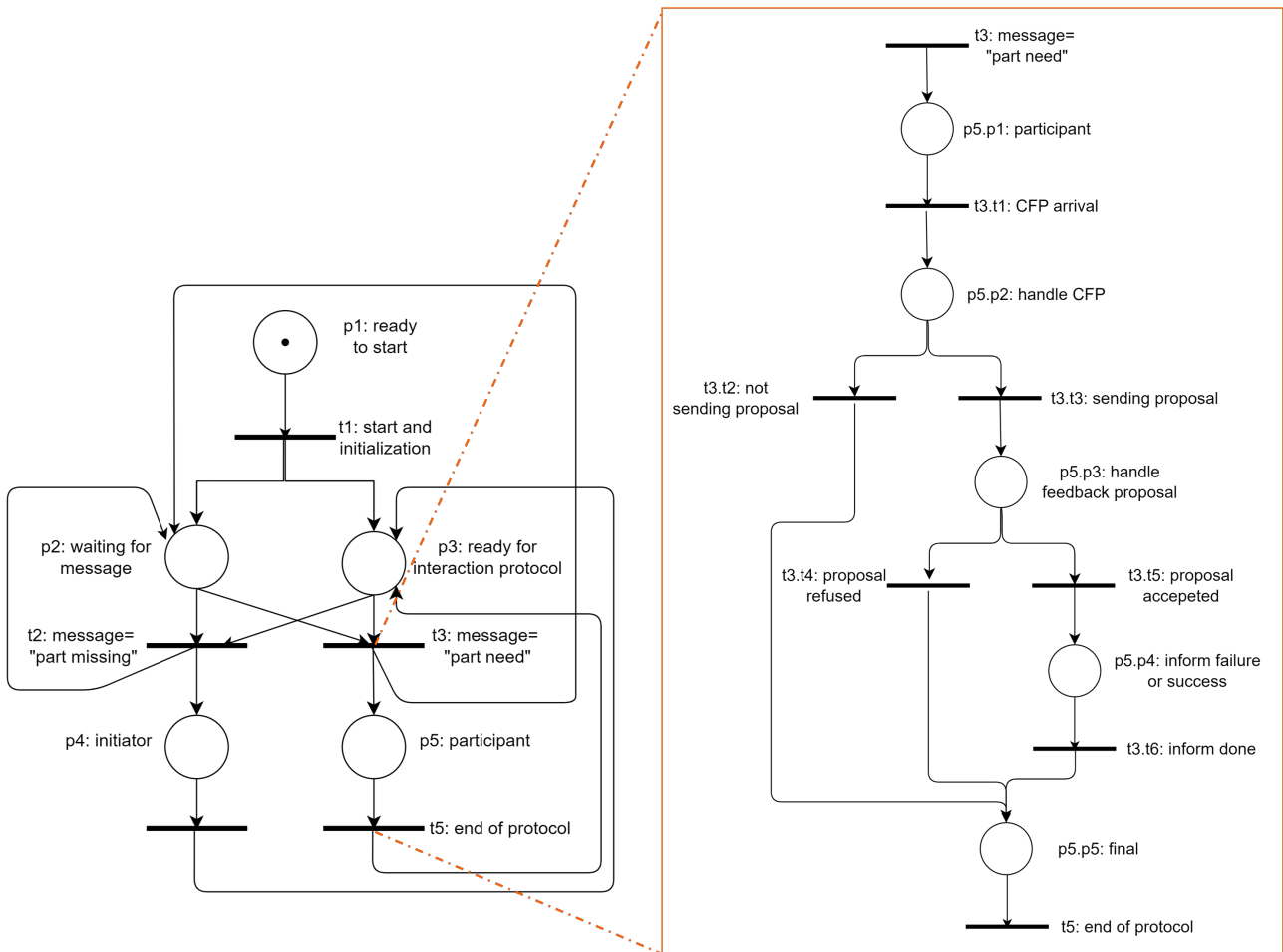


FIGURE 2.7 – Modèle de comportement des agents kit [Montoya Zapata et al., 2024a]

La transition  $t_1$  représente la procédure d'initialisation. Elle comprend le chargement du profil de l'agent, la lecture des fichiers d'entrée et l'enregistrement des services de l'agent auprès du DF. Après avoir déclenché cette transition, l'agent kit entre dans les états  $p_2$  et  $p_3$ , où il attend des messages d'autres agents et où il est disponible pour participer à un protocole d'interaction Contract-Net. Selon l'alerte déclenchée par l'opérateur de la ligne d'assemblage, l'agent kit atteindra les états d'initiateur ( $p_4$ ) ou de participant ( $p_5$ ).

L'agent kit avec la pièce manquante est informé par la déclaration d'alerte et prend le rôle d'initiateur. À ce moment, l'agent initiateur envoie un message "besoin de pièce" au reste des agents kit dans le système pour les informer du problème. Ensuite, le reste des agents kit dans le système prend

## 2.3. MÉTHODE DE MODÉLISATION MULTI-AGENTS PROPOSÉE

---

le rôle de participants. Ainsi, tous les agents kit dans le système peuvent communiquer afin de trouver des kits qui pourraient fournir la pièce manquante à l'agent initiateur.

Le protocole d'interaction global est expliqué dans la section 2.3.2. Cependant, notez que l'objectif principal des agents kit participant est de déterminer s'ils peuvent ou non fournir la pièce manquante, et de calculer le coût de fourniture de cette pièce (état  $p_5.p_2$ ). L'estimation du coût est présentée dans la section 2.3.4.

Notez également que les agents kit sont constamment disponibles pour échanger des messages avec d'autres agents dans le système, en parallèle avec le protocole d'interaction. Cette fonctionnalité n'est pas exploitée dans ces travaux de thèse, mais elle devrait être utilisée dans les développements futurs du système actuel.

### 2.3.4 Méthodologie d'estimation des coûts

Dans notre approche, un opérateur doit effectuer deux déplacements pour réparer un kit. Tout d'abord, l'opérateur doit effectuer un aller-retour entre le kit  $K_D$  et le kit candidat choisi ( $K_C$ ) pour récupérer la pièce manquante du kit  $K_C$  (Figure 2.8a). Ensuite, l'opérateur doit se déplacer du kit  $K_D$  vers le kit  $K_E$  et du kit  $K_E$  vers le kit  $K_C$ , afin de remplacer la pièce manquante dans le kit  $K_C$  (Figure 2.8b). La procédure pour estimer le temps de déplacement total ( $\Delta t_{total}$ ) est la suivante. Veuillez noter que cette procédure doit être appliquée à chaque  $K_C$ .

La position initiale des kits  $K_D$  et  $K_C$  est identifiée (positions à  $t_0$ ) et la distance entre ces deux kits est calculée. Ensuite, en supposant une vitesse de déplacement constante de l'opérateur ( $S_O$ ), le temps de déplacement aller-retour ( $\Delta t_0$ ) est estimé. À  $t_1$ , où  $t_1 = t_0 + \Delta t_0$ , les positions des kits  $K_D$ ,  $K_E$  et  $K_C$  sont mises à jour. Selon la position de chaque kit, une vitesse moyenne des AGV ( $S_{AGV}$ ) et la vitesse de la ligne de production ( $S_{PL}$ ) sont utilisées pour effectuer la mise à jour de la position. Les nouvelles positions sont ensuite utilisées pour calculer la distance entre les kits  $K_D$  et  $K_E$  et entre  $K_E$  et  $K_C$ . Ainsi, avec la vitesse  $S_O$ , nous pouvons estimer les temps de déplacement de l'opérateur entre ces kits ( $\Delta t_1$ ). De cette manière,  $\Delta t_{total} = \Delta t_0 + \Delta t_1$ .

### 2.3.5 Méthodologie de collecte de données et d'identification de l'état du système

Comme mentionné dans la Section 2.3.2, contrairement à la plupart des travaux de la littérature tels que [Gamboa Quintanilla et al., 2016], [Sanogo et al., 2023], [Trentesaux et al., 2013], nous n'utilisons pas de simulation pour effectuer des tests dans notre système. Nous utilisons des données collectées directement sur le terrain industriel afin de définir l'état du système.

Selon les définitions fournies par [Trentesaux et al., 2013], ces expériences sont réalisées hors ligne, c'est-à-dire que le temps nécessaire au système pour réagir et proposer des solutions n'est pas pris en



### 2.3. MÉTHODE DE MODÉLISATION MULTI-AGENTS PROPOSÉE

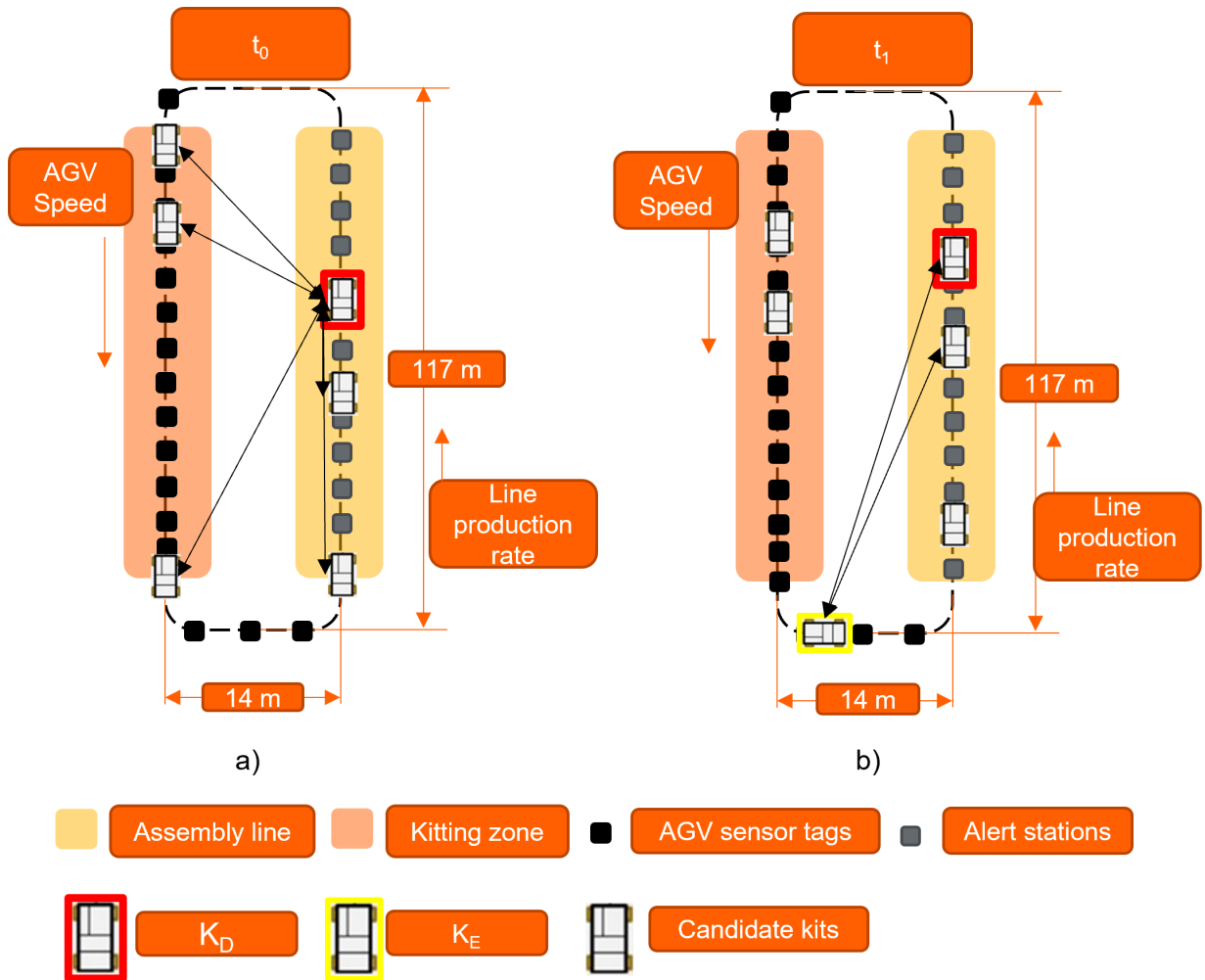


FIGURE 2.8 – Méthodologie d’estimation des coûts a) de  $K_D$  à chaque kit candidat et b) de  $K_D$  à  $K_E$

compte. Les données utilisées pour effectuer les tests correspondent à des données historiques stockées dans des bases de données. Ainsi, nous avons pu préparer les données nécessaires - les fichiers d’entrée pour notre MAS - sans perturber le fonctionnement du système réel. Afin d’effectuer des tests en ligne dans des travaux futurs, le temps nécessaire pour réaliser des requêtes à ces bases de données et sélectionner les informations nécessaires doit être pris en compte et minimisé autant que possible.

L’état d’un système “consiste en une collection de variables contenant toutes les informations nécessaires pour décrire le système à tout instant”, selon [Banks et al., 1996]. D’autres auteurs [Cardin, 2007] préfèrent la définition de [Pooch et Wall, 1992], la trouvant moins vague et plus explicite quant au niveau de détail nécessaire pour décrire l’état du système. [Pooch et Wall, 1992] définit l’état du système comme : “Collection minimale d’informations avec lesquelles le comportement futur du système peut être déterminé de façon unique en l’absence de hasard.” Nous nous basons sur ces définitions pour

### 2.3. MÉTHODE DE MODÉLISATION MULTI-AGENTS PROPOSÉE

---

identifier les variables ou informations nécessaires pour définir l'état de notre système, à un niveau de détail tel que la méthode multi-agents proposée puisse être testée.

Pour ce faire, nous utilisons des informations directement issues du système réel, notamment du MES et d'autres systèmes d'information. Nous sommes conscients que cette méthode ne permet pas de faire une description tout à fait exacte du système réel à chaque instant, à cause des incertitudes spatiales et temporelles du MES [Cardin et Castagna, 2006]. Néanmoins, elle proportionne une description suffisamment détaillée et fiable pour l'évaluation de la méthode proposée.

Les Tableaux 2.2 à 2.4 présentent les données que nous avons collectées à travers les différents systèmes d'informations du site de fabrication étudié : le serveur des alertes, le serveur des AGVs et le MES. Ces données sont stockées dans des serveurs et des bases de données séparées. Un travail d'agrégation préliminaire a dû être effectué avant de pouvoir les exploiter. Quant à la récupération, certaines données étaient récupérables uniquement en se branchant physiquement au serveur local. Ce travail est plus richement décrit dans le Chapitre 4.

TABLEAU 2.2 – Extrait des données fournies par le système d'information des alertes

Segment	Work station	Call type	Call subtype	Call status	Call date	Vehicle ID
UET DOS	810	Quality	Cache rétro	Closed	01/06/2023 12 :58	2250059
UET DOS	820	Quality	Cache rétro	Closed	01/06/2023 12 :48	2230174
UET DOS	15	Question		Canceled	01/06/2023 12 :47	
UET DOS	40	Means	Batterie HS	Canceled	01/06/2023 12 :46	
UET DOS	80	Logistics		Canceled	01/06/2023 12 :43	
UET DOS	105	Logistics		Closed	01/06/2023 12 :41	
UET DOS	810	Quality	Coulisse ARG	Closed	01/06/2023 12 :38	2310179
UET DOS	820	Quality	Cache rétro	Closed	01/06/2023 12 :38	2250146
UET DOS	125	Question		Canceled	01/06/2023 12 :36	

A partir de ces tableaux nous pouvons obtenir les informations suivantes :

- Depuis le serveur des alertes :
  - L'identification du véhicule qui génère chaque alerte
  - La date et l'heure de déclaration de chaque alerte
  - Le poste de travail qui déclenche l'alerte
- Depuis le serveur des AGVs :
  - La référence du dernier capteur de position (Tag) que chaque AGV a lu
  - La date et l'heure de la lecture
- Depuis le MES :
  - La séquence de production

### 2.3. MÉTHODE DE MODÉLISATION MULTI-AGENTS PROPOSÉE

---

TABLEAU 2.3 – Extrait des données fournies par le système d’information des AGVs

Type	Date	AGV	Circuit	Tag
Tag	01/06/2023 4 :43 :42	3345	DOS	19126
Tag	01/06/2023 4 :43 :43	3345	DOS	19675
Tag	01/06/2023 4 :44 :00	3345	DOS	18121
Tag	01/06/2023 4 :44 :09	3345	DOS	18725
Tag	01/06/2023 4 :48 :05	2953	DOS	16673
Tag	01/06/2023 4 :48 :40	2953	DOS	16598
Tag	01/06/2023 4 :48 :53	2953	DOS	19533
Tag	01/06/2023 4 :49 :00	2953	DOS	19610
Tag	01/06/2023 5 :11 :59	3344	DOS	17977
Tag	01/06/2023 5 :12 :38	3344	DOS	16882
Tag	01/06/2023 5 :12 :57	3345	DOS	11682
Tag	01/06/2023 5 :13 :11	3345	DOS	18017

TABLEAU 2.4 – Extrait des données fournies par le MES

Vehicle ID	Order date	Product type
1710096	06/04/2023 04 :46	Type 13
1740027	06/04/2023 04 :48	Type 13
1410078	06/04/2023 04 :49	Type 6
1730011	06/04/2023 05 :21	Type 7
1750011	06/04/2023 05 :23	Type 7
710392	06/04/2023 05 :24	Type 8
1520470	06/04/2023 05 :25	Type 13
1530032	06/04/2023 05 :26	Type 13
940008	06/04/2023 05 :27	Type 3
1750007	06/04/2023 05 :28	Type 7
1750027	06/04/2023 05 :29	Type 24
1310070	06/04/2023 05 :30	Type 26
1740034	06/04/2023 05 :31	Type 13
1530042	06/04/2023 05 :32	Type 13
1230057	06/04/2023 05 :33	Type 26

Nous nous servons de ces informations pour reconstituer l’état du système au moment du déclenchement de l’alerte. Avec les informations issues du serveur des alertes nous récupérons la date et l’heure de déclaration de l’alerte, le véhicule concerné et la station de travail qui déclare l’alerte. Puis nous regardons le dernier tag lu par les AGVs à l’heure la plus proche du moment de l’alerte. Nous repérons ainsi la position des AGVs situés entre la zone de kitting et la ligne de montage. Ensuite avec la position du poste qui déclare l’alerte, nous sommes capables d’identifier le nombre de postes de travail et donc de kits qui sont sur la ligne de production et qui sont en-cours de livraison sur les AGVs.

## 2.4. CONCLUSION

---

En agrégeant ces éléments avec la séquence de production nous pouvons avoir à la fin, pour chacune des alertes, un scénario comme celui présenté par la Figure 2.9 où nous connaissons la position, l'ID véhicule et la diversité du kit défectueux ainsi que des AGVs portant les kits en amont.

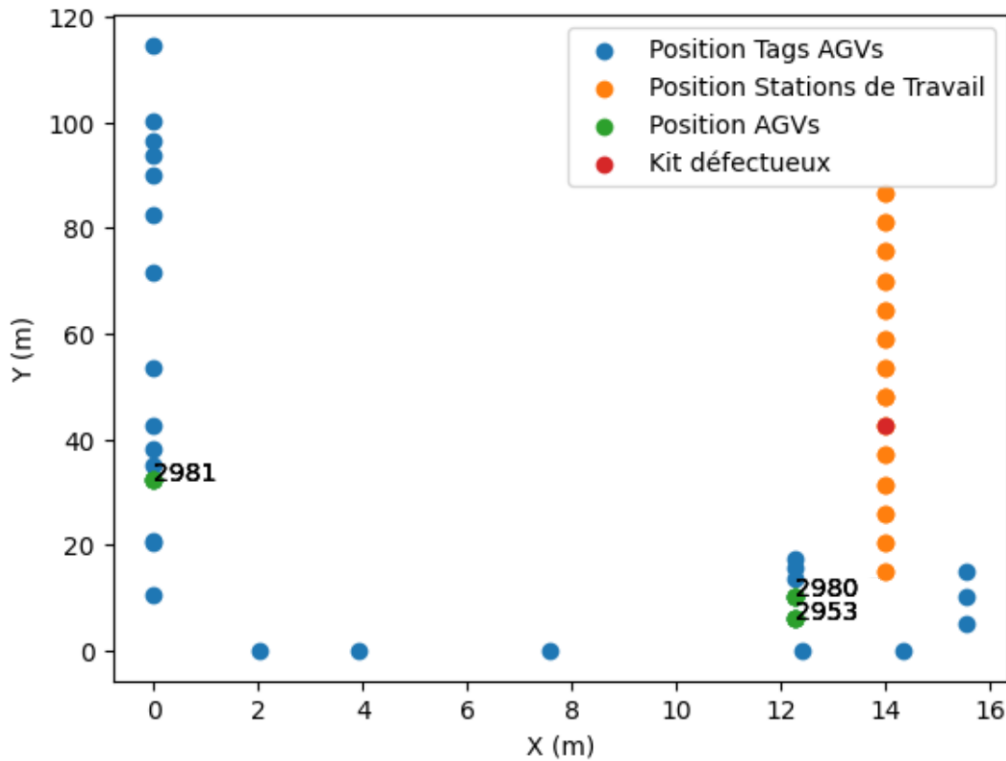


FIGURE 2.9 – Disposition physique d'un scénario ponctuel

## 2.4 Conclusion

Ce chapitre a présenté un état général de la littérature sur plusieurs aspects des systèmes multi-agents. D'une part, il a apporté un référentiel académique permettant de positionner les travaux de recherche effectués dans le cadre de cette thèse, et de l'autre, il a permis d'expliquer les concepts de base ainsi que les applications existantes dans plusieurs secteurs d'activité. D'abord, le concept de systèmes multi-agents a été présenté comme un type d'intelligence artificielle distribuée. Ensuite, la définition d'agent et les types d'agents ont été introduits. Pour finir, le mode d'apprentissage des systèmes multi-agents ainsi que les outils de développement et les secteurs d'applications de ces systèmes ont été discutés. Ici, le grand nombre de plateformes, d'outils, et d'environnements de développements utilisés pour la modélisation de systèmes multi-agents, montre qu'il n'existe pas de méthodologie définie ou de processus standard pour la création de ces systèmes. En outre, nous avons pu constater que ces systèmes peuvent être appliqués à une grande diversité de secteurs d'activité. Néanmoins, le

## 2.4. CONCLUSION

---

faible nombre de publications présentant des cas d'études réels nous laisse penser que l'application de ce domaine de recherche et son impact dans l'industrie restent réduits. Ainsi, des liens plus forts avec l'industrie doivent être construits, et une seule ou un nombre restreint de méthodologies standards doivent être définies.

Aucune étude considérant l'application des techniques des systèmes multi-agents dans une zone de kitting d'un processus industriel n'a été trouvée. Par conséquent, l'objectif de proposer une architecture de ce type, afin d'assurer la bonne performance du processus de logistique interne d'une usine automobile, se révèle pertinent d'un point de vue scientifique. Cela contribue ainsi à l'application des systèmes multi-agents dans des cas d'études concrets.

Dans ce cadre, la deuxième partie de ce chapitre a présenté une méthode basée sur les systèmes multi-agents pour réduire le temps de réaction à une alerte de pièce manquante, déclenchée par un opérateur sur une ligne d'assemblage utilisant le processus d'approvisionnement de kitting. La méthode proposée exploite les propriétés de la modélisation basée sur des agents, pour fournir des solutions dynamiques dans une approche décentralisée.

Afin de mettre en œuvre la méthode proposée sur le terrain industriel, nous avons pris en compte la procédure actuelle utilisée par les opérateurs pour gérer l'alerte de pièce manquante, et nous avons souhaité la modifier le moins possible. De plus, dans notre approche, les opérateurs jouent un rôle actif dans la phase de prise de décision.

Le chapitre suivant a pour objectif de présenter l'application de cette architecture dans un cas d'étude réel, provenant d'une usine d'assemblage du Groupe Renault.

## Chapitre 3

# Système multi-agents pour réagir aux perturbations d'un processus de kitting d'une ligne d'assemblage automobile : application

### Contenu

---

3.1	Introduction . . . . .	74
3.2	Illustration de la méthode proposée à travers une étude de cas ponctuel . . . . .	74
3.2.1	Description de l'étude de cas ponctuel . . . . .	74
3.2.2	Résultats expérimentaux . . . . .	77
3.3	Analyse de la distribution de temps entre deux alertes consécutives . . . . .	79
3.4	Méthodologie d'évaluation de la méthode multi-agents proposée sur une période de temps . . . . .	82
3.4.1	Identification de l'état du système en utilisant des informations directement issues du système réel . . . . .	82
3.4.2	Identification des kits candidats et calculs des critères de sélection du meilleur endroit de récupération de la pièce manquante . . . . .	84
3.4.3	Méthode de sélection du meilleur kit candidat . . . . .	86
3.5	Définition des KPIs globaux . . . . .	87
3.5.1	Alertes comme événements indépendants . . . . .	87
3.5.2	Alertes comme événements non indépendants . . . . .	88
3.6	Validation expérimentale . . . . .	91
3.6.1	Sélection de l'endroit de récupération de la pièce manquante . . . . .	92
3.6.2	KPIs résultants en traitant les alertes comme événements indépendants . . . . .	95
3.6.3	KPIs résultants en traitant les alertes comme événements non indépendants . . . . .	95
3.7	Conclusion . . . . .	98

---

### 3.1 Introduction

Le Chapitre 2 a présenté le cadre scientifique de ces travaux de thèse, exposant les concepts fondamentaux des systèmes multi-agents ainsi que leurs applications existantes dans plusieurs secteurs d'activité. Ce chapitre a également proposé une méthode multi-agents visant à répondre au déclenchement d'alertes en cas de pièce manquante dans un kit, sur une ligne de montage automobile.

L'objectif du Chapitre 3 est d'illustrer la méthode multi-agents proposée, à travers son application sur un scénario réel provenant d'une usine d'assemblage automobile. De plus, ce chapitre expose une démarche pour évaluer cette méthode, non seulement sur un scénario ponctuel, mais aussi sur un ensemble de scénarios d'alerte, qui ont lieu au cours d'une journée de production type au sein de l'usine étudiée. Des données réelles ont été collectées à travers différents systèmes d'information d'un site de fabrication du Groupe Renault. Grâce à ces données, nous avons reconstitué l'état du système de production à tous les moments où une alerte a été déclenchée par un opérateur. Ensuite, la méthode multi-agents a été testée sur chacun des scénarios, en traitant les alertes comme (i) un ensemble de scénarios indépendants et (ii) comme des scénarios non indépendants. Finalement, deux variantes de la méthode multi-agents ont été comparées avec la méthodologie utilisée actuellement par le site de production, à travers trois indicateurs calculés dans les deux configurations. Les travaux précédents sur le benchmarking et le test de systèmes multi-agents ont fortement inspiré la méthode d'évaluation proposée, en particulier [Trentesaux et al., 2013], [Cardin et L'Anton, 2018] et [Pinquie et al., 2022].

### 3.2 Illustration de la méthode proposée à travers une étude de cas ponctuel

#### 3.2.1 Description de l'étude de cas ponctuel

Pour tester notre méthode basée sur les MAS dans un scénario réel, nous avons utilisé des données historiques collectées dans une véritable usine d'assemblage automobile. Ces données ont servi à modéliser l'usine d'assemblage au moment où un opérateur a déclenché l'alerte pour une pièce manquante dans l'un des kits. Le système d'alerte DWS, le système MES et le serveur des AGVs ont permis la collecte des données.

L'usine d'assemblage de notre étude est divisée en deux zones principales : la zone de kitting (où les kits sont préparés) et la ligne d'assemblage des portes. La Figure 3.1 présente la disposition globale de l'usine d'assemblage, montrant l'emplacement de (i) la zone de préparation des kits, (ii) la ligne d'assemblage des portes et les postes de travail, et (iii) les capteurs qui détectent le passage des AGVs (AGV sensor tags).

Les Tableaux 3.1 à 3.3 présentent les données collectées par les systèmes d'information industriels nécessaires à la configuration de notre modèle MAS. Le Tableau 3.1 rapporte les informations relatives

### 3.2. ILLUSTRATION DE LA MÉTHODE PROPOSÉE À TRAVERS UNE ÉTUDE DE CAS PONCTUEL

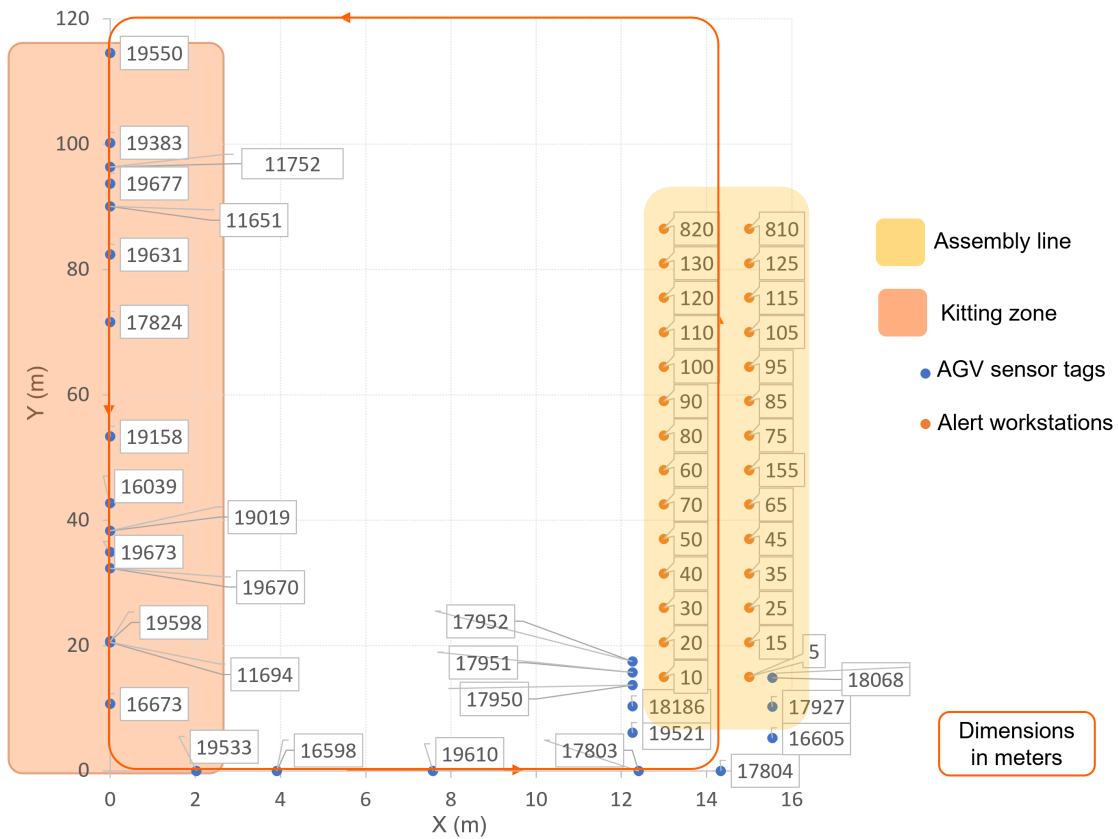


FIGURE 3.1 – Disposition physique du cas d'étude

à l'heure de l'alerte, au kit avec la pièce manquante et à la station de travail avec le kit défectueux. Le Tableau 3.2 informe sur le dernier capteur d'AGV détecté par les AGVs entre la zone de préparation des kits et la ligne d'assemblage. Cette information permet d'estimer l'emplacement des AGVs au moment de l'alerte. Le Tableau 3.3 fournit la séquence ordonnée des kits déjà situés dans la ligne d'assemblage et ceux transportés par les AGVs (chaque AGV peut transporter 4 chariots de kit). Veuillez noter que le premier kit dans le tableau est le kit  $K_D$ . Le Tableau 3.3 fournit également le type de produit associé à chaque kit, définissant les pièces que chaque kit devrait contenir. Les informations dans les Tableaux 3.1 à 3.3 permettent de définir l'état du système au moment où l'alerte est déclenchée.

TABLEAU 3.1 – Données fournies par le système d'information d'alerte

Kit ID	Station de travail	Date et heure de l'alerte
2230108	85	31/05/2023 10 :06



### 3.2. ILLUSTRATION DE LA MÉTHODE PROPOSÉE À TRAVERS UNE ÉTUDE DE CAS PONCTUEL

---

TABLEAU 3.2 – AGVs en cours entre la zone de préparation des kits et la ligne d’assemblage

Identification de l’AGV	Dernier tag de capteur d’AGV lu
2955	18068
3344	16673
2973	19158
2978	19631
2966	11752

TABLEAU 3.3 – Séquence et diversité des ordres de fabrication

Kit ID	Groupe de types de produits	Type de produit
2230108	Type 13	Type_13_14
2120068	Type 5	Type_5_0
2240101	Type 13	Type_13_14
2240016	Type 13	Type_13_28
2230139	Type 5	Type_5_26
2120098	Type 7	Type_7_13
2120115	Type 13	Type_13_56
2230141	Type 5	Type_5_26
2120085	Type 18	Type_18_10
2150006	Type 18	Type_18_9
2230146	Type 5	Type_5_26
2240023	Type 13	Type_13_42
2230093	Type 13	Type_13_14
2230111	Type 13	Type_13_14
2230164	Type 5	Type_5_26
2230092	Type 13	Type_13_14
2240024	Type 13	Type_13_42
2220075	Type 5	Type_5_26
2320060	Type 13	Type_13_33
2240044	Type 18	Type_18_4
2320068	Type 13	Type_13_33
2230149	Type 5	Type_5_26
2240008	Type 13	Type_13_30
2230167	Type 5	Type_5_26
2230098	Type 13	Type_13_14
2230102	Type 13	Type_13_14
2220084	Type 5	Type_5_26
2240018	Type 13	Type_13_42
2310046	Type 7	Type_7_7
2250006	Type 13	Type_13_50

Nous avons implémenté notre approche sur JADE [Bellifemine et al., 2007]. Ainsi, les agents kit

### 3.2. ILLUSTRATION DE LA MÉTHODE PROPOSÉE À TRAVERS UNE ÉTUDE DE CAS PONCTUEL

---

communiquent suivant le langage de communication d'agent FIPA (Foundation for Intelligent Physical Agents). Un agent JADE a été lancé pour chaque agent de kit présent dans le système. Nous avons également modifié le protocole traditionnel de Contract-Net et l'avons implémenté avec celui proposé dans la Section 2.3.2.

#### 3.2.2 Résultats expérimentaux

Nous avons utilisé notre méthode pour résoudre la perturbation de la chaîne de montage indiquée dans le Tableau 3.1. Le Tableau 3.4 présente le premier résultat de la phase de négociation entre les agents kit : la sélection des agents kit pouvant fournir la pièce manquante et leur emplacement actuel dans l'usine. La configuration actuelle de l'usine d'assemblage est également illustrée dans la Figure 2.8a.

TABLEAU 3.4 – Résultat de la phase de négociation : identification des kits candidats

Kit ID	Groupe de types de produits	Type de produit	AGV ou station de travail DWS	Position X (m)	Position Y (m)
2230108	Type 13	Type_13_14	85	14	59
2240101	Type 13	Type_13_14	55	14	48
2230108	Type 13	Type_13_14	2955	14	14.8
2230111	Type 13	Type_13_14	3344	0	10.7
2230092	Type 13	Type_13_14	3344	0	10.7
2230098	Type 13	Type_13_14	2978	0	82.4
2230102	Type 13	Type_13_14	2966	0	96.3

Le Tableau 3.5 rapporte, pour chaque kit candidat  $K_C$ , (i) le temps nécessaire à l'opérateur pour effectuer un aller-retour entre le kit  $K_D$  et le kit  $K_C$ , (ii) le temps nécessaire pour récupérer la pièce manquante du kit  $K_E$  et la remplacer dans le kit  $K_C$ , et (iii) la somme des deux temps. Le temps requis pour effectuer un aller-retour entre le kit  $K_D$  et le centre de la zone de kitting est également présenté. Les vitesses de l'opérateur, de l'AGV et de la ligne de production utilisées dans les calculs sont respectivement de  $S_O = 1.2$  m/s,  $S_{AGV} = 0.25$  m/s et  $S_{PL} = 0.117$  m/s.

En considérant ces résultats, le temps nécessaire pour récupérer la pièce manquante des kits  $K_C$  2240101, 2230098, et 2230102 (premier temps de trajet aller-retour) est inférieur au temps nécessaire pour un aller-retour vers la zone de kitting. Pendant la phase d'évaluation, l'agent du kit  $K_D$  classe ces options et les propose à l'opérateur de la ligne d'assemblage. Étant donné que l'interprétation des résultats présentés dans le Tableau 3.5 pourrait représenter une charge cognitive pour l'opérateur et pourrait prendre du temps, la proposition est également présentée sous forme graphique à l'opérateur de la ligne d'assemblage, comme illustré dans la Figure 3.2.

Les résultats pour les 3 premiers kits  $K_C$  (2240101, 2230098, et 2230102) démontrent l'efficacité

### 3.2. ILLUSTRATION DE LA MÉTHODE PROPOSÉE À TRAVERS UNE ÉTUDE DE CAS PONCTUEL

TABLEAU 3.5 – Résultats : temps de déplacement de l'opérateur pour chaque kit candidat

Kit ID	AGV ou station DWS	Temps 1 <sup>er</sup> aller-retour (s)	Temps 2 <sup>ème</sup> aller (s)	Somme des temps (s)	Temps aller zone de kitting (s)
2230108	85	0	0	0	69.0
2240101	55	18.3	94.7	113.0	
2230108	2955	73.6	113.1	186.7	
2230111	3344	83.9	114.8	198.7	
2230092	3344	83.9	114.8	198.7	
2230098	2978	45.4	109.5	154.9	
2230102	2966	66.5	112.0	178.5	

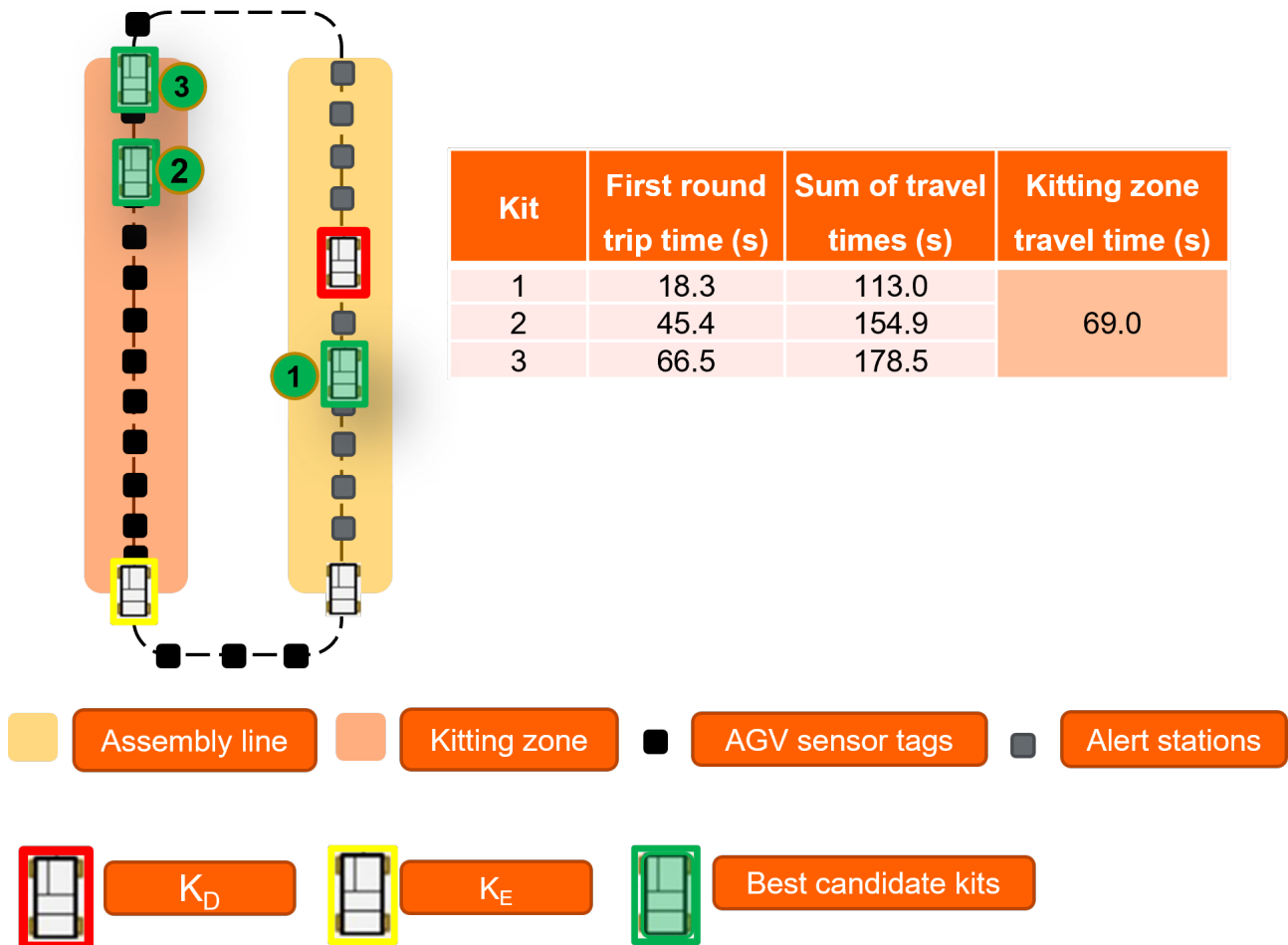


FIGURE 3.2 – Meilleurs kits candidats proposés à l'opérateur de la ligne d'assemblage

de l'approche proposée pour fournir des options pertinentes aux opérateurs de la ligne d'assemblage afin de réduire les temps de réaction lorsqu'un kit est défectueux, prévenant ainsi des arrêts inutiles de la ligne de production. En se concentrant sur la première option suggérée, nous pouvons souligner

### 3.3. ANALYSE DE LA DISTRIBUTION DE TEMPS ENTRE DEUX ALERTES CONSÉCUTIVES

---

deux éléments importants. Premièrement, en comparant les temps de trajet entre (i)  $K_D$  et le premier candidat vs (ii)  $K_D$  et la zone de kitting, on observe que le temps de réaction diminue de plus de 70%. Deuxièmement, l'opérateur de la ligne d'assemblage dispose d'environ 2 minutes pour remplacer la pièce manquante dans le kit  $K_C = 2240101$ , sachant que le taux de production est d'un véhicule par minute et que le kit candidat est situé à 2 postes de travail du kit  $K_D$ . Nous pouvons noter que ce temps disponible est plus long que la somme des temps de trajet pour la solution proposée (113 s).

### 3.3 Analyse de la distribution de temps entre deux alertes consécutives

L'objectif de cette section est de déterminer si nous pouvons étudier les alertes et la réaction aux alertes comme des événements ponctuels et indépendants, comme le cas étudié dans la Section 3.2, ou si une alerte et le temps de réaction ont des impacts sur les alertes suivantes. En d'autres termes, nous voulons vérifier si l'opérateur en charge de répondre aux alertes est disponible à chaque fois qu'il y a une nouvelle alerte.

Selon les résultats de la Section 3.2.2, nous pouvons constater que la méthode proposée permet d'améliorer le temps de réaction face à l'alerte et donc d'éviter des potentiels arrêts de production mais elle augmente aussi le temps de déplacement et, par conséquent, la charge de travail des opérateurs dédiés à la réponse aux alertes. Il est donc nécessaire d'étudier le temps qui s'écoule entre la déclaration de deux alertes consécutives pour vérifier la disponibilité de l'opérateur.

Grâce au serveur d'alertes, nous avons pu récupérer des données des alertes qui ont eu lieu entre mai et juin 2023. Dans ce fichier, nous pouvons trouver des informations comme le poste qui déclare l'alerte, le type d'alerte (qualité, logistique, question, moyens et sécurité), la date et l'heure de la déclaration de l'alerte et l'identifiant du véhicule et donc du kit concerné. Sachant que la méthode que nous proposons concerne le problème de pièce manquante dans un kit, nous nous intéressons à la catégorie d'alertes de type "Logistique" provenant de l'atelier de montage des portes.

Comme il est possible de voir sur la Figure 3.1 l'atelier de montage des portes est composé de 28 postes différents et chacun de ces postes peut déclarer des alertes. Dans un premier temps nous servons des données récupérées pour déterminer le temps entre deux alertes provenant du même poste. Pour effectuer cela, nous regroupons les données par poste de travail et ensuite nous calculons la différence entre la date et l'heure de déclaration de l'alerte  $A$  avec l'alerte suivante  $A + 1$ . La Figure 3.3 présente la distribution de l'écart de temps entre deux alertes consécutives pour chacun des postes. Pour un souci de clarté, uniquement des écarts de temps inférieurs à 10 minutes (600 secondes) ont été pris en compte. Ainsi, les stations de travail qui ne sont pas affichées sur la Figure 3.3 peuvent présenter des alertes mais avec des écarts de temps entre alertes supérieurs à 10 minutes. Le nombre total d'alertes déclarées par chaque poste de travail sur la période d'analyse est présenté par le Tableau 3.6.

### 3.3. ANALYSE DE LA DISTRIBUTION DE TEMPS ENTRE DEUX ALERTES CONSÉCUTIVES

---

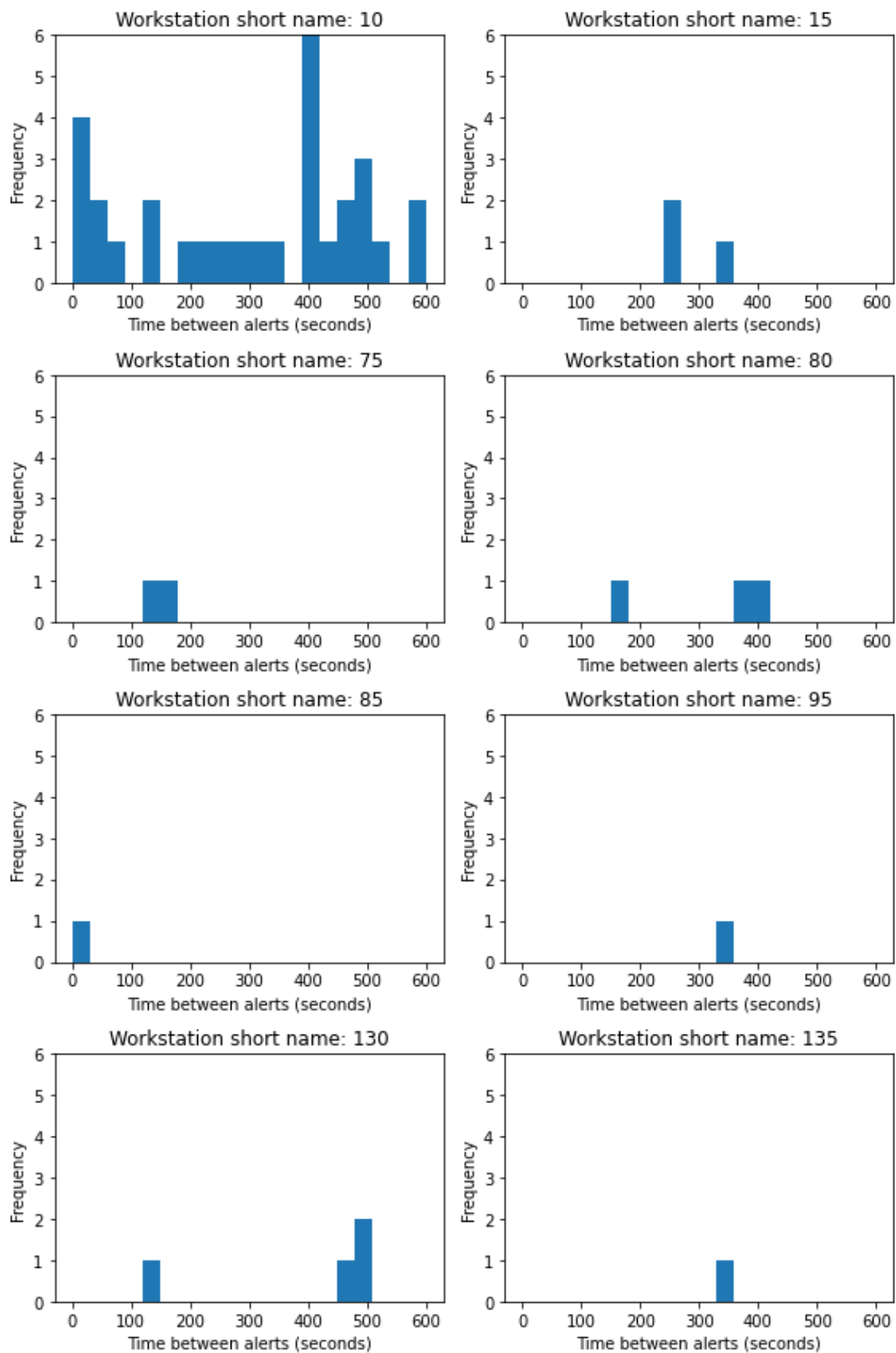


FIGURE 3.3 – Ecart de temps entre deux alertes consécutives pour chaque poste de travail

### 3.3. ANALYSE DE LA DISTRIBUTION DE TEMPS ENTRE DEUX ALERTES CONSÉCUTIVES

---

TABLEAU 3.6 – Nombre d’alertes par station de travail sur la période d’analyse

Station de travail	Nombre d’alertes
5	33
10	348
15	111
25	5
35	6
40	7
45	6
50	8
55	10
60	5
65	3
75	14
80	30
85	22
90	11
95	6
100	11
105	5
110	6
115	6
120	1
125	14
130	47
135	12
145	2
155	2
Total	729

Les résultats du scénario présenté dans la Section 3.2.2 exposent que la durée totale de la perturbation et donc de l’indisponibilité de l’opérateur est d’environ 113 secondes. La Figure 3.3 permet de constater que le nombre d’écarts de temps entre deux alertes consécutives inférieurs à 113 secondes est très faible par rapport au total des alertes par poste de travail. Uniquement les postes 10 et 85 présentent des écarts inférieurs à 113 secondes, le poste 10 étant celui avec le plus d’occurrences. Néanmoins, la probabilité d’apparition ou déclaration des deux alertes consécutives dans cette fenêtre de temps reste assez faible ( $7/348 = 2\%$ ). Ainsi, nous pouvons considérer que par poste de travail nous pouvons étudier les alertes en tant qu’événements indépendants.

Etant donné que le même groupe d’opérateurs doit réagir aux alertes pour tout l’atelier de portes et non pas pour chaque poste en particulier, dans un deuxième temps, nous effectuons la même analyse d’écart de temps entre deux alertes pour tout l’atelier des portes de façon globale. Ici, nous ne

### 3.4. MÉTHODOLOGIE D'ÉVALUATION DE LA MÉTHODE MULTI-AGENTS PROPOSÉE SUR UNE PÉRIODE DE TEMPS

---

regroupons pas les alertes par poste de travail mais nous les considérons toutes venant de l'atelier en question. Ensuite, nous calculons l'écart entre deux alertes consécutives en suivant la même démarche que précédemment. La Figure 3.4 met en évidence la distribution de ces écarts pour des écarts inférieurs à 10 minutes.

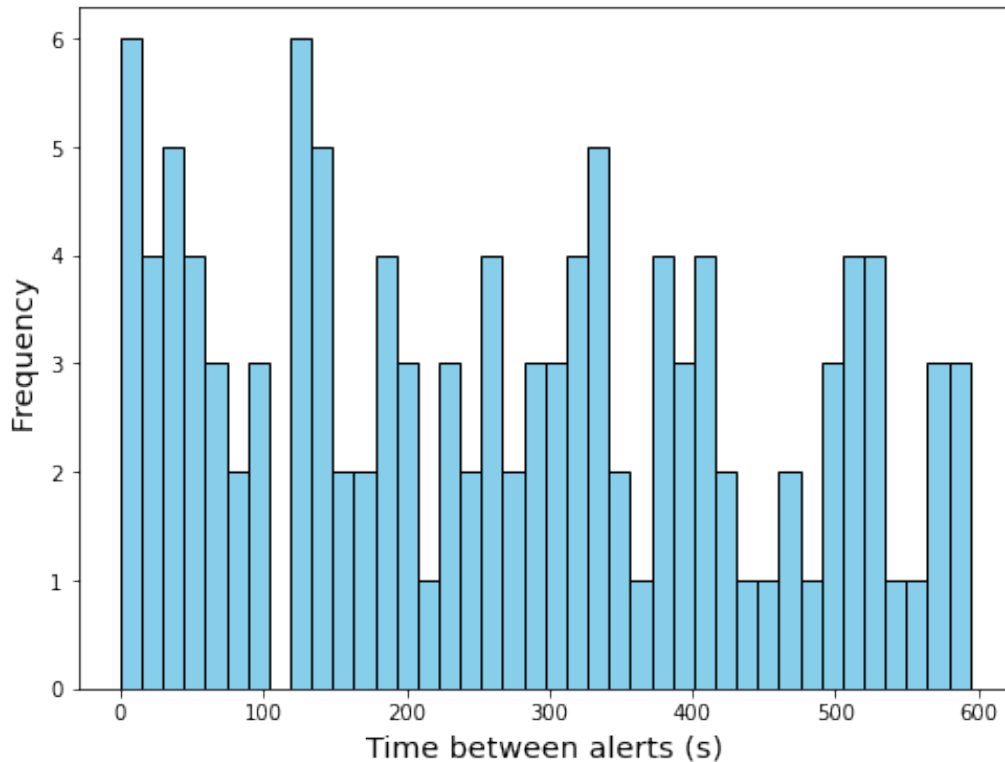


FIGURE 3.4 – Ecart de temps entre deux alertes consécutives pour l'atelier d'étude

Nous pouvons observer sur cette figure que, d'un point de vue global, le nombre d'écarts inférieurs à 10 minutes augmente et surtout, que le nombre d'écarts inférieurs à 120 secondes est considérable. Par conséquent, l'évaluation de la méthode proposée en considérant les alertes comme des événements non indépendants est nécessaire.

## 3.4 Méthodologie d'évaluation de la méthode multi-agents proposée sur une période de temps

### 3.4.1 Identification de l'état du système en utilisant des informations directement issues du système réel

Pour réaliser l'évaluation de la méthode proposée sur une journée de production, nous réalisons le processus de reconstitution de l'état du système (Section 2.3.5) pour chaque alerte de façon automa-

### 3.4. MÉTHODOLOGIE D'ÉVALUATION DE LA MÉTHODE MULTI-AGENTS PROPOSÉE SUR UNE PÉRIODE DE TEMPS

---

tique. Pour ce faire, nous lisons les trois bases de données, associées au MES, au serveur d'alertes et au serveur des AGVs, à l'aide d'un script Python. Veuillez noter que dans les données fournies par le système d'alerte (Tableau 2.2), l'identifiant du véhicule déclenchant l'alerte est parfois manquante. Etant donné que l'identifiant du véhicule est indispensable pour mener le processus de reconstitution de l'état de système, les alertes qui n'ont pas cette information renseignée dans la base de données ne sont pas considérées pour l'analyse. La réduction du nombre d'alertes à traiter peut être favorable au bon fonctionnement de notre méthode. Ainsi, afin de ne pas biaiser notre étude par rapport à un fonctionnement normal, ici nous ne filtrons pas les alertes par leur type (Logistique par exemple) mais nous considérons toutes les alertes de la journée comportant l'identifiant du véhicule.

Afin de pouvoir traiter les informations de toutes les alertes, nous stockons l'état du système sous forme de tableau et informatiquement sous forme d'une structure de données abstraite (DataFrame en Python). A titre d'exemple, le Tableau 3.7 présente l'état du système pour l'alerte levée à 05 :20 :23 pour le kit défectueux N°2220082. Dans la suite de ce document, l'état du système de chaque alerte sera également appelé scénario. L'application de la méthodologie ici décrite sur l'ensemble des alertes d'une journée de production donne comme résultat un ensemble de tableaux comme le Tableau 3.7, agrégés tous sur un même DataFrame.

TABLEAU 3.7 – Etat du système à l'instant t où le kit N°2220082 a été reconnu défectueux

Vehicle /kit ID	Product type	Workstation /AGV	AGV Tag	X (m)	Y (m)
2230134	Type 13	45		14	37.0
2250011	Type 13	35		14	31.5
2230012	Type 5	25		14	26.0
2230131	Type 13	15		14	20.5
2230118	Type 13	5		14	15.0
2240039	Type 5	2980	18186	12.3	10.3
2120090	Type 18	2980	18186	12.3	10.3
2230128	Type 13	2980	18186	12.3	10.3
2430004	Type 5	2980	18186	12.3	10.3
2230130	Type 13	2953	19521	12.3	6.0
2240002	Type 13	2953	19521	12.3	6.0
2220076	Type 5	2953	19521	12.3	6.0
2120080	Type 18	2953	19521	12.3	6.0
2230140	Type 5	2981	19670	0.0	32.3
2240003	Type 13	2981	19670	0.0	32.3
2230169	Type 5	2981	19670	0.0	32.3
2310032	Type 13	2981	19670	0.0	32.3



### 3.4.2 Identification des kits candidats et calculs des critères de sélection du meilleur endroit de récupération de la pièce manquante

#### 3.4.2.1 Identification des kits candidats

Une fois que l'état du système est défini pour chacune des alertes, les agents kit ayant le modèle de comportement décrit dans la Section 2.3.3, utilisent le protocole d'interaction présenté dans la Section 2.3.2 afin d'identifier les kits candidats ( $K_C$ ) qui pourraient fournir la pièce manquante au kit défectueux ( $K_D$ ).

Pour ce faire, une phase de négociation est démarrée par le  $K_D$ . Ce dernier envoie un appel à propositions à tous les agents kits qui se trouvent dans le système (agents kits participants) afin de vérifier s'ils possèdent la pièce manquante. Chaque agent kit participant répond refuse ou accepte l'appel à propositions. En cas d'acceptation, ils envoient le coût associé à la récupération de la pièce. Les kits participants ayant la pièce manquante sont catalogués donc en tant que kits candidats.

Ainsi, en reprenant l'exemple de la Section 3.4.1, la liste de kits présentés par le Tableau 3.7 représente la liste des kits participants dans le protocole d'interaction pour le kit défectueux N°2220082. Les kits candidats seraient donc les kits ayant la même diversité que le kit défectueux, à savoir diversité Type 5. Le Tableau 3.8 présente la liste de kits candidats pour le kit défectueux en question.

De même que pour la Section 3.4.1 cette démarche est appliquée à l'ensemble des scénarios reconstitués et les listes de kits candidats sont agrégées toutes dans un même DataFrame.

TABLEAU 3.8 – Liste des kits candidats pour le kit défectueux N°2220082

Vehicle /Kit ID	Product type	Workstation /AGV	Tag	X (m)	Y (m)	Type
2230012	Type 5	25.0		14.0	26.0	Candidate
2240039	Type 5	2980	18186	12.3	10.3	Candidate
2430004	Type 5	2980	18186	12.3	10.3	Candidate
2220076	Type 5	2953	19521	12.3	6.0	Candidate
2230140	Type 5	2981	19670	0.0	32.3	Candidate
2230169	Type 5	2981	19670	0.0	32.3	Candidate

#### 3.4.2.2 Estimation des coûts de récupération de la pièce manquante

Les kits candidats ayant la pièce manquante doivent informer le kit défectueux du coût de récupération de la pièce en fonction de leurs positions dans l'atelier. Chaque kit candidat suit la méthodologie présentée par la Section 2.3.4 pour réaliser l'estimation du coût de récupération. Cependant, il y a une variante qui est incorporée sur le deuxième déplacement afin de simplifier la démarche. Le deuxième déplacement ne se fera pas vers un autre kit mais directement vers la zone de kitting.

### 3.4. MÉTHODOLOGIE D'ÉVALUATION DE LA MÉTHODE MULTI-AGENTS PROPOSÉE SUR UNE PÉRIODE DE TEMPS

---

Ainsi, l'approche proposée est la suivante : d'abord, l'opérateur doit effectuer un aller-retour entre le kit  $K_D$  et le kit candidat choisi ( $K_C$ ) pour récupérer la pièce manquante (Figure 2.8a). Ensuite, l'opérateur doit se déplacer du kit  $K_D$  vers la zone de kitting afin de remplacer la pièce manquante dans le kit  $K_C$ .

La procédure pour estimer le coût c'est-à-dire, le temps de déplacement total de l'opérateur ( $\Delta t_{total}$ ) pour chaque kit candidat, serait donc la somme du temps de déplacement entre  $K_D$  et  $K_C$  ( $\Delta t_0$ ) et le temps de déplacement entre la nouvelle position de  $K_D$  et la zone de kitting ( $\Delta t_1$ ). Également, un temps constant de 25 secondes est additionné afin de prendre en compte le déplacement à l'intérieur de la zone de kitting. De cette manière,  $\Delta t_{total} = \Delta t_0 + \Delta t_1 + 25$ . Veuillez noter que la nouvelle position de  $K_D$  est calculée en fonction de  $\Delta t_0$  et la vitesse d'avancement de la ligne de production  $S_{PL}$ .

En appliquant cette méthodologie à l'ensemble des scénarios, nous pouvons ajouter à notre Data-Frame 3 colonnes contenant  $\Delta t_0$ ,  $\Delta t_1$  et  $\Delta t_{total}$ . En complément, afin d'être en mesure de comparer la méthode multi-agents proposée avec la méthodologie utilisée actuellement sur le site production, nous estimons aussi le temps de déplacement d'un aller-retour à la zone de kitting dès que l'alerte est déclenchée ( $\Delta t_{kitting}$ ). Le Tableau 3.9 présente le résultat pour le scénario du kit défectueux N°2220082.

TABLEAU 3.9 – Coûts de récupération de la pièce manquante pour le kit défectueux N°2220082

Vehicle /Kit ID	Product type	Alert date	Type	Journey kitting zone $\Delta t_{kitting}$ (s)	Journey candidate kit $\Delta t_0$ (s)	Journey kitting zone $\Delta t_1$ (s)	Global perturbation $\Delta t_{total}$ (s)
2220082	Type 5	2023-06-01 05 :20 :23	Defective	88.98			
2230012	Type 5		Candidate	88.98	18.33	85.72	104.06
2240039	Type 5		Candidate	88.98	45.83	80.92	126.75
2430004	Type 5		Candidate	88.98	53.75	79.55	133.30
2220076	Type 5		Candidate	88.98	60.83	78.34	139.17
2230140	Type 5		Candidate	88.98	28.86	83.87	112.73
2230169	Type 5		Candidate	88.98	28.86	83.87	112.73

#### 3.4.2.3 Estimation des temps d'arrêt de production et d'impact sur les kits candidats

En ayant les temps de déplacement pour chaque kit candidat, il est possible d'estimer le temps d'arrêt de la ligne d'assemblage (i) dans le cas où on choisit la méthode actuelle (aller-retour à la zone de kitting) et (ii) dans le cas où on choisit la méthode multi-agents proposée (aller au kit candidat). De même, pour tous les kits candidats de l'option (ii) qui se trouvent déjà sur la ligne de montage, nous estimons s'ils seront impactés si jamais ils sont choisis comme le kit fournisseur de la pièce manquante. C'est-à-dire, nous vérifions si au moment d'arriver à la position actuelle du kit défectueux,

### 3.4. MÉTHODOLOGIE D'ÉVALUATION DE LA MÉTHODE MULTI-AGENTS PROPOSÉE SUR UNE PÉRIODE DE TEMPS

---

ils déclencheront une alerte à cause d'une pièce manquante.

Pour effectuer ces estimations nous utilisons les équations (3.1) à (3.3). Le Tableau 3.10 présente ces estimations pour le scénario du kit défectueux N°2220082.

$$\text{Temps arrêt kitting} = \Delta t_{\text{kitting}} - \text{Cycle time} \quad (3.1)$$

$$\text{Temps arrêt candidat} = \Delta t_0 - \text{Cycle time} \quad (3.2)$$

$$\text{Temps impact candidat} = \Delta t_{\text{total}} - \text{Distance kit candidat} / S_{PL} \quad (3.3)$$

TABLEAU 3.10 – Temps d'arrêt de production et impact sur les kits candidats pour le kit défectueux N°2220082

Vehicle /Kit ID	Product type	Alert date	Type	Time line stop kitting (s)	Time line stop candidate kit (s)	Time impact candidate kit (s)
2220082	Type 5	2023-06-01 05 :20 :23	Defective	28.98		
2230012	Type 5		Candidate	28.98	0.00	9.77
2240039	Type 5		Candidate	28.98	0.00	0.00
2430004	Type 5		Candidate	28.98	0.00	0.00
2220076	Type 5		Candidate	28.98	0.83	0.00
2230140	Type 5		Candidate	28.98	0.00	0.00
2230169	Type 5		Candidate	28.98	0.00	0.00

#### 3.4.3 Méthode de sélection du meilleur kit candidat

Ici, il est pertinent de préciser que la méthode multi-agents proposée se limite à présenter une liste réduite des résultats des calculs effectués dans la Section 3.4.2 à l'opérateur, pour qu'il puisse prendre une décision de façon éclairée. Néanmoins, afin de pouvoir estimer l'impact de notre méthode multi-agents sur une journée de production, nous devons "automatiser" cette prise de décision.

D'abord, il est nécessaire de décider s'il est mieux d'aller récupérer la pièce manquante à (i) la zone de kitting ou (ii) auprès d'un kit candidat. Ensuite, si (ii), il faut choisir quel serait le meilleur kit candidat.

La priorité dans le site de production est d'éviter l'arrêt de la ligne d'assemblage. Le critère de choix entre (i) et (ii) est donc le temps d'arrêt de la ligne généré par chaque option. Ainsi, si le temps d'arrêt de (i) est supérieur ou égal au temps d'arrêt d'un kit candidat de (ii), l'option choisie sera

### 3.5. DÉFINITION DES KPIS GLOBAUX

---

(ii) : récupérer la pièce auprès d'un des kits candidats. Si le temps d'arrêt de (i) est inférieur au temps d'arrêt d'un kit candidat de (ii), l'option choisie sera (i) : récupérer la pièce directement dans la zone de kitting.

Ensuite, si l'option (ii) est choisie, le kit candidat sélectionné sera le kit candidat le plus proche, c'est-à-dire, le kit candidat avec  $\Delta t_0$  le plus faible à condition que l'impact sur le kit candidat soit inférieur à un temps de cycle de la ligne de production. Cela veut dire que l'on autorise au kit candidat choisi de déclencher une alerte au moment de son arrivée à la position du kit défectueux sous réserve que la pièce manquante arrive durant son temps de cycle sur le poste. En respectant ces critères, l'option choisie pour le kit défectueux N°2220082 serait donc de récupérer la pièce manquante auprès du kit candidat N°2230012.

## 3.5 Définition des KPIS globaux

Nous comparons la méthode multi-agents proposée avec la méthodologie utilisée actuellement dans le site de production à travers le calcul des KPIS suivants sous deux configurations différentes (i) en supposant que la liste d'alertes est un ensemble d'événements indépendants et (ii) en prenant en compte l'impact que le déclenchement d'une alerte et le temps nécessaire à sa solution a sur la suivante.

- Temps d'arrêt total de la ligne d'assemblage
- Temps de réaction à l'alerte
- Temps de déplacement total de l'opérateur

### 3.5.1 Alertes comme événements indépendants

La considération des alertes comme des événements indépendants permet de réaliser les calculs de KPIS globaux de la journée de production avec les estimations obtenues avec les calculs de la Section 3.4.2 et avec la sélection effectuée en suivant la Section 3.4.3.

Pour la situation actuelle nous calculons les KPIS globaux en utilisant les estimations réalisées liées à l'aller-retour à la zone de kitting. Le temps d'arrêt global correspond à la somme des temps d'arrêt de chaque scénario (cinquième colonne du Tableau 3.10). Le temps de réaction et le temps de déplacement total de l'opérateur correspondent à la somme du temps de déplacement à la zone de kitting de tous les scénarios (cinquième colonne du Tableau 3.9).

Pour la méthode multi-agents proposée, les sommes se feront en fonction de l'option choisie pour chaque scénario : récupération de la pièce manquante (i) de la zone de kitting ou (ii) d'un kit candidat. Le temps d'arrêt considéré pour chaque scénario est a) le temps d'arrêt lié au déplacement vers la zone de kitting (cinquième colonne du Tableau 3.10) si l'option (i) a été sélectionnée, ou b) le temps d'arrêt lié au déplacement vers le kit candidat (sixième colonne du Tableau 3.10) si l'option (ii) a été

### 3.5. DÉFINITION DES KPIS GLOBAUX

---

choisie. Le temps d'arrêt total de la ligne est ainsi la somme du temps d'arrêt considéré pour tous les scénarios d'alerte.

De façon équivalente, pour le temps de réaction, nous considérons soit le temps de déplacement à la zone de kitting si (i), soit le temps de déplacement au kit candidat ( $\Delta t_0$ ) si (ii). Pour le temps de déplacement total de l'opérateur nous considérons le temps de déplacement à la zone de kitting si (i) ou le temps de déplacement total de l'opérateur ( $\Delta t_{total}$ ) si (ii).

#### 3.5.2 Alertes comme événements non indépendants

Dans cette section, nous enlevons l'hypothèse du traitement des alertes en tant qu'événements indépendants afin de réaliser une estimation des KPIs plus proche de la réalité. Pour ce faire, avec les données calculées dans la Section 3.4.2.2 nous estimons l'impact qu'une alerte peut générer sur l'alerte suivante, en fonction de la date de début de l'alerte suivante et le temps de déplacement total de l'opérateur pour résoudre l'alerte initiale. Cet impact est calculé avec l'équation (3.4) pour l'option aller-retour à la zone de kitting et avec l'équation (3.5) pour chaque kit candidat.

$$(\text{Impact alerte})_{i+1} = (\text{Date début alerte})_i + \Delta t \textit{ kitting}_i - (\text{Date début alerte})_{i+1} \quad (3.4)$$

$$(\text{Impact alerte})_{i+1} = (\text{Date début alerte})_i + \Delta t \textit{ total}_i - (\text{Date début alerte})_{i+1} \quad (3.5)$$

En définissant (3.6), nous pouvons aussi utiliser l'équation (3.7) pour calculer l'impact sur l'alerte suivante.

$$(\text{Date fin alerte})_i = \begin{cases} (\text{Date début alerte})_i + \Delta t \textit{ kitting}_i, & \text{si récupération de la pièce dans la zone} \\ & \text{de kitting} \\ (\text{Date début alerte})_i + \Delta t \textit{ total}_i, & \text{si récupération de la pièce dans un kit} \\ & \text{candidat} \end{cases} \quad (3.6)$$

$$(\text{Impact alerte})_{i+1} = (\text{Date fin alerte})_i - (\text{Date début alerte})_{i+1} \quad (3.7)$$

Il est nécessaire de tenir compte de l'aspect cumulatif dans le temps de cet impact afin de pouvoir appliquer la démarche à l'ensemble de scénarios. Pour ce faire, après la sélection de l'endroit de récupération de la pièce manquante, nous calculons l'impact cumulé d'une alerte sur la suivante avec l'équation (3.8) et donc l'impact cumulé pour l'alerte actuelle avec l'équation (3.9).

$$(\text{Impact cumulé alerte})_{i+1} = (\text{Date fin alerte})_i - (\text{Date début alerte})_{i+1} + (\text{Impact cumulé alerte})_i \quad (3.8)$$

$$(\text{Impact cumulé alerte})_i = (\text{Date fin alerte})_{i-1} - (\text{Date début alerte})_i + (\text{Impact cumulé alerte})_{i-1} \quad (3.9)$$

En intégrant l'impact cumulé sur les alertes, le calcul des KPIs pour la méthode actuellement utilisée en usine se ferait à l'aide des équations (3.10) et (3.11) pour le temps d'arrêt total de la ligne d'assemblage et le temps de réaction, respectivement. Le temps de déplacement total de l'opérateur reste inchangé. Il se calcule comme expliqué à la Section 3.5.1.

— Temps d'arrêt total de la ligne d'assemblage :

$$\text{Temps arrêt kitting} = \sum_{i=1}^n (\Delta t \text{ kitting}_i + (\text{Impact cumulé alerte kitting})_i - \text{Cycle time}) \quad (3.10)$$

— Temps de réaction à l'alerte :

$$\text{Temps réaction kitting} = \sum_{i=1}^n (\Delta t \text{ kitting}_i + (\text{Impact cumulé alerte kitting})_i) \quad (3.11)$$

Pour la méthode multi-agents proposée, deux cas de figure sont étudiés a) avec un seul opérateur qui fait les deux déplacements (l'aller-retour au kit candidat et l'aller-retour à la zone kitting) et b) avec deux opérateurs répondant aux alertes, l'opérateur 1 (OP1) et l'opérateur 2 (OP2). L'opérateur 1 s'occupe d'effectuer le premier trajet (le trajet au kit candidat ou le trajet à la zone de kitting, en cas de récupération de la pièce directement de la zone de kitting) et l'opérateur 2 se charge d'effectuer le trajet à la zone de kitting, en cas de récupération de la pièce manquante auprès d'un kit candidat.

Le temps d'arrêt total de la ligne d'assemblage et le temps de réaction à l'alerte pour ces deux configurations se calculeraient comme suit. La méthode de calcul pour le temps de déplacement total du ou des opérateurs reste celle décrite à la Section 3.5.1.

#### **méthode multi-agents avec un seul opérateur**

— Temps d'arrêt total de la ligne d'assemblage :

$$\text{Temps arrêt méthode MAS} = \sum_{i=1}^n ((\text{Temps arrêt})_i + (\text{Impact cumulé alerte})_i) \quad (3.12)$$

— Temps de réaction à l'alerte :

$$\text{Temps réaction méthode MAS} = \sum_{i=1}^n ((\text{Temps réaction})_i + (\text{Impact cumulé alerte})_i) \quad (3.13)$$

Où le *Temps arrêt<sub>i</sub>* et le *Temps réaction<sub>i</sub>* dépendent de l'endroit choisi pour la récupération de la pièce manquante pour chaque scénario *i*.

#### méthode multi-agents avec deux opérateurs

Dans ce cas, les deux opérateurs étant indépendants, nous devons calculer l'impact cumulé d'une alerte sur l'autre pour chaque opérateur. Ce calcul dépend de l'option prise quant à l'endroit pour récupérer la pièce manquante : (i) la zone de kitting ou (ii) un des kits candidats. Ainsi les équations (3.14) et (3.15) ou (3.16) et (3.17) sont utilisées selon le cas.

— Si l'option choisie est de récupérer la pièce dans la zone de kitting

$$\begin{aligned} (\text{Impact cumulé alerte OP1})_i &= (\text{Date fin alerte OP1})_{i-1} - (\text{Date début alerte})_i \\ &+ (\text{Impact cumulé alerte OP1})_{i-1} \end{aligned} \quad (3.14)$$

Où

$$(\text{Date fin alerte OP1})_i = (\text{Date début alerte})_i + \Delta t \textit{ kitting}_i$$

$$\begin{aligned} (\text{Impact cumulé alerte OP2})_i &= (\text{Date début alerte})_{i-1} - (\text{Date début alerte})_i \\ &+ (\text{Impact cumulé alerte OP2})_{i-1} \end{aligned} \quad (3.15)$$

— Si l'option choisie est de récupérer la pièce auprès d'un kit candidat

$$\begin{aligned} (\text{Impact cumulé alerte OP1})_i &= (\text{Date fin alerte OP1})_{i-1} - (\text{Date début alerte})_i \\ &+ (\text{Impact cumulé alerte OP1})_{i-1} \end{aligned} \quad (3.16)$$

Où

$$(\text{Date fin alerte OP1})_i = (\text{Date début alerte})_i + \Delta t_0$$

$$\begin{aligned} (\text{Impact cumulé alerte OP2})_i &= (\text{Date fin alerte OP2})_{i-1} - (\text{Date début alerte})_i \\ &+ (\text{Impact cumulé alerte OP2})_{i-1} \end{aligned} \quad (3.17)$$

Où

$$(\text{Date fin alerte OP2})_i = (\text{Date début alerte})_i + \Delta t \textit{ kitting}_i$$

Après avoir calculé les impacts pour les deux opérateurs et pour chaque scénario (ou pour chaque alerte), nous pouvons calculer le temps d'arrêt de la ligne de production correspondant à l'activité de chaque opérateur pour toutes les alertes avec l'équation (3.18) pour l'opérateur 1 et avec l'équation (3.19) pour l'opérateur 2. En lisant les équations, nous pouvons remarquer deux éléments. D'abord, que le temps d'arrêt pour l'opérateur 1 est lié au temps de cycle de la ligne de production parce qu'il répond à l'alerte du kit défectueux. Deuxièmement, que le temps d'arrêt pour l'opérateur 2 dépend du temps de cycle ainsi que du temps nécessaire pour que le kit candidat arrive à la position du kit défectueux. Par conséquent, l'opérateur 2 a plus de temps disponible avant de perturber le fonctionnement de la ligne d'assemblage.

$$(\text{Temps arrêt OP1})_i = \Delta t_1 + (\text{Impact cumulé alerte OP1})_i - \text{Cycle time} \quad (3.18)$$

$$(\text{Temps arrêt OP2})_i = \Delta \textit{kitting}_i + (\text{Impact cumulé alerte OP2})_i - (\text{Cycle time} + \text{Temps pos défectueuse}) \quad (3.19)$$

Où

$$\text{Temps pos défectueuse} = \text{Distance kit candidat} / S_{PL}$$

Finalelement les éléments  $\textit{Temps arrêt}_i$  correspondants à chaque scénario, utilisés pour calculer le temps d'arrêt global de la journée (équation (3.21)) sont définis par l'équation (3.20) en fonction de l'endroit de récupération de la pièce manquante.

$$(\text{Temps arrêt})_i = \begin{cases} (\text{Temps arrêt kitting})_i + (\text{Impact cumulé alerte OP1})_i, & \text{si kitting} \\ \max((\text{Temps arrêt OP1})_i, (\text{Temps arrêt OP2})_i), & \text{si kit candidat} \end{cases} \quad (3.20)$$

$$\text{Temps arrêt MAS 2} = \sum_{i=1}^n (\text{Temps arrêt})_i \quad (3.21)$$

Quant au temps de réaction à la suite de l'alerte, nous le calculons avec l'équation (3.22), en prenant compte du temps de réaction calculé à la Section 3.5.1 et de l'impact cumulé des activités de l'opérateur 1.

$$\text{Temps réaction MAS 2} = \sum_{i=1}^n (\text{Temps réaction})_i + (\text{Impact cumulé alerte OP1})_i \quad (3.22)$$

Où le  $\textit{Temps réaction}_i$  dépend de l'endroit choisi pour la récupération de la pièce manquante pour chaque scénario  $i$ .

### 3.6 Validation expérimentale

Pour tester notre méthode basée sur les MAS dans un scénario réel, nous avons utilisé des données historiques collectées dans une véritable usine d'assemblage de l'industrie automobile. Ces données ont servi à modéliser l'usine d'assemblage aux moments où un opérateur a déclenché des alertes à cause d'une pièce manquante dans l'un des kits. A la différence de l'étude de cas présentée à la Section 3.2, ici nous ne nous focalisons pas uniquement sur un scénario ponctuel mais sur l'ensemble des scénarios



### 3.6. VALIDATION EXPÉRIMENTALE

d’alerte qui ont eu lieu au cours d’une journée de production donnée. L’usine d’assemblage de notre étude est celle décrite dans la Section 3.2.1 et représentée à nouveau par la Figure 3.5.

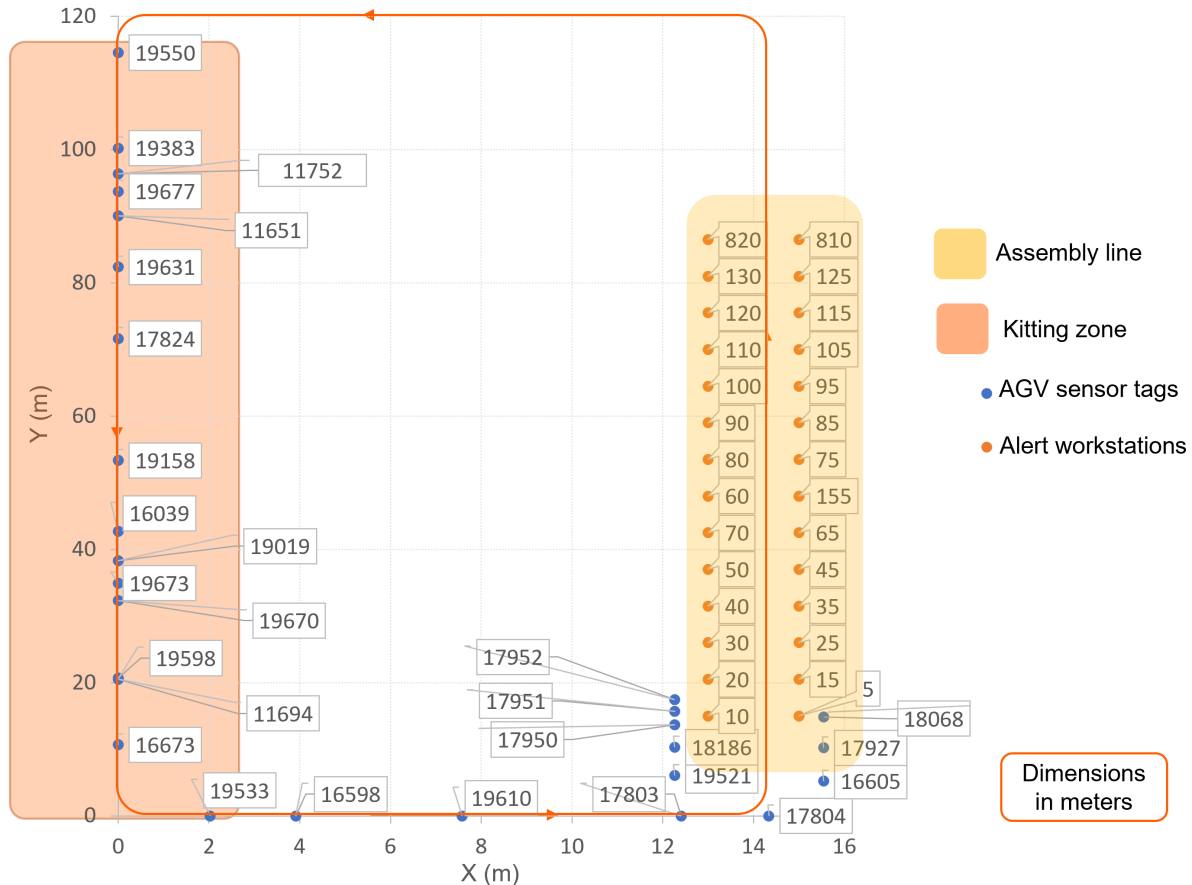


FIGURE 3.5 – Disposition physique de l’étude de cas

#### 3.6.1 Sélection de l’endroit de récupération de la pièce manquante

Nous avons suivi la méthodologie de la Section 3.4.1 pour constituer l’état du système de production à l’instant du déclenchement de chaque alerte, celle de la Section 3.4.2.1 pour identifier les kits candidats, les coûts de récupération de la pièce manquante auprès de chaque kit candidat et finalement, basées sur ces calculs, celle de la Section 3.4.3 pour sélectionner le meilleur endroit pour récupérer la pièce manquante. Le Tableau 3.11, présente un extrait du DataFrame résultant après l’application de ces méthodologies sur des données historiques de la journée de production étudiée.

Le Tableau 3.11 rapporte des informations relatives au nombre des scénarios apparus et traités durant la période d’analyse, l’identification et le type de produit du kit défectueux ou du meilleur kit candidat sélectionné et la date de déclenchement de chaque alerte. Il fournit également, les résultats

### 3.6. VALIDATION EXPÉRIMENTALE

TABLEAU 3.11 – Extrait du DataFrame résultant suite à la prise de décision d’endroit de récupération de la pièce manquante (zone de kitting ou au meilleur kit candidat)

Sce	Vehicle /kit ID	Product type	Alert date	Journey kitting zone $\Delta t_{kitting}$ (s)	Time line stop kitting (s)	Journey candid. kit $\Delta t_0$ (s)	Time line stop candid. kit (s)	Journey kitting zone $\Delta t_1$ (s)	Global pert. $\Delta t_{total}$ (s)	Time impact candid. kit (s)	Option
1	2230012	Type 5	01/06/2023 05 :20	88.98	28.98	27.50	0.00	84.11	111.61	0.00	candidat
2	2220082	Type 5	01/06/2023 05 :22	65.43	5.43	27.50	0.00	61.61	89.11	0.00	candidat
3	2120090	Type 18	01/06/2023 05 :24	60.20	0.20	100.83	40.83	75.25	176.09	0.00	kitting
4	2240032	Type 5	01/06/2023 05 :27	60.20	0.20	76.66	16.66	71.30	147.96	0.00	kitting
5	2230131	Type 13	01/06/2023 05 :32	60.20	0.20	27.50	0.00	63.84	91.34	0.00	candidat
6	2120080	Type 18	01/06/2023 05 :38	60.20	0.20	64.17	4.17	69.32	133.48	0.00	kitting
...	...	...	...	...	...	...	...	...	...	...	...
80	2430014	Type 5	01/06/2023 12 :28	60.20	0.20	45.83	0.00	66.51	112.34	0.00	candidat
81	2310131	Type 7	01/06/2023 12 :32	60.20	0.20	76.66	16.66	71.30	147.96	0.00	kitting
82	2310101	Type 18	01/06/2023 12 :38	60.20	0.20	24.32	0.00	63.40	87.72	0.00	candidat
83	2310059	Type 18	01/06/2023 12 :38	60.20	0.20	18.33	0.00	62.58	80.91	0.00	candidat
84	2250078	Type 5	01/06/2023 12 :48	60.20	0.20	59.96	0.00	68.66	128.62	0.00	candidat
85	2310163	Type 13	01/06/2023 12 :58	60.20	0.20	36.67	0.00	65.16	101.82	0.00	candidat

### 3.6. VALIDATION EXPÉRIMENTALE

---

des estimations des temps de déplacement de l'opérateur ainsi que les impacts quant aux temps d'arrêt de la ligne d'assemblage ou de perturbation sur le kit candidat sélectionné et, sur la dernière colonne, l'endroit choisi pour servir de fournisseur de la pièce manquante : "candidat" quand le fournisseur est un kit candidat et "kitting" quand la pièce est récupérée directement de la zone de kitting.

Le DataFrame complet qui a servi à faire le Tableau 3.11 nous permet de souligner plusieurs éléments. Premièrement, parmi les 85 scénarios d'alerte étudiés, 9 scénarios ne comportaient pas de kits candidats, c'est-à-dire le type de kit défectueux n'était similaire à aucun des kits qui se trouvaient entre le poste de travail déclenchant l'alerte et la sortie de la zone de kitting. Pour ces cas-ci, la méthode multi-agents proposée n'était donc pas très pertinente et la pièce manquante devait être récupérée de la zone de kitting. Deuxièmement, sur l'ensemble des 85 scénarios, l'option de récupérer la pièce manquante auprès d'un kit candidat au lieu de la récupérer de la zone de kitting est plus convenable pour 61 scénarios, soit 71.8% des cas (cf. Figure 3.6). De plus, si l'on ne considère pas les 9 scénarios ne comportant pas de kits candidats, le rapport monterait à 80.3%. Par conséquent, la méthode multi-agents proposée permettrait de mieux répondre à 71.8% des alertes de la journée de production ou à 80.3% des alertes où cette méthode serait applicable. Troisièmement, l'impact sur les kits candidats fournissant la pièce manquante est majoritairement inexistant. Sur l'ensemble de scénarios étudiés, uniquement 11 kits candidats ont présenté un impact.

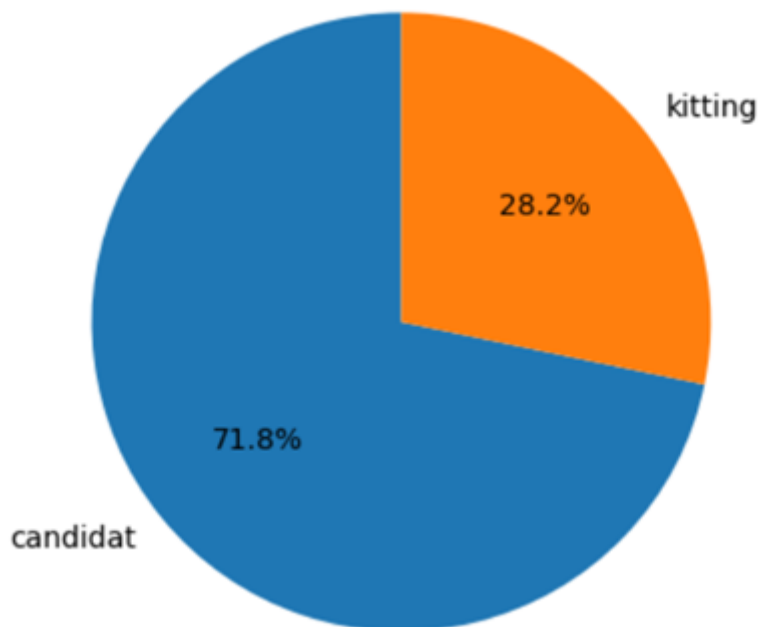


FIGURE 3.6 – Distribution des choix de l'endroit de récupération de la pièce manquante sur l'ensemble des scénarios

### 3.6.2 KPIs résultants en traitant les alertes comme événements indépendants

Grâce aux informations du Tableau 3.11 nous pouvons calculer les KPIs globaux de la journée de production en suivant la méthodologie décrite à la Section 3.5. Le Tableau 3.12 présente les KPIs résultants pour la méthode multi-agents proposée et de la méthode actuelle dans le cas où les alertes sont traitées comme un ensemble d'événements indépendants. Le Tableau 3.12 présente donc le temps d'arrêt, le temps de réaction et le temps de déplacement totaux de la journée en minutes, pour chacune des méthodes et les rapports entre leurs résultats.

En confrontant les résultats des deux méthodes, nous pouvons apercevoir que sous ces conditions, la méthode proposée est plus intéressante que la méthodologie utilisée actuellement. Le temps d'arrêt est 85% inférieur pour la méthode proposée que pour la méthode actuelle. Le temps de réaction, pour sa part, est 47% inférieur. Le seul KPI qui a une valeur supérieure pour la méthode proposée que pour la méthodologie actuelle est le temps de déplacement de l'opérateur. Ce résultat est tout à fait cohérent puisqu'avec la méthode multi-agents l'opérateur doit effectuer deux déplacements au lieu d'un seul, comme préconisé par la méthodologie actuelle. Malgré l'augmentation du temps de travail de l'opérateur, la méthode reste financièrement bénéfique grâce à la réduction du temps d'arrêt de production.

Ces résultats confirment les résultats et conclusions de la Section 3.2 et, même si l'hypothèse du traitement des alertes comme des événements indépendants n'est pas tout à fait représentative de la réalité, ils permettent d'avoir un premier aperçu prometteur du potentiel de la méthode multi-agents proposée face à la méthodologie actuelle.

TABLEAU 3.12 – KPIs résultants pour la méthode multi-agents proposée et la méthodologie actuelle en traitant les alertes comme des événements indépendants

Méthode	Temps d'arrêt (min)	Temps de réaction (min)	Temps déplacement opérateur (min)
Proposée	2.87	55.39	133.70
Actuelle	19.60	104.52	104.52
Rapport	0.15	0.53	1.28

### 3.6.3 KPIs résultants en traitant les alertes comme événements non indépendants

Avant de pouvoir calculer les KPIs globaux pour la méthode proposée et pour la méthode actuelle en traitant les alertes comme des événements non indépendants, nous devons calculer l'impact en termes de temps qu'une alerte peut avoir sur l'alerte suivante ainsi que le cumul de cet impact au cours du temps.

La Figure 3.7 présente le résultat du calcul de cet impact sur les 85 scénarios étudiés en suivant la

### 3.6. VALIDATION EXPÉRIMENTALE

méthodologie décrite à la Section 3.5.2. Sur cette figure, la courbe bleue représente l'impact individuel du scénario en question sur l'alerte ou le scénario suivant et la courbe orange représente le cumul de cet impact au cours du temps. La figure met également en évidence comment, en prenant en compte le cumul, une alerte peut non seulement affecter l'alerte suivante mais aussi celles d'après. Ceci est particulièrement évident autour des scénarios 18, 38 et 78.

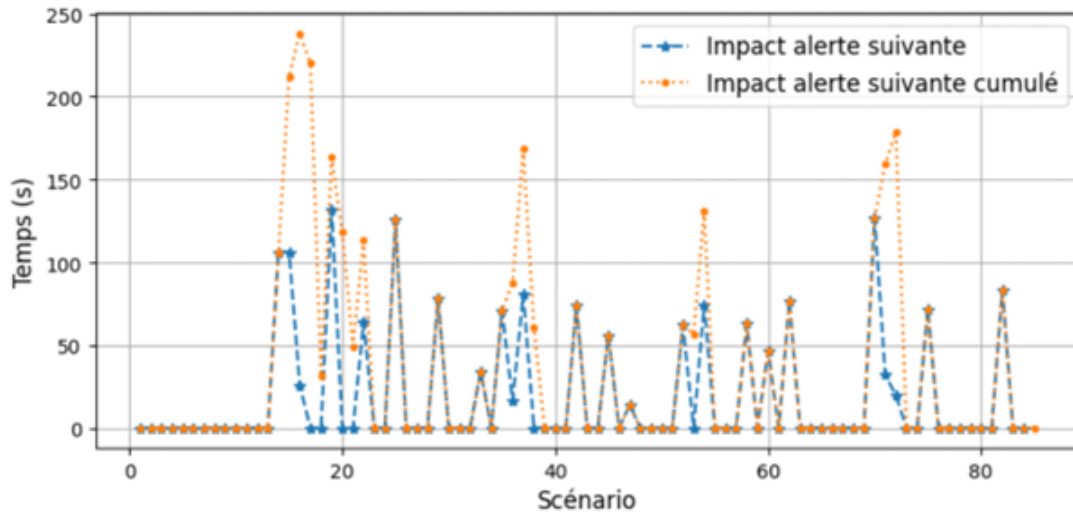


FIGURE 3.7 – Impact individuel et cumulé d'une alerte sur l'alerte suivante après la sélection du meilleur kit candidat ou de la récupération de la pièce de la zone de kitting

Avec les informations des impacts cumulés sur chaque scénario, nous pouvons calculer les KPIs pour la méthode actuelle, la méthode multi-agents proposée avec un seul opérateur et la méthode multi-agents proposée avec deux opérateurs. Les Tableaux 3.13 et 3.14 rapportent ces résultats.

TABLEAU 3.13 – KPIs résultants pour la méthode multi-agents proposée avec un seul opérateur et la méthodologie actuelle en traitant les alertes comme des événements non indépendants

Méthode	Temps d'arrêt (min)	Temps de réaction (min)	Temps déplacement opérateur (min)
Proposée 1 opérateur	54.16	106.23	133.70
Actuelle	44.71	127.62	104.52
Rapport	1.21	1.02	1.28

En comparant les résultats présentés par le Tableau 3.12 et ceux des Tableaux 3.13 et 3.14, nous pouvons remarquer qu'il y a des différences importantes et donc que la prise en compte de l'impact cumulé entraîne des répercussions significatives.

Concernant les résultats sur la méthode actuelle, la différence principale se trouve sur les estimations de temps d'arrêt de la ligne d'assemblage. Les résultats répertoriés par les Tableaux 3.13 informent

### 3.6. VALIDATION EXPÉRIMENTALE

---

TABLEAU 3.14 – KPIs résultants pour la méthode multi-agents proposée avec deux opérateurs et la méthodologie actuelle en traitant les alertes comme des événements non indépendants

Méthode	Temps d'arrêt (min)	Temps de réaction (min)	Temps déplacement opérateur (min)
Proposée 2 opérateurs	12.57	68.49	133.70
Actuelle	44.71	127.62	104.52
Rapport	0.28	0.66	1.28

que la ligne d'assemblage aurait dû être arrêtée pendant 44.71 minutes alors que le Tableau 3.12 présente un temps total d'arrêt de 19.60 minutes. Le temps d'arrêt montre une augmentation de plus du double de la valeur initialement estimée. Malgré le fait que la valeur de 44.71 minutes ne reflète pas le temps d'arrêt réel de ligne de production pour la journée analysée, l'augmentation par rapport à la valeur du Tableau 3.12 s'explique par la prise en compte de l'impact cumulé dans les calculs. En d'autres termes, il est normal que la ligne d'assemblage s'arrête plus car l'opérateur tarde plus pour répondre aux alertes. A l'apparition de certaines alertes, l'opérateur ne peut pas répondre immédiatement car il est toujours occupé à répondre à l'alerte précédente.

La différence entre le temps d'arrêt estimés et la réalité du système de production est due à plusieurs facteurs ou hypothèses prises pour la réalisation des calculs. Pour les calculs, un seul opérateur répond de façon séquentielle à chaque alerte, le temps d'arrêt est comptabilisé dès qu'une pièce est manquante pendant un temps supérieur au temps de cycle de la ligne et les temps de pauses des opérateurs sur la journée ne sont pas pris en compte. Dans la réalité, il peut y avoir plusieurs opérateurs qui répondent aux alertes, un opérateur peut chercher plusieurs pièces manquantes au même temps, le montage de la pièce manquante peut être effectué en dehors du poste qui déclare l'alerte et dans certains cas les pauses servent à répondre à certaines alertes.

Ainsi, malgré l'écart entre la réalité et les résultats des Tableaux 3.13 et 3.14 pour la méthode actuellement utilisée sur le site de production, les résultats des tableaux en question restent cohérents et la méthode de calcul permet de comparer la méthodologie de réponse aux alertes actuelles et les méthodes proposées sous les mêmes conditions et sous des conditions basées sur des données réelles.

Concernant les résultats pour la méthode proposée avec un seul opérateur, nous pouvons remarquer le même phénomène d'augmentation de temps d'arrêt et de réaction des résultats du Tableau 3.13 par rapport aux résultats du Tableau 3.12. Pour la méthode multi-agents proposée, la prise en compte de l'impact cumulé des alertes a une forte incidence sur les résultats.

Le temps d'arrêt et le temps de réaction croissent de manière très importante. De la même façon que pour les résultats de la méthode actuelle, le fait que l'opérateur soit occupé au moment de déclenchement de l'alerte produit des arrêts de la ligne d'assemblage. De plus, étant donné que la

### 3.7. CONCLUSION

---

méthode proposée augmente le temps de déplacement total de l'opérateur par rapport à la méthode actuelle, l'augmentation de temps d'arrêt et du temps de réaction et d'autant plus importante. Ainsi, en comparant les résultats de la méthode proposée avec un seul opérateur avec la méthode actuelle, la méthode actuelle semble la plus pertinente parmi les deux.

Cependant, vus les bons résultats de la méthode proposée présentés au Tableau 3.12 et que cette méthode est pénalisée principalement par l'augmentation du temps de déplacement total de l'opérateur et donc le cumul de l'impact d'une alerte sur les suivantes, l'évaluation de cette méthode en considérant un partage des déplacements entre deux opérateurs se révèle pertinent.

Les résultats rapportés par le Tableau 3.14 concernant l'évaluation de la méthode proposée avec deux opérateurs indiquent aussi une augmentation sur les temps d'arrêt de production et de réaction par rapport aux résultats du Tableau 3.12. Néanmoins, cette augmentation est beaucoup moins importante que pour le cas considérant un seul opérateur. Dans cette configuration, la méthode proposée s'avère plus intéressante que la méthode utilisée actuellement. Malgré le fait d'avoir un temps de déplacement total des deux opérateurs plus important que la méthode actuelle, elle permet de réduire le temps d'arrêt de la ligne d'environ 72 % et le temps de réaction d'environ 46 %.

Il est important de préciser que, comme commenté précédemment, une solution qui donne lieu à une baisse si importante des arrêts de la ligne d'assemblage reste intéressante financièrement nonobstant l'augmentation de la charge de travail d'un ou des opérateurs. Les gains potentiels ont été estimés mais par souci de confidentialité, ces données ne seront pas rapportées dans le présent manuscrit.

### 3.7 Conclusion

Dans ce chapitre, nous avons appliqué la méthode multi-agents proposée dans le Chapitre 2, pour répondre aux alertes de pièce manquante sur une ligne d'assemblage du Groupe Renault, utilisant le processus d'approvisionnement de kitting.

Dans un premier temps, nous avons testé notre méthode multi-agents dans un scénario d'alerte ponctuel. Les résultats obtenus pour le scénario étudié ont démontré l'efficacité de l'approche proposée pour fournir des options pertinentes aux opérateurs de la ligne d'assemblage afin de réduire les temps de réaction lorsqu'un kit est défectueux, prévenant ainsi des arrêts inutiles de la ligne de production. La meilleure option suggérée a permis de réduire le temps de réaction de plus de 70%.

Malgré les bons résultats, étant donné qu'au cours d'une journée de production il y peut y avoir plus d'une alerte, l'évaluation de la méthode proposée sur un seul scénario ponctuel n'est pas suffisante pour prouver la valeur ajoutée de son application sur le terrain industriel.

Ainsi, dans un deuxième temps, nous avons présenté une démarche pour évaluer la méthode multi-agents proposée, non seulement sur un scénario d'alerte ponctuel, mais sur un ensemble de scénarios

### 3.7. CONCLUSION

---

qui ont lieu au cours d'une journée de production.

Nous avons collecté des données historiques provenant de plusieurs systèmes d'informations d'une usine d'assemblage du Groupe Renault. Nous avons agrégé et traité ces données réelles afin de reconstituer l'état du système de production à tous les moments où une alerte a été déclenchée par un opérateur. Ensuite, nous nous sommes servis de la méthode multi-agents pour identifier les kits candidats permettant de répondre à l'alerte levée sur chacun des scénarios. Finalement, nous avons sélectionné la solution la plus pertinente pour chaque scénario et 3 KPIs ont été calculés pour la globalité de la période d'analyse, en traitant les alertes comme des scénarios indépendants et comme des scénarios non indépendants. Grâce à ces KPIs nous avons pu comparer la méthodologie actuellement utilisée pour répondre à ce type d'alertes par le site de production avec 2 variantes de la méthode multi-agents proposée : (i) avec un opérateur et (ii) avec deux opérateurs.

Les résultats obtenus pour l'ensemble des 85 scénarios étudiés confirment le potentiel de l'approche multi-agents proposée pour fournir des options pertinentes aux opérateurs de la ligne d'assemblage. L'option de récupérer la pièce manquante auprès d'un kit candidat, au lieu de la récupérer de la zone de kitting, s'est révélée plus convenable pour 71.8% des scénarios. Dans le cas où les alertes ont été traitées comme des scénarios indépendants, la méthode multi-agents a présenté un temps d'arrêt et un temps de réaction respectivement inférieurs de 85% et 47% à ceux de la méthodologie actuelle. Seul l'indicateur du temps de déplacement de l'opérateur a été supérieur à une hauteur de 28%. Dans le cas où les alertes ont été traitées comme des scénarios non indépendants, la méthode multi-agents présente des résultats moins intéressants que la méthodologie actuelle à cause de l'augmentation du temps de déplacement des opérateurs. Néanmoins, en appliquant la méthode multi-agents avec deux opérateurs, les résultats sont à nouveau, 72% pour le temps d'arrêt et 46% quant au temps de réaction, meilleurs pour la méthode multi-agents comparée à la méthodologie actuelle.

En ayant constaté la pertinence des approches multi-agents pour des situations de réaction face à une perturbation, les travaux futurs seront consacrés à l'étude des approches multi-agents pour la détection et l'anticipation d'anomalies grâce à l'IA.



### 3.7. CONCLUSION

---

## Chapitre 4

# Modèle de détection anticipée de la perturbation d'une pièce manquante dans un kit

### Contenu

---

4.1	Introduction . . . . .	<b>102</b>
4.2	Objectif de l'agrégation des données . . . . .	<b>102</b>
4.3	Collecte des données . . . . .	<b>104</b>
4.3.1	Spotfire pour les données logistiques . . . . .	104
4.3.2	Le serveur Synersight des AGVs . . . . .	105
4.3.3	Les tableaux manufacturing sur GCP . . . . .	106
4.4	Stratégie d'agrégation . . . . .	<b>107</b>
4.4.1	Agrégation flux rang 0 et rang 1 . . . . .	107
4.4.2	Composition et passage des véhicules dans la zone de montage des portes . . . . .	109
4.4.3	Extraction des alertes à partir des appels DWS . . . . .	111
4.5	Création du <i>data set</i> . . . . .	<b>112</b>
4.5.1	Liaison code identifiant véhicule -support physique . . . . .	112
4.5.2	Liaison code véhicule - alerte . . . . .	113
4.5.3	Pré-traitement et transformation des données . . . . .	114
4.5.4	Limitations et difficultés . . . . .	115
4.6	Environnement et outils d'entraînement . . . . .	<b>116</b>
4.7	Analyse exploratoire des données . . . . .	<b>117</b>
4.7.1	Analyse du comportement des variables . . . . .	117
4.7.2	Distribution des classes . . . . .	119
4.8	Métriques d'évaluation des modèles . . . . .	<b>123</b>
4.9	Stratégie de modélisation . . . . .	<b>124</b>
4.9.1	Modélisation avant équilibrage des données . . . . .	124
4.9.2	Modélisation après équilibrage des données . . . . .	128
4.10	Conclusion . . . . .	<b>136</b>

---

### 4.1 Introduction

Les Chapitres 2 et 3 ont présenté une méthode multi-agents ayant pour objectif de réagir aux alertes de type pièce manquante dans un kit. Ces chapitres ont également présenté les résultats de l'application de cette méthode sur un ensemble de scénarios d'alerte réels provenant d'une usine de montage automobile.

Etant donné que le problème de la pièce manquante dans un kit est souvent identifié quand le kit est déjà sur la ligne de montage, l'objectif de ce chapitre est de proposer une démarche pour le détecter de manière anticipée, juste à la sortie de la zone de kitting. Pour ce faire, nous exploitons les données provenant du système multi-agents présenté dans les chapitres précédents. De cette façon, le système multi-agents proposé pourra réagir non seulement aux alertes déclenchées par les opérateurs de la ligne de montage mais aussi par le système de détection proposé ici.

La première partie de ce chapitre est dédiée à la présentation de la méthodologie suivie pour la constitution d'un *data set* ayant pour but l'entraînement des modèles de Machine Learning (ML) pour la détection de ces anomalies. Dans cette partie, nous décrivons les systèmes d'information utilisés pour collecter les données, la stratégie de collecte et agrégation des données ainsi que les manipulations effectuées pour la construction du *data set*.

La deuxième partie du chapitre présente la phase de modélisation et exploitation des données. Nous détaillons l'environnement et les outils utilisés pour la création des modèles ML, nous présentons une analyse exploratoire des données et finalement nous présentons les résultats de l'évaluation des modèles créés sous différentes configurations.

### 4.2 Objectif de l'agrégation des données

Les systèmes d'information présentés dans la Section 1.3.5 permettent de capter des informations d'un nombre important d'opérations du processus logistique en étude, ainsi que de leurs répercussions sur la ligne de montage de portes.

Pour rappel, du côté du processus logistique, d'abord, grâce à Reflex et au Data Lake de Google, il est possible d'être au courant de mouvements des palettes jusqu'à l'arrivée à la zone de kitting. Deuxièmement, avec le serveur Synersight, la dynamique des AGV peut être suivie. Enfin, la consommation de pièces peut être déduite en utilisant les informations des séquences de production stockées sur PSFV.

Malheureusement, toutes les données présentes dans ces bases de données ou dans ces systèmes d'information sont stockées séparément. La fusion de ces flux de données permettrait d'avoir une vision d'ensemble du processus logistique. Néanmoins, dû à l'hétérogénéité des flux et des sources de données,

## 4.2. OBJECTIF DE L'AGRÉGATION DES DONNÉES

cette agrégation représente un défi important [Zenisek et al., 2022].

La Figure 4.1 présente la stratégie générale de la solution d'agrégation que nous proposons, afin d'effectuer la liaison de ces bases de données. L'image met en évidence, notamment, les clés principales qui mettent en relation les différents flux de données. Ce processus a pour objectif de créer une seule base de données, contenant la globalité des informations d'approvisionnement définissant chacun des kits livrés à la chaîne de montage.

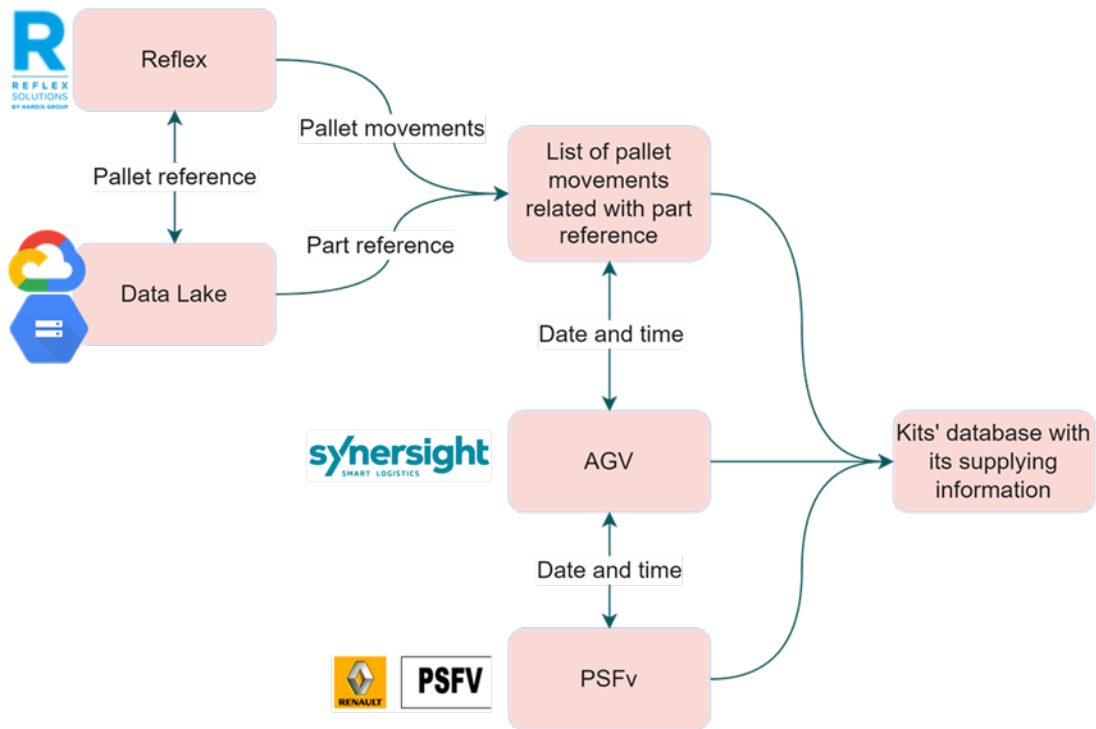


FIGURE 4.1 – Agrégation des données d'approvisionnement des kits

Du côté de la ligne de montage, grâce au système DWS, nous pouvons être au courant de certains des problèmes que les opérateurs de la ligne de montage peuvent rencontrer pendant la réalisation de leurs tâches. De façon similaire que pour la partie logistique, avec les informations de DWS nous pouvons créer une base de données contenant les informations d'alerte ou de conformité des kits, une fois qu'ils ont été livrés et utilisés pendant le processus de montage.

Le but final de la création de ces deux bases de données est de les mettre en relation en utilisant le numéro de référence unique du véhicule associé à chaque kit (numéro PJI). De cette manière, nous aurons d'un côté, toutes les informations d'approvisionnement concernant le kit avant sa livraison et de l'autre côté, toutes les informations venant du processus de montage après la livraison des kits. Nous pourrions donc nous servir de ces deux bases de données pour créer un *data set* pour chercher

### 4.3. COLLECTE DES DONNÉES

---

à trouver des corrélations entre les alertes déclarées sur la ligne de montage concernant les kits et la configuration d’approvisionnement des kits (Figure 4.2). Ce *data set* sera employé ultérieurement pour entraîner des modèles IA afin de détecter des anomalies de manque de pièce dans les kits, en se basant sur les informations du processus de livraison des kits.

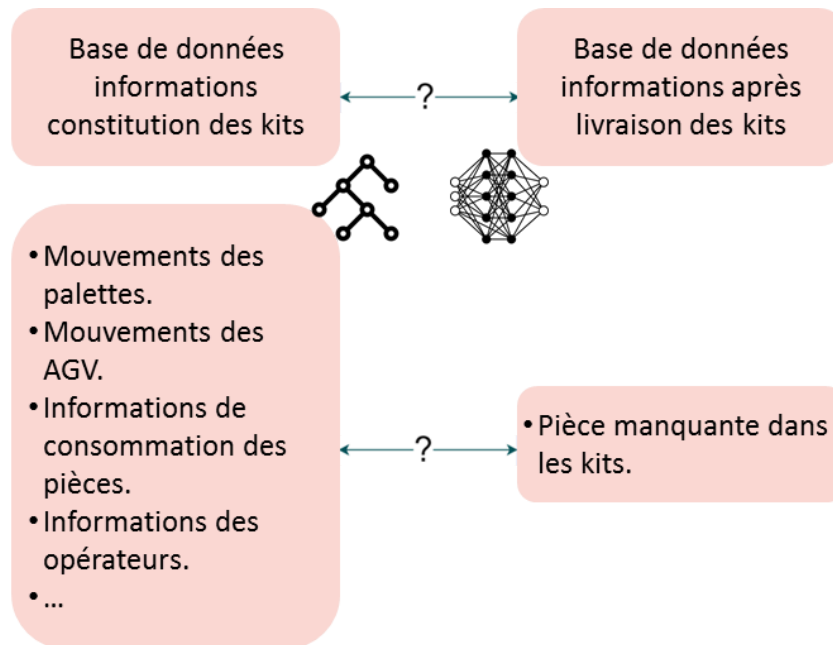


FIGURE 4.2 – Mise en relation des données d’approvisionnement et des données après la livraison des kits

## 4.3 Collecte des données

La collecte des données constitue une étape essentielle au début de l’opération d’agrégation. L’ensemble de données que nous utilisons pour l’étude actuelle provient de trois sources principales : Spotfire pour les données logistiques, le serveur Synersight des AGVs et les tableaux manufacturing sur GCP.

### 4.3.1 Spotfire pour les données logistiques

TIBCO Spotfire est une plate-forme d’analyse visuelle et de business intelligence développée par TIBCO Software, une société de logiciels informatiques. Spotfire permet aux utilisateurs de découvrir, visualiser, analyser et partager des données de manière interactive et intuitive. Largement utilisé dans les entreprises, comme Renault, il facilite la prise de décisions éclairées en utilisant des visualisations de données interactives, des tableaux de bord et des analyses prédictives. Concernant les données

### 4.3. COLLECTE DES DONNÉES

---

logistiques pertinentes pour notre cas d'étude, notamment les mouvements physiques de palettes de pièces ainsi que la liste des références utilisées dans la zone de kitting des portes, elles sont présentées sous forme de tableaux de bord préparés par des data analystes. Cela simplifie l'accès à ces informations sans avoir recours à l'outil Reflex. Nous avons pu effectuer l'extraction de plusieurs données.

Le premier extrait renseigne sur la liste des mouvements physiques dans le site industriel. Un mouvement physique est le déplacement d'un support physique par les caristes ou par les AGVs. Un support physique, ou palette, porte l'ensemble des emballages des références dans le site. Dans cet extrait, on trouve : les codes références, les numéros des supports qui portent ces références, les types des mouvements (mouvements de stockage ou de transfert) et les motifs des mouvements. Pour les mouvements de type stockage, le motif peut être une mise en stock interne ou une mise en kitting directement. Pour les mouvements de type transfert, il peut s'agir d'un transfert dans le stock ou d'une demande de réapprovisionnement des meubles de kitting à partir des supports placés déjà dans les stocks. L'extrait fournit également des informations sur les emplacements de départ, intermédiaires s'ils existent, et d'arrivée de chaque mouvement, ainsi que les dates de prise en charge du mouvement (lorsque les caristes prennent les supports depuis leurs emplacements initiaux) et les dates de validation du mouvement (lorsque les supports sont déposés dans leurs emplacements finaux).

Le deuxième extrait renseigne sur la liste de mouvements des emballages portés par chaque support physiques, que ce soit de petite taille (PE) ou de grande taille (GE). Ces mouvements sont appelés mouvements de Groupe d'Eléments Interchangeables (GEI). Comme pour les mouvements physiques, il existe plusieurs motifs de mouvements de GEI. Pour répondre à ce cas d'étude, on s'intéresse uniquement à ceux d'approvisionnement des meubles de kitting qui ont le code 430. A partir de cet extrait, on peut savoir pour chaque référence la quantité du mouvement de GEI et donc la quantité des pièces mises dans l'emplacement de kitting concerné.

Le troisième extrait renseigne sur la liste des références présentes dans la zone de kitting des portes. De plus, cet extrait renseigne sur la taille d'emballage de chaque référence.

#### 4.3.2 Le serveur Synergisight des AGVs

Comme expliqué dans la section précédente, ce serveur est connecté à l'ensemble des Tags au sol, localisés sur tout le site de production pour surveiller l'activité des AGV. Ce serveur remonte les informations d'une manière continue mais elles sont récupérables d'une façon hebdomadaire sous format CSV. Ce serveur donne des informations sur (i) les tracteurs autonomes présents entre l'entrée de la gare routière du site et les stocks internes et (ii) sur les AGVs qui circulent dans la zone de kitting.

### 4.3.3 Les tableaux manufacturing sur GCP

Ces tableaux sont des représentations des données PSFV et DWS créés par les équipes data analyste de Renault et hébergés dans GCP. Ces tableaux sont distribués dans des bases de données selon le processus qu’elles traitent. On peut y trouver une base de données pour les données PSFV concernant la composition en pièces des véhicules et flux de production, et une base de donnée pour les données DWS pour tout ce qui est appel d’alerte au bord du ligne. Il existe deux types de tableaux : les tableaux “raw” et les tableaux “gold”. Les tableaux “raw” contiennent des données brutes remontées tels qu’elles sont, par des serveurs et automates industriels. Au contraire, les tableaux “gold” affichent des nouvelles données résultantes, après avoir effectué des opérations de pré-traitement sur des données brutes.

Concernant les données PSFV, la première capture représente la table de composition en pièces de chaque véhicule, ou ordre de fabrication, dans les différents sites de production Renault. Cette table affiche alors les informations suivantes : le code du site de fabrication, l’année de production, le code d’identification du véhicule, pour chaque véhicule, les codes des pièces qui le composent sur l’intégralité du processus de fabrication, le nombre de pièces de chaque référence ainsi que son code famille PPUM. Une famille PPUM est un ensemble des pièces qui possèdent les même caractéristiques physiques et mécaniques mais qui se distinguent par leurs couleurs ou leurs matières primaires, par exemple un poignée noir et un poignée blanc.

La deuxième capture présente la table des séquences de passage de chaque véhicule dans un tronçon du flux principal de production délimité par un point de départ et un point d’arrivée. Cette table contient des informations sur : le code du site, l’année de production, le code d’identification des véhicules, la date de passage du véhicule dans le tronçon et le codes des zones de départ et arrivé du tronçon.

Par rapport aux données DWS, la première capture représente la table des appels d’alerte remontés par les opérateurs de la ligne de montage. Cette table contient des informations sur : l’identifiant de l’appel, la date de création de l’appel, le type de l’appel, l’identification de la station de travail au niveau de laquelle l’appel a été déclaré. La deuxième capture présente la table de liaison alerte-véhicule. Ce tableau associe à chaque appel d’alerte l’ordre de fabrication concernée. Il contient alors l’identifiant de l’appel et le code d’identification du véhicule.

Dans cette étude nous avons utilisé BigQuery pour accéder aux tables mentionnées ci-dessus, pour importer des fichiers CSV et les stocker dans la plate-forme et pour développer des requêtes SQL afin d’agrèger les données. En complément, nous avons utilisé la librairie Pandas de Python pour effectuer quelques opérations d’agrégation difficiles à faire à travers SQL. Ces outils nous ont permis de manipuler des données sous format CSV ainsi que des données récupérées directement sur le lake de données de GCP.

## 4.4 Stratégie d'agrégation

Pour répondre à l'objectif mentionné en Section 4.2, notre stratégie d'agrégation de données sera divisée en trois parties majeures. D'abord on commence par réaliser **l'agrégation des flux rang 1 et rang 0**, c'est à dire faire la traçabilité des références dès qu'ils arrivent sur la gare routière jusqu'à leurs arrivées aux meubles de la zone de kitting. La deuxième partie consiste à déterminer la **composition et passage des véhicules dans la zone de montage des portes**. Nous déterminons donc pour chaque véhicule la liste de références - des pièces demandées pour le montage des portes. Les résultats obtenus dans ces 2 parties seront combinés pour obtenir la base de données des kits avant départ vers zone du montage. La troisième et dernière étape de cette opération d'agrégation consiste à **extraire les alertes à partir des appels DWS** pour former la base de données des kits après leur livraison à la zone du montage.

### 4.4.1 Agrégation flux rang 0 et rang 1

#### Principe :

En se basant sur les données des mouvements physiques dans le site ainsi que sur la cartographie de répartition des tags des AGVs (Figure 4.3), cette partie consiste à réaliser le suivi des trajets des références sur l'ensemble des flux rang 0 et rang 1. Les informations importantes qu'on souhaite obtenir à partir de cette première approche sont : les codes articles (références des pièces), les numéros des supports physiques, les AGVs tracteurs qui portent chaque support ainsi que les emplacements de chaque référence dans les meubles de la zone de kitting.

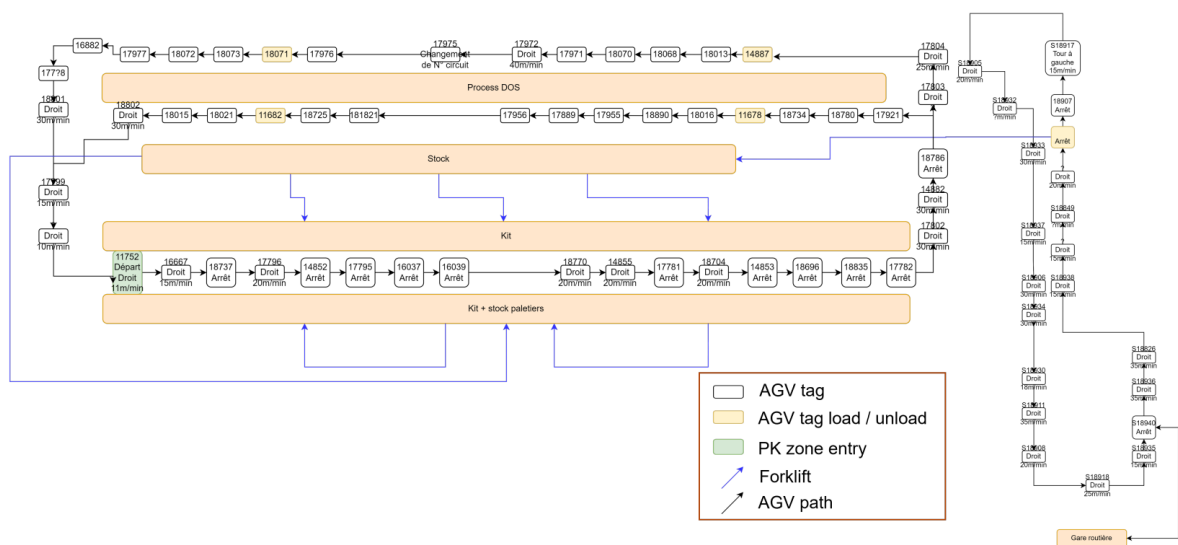


FIGURE 4.3 – Distribution des tags AGV dans la zone de montage des portes



#### 4.4. STRATÉGIE D'AGRÉGATION

##### Architecture :

Dans un premier temps, nous réalisons l'agrégation des supports avec les AGVs tracteurs, en se basant sur l'extrait Spotfire des données logistiques de Reflex et les données AGV remontées par le serveur Synersight. Pour ce faire, nous identifions le tag de chargement et de déchargement des AGVs tracteurs (S18940 et S18936, respectivement) et extrayons les informations de passages des AGVs par ces deux tags. Ensuite, nous ajoutons une nouvelle colonne indiquant le numéro de passage de chaque AGV afin de joindre les 2 livrables. Dans un deuxième temps, à travers requêtes SQL appliquées sur l'ensemble des mouvements physiques, nous générons deux autres listes qui représentent les mouvements de type "Mise en Picking" et "Transfert en Picking". L'ensemble des références de ces 2 mouvements constitue le flux rang 1. Enfin, une simple jointure par code article et support est faite entre les deux résultats pour obtenir l'agrégation désirée. Cette démarche est illustrée par la Figure 4.4.

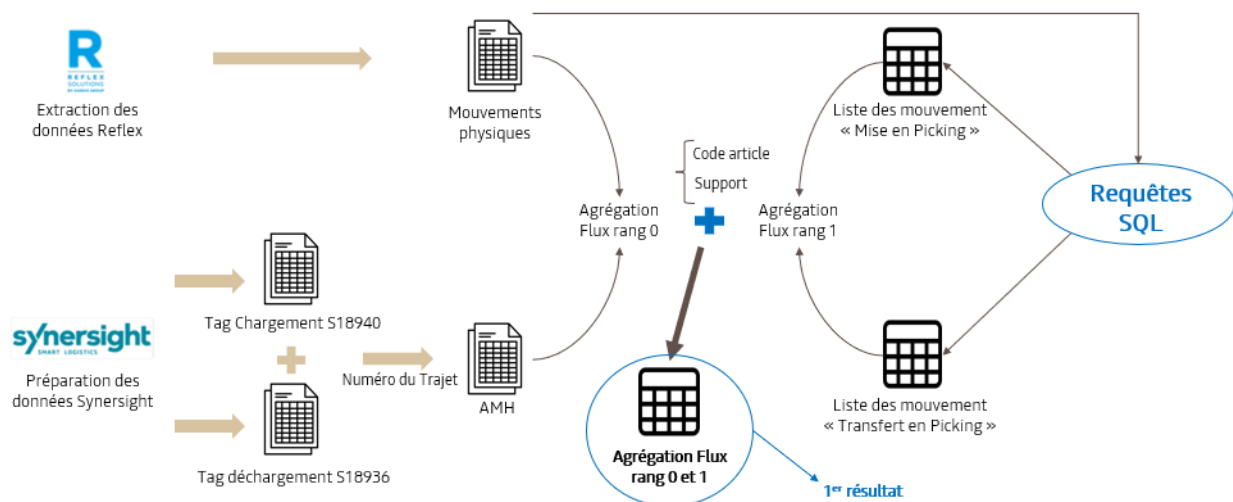


FIGURE 4.4 – Architecture de l'agrégation flux rang 0 et flux rang 1

##### Implémentation et résultats :

Les données brutes des AGVs remontées par le serveur Synersight ne permettent pas dans leur état initial de répondre à notre cas d'étude (1.8). Une étape de pré-traitement est alors indispensable avant de passer à l'agrégation des données. Ce pré-traitement consiste à calculer les dates d'entrée et de sorties de chaque AGV dans les TAGs de chargement et déchargement définis dans la section précédente en utilisant les données des tags suivants. Pour déterminer la date d'entrée dans le tag, nous considérons la différence entre la date de lecture du tag et la durée entre les lectures actuels et précédente. Pour déterminer celle de sortie, nous considérons la différence entre la date de lecture et la durée du trajet entre les deux tags, qui est de l'ordre de 6 secondes. Puis, on ordonne les données des deux tags par ordre croissant des dates d'entrée et on les agrège en utilisant la colonne numéro du

#### 4.4. STRATÉGIE D'AGRÉGATION

---

trajet qui a été générée.

On agrège les mouvements physiques des supports avec l'ensemble des AGVs tracteurs en se basant sur les hypothèses suivantes : il faut que l'emplacement intermédiaire du mouvement soit égale à "Tamp 1 DOS" ou "Tamp 2 DOS" et, pour déterminer l'AGV de chaque mouvement, il faut que la date de validation du mouvement soit entre les dates d'entrée et de sortie de l'AGV dans le tag de chargement. "Tamp 1 DOS" et "Tamp 2 DOS" représentent les adresses des zones tampons entre la gare routière et les stocks internes, relatives à la zone du montage des portes.

A partir des zones tampon, les pièces peuvent être mises à disposition directement à la zone de kitting ou être mises en stock. Pour suivre les pièces que sont mises directement à la zone de kitting, nous considérons les supports physiques ayant comme emplacement de départ soit "Tamp 1 DOS" soit "Tamp 2 DOS" avec le motif du mouvement 180.

D'autre part, pour obtenir la liste des supports mis en stock, nous considérons ceux provenant des zones tampons et ceux provenant de la gare routière. Nous réalisons une jointure par code d'article sur ces deux listes de mouvements. Les supports provenant des zones tampons sont représentés par les références ayant comme emplacement de départ soit "Tamp 1 DOS" soit "Tamp 2 DOS" avec un code motif mouvement égale à 140. La liste des mouvements directs entre la gare routière et la zone de stockage de la zone de kitting des portes, comprend les références ayant comme emplacement final des adresses commençant par soit "F001 1" soit "D005 2" et ayant comme code motif 410. Les dates de validation des mouvements représentent les dates de mise en stock ou mise en zone de kitting, en fonction du motif du mouvement. Par une simple union entre les 2 résultats, on obtient le résultat désiré. Une capture de ce résultat est donnée dans l'annexe (Figure A.3).

Pour suivre clairement le cheminement des supports et des références depuis la gare routière de l'usine jusqu'aux meubles de kitting, les résultats de l'agrégation des flux de rang 0 et rang 1 sont combinés. Cela se fait en utilisant les doublés code article-support physique communs entre les 2 tables. Une capture du résultat de cette agrégation finale est donnée dans les annexes à travers la Figure A.4. À la fin de cette première partie, nous avons réussi à établir une connexion entre les données des AGVs et les données logistiques, telles que les références, les supports physiques, ainsi que les différents endroits de stockage et de kitting.

#### 4.4.2 Composition et passage des véhicules dans la zone de montage des portes

##### Principe :

Cette deuxième étape consiste à déterminer, pour chaque véhicule : la liste de références, et leurs familles PPUM, composant son kit, le nombre des pièces de chaque référence ainsi que la date de passage du véhicule sur la ligne de montage des portes.

##### Architecture :

Afin d'atteindre l'objectif de cette deuxième partie, nous avons suivi la démarche illustrée par la Figure 4.5. L'approche adoptée a débuté par l'extraction de données du MES (PSFV), lesquelles ont été divisées en deux tables distinctes : la table de composition et la table des itinéraires, présentées en Section 4.3. À partir de la première table, une combinaison avec la liste des références utilisées exclusivement dans la zone de montage des portes a permis de générer la composition des véhicules dans cette zone. Ainsi, nous avons défini la structure de chaque kit préparé, comprenant les références, le nombre de pièces pour chaque référence, ainsi que la taille d'emballage associée. En ce qui concerne la deuxième table, elle a servi à déterminer les dates de passage de chaque véhicule dans la zone. Enfin, pour obtenir le résultat désiré, une jointure entre les deux tables a été réalisée en utilisant le code d'identification des véhicules (PJI).

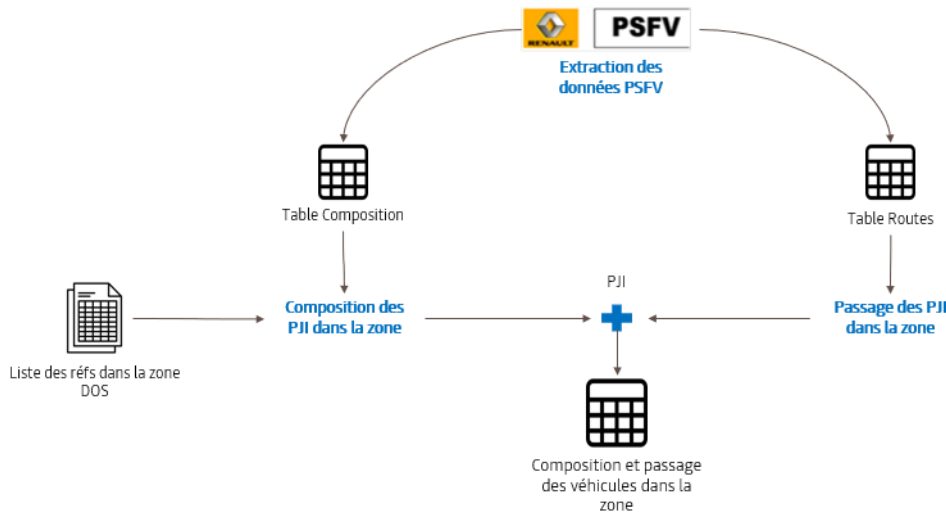


FIGURE 4.5 – Architecture de l'agrégation composition et passage des véhicules dans la zone de montage des portes

##### Implémentation et résultats :

Pour la partie composition des PJI dans la zone DOS, les tables qui listent des références dans les zones du site et composition, sont combinés sous les hypothèses suivantes : afin de considérer uniquement les références qui arrivent dans la zone de montage des portes, il faut que la colonne "secteur\_kit" soit égale à "DOS DA" ou "DOS DB1". Ensuite, pour prendre seulement la composition des véhicules dans le site d'intérêt pour l'année 2023, les variables "Plant\_code" et "year" doivent respectivement être égales à "039" et "23". La table résultante de cette agrégation fournit alors les informations suivantes : code du site, l'année de production, le code d'identification véhicule, le code famille PPUM, le code de la référence et le nombre de pièces de chaque référence présente dans le kit.

#### 4.4. STRATÉGIE D'AGRÉGATION

---

Pour le passage des véhicules dans la zone, et pour avoir uniquement les véhicules traités pendant notre période d'étude, la date de passage doit être comprise entre "2022-04-03" et "2022-06-12". Pour sélectionner uniquement la séquence de passage dans le site d'intérêt, la variable "plant\_code" doit être égale à "039", et pour limiter les ordres de passages à la zone de montage, le code de la zone de départ doit être égal à "MCT1". La table résultante de cette agrégation inclut les informations suivantes : le code du site, l'année de production, le code d'identification véhicule et la date de passage.

Enfin, les deux résultats sont agrégés en utilisant la colonne clé "code PJI" (code d'identification véhicule), pour sélectionner les éléments communs entre eux. Une capture du résultat de cette agrégation est illustrée dans la partie annexe (Figure A.5).

#### 4.4.3 Extraction des alertes à partir des appels DWS

##### Principe :

Le principe de cette dernière partie consiste à identifier tous les véhicules présentant des états d'alerte. Nous nous basons sur les appels DWS effectués par les opérateurs de la ligne de montage. Ces alertes se répartissent en quatre types distincts : logistique, moyens, sécurité et question. Dans notre cas d'étude, nous nous concentrons exclusivement sur les alertes de type logistique et moyens.

##### Architecture :

Pour répondre à cette problématique, nous suivons l'architecture présentée par la Figure 4.6. La démarche consiste à agréger la table des appels DWS avec les véhicules concernés. Pour ce faire, nous utilisons la colonne clé ID de l'appel présente dans les deux tables. Ces deux tables de données manufacturing sont présentées en Section 4.3.

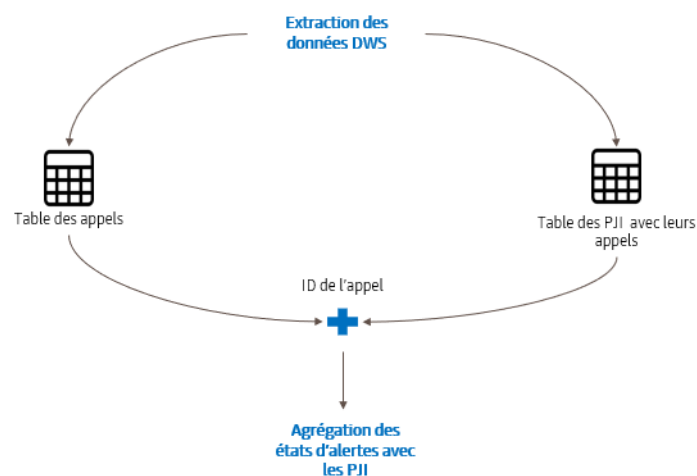


FIGURE 4.6 – Architecture d'extraction des alertes DWS

### **Implémentation et résultats :**

Pour effectuer l'agrégation des états d'alertes avec les PJI, les tables des appels et appels\_pji sont combinées sous les hypothèses suivantes. D'une part, on sélectionne uniquement les alertes de type logistique et moyen de la table des appels. D'autre part, on considère uniquement les appels remontés entre le 3 avril et 12 juin 2023, qui représente notre période d'étude. Finalement, nous effectuons une jointure par l'ID de l'appel pour obtenir la table résultante qui renseigne sur : l'ID de l'appel, le type de l'appel, le code du PJI concerné par l'appel, la date de création de l'appel ainsi que l'ID de la station du travail. Une capture du résultat obtenu est donnée dans la partie annexe (Figure A.6).

## **4.5 Création du *data set***

Jusqu'à présent, nous avons réussi à obtenir des résultats nécessaires pour constituer un *data set*. Cela a été réalisé grâce à l'agrégation des flux de rang 0 et rang 1 avec les données des AGVs, la récupération de la composition des kits des véhicules dans la zone de montage des portes, ainsi que leur séquence de passage, et enfin l'extraction des états d'alerte des véhicules sur la ligne.

Cependant, pour parvenir au jeu de données final qui sera utilisé dans la partie de modélisation, quelques étapes supplémentaires doivent encore être entreprises. Tout d'abord, nous devons effectuer une dernière opération d'agrégation qui relie toutes les tables générées. Cela se fera à travers deux approches : (i) la liaison code d'identification véhicule- support physique pour agréger la liste des compositions des kits avec les flux de rang 0 et rang 1 et (ii) la liaison véhicule-alerte pour agréger la liste des compositions avec les états d'alerte d'autre part. Ensuite, après la combinaison de toutes les données, des opérations de pré-traitement et de transformation doivent être appliquées à l'ensemble des informations afin d'obtenir de nouvelles données, ou caractéristiques, qui seront utiles dans la prise de décision des modèles d'IA.

### **4.5.1 Liaison code identifiant véhicule -support physique**

Cette partie vise à agréger la liste des compositions des kits de chaque véhicule avec les flux de rang 0 et rang 1, dans le but de déterminer l'emplacement de chaque référence présente dans le kit, à la fois dans le meuble et sur le support physique sur lequel elle a été chargée.

La complexité de cette opération d'agrégation réside dans le fait que, pour une même période d'étude, il existe plusieurs mouvements d'approvisionnement de la même référence dans la zone de kitting. Ces mouvements sont demandés par les opérateurs de kitting lorsque le stock est épuisé, mais les pièces peuvent provenir d'un support physique différent. Cet aspect est observable dans l'extrait du rapport des mouvements des GEI, filtré sur un seul article (Figure A.7).

Afin que le résultat d'agrégation soit en adéquation avec la réalité de la zone, une approche doit

## 4.5. CRÉATION DU *DATA SET*

---

être trouvée pour déterminer précisément de quel support physique provient chaque référence présente dans chaque kit d'ordre de fabrication. Cette opération d'agrégation est complexe à réaliser à travers les requêtes SQL, car il n'existe pas de clés directes communes entre les tables. Nous avons donc fait appel à la bibliothèque de traitement des données Pandas pour résoudre cette complexité, ce que nous avons réussi à faire à travers un script Python. Ce script se base sur la quantité de pièces de chaque référence présente dans les kits, fournie dans la table des compositions et passage des véhicules, ainsi que sur la quantité de mouvements de chaque référence présente dans la table des mouvements des GEI. Le code suit les étapes suivantes :

1. Télécharger les tables mouvements des GEI et composition et passage des PJI de BigQuery.
2. Transformer les deux tables en DataFrames.
3. Ordonner les passages des véhicules et les mouvements dans les deux tables, des plus anciens aux plus récents, en utilisant respectivement les colonne date de passage et date de création du mouvement.
4. Parcourir chaque référence dans les deux tables et calculer la différence entre la quantité du mouvement et la quantité des pièces de la référence présente dans le kit. Si le résultat est positif, indiquant que le stock est encore plein, cela entraîne l'affectation du support au code véhicule concerné, la mise à jour de la quantité de stock, et le passage au code véhicule suivant. Si le résultat est négatif, signifiant que le stock à cet instant est épuisé, cela conduit à l'affectation du support au code véhicule concerné, à l'élimination de ce mouvement de la liste, et au passage au code véhicule suivant. Ce processus est répété jusqu'à ce qu'un support soit affecté à tous les codes véhicules présents dans la table de composition.

Maintenant qu'on a obtenu cette liaison code véhicule - support physique, on peut facilement agréger la table de composition et passage des véhicules avec les flux rang 0 et rang 1, par des simples requêtes. Nous utilisons la combinaison code article support physique actuellement présente dans les deux tables. Cela permet d'ajouter les emplacements de kitting dans la table de composition. Une capture du résultat de cette agrégation est donnée dans l'annexe (Figure A.8).

### 4.5.2 Liaison code véhicule - alerte

Avec cette étape on clôture la phase d'agrégation des données. Elle consiste à faire la liaison entre la liste de composition et passage des véhicules, avec les extractions des états d'alertes à partir des appels DWS. Cette liaison est effectuée à travers la création d'une nouvelle colonne qui porte le nom "Alerte" dans la table de composition. Cette colonne prend la valeur "Oui" si le véhicule concerné existe dans la table d'extraction des états d'alertes. Si non, elle prend la valeur "Non". Cette colonne sera prochainement utilisée comme label pour l'entraînement et la validation des modèles d'apprentissage automatique.

### 4.5.3 Pré-traitement et transformation des données

Pour avoir un *data set* final prêt à être déployé dans la partie modélisation, il faut appliquer des opérations de pré-traitement et transformation sur les données agrégées. Ces opérations visent à améliorer la forme général du *data set* et extraire d'autres informations utiles. Ces opérations sont traduites par les trois étapes suivantes :

1. **Création des listes de diversités PPUM et référence** : comme affiché sur la table de composition et passage des véhicules, chaque kit de chaque ordre de fabrication est formé par un ensemble des références caractérisées chacune par son code article et code famille PPUM. Les listes des codes famille PPUM peuvent se répéter d'un kit à un autre, selon le type ou la diversité du véhicule. Dans ce cas, on dit que les kits ont les mêmes diversités PPUM. Dans une même diversité PPUM, les mêmes codes articles peuvent se répéter d'un kit à un autre. On dit alors que les kits ont les mêmes diversités PPUM et références de pièces.

Dans cette première étape de pré-traitement, on a développé un script Python/Pandas qui regroupe les véhicules en fonction de leurs codes de famille PPUM et leurs codes articles. Premièrement, à partir de la table composition, on détermine les listes des diversités PPUM et référence. Puis, on affecte à chaque ordre de fabrication deux diversités dépendant de ces listes. De cette façon, on peut connaître les diversités de chaque kit sans recours aux différents codes.

Après la création des différentes listes, on a découvert que le jeu de données contient en total 29 diversités PPUM et 269 diversités références distribuées d'une manière déséquilibrée sur l'ensemble des diversités PPUM.

2. **Extraction de l'heure de passage** : cette information est récupérée à partir de la date de passage dans la table de composition. Elle peut être utile dans la prise de décision des modèles plus tard, car selon l'heure la probabilité d'apparition des alertes peut augmenter ou diminuer. Par exemple, peut-être que les ordres de fabrication de l'après-midi sont plus susceptibles d'avoir des mauvais kits, vu que le niveau de concentration des opérateurs qui les préparent commence à se dégrader peu à peu.
3. **Groupement et transformation des données** : le groupement des données consiste à rassembler les ensembles de lignes qui portent une même clé pour créer une seule observation résultante. Cela transforme toutes les informations de ces lignes et permet d'extraire d'autres valeurs significatives. Concernant notre cas d'étude, nous avons appliqué le groupement des données sur le *data set* agrégé en utilisant les codes d'identification des véhicules comme clé de groupement. Nous avons pu générer l'ensemble des données suivantes :

— **Pourcentage PE** : cette information renseigne sur le pourcentage des pièces provenant de petits emballages dans chaque kit. Ce pourcentage est calcul à travers la colonne "UC" dans

la table initiale des mouvements de GEI. Si elle porte la valeur “UM” la référence provient d’un grand emballage sinon elle provient d’un petit emballage. L’augmentation du pourcentage PE dans le kit peut augmenter les probabilités d’apparition d’alertes, car il peut être plus difficile pour le préparateur du kit de récupérer plusieurs petites pièces que des grandes pièces.

- **Nombre de références provenant des meubles situés à droite ou à gauche de la zone de kitting :** dans la zone de kitting des portes, les meubles se trouvent à droite et à gauche des opérateurs. Le groupement des données permet alors de diviser le nombre total des références constituant chaque kit, en nombre de références provenant des meubles placés à droite et en nombre de références provenant de meubles placés à gauche. Pour ce faire, on utilise la valeur donnée dans la colonne emplacement de kitting et plus spécifiquement les cinq premiers caractères (“F001 1” pour l’étagère droit et “D005 2” pour l’étagère gauche).
- **Nombre des supports :** cette information indique le nombre total de supports physiques d’où proviennent les références présentes dans chaque kit. Un nombre élevé de supports peut augmenter les probabilités d’alerte, car les opérateurs doivent chercher les pièces dans plusieurs endroits différents, augmentant ainsi le risque d’erreurs.
- **Nombre de références provenant des différents étages des meubles de kitting :** pour extraire cette information nous avons suivi la même logique que pour déterminer le nombre de références provenant des deux cotés de la zone de kitting. Dans ce cas, il s’agit de déterminer pour chaque kit, le nombre de référence provenant des différents étages dans le meuble. Nous utilisons la colonne emplacement de kitting et plus précisément les deux avant-dernier caractères. Il existe quatre étages sur chaque meuble : “00” pour l’étage au sol, “01” pour le premier étage, “02” pour le deuxième étage et “03” pour le troisième étage. L’emplacement des pièces dans les étages peut être une source d’alerte, car il est plus compliqué pour l’opérateur d’accéder à certains étages par rapport à d’autres.

Le *data set* final agrégé et groupé contient alors tous les ordres de fabrication véhicule qui passent sur le flux principal entre les dates de 3 avril et 12 juin 2023. Ce *data set* est composé d’un ensemble de 11 colonnes qui caractérisent les kits et une colonne qui renseigne sur l’état d’alerte du véhicule. En termes de taille, ce *data set* est constitué d’environ 5200 observations, dont 500 portent l’étiquette “Oui”. C’est à dire, 500 véhicules présentent des états d’alerte. Une capture du *data set* final est illustrée par la Figure 4.7.

### 4.5.4 Limitations et difficultés

L’une des principales difficultés rencontrées pendant cette phase d’agrégation des données concerne la collecte des informations des AGVs. Comme expliqué dans la Section 4.3, ces données sont remontées par le serveur Synersight de manière hebdomadaire, il faut donc attendre jusqu’à la fin de chaque



## 4.6. ENVIRONNEMENT ET OUTILS D'ENTRAÎNEMENT

PJI	DIVER_PPUM	DIVER_PART	POURCENT	NUM_REF_ME	NUM_REF	NUM_SUPP	NUM_REF_ET	NUM_REF_ET	NUM_REF_ET	NUM_R	ROUTE_HC	ALERTE
1740027	diversité 13	diversité_13_32	79.1	32	25	59	13	12	19	16	4	non
1410078	diversité 6	diversité_6_0	86.76	33	26	55	4	15	34	7	4	non
1730011	diversité 7	diversité_7_18	75.81	29	24	54	14	9	15	18	5	non
1750011	diversité 7	diversité_7_18	75.81	29	24	54	14	9	15	18	5	non
1520470	diversité 13	diversité_13_48	79.1	32	25	59	13	12	19	16	5	non
0940008	diversité 3	diversité_3_3	80.0	17	6	17	4	5	14	0	5	non
1750007	diversité 7	diversité_7_22	76.92	31	24	55	15	8	15	21	5	non
1750027	diversité 24	diversité_24_5	75.0	31	23	55	14	9	14	20	5	oui
1740034	diversité 13	diversité_13_3	78.79	32	24	58	13	11	19	16	5	non
1530042	diversité 13	diversité_13_44	80.28	34	26	60	14	14	20	16	5	non
1530041	diversité 13	diversité_13_44	80.28	34	26	60	14	14	20	16	5	non
1530040	diversité 13	diversité_13_44	80.28	34	26	60	14	14	20	16	5	non
1530056	diversité 13	diversité_13_41	80.28	34	27	60	15	14	20	16	5	non

FIGURE 4.7 – Jeu de données final

semaine pour obtenir les extraits. Cette périodicité rend difficile l'agrégation en continu avec les autres tables remontées sur le Cloud. Nous disposons de quelques extraits couvrant quelques jours de travail sur une période d'étude de 3 mois, et par conséquent, nous avons décidé de ne pas prendre en compte ces données pour cette première approche d'agrégation.

En ce qui concerne les limitations, la taille du *data set* généré est insuffisante pour entraîner des modèles d'apprentissage profond de grande taille. Le *data set* compte environ 5200 observations, alors que dans le cas idéal, il faudrait au moins 10000 observations pour des modèles robustes. De plus, il y a un déséquilibre significatif entre les deux classes présentes dans le jeu, avec seulement 500 observations d'alerte sur un total de 5200. Ce déséquilibre incite les modèles à privilégier la classe majoritaire dans leurs prises de décision, entraînant une mauvaise performance. Des solutions à ces problèmes seront envisagées dans les prochaines parties.

## 4.6 Environnement et outils d'entraînement

Dans cette phase de modélisation, nous avons travaillé sur l'environnement Google Cloud Platform (GCP), plus spécifiquement sur les services BigQuery. Nous avons utilisé l'outil BigQuery ML ainsi que Vertex AI. Cette dernière est une plate-forme de machine learning visant à simplifier et accélérer le développement, le déploiement et la gestion de modèles d'apprentissage automatique.

En plus de sa fonctionnalité principale, Vertex AI offre aux utilisateurs la possibilité de créer des Jupyter Notebooks sur des machines virtuelles personnalisées. Pour mettre en place notre environnement de développement et d'entraînement, nous avons commencé par créer une instance de Jupyter Notebook sur une machine virtuelle en spécifiant différentes caractéristiques telles que le système d'exploitation, la capacité RAM, la puissance du processeur, et la présence d'une unité de traitement graphique (GPU), le cas échéant.

BigQuery ML, une extension de Google BigQuery, permet aux utilisateurs d'exécuter des tâches d'apprentissage automatique (machine learning ou deep learning) directement dans le cadre de requêtes SQL, éliminant ainsi la nécessité de transférer des données vers une autre plate-forme. Cela nous a permis d'effectuer des analyses prédictives et de construire des modèles de machine learning en utilisant les données stockées dans BigQuery, évitant les complexités liées aux transferts de données et aux intégrations avec des outils externes.

### 4.7 Analyse exploratoire des données

Avant de commencer un projet d'apprentissage automatique, il est primordial de s'assurer que les données sont prêtes pour le travail de modélisation. L'analyse exploratoire des données permet de s'assurer que les données sont prêtes et aptes à être utilisées dans un système d'apprentissage automatique et permet de mieux comprendre le domaine du problème. Ainsi, l'on a effectué deux analyses : une analyse du comportement des variables et une analyse de la distribution des échantillons sur les différentes classes présentes.

#### 4.7.1 Analyse du comportement des variables

L'analyse visuelle des données constitue l'une des techniques d'analyse initiale des données les plus fréquemment utilisées dans le développement de systèmes intelligents. En analysant visuellement les données, et/ou en examinant le comportement des variables qui serviront de base à l'élaboration du modèle, le développeur peut mieux comprendre le problème et prendre une décision éclairée sur les algorithmes nécessaires à déployer. Ce type d'analyse est généralement effectué à l'aide de graphiques dont leurs types dépend du type de la variable à analyser.

Concernant le jeu de données de ce cas d'étude, nous analysons le comportement de trois variables : les diversités des kits (Diver\_PPUM), l'heure de passage des véhicules sur le flux principal de montage (Route\_Hour) et le pourcentage de pièces provenant de petits emballages (PE) dans chaque kit.

La Figure 4.8 présente un histogramme pour les diversités des kits (à gauche) et les heures de passage (à droite). Ces graphiques montrent le nombre d'occurrences appartenant à la classes des alertes en fonction de la diversité des kits et de l'heure de passage. On constate que seulement 17 des 29 diversités qui constituent le *data set* présentent des états d'alerte. Ces diversités sont données sur l'axe des abscisses et le nombre d'occurrences sont distribuées d'une manière déséquilibrée sur les différentes diversités des kits. Les kits de diversité 13 et de diversité 5 sont ceux qui présentent les plus souvent d'anomalies. Par rapport aux heures de passage, comportement similaire est observé. On constate que les alertes sont distribuées de façon déséquilibrée et qu'elles sont plus fréquentes à 12h, 14h et 17h.

#### 4.7. ANALYSE EXPLORATOIRE DES DONNÉES

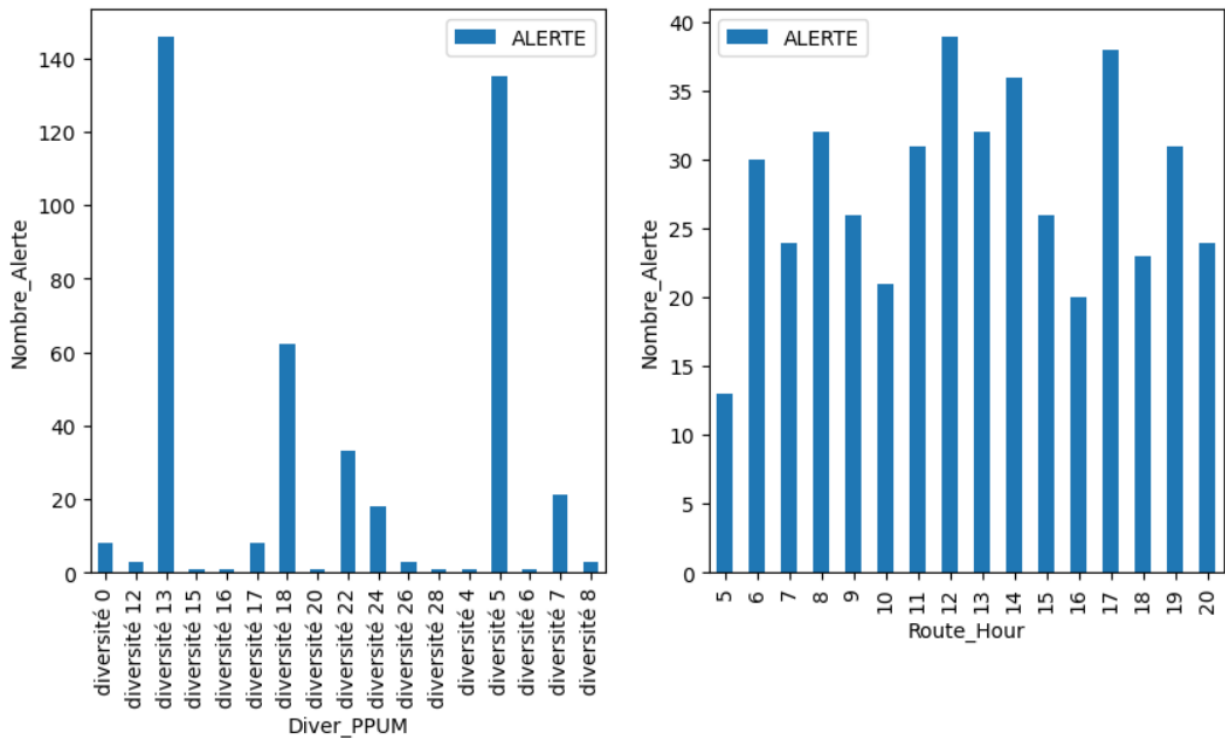


FIGURE 4.8 – Nombre d’occurrences d’alertes en fonction des diversités des kits et des heures de passage

Pour analyser le comportement de la variable pourcentage PE vis-à-vis des états d’anomalie, nous utilisons une courbe de densité. Elle permet d’obtenir une compréhension visuelle rapide de la distribution des valeurs de cette caractéristique dans la classe concernée. Dans ce type de graphique, l’axe des x représente la distribution des valeurs de la variable dans le jeu de données entier, comme dans un histogramme, et l’axe des y est la fonction de densité de probabilité pour une classe donnée. La Figure 4.9 illustre la courbe résultante. Cette figure met en évidence que les kits qui présentent des anomalies possèdent des pourcentage des pièces provenant des PE entre 75% et 85%, environ. Ce fait peut laisser penser que la probabilité d’alerte augmente avec l’augmentation du pourcentage de pièces provenant de PEs dans le kit.

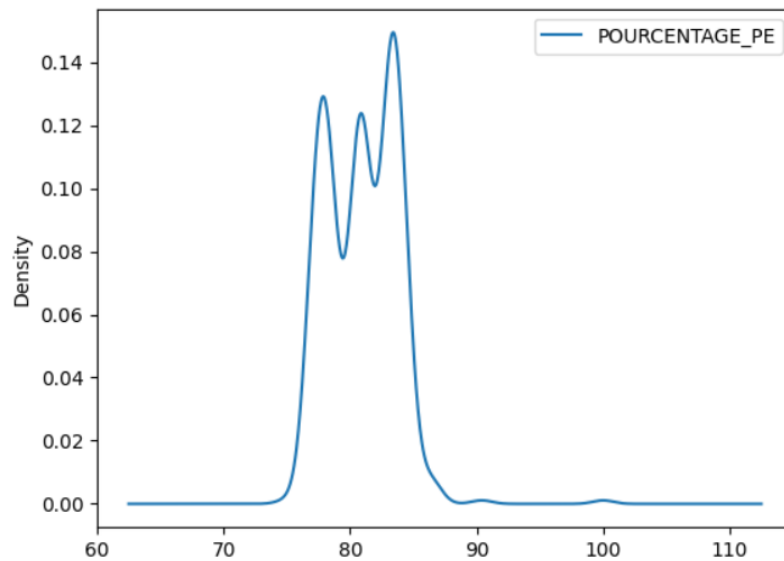
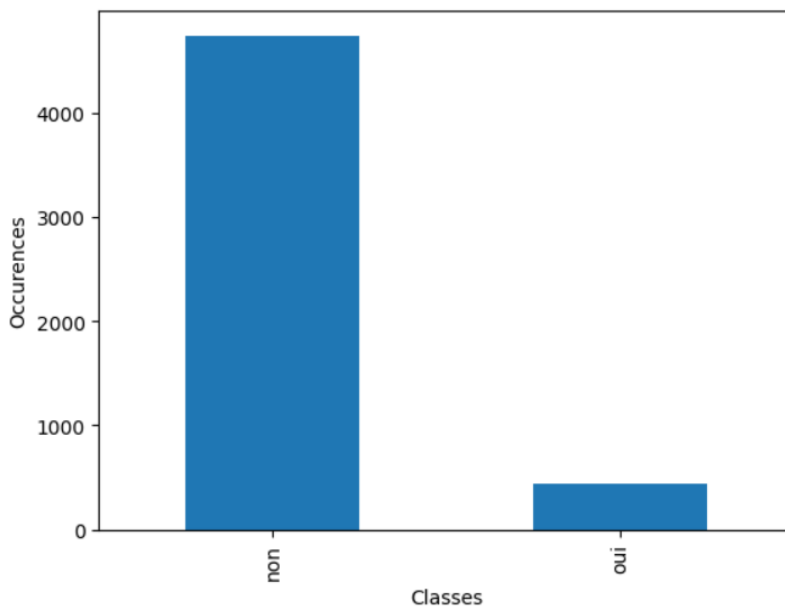


FIGURE 4.9 – Courbe de densité de distribution des pourcentage PE dans les états d’alerte

#### 4.7.2 Distribution des classes

Lors de l’analyse des données recueillies à partir des informations manufacturing, il faut tenir compte du fait que la plupart des kits qui arrivent sur la zone de montage sont en bonne condition. Par conséquent, l’apparition d’alertes sur la ligne est très peu fréquente. Cette situation se traduit par un ensemble de données très déséquilibré. Le *data set* agrégé comporte uniquement 500 observations qui portent l’étiquette ”Oui” de l’alerte sur un total de 5200 observations (Figure 4.10). L’importance de l’équilibrage des données réside dans le fait que de la plupart des modèles de classification sont conçus pour fonctionner avec à peu près la même quantité d’échantillons pour chaque [Prytz, 2014]. En outre, il existe plusieurs autres facteurs de dégradation des performances de la classification qui sont associés à des données non équilibrées. S’il y a peu de données à partir desquelles le modèle peut apprendre, un algorithme prédictif peut avoir des difficultés à extraire des caractéristiques importantes qui permettent de distinguer les classes de problèmes et d’établir une limite de décision fiable. Cela interfère directement avec les performances du modèle en matière de généralisations futures sur des observations de données inconnues.

FIGURE 4.10 – Distribution des classes dans le *data set* initial

Les méthodes basées sur l'échantillonnage sont les plus simples utilisées pour équilibrer les ensembles de données [Dogan et Birant, 2021]. Ces catégories de méthodes sont également connues sous le nom d'approches au niveau des données, étant donné que la solution complète pour équilibrer les données est appliquée directement sur l'ensemble de données, en isolant les modèles de classification. Vu que ce processus intervient généralement avant que les modèles de classification ne soient formés, les méthodes d'échantillonnage sont liées à l'étape de pré-traitement des données.

Pour répondre au cas d'étude actuel, nous avons utilisé des méthodes de sous-échantillonnage, de sur-échantillonnage et une combinaison des deux sur le *data set* d'entraînement des modèles.

#### 4.7.2.1 Sous-échantillonnage (Undersampling) :

Le sous-échantillonnage consiste à réduire le nombre d'exemples de la classe majoritaire pour équilibrer le ratio entre les classes. Cela peut être réalisé en sélectionnant aléatoirement un sous-ensemble d'exemples de la classe majoritaire ou en supprimant certaines lignes des données.

Nous avons donc sélectionné et supprimé de manière aléatoire un ensemble de données de la catégorie ne comportant pas d'alerte afin de réduire l'écart entre le nombre de données des deux catégories. En suivant cette démarche, nous avons créé plusieurs *data sets* d'entraînement comportant différentes tailles. Pour ce faire, le *data set* de la classe non-alerte a été réduit à 351, 702, 877 et 1053 lignes. Des tests ont été effectués sous ces différentes configurations du *data set*.

### 4.7.2.2 Sur-échantillonnage (Oversampling) :

Le sur-échantillonnage consiste à augmenter le nombre d'exemples de la classe minoritaire en dupliquant des exemples existants ou en générant de nouveaux exemples similaires dits synthétiques.

Pour le cas d'étude nous avons utilisé la méthode SMOTE (Synthetic Minority Over-sampling Technique) sur la catégorie de données comportant des alertes. Cette méthode consiste à entraîner un modèle qui génèrent synthétiquement de nouvelles données basées sur les répercussions des données originales [Chawla et al., 2002]. En suivant cette méthode, nous avons pu créer un *data set* d'entraînement composé de 3417 données pour chaque classe.

Pour implémenter cette approche nous avons suivi la démarche illustrée par la Figure 4.11. Cette démarche est constituée de trois étapes principales qui prennent en entrée un *data set* initial et génère un *data set* synthétique prêt pour la phase de modélisation :

- **Pré-traitement** : cette étape consiste à préparer les données avant de les alimenter au modèle SMOTE. Vu que cet algorithme réalise des calculs de distances et manipule des données numériques, il faut dans un premier temps transformer les colonnes catégorielles (Diversités PPUM, Diversités références et la colonne cible alerte) par un encodage des labels puis changer les types des données en réel.
- **Entraînement et équilibrage** : cette étape consiste à faire appel au modèle SMOTE. Il faut d'abord séparer les colonnes caractéristiques de la colonne cible puis lancer l'entraînement du modèle sur l'intégralité des données. Les configurations par défaut des paramètres sont conservées. Le nombre des voisins les plus proches utilisés pour définir le voisinage des échantillons à utiliser pour générer les observations synthétiques est égal à 5. Enfin, le modèle entraîné transforme le *data set* original.
- **Transformation inverse** : Cette partie consiste à inverser les transformations effectuées pendant l'étape de pré-traitement pour revenir à la configuration initiale du jeu de données avec toutes les valeurs originales. Après cette étape, on dispose alors d'un ensemble de données équilibré avec deux situations d'exploitation nécessaires pour créer les modèles.

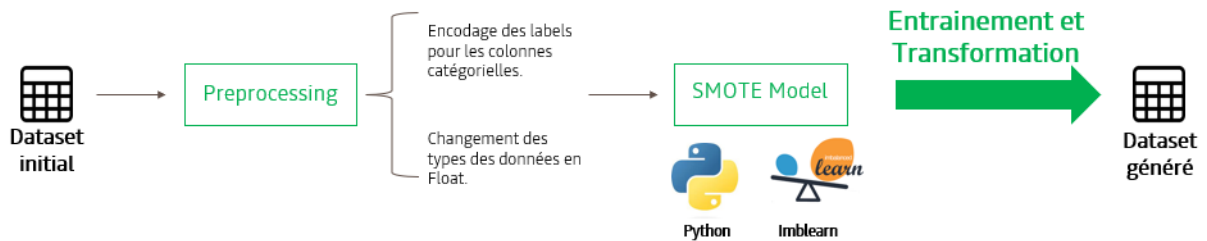


FIGURE 4.11 – Démarche suivie pour équilibrer les proportions des classes.

La Figure 4.12 présente le résultat de l’application de cette démarche sur l’ensemble de données du cas d’étude actuel. On peut constater que la distribution finale des échantillons apporte une structure de données équilibrée.

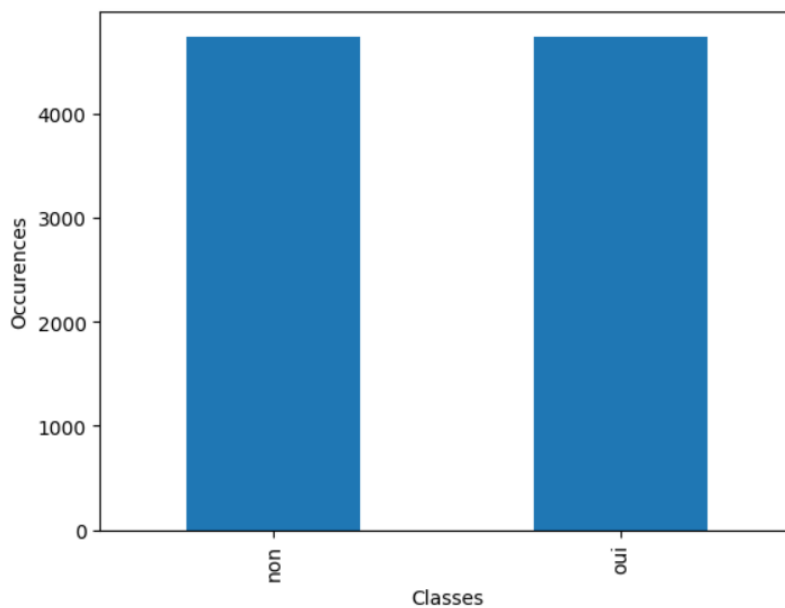


FIGURE 4.12 – Distribution des classes dans le *data set* généré

#### 4.7.2.3 Combinaison sous-échantillonnage et sur-échantillonnage :

Nous avons utilisé les méthodes de sous-échantillonnage et sur-échantillonnage décrites précédemment pour créer de *data sets* équilibrés à des valeurs intermédiaires entre 351 et 3417 données sur chaque

classe. Le sous-échantillonnage a été appliqué sur la catégorie comportant les données de non-alertes et le sur-échantillonnage sur les données comportant des alertes. Nous avons ainsi créé des *data sets* équilibrés, composés de 702, 1053, 1404 et 1755 données sur chaque catégorie.

### 4.8 Métriques d'évaluation des modèles

Les métriques d'évaluation sont déployées pour mesurer les performances d'un modèle de machine learning. Elles permettent de quantifier à quel niveau le modèle est efficace dans la résolution d'une tâche déterminée. Pour évaluer les modèles de classification développés dans cette étude, nous utilisons les métriques *accuracy*, précision, rappel et score  $F_\beta$ , qui sont fréquemment employées dans des applications d'apprentissage automatique de façon générale. Cependant, nous utilisons aussi les métriques taux de fausses alertes et taux de non-détections, car elles sont plus parlantes pour nos interlocuteurs du terrain industriel.

Étant donné qu'une problématique de pièce manquante dans un kit sera détectée au plus tard sur la ligne de montage et qu'il existe déjà une méthodologie pour y réagir, pour ce cas d'étude nous considérons que le taux de fausses alertes est prioritaire face au taux de non-détections. Ainsi, sur la dernière partie de ce chapitre, nous calculerons non seulement le score  $F_1$  ( $\beta = 1$ ), mais aussi le score  $F_{0,4}$  ( $\beta = 0,4$ ) pour tenir compte de cette priorisation.

La Figure 4.13 présente la matrice de confusion type qui sert de base de calcul pour les métriques d'évaluation. Cette matrice regroupe le nombre de vrais positifs (TP), de faux positifs (FP), de vrais négatifs (TN) et de faux négatifs (FN) résultants de la classification effectuée par le modèle. Elle permet également de donner un aperçu des performances d'un modèle de classification.

Confusion Matrix		Predicted Class	
		Negative	Positive
Actual Class	Negative	All clear TN	False Alert FP
	Positive	Non-detection FN	Right detection TP

FIGURE 4.13 – Matrice de confusion pour un modèle de classification binaire

Le calcul de ces métriques est effectué avec les équations suivantes :

$$\text{Accuracy} = \frac{\text{Nb Vrais Positifs} + \text{Nb Vrais Négatifs}}{\text{Nb Vrais Positifs} + \text{Nb Vrais Négatifs} + \text{Nb Faux Positifs} + \text{Nb Faux Négatifs}} \quad (4.1)$$



$$\text{Précision} = \frac{\text{Nb Vrais Positifs}}{\text{Nb Vrais Positifs} + \text{Nb Faux Positifs}} \quad (4.2)$$

$$\text{Rappel} = \frac{\text{Nb Vrais Positifs}}{\text{Nb Vrais Positifs} + \text{Nb Faux Négatifs}} \quad (4.3)$$

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Précision} \cdot \text{Rappel}}{(\beta^2 \cdot \text{Précision}) + \text{Rappel}} \quad (4.4)$$

$$\text{Taux fausses alertes} = \frac{\text{Nb Faux Positifs}}{\text{Nb Faux Positifs} + \text{Nb Vrais Négatifs}} \quad (4.5)$$

$$\text{Taux non-détections} = \frac{\text{Nb Faux Négatifs}}{\text{Nb Faux Négatifs} + \text{Nb Vrais Positifs}} \quad (4.6)$$

## 4.9 Stratégie de modélisation

Avant d'entrer dans la conception des modèles choisis, il est important de rappeler que l'objectif principal de ce chapitre est de développer un système capable de trier efficacement les kits des voitures préparés dans la zone de kitting en deux catégories : défectueux et non-défectueux. Cette section détaille les différentes démarches suivies pour implémenter des algorithmes d'IA de classification sur le jeu de données agrégé, obtenu à la fin de la Section 4.5. Elle présente les modélisations effectuées avant et après l'équilibrage du *data set*.

### 4.9.1 Modélisation avant équilibrage des données

Dans cette phase, nous avons utilisé l'outil BigQuery ML pour créer, entraîner et évaluer des modèles machines learning d'une manière rapide. L'objectif est de vérifier si le déséquilibre des classes pose un problème dans les prédictions des modèles.

#### 4.9.1.1 Choix et conception des modèles

Nous avons implémenté trois algorithmes de classification : un arbre de décision boosté, une régression logistique et un réseau de neurones. Ces modèles vérifient trois critères importants : ils représentent des approches de classification binaire supervisé capables de traiter deux classes uniquement dans le système, ils sont censés gérer efficacement des ensembles de données déséquilibrés avec une grande dimension de caractéristiques et ils sont dans le catalogue des modèles prédéfinis fournis par BigQuery ML.

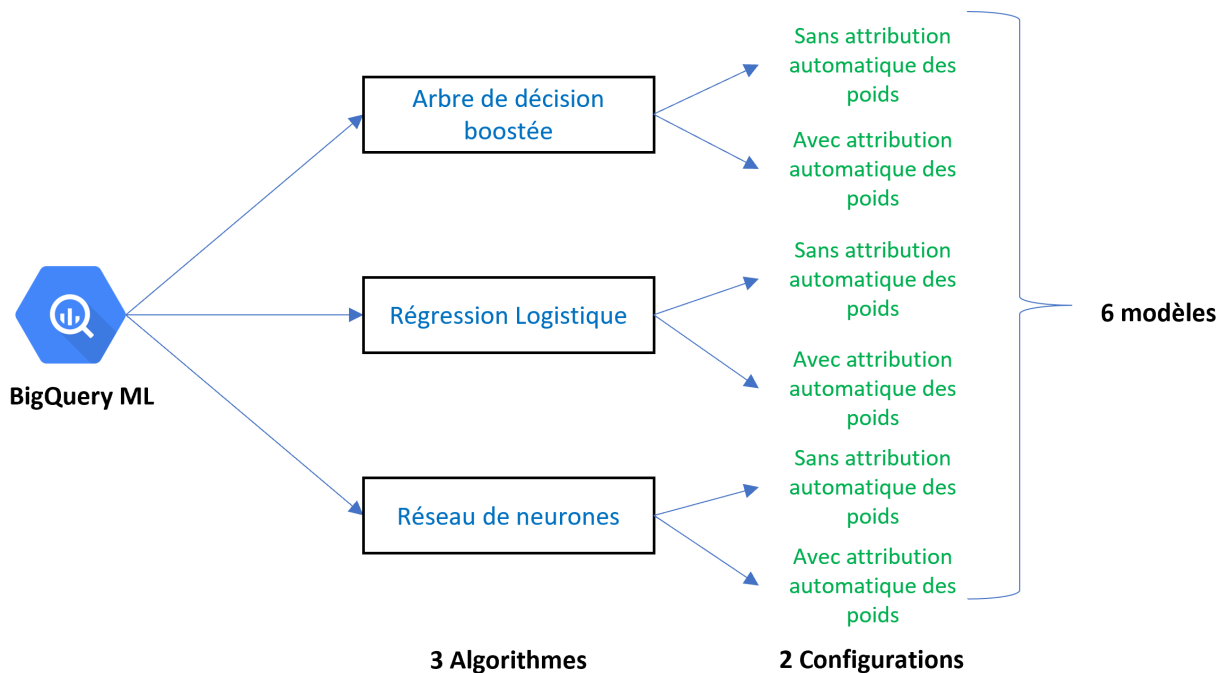


FIGURE 4.14 – Conception des modèles avec BigQuery

Concernant la partie conception et comme il est montré par la Figure 4.14, 2 modèles sont développés à partir de chaque algorithme. Nous utilisons 2 configurations différentes pour chaque algorithme : soit sans attribution automatique des poids des classes, soit avec attribution automatique des poids des classes. La pondération automatique des classes signifie que lors de son entraînement, le modèle calcule et affecte des poids différents pour chaque classe en fonction de sa fréquence dans le jeu de données. Généralement, les classes minoritaires peuvent se voir attribuer des poids plus élevés, tandis que les classes majoritaires peuvent se voir attribuer des poids plus bas. Cette approche vise à atténuer les effets du déséquilibre et à améliorer les performances du modèle, en particulier en ce qui concerne la prédiction des classes minoritaires.

#### 4.9.1.2 Détail d'entraînement

Avant d'entamer la partie entraînement des modèles, il faut séparer les données dans le *data set* en lots d'entraînement et lots de test. Nous utilisons 90% des observations pour l'entraînement et la validation de l'entraînement et 10% pour les tests.

Nous utilisons la fonction `ML.CREATE MODEL` de BigQuery ML pour entraîner chaque modèle en précisant le type d'algorithme à implémenter, la colonne cible et les colonnes caractéristiques. Nous précisons également, si la pondération des classes doit être activée ou pas.

## 4.9. STRATÉGIE DE MODÉLISATION

---

Concernant le choix des hyperparamètres, étant donné que le jeu de données traité est peu volumineux, le réseau de neurones développé contient uniquement une seule couche cachée avec 20 neurones avec des fonctions d'activations de types "relu" et une couche de sortie contenant une seule neurone avec une fonction d'activation de type "sigmoid" pour calculer la probabilité d'appartenance. Les hyperparamètres des autres algorithmes gardent leurs valeurs par défaut.

Tout pré-traitement nécessaire à appliquer sur le jeu de données avant l'entraînement et l'évaluation des modèles, notamment l'encodage des colonnes catégorielles et la mise en échelle des valeurs, se fait d'une manière automatique par BigQuery ML, et c'est l'un des avantages les plus importants de l'outil. Par défaut, la fonction `ML.CREATE MODEL` divise les données d'entrée en 80% des observations pour l'entraînement et 20% pour l'évaluation. Les 6 modèles créés et entraînés avec leurs résultats d'évaluation obtenus sont ensuite enregistrés dans la base de données créée pour sauvegarder les tables d'agrégation.

### 4.9.1.3 Évaluation et interprétation des résultats

Pendant la partie de pré-traitement, la clause `ML.CREATE MODEL` transforme la colonne label qui contient initialement les étiquettes "oui" (alerte) et "non" (non-alerte) pour les états d'alerte en 0, qui représente la classe contenant des alertes, et 1, qui représente la classe de non-alertes.

Le Tableau 4.1 présente les résultats obtenus sur les données de validation, pour les modèles entraînés sans attribution automatique des poids pour les classes.

TABLEAU 4.1 – Métriques d'évaluation pour les modèles de la première configuration

Modèle	<i>Accuracy</i>	Précision	Rappel	Score F1	Taux fausses alertes	Taux non détections
Arbre de décision	0,92	0,43	0,04	0,07	0,00	0,96
Régression logistique	0,92	0,44	0,05	0,09	0,01	0,95
Réseau de neurones	0,92	0,00	0,00	0,00	0,00	1,00

En examinant ces résultats, on peut constater que les valeurs des métriques sont très proches pour tous les 3 modèles. Cela montre que malgré les grandes différences dans les mécanismes de fonctionnement de ces modèles, ils se comportent de la même façon sur le jeux de données agrégé.

Dans ce cas, malgré que les modèles réalisent de très bons résultats en termes d'*accuracy*, cela ne signifie pas qu'ils sont performants de façon global. C'est l'un des problèmes majeurs causés par les jeux de données avec un déséquilibre entre les classes. Les modèles ont été victimes de ce qu'on appelle le paradoxe de l'"*accuracy*" [Mitchell, 1997]. Malgré des métriques d'évaluation élevées, les algorithmes ne sont pas en mesure de différencier la classe minoritaire de l'autre catégorie. Ce manque de distinction peut poser de sérieux problèmes car l'identification de ces classes minoritaires peut être

## 4.9. STRATÉGIE DE MODÉLISATION

---

la cible du challenge à résoudre. C'est le cas dans la détection de kits défectueux. Ce comportement est visible surtout à partir des matrices de confusion sur les données de validation illustré par la Figure 4.15.

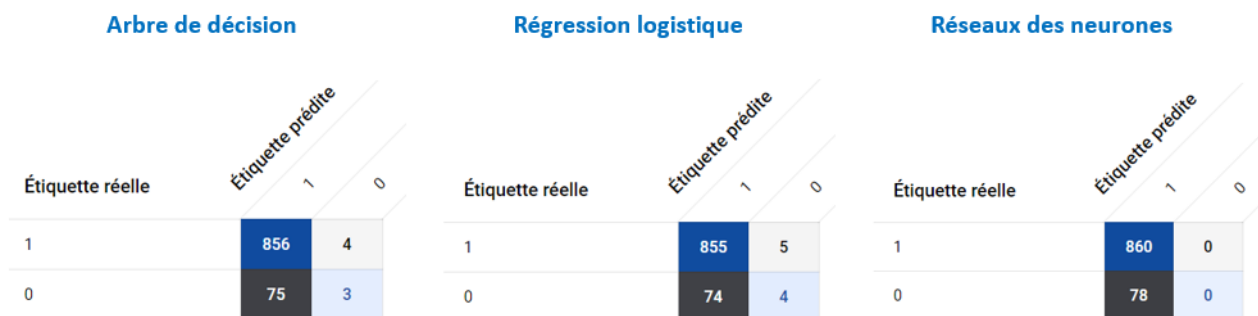


FIGURE 4.15 – Matrices de confusion sur les données de validation pour les modèles de la première configuration

La Figure 4.15 met en évidence que les modèles ratent complètement la tâche d'identification de la classe des alertes et favorisent la classe de non-alertes, plus fréquente dans le *data set* d'entraînement et dans leur prédiction. Le taux des vrais négatifs (les bons kits prédits correctement par les modèles) est très élevé par rapport aux autres valeurs de la matrice. Cela explique la valeur élevée au niveau de l'“*accuracy*”.

Ces modèles ne répondent pas au cas d'étude réel que nous traitons. On doit apporter des modifications aux modèles pour augmenter le taux des vrais positifs par rapport aux faux négatifs et faire apparaître la classe minoritaire dans leurs prises de décision.

La Tableau 4.2 présente les métriques d'évaluation résultantes pour les modèles avec attribution automatique des poids. A partir de ces métriques, on constate que le rappel des modèles augmente par rapport aux valeurs de la première configuration. Cela signifie que les modèles deviennent plus performants dans la classification de la classe des alertes. Ce fait conduit à la baisse du taux des non-détections et par conséquent à la hausse du taux des fausses alertes. Cependant, les modèles montrent une amélioration significative sur la distinction des deux classes. On peut vérifier ce comportement sur les matrices de confusion illustrées par la Figure 4.16.

## 4.9. STRATÉGIE DE MODÉLISATION

TABLEAU 4.2 – Métriques d'évaluation pour les modèles de la deuxième configuration

Modèle	<i>Accuracy</i>	Précision	Rappel	Score F1	Taux fausses alertes	Taux non détections
Arbre de décision	0,71	0,18	0,69	0,29	0,28	0,31
Régression logistique	0,58	0,14	0,83	0,25	0,45	0,17
Réseau de neurones	0,69	0,18	0,76	0,29	0,31	0,24

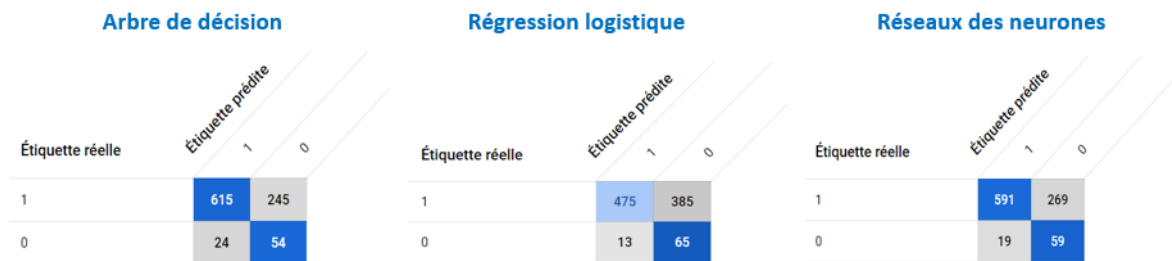


FIGURE 4.16 – Matrices de confusion sur les données de validation pour les modèles de la deuxième configuration

Selon ces résultats, la pondération des classes permet de favoriser la classe minoritaire par rapport aux autres classes et donc de réaliser une meilleure classification de façon globale. Les résultats obtenus ne permettent pas encore d'envisager un déploiement du modèle sur le terrain industriel. Cependant, ils renforcent l'hypothèse de l'obtention de meilleurs résultats en équilibrant la distribution des classes avant d'entamer l'étape de modélisation.

### 4.9.2 Modélisation après équilibrage des données

Les résultats des performances des modèles entraînés sur le *data set* original obtenus à la fin de la partie précédente mettent en évidence l'importance de l'étape d'équilibrage des données comme étape préliminaire avant de commencer l'entraînement des modèles. Cette section présente alors la phase de modélisation après l'étape de manipulation des données sur le *data set* d'entraînement. Le comportement de 6 modèles de classification (Naive Bayes, régression logistique, support vector machine, arbre de décision, forêt aléatoire et K plus proches voisins) est présenté.

#### 4.9.2.1 Détail d'entraînement

Afin de créer et entraîner les différents modèles, nous avons suivi la démarche illustrée dans la Figure 4.17. Tout d'abord, des opérations de pré-traitement nécessaires sur le jeu de données ont été appliquées. Notamment, un encodage de labels pour les colonnes catégorielles afin de les transformer en valeurs numériques et une mise en échelle des valeurs des caractéristiques ("scaling" en Anglais)

## 4.9. STRATÉGIE DE MODÉLISATION

pour transformer les variables de manière à ce qu'elles se situent dans une plage de valeurs entre -1 et 1. Cette deuxième étape est importante pour garantir que les caractéristiques ont des ordres de grandeur comparables, ce qui peut améliorer la performance des modèles d'apprentissage automatique. Il faut rappeler que cette phase de pré-traitement se fait d'une manière automatique dans BigQuery ML.

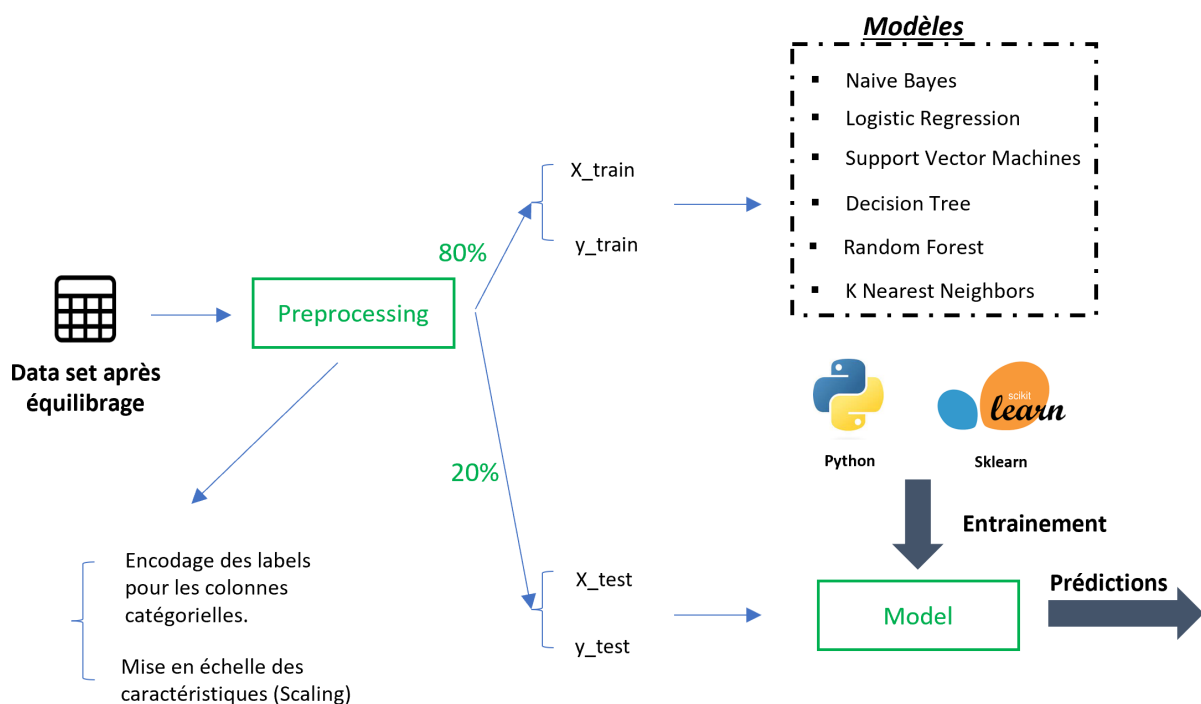


FIGURE 4.17 – Architecture d'entraînement

Ensuite la colonne des alertes est séparée des autres colonnes pour obtenir les caractéristiques et la cible. Puis, l'ensemble total est subdivisé en 80% des observations dédiées pour l'entraînement des modèles et 20% réservées pour le test et l'évaluation. C'est donc sur 80% des données que l'on a mené des actions d'équilibrage sur le *data set*.

Enfin, l'ensemble d'entraînement est alimenté à un *pipeline* qui contient un ensemble de modèles possédant chacun leur propre instance implémentée. Ultérieurement, les modèles sont testés avec les données de test.

### 4.9.2.2 Évaluation et interprétation des résultats

#### 4.9.2.2.1 Résultats de classification

Nous avons appliqué des techniques de sur-échantillonnage, sous-échantillonnage et une combinaison des deux pour équilibrer les données du *data set* utilisé pour l'entraînement des modèles. Les modèles ont été entraînés sous différentes configurations de *data set* mais ils ont tous été testés avec le même *data set* d'évaluation. Les Tableaux 4.3 à 4.5 présentent les métriques d'évaluation résultantes pour les modèles testés suite à l'équilibrage des *data sets*. Plus précisément, le Tableau 4.3 présente les résultats suite à l'équilibrage des données en utilisant SMOTE pour générer des données présentant des alertes afin d'égaliser le nombre de données ne présentant pas d'alerte. Le *data set* d'entraînement final est donc composé de 3417 données conformes et 3417 données non conformes, dont 351 sont des données réellement non conformes. Le Tableau 4.4 présente les métriques d'évaluation résultantes pour des modèles entraînés avec un *data set* composé de 702 données ne présentant pas d'alerte et 351 données présentant des alertes. Nous avons dû sélectionner aléatoirement 702 données conformes parmi les 3417 données conformes disponibles. Le Tableau 4.5 montre les métriques d'évaluation pour des modèles entraînés avec un *data set* équilibré à 1404 données sur chaque catégorie. D'abord, un sous-échantillonnage aléatoire de 1404 données conformes a été effectué et puis un sur-échantillonnage avec SMOTE a été mis en place sur les données présentant des alertes afin d'obtenir 1404 données.

A partir de ces résultats, nous pouvons constater que les modèles K plus proches voisins et Forêt aléatoire présentent les meilleures performances en termes d'*accuracy*, score F1 et score  $F_\beta$ . K plus proches voisins pour le cas de sur-échantillonnage et forêt aléatoire pour les cas des sous-échantillonnage et de la combinaison de sur et sous-échantillonnage. Parmi ces trois configurations, celle qui fournit les meilleurs résultats est la configuration de sous-échantillonnage. Sous cette configuration, le modèle forêt aléatoire atteint une *accuracy*=0,78. Ainsi, 78% des décisions prises par le modèle sont correctes. En comparant ces résultats avec les résultats présentés par le Tableau 4.2, nous pouvons remarquer que les résultats obtenus par le modèle forêt aléatoire sont meilleurs que les résultats présentés par les trois modèles du Tableau 4.2, en termes d'*accuracy* et de score F1. Nous pouvons remarquer une baisse au niveau du rappel et donc une montée du taux de non-détections. En revanche, la hausse dans les valeurs de précision montre que la performance sur l'identification de la classe des non-alerte est meilleure que celle vue dans la Section 4.9.1. Cela se traduit par une baisse du taux de fausses alertes. Pour le cas d'étude actuel, le taux de fausses alertes est prioritaire face au taux de non-détections. Ainsi, les résultats obtenus par le modèle forêt aléatoire se révèlent convenables pour le présent cas d'étude.

#### 4.9. STRATÉGIE DE MODÉLISATION

TABLEAU 4.3 – Métriques d'évaluation pour les modèles entraînés avec un *data set* équilibré en utilisant SMOTE

Modèle	<i>Accuracy</i>	Précision	Rappel	Score F1	Score $F_\beta$ $\beta = 0,4$	Taux fausses alertes	Taux non détections
Régression logistique	0,63	0,18	0,86	0,30	0,70	0,40	0,14
Support Vector Machines	0,64	0,18	0,86	0,30	0,71	0,38	0,14
Arbre de décision	0,60	0,14	0,66	0,23	0,68	0,41	0,34
Forêt aléatoire	0,61	0,14	0,64	0,23	0,70	0,39	0,36
Naive Bayes	0,35	0,12	0,94	0,21	0,42	0,71	0,06
K plus proches voisins	0,75	0,21	0,64	0,32	0,80	0,24	0,36

TABLEAU 4.4 – Métriques d'évaluation pour les modèles entraînés suite à un sous-échantillonnage du *data set* (702 non alertes et 351 alertes)

Modèle	<i>Accuracy</i>	Précision	Rappel	Score F1	Score $F_\beta$ $\beta = 0,4$	Taux fausses alertes	Taux non détections
Régression logistique	0,73	0,21	0,71	0,32	0,78	0,27	0,29
Support Vector Machines	0,73	0,21	0,72	0,33	0,79	0,27	0,28
Arbre de décision	0,69	0,16	0,55	0,24	0,75	0,30	0,45
Forêt aléatoire	0,78	0,25	0,67	0,36	0,82	0,20	0,33
Naive Bayes	0,46	0,14	0,91	0,23	0,55	0,58	0,09
K plus proches voisins	0,77	0,23	0,64	0,34	0,81	0,22	0,36

TABLEAU 4.5 – Métriques d'évaluation pour les modèles entraînés avec un *data set* équilibré en utilisant une combinaison de sous et sur-échantillonnage (1404 non alertes et 1404 alertes)

Modèle	<i>Accuracy</i>	Précision	Rappel	Score F1	Score $F_\beta$ $\beta = 0,4$	Taux fausses alertes	Taux non détections
Régression logistique	0,65	0,19	0,85	0,31	0,72	0,37	0,15
Support Vector Machines	0,66	0,19	0,83	0,31	0,73	0,36	0,17
Arbre de décision	0,64	0,14	0,56	0,22	0,72	0,35	0,44
Forêt aléatoire	0,71	0,19	0,67	0,30	0,77	0,29	0,33
Naive Bayes	0,36	0,12	0,94	0,21	0,43	0,70	0,06
K plus proches voisins	0,70	0,19	0,69	0,29	0,76	0,30	0,31



#### 4.9. STRATÉGIE DE MODÉLISATION

Nous avons testé le comportement du modèle forêt aléatoire suite à un entraînement avec différentes tailles et configurations du *data set*. Les Tableaux 4.6 à 4.7 présentent les résultats obtenus avec les différentes configurations. Le Tableau 4.6 présente les résultats pour des entraînements effectués avec un *data set* non équilibré, composé de 351 données d’alertes et un nombre variable de données de non-alertes. Le nombre de données conformant le *data set* des non-alertes a toujours été inférieur à 3417. Il a fallu donc faire un processus de sous-échantillonnage. Ces résultats mettent en évidence que plus le nombre de données dans le *data set* des non-alertes est important, plus l’*accuracy* et le score  $F_\beta$  sont importants. Ainsi, le taux de fausses alertes diminue et le taux de non-détections augmente.

TABLEAU 4.6 – Métriques d’évaluation pour le modèle forêt aléatoire entraîné avec un *data set* composé de 351 données alertes et un nombre variable de non alertes

Taille <i>data set</i> non alertes	<i>Accuracy</i>	Précision	Rappel	Score F1	Score $F_\beta$ $\beta = 0,4$	Taux fausses alertes	Taux non détections
1053	0,82	0,28	0,60	0,38	0,85	0,15	0,40
877	0,80	0,26	0,62	0,36	0,83	0,18	0,38
702	0,78	0,25	0,67	0,36	0,82	0,20	0,33
351	0,61	0,17	0,80	0,27	0,69	0,41	0,20

Le Tableau 4.7 présente les résultats obtenus pour le modèle entraîné avec un *data set* équilibré à un nombre variable de données sur chaque catégorie. Il a fallu mettre en place une démarche de sous et de sur-échantillonnage. Les résultats montrent qu’en augmentant la taille du *data set* jusqu’à 1404, l’*accuracy* et le score  $F_\beta$  augmentent. Néanmoins, à partir de 1404, ces métriques commencent à diminuer à nouveau. Cela peut être dû au fait que, plus la taille du *data set* augmente, plus la proportion de données réelles dans la catégorie d’alertes diminue. Le modèle s’entraîne donc sur des données moins fiables.

TABLEAU 4.7 – Métriques d’évaluation pour le modèle forêt aléatoire entraîné avec un *data set* équilibré composé d’un nombre variable de données

Taille <i>data set</i>	<i>Accuracy</i>	Précision	Rappel	Score F1	Score $F_\beta$ $\beta = 0,4$	Taux fausses alertes	Taux non détections
1755	0,65	0,15	0,62	0,25	0,73	0,34	0,38
1404	0,71	0,19	0,67	0,3	0,77	0,29	0,33
1053	0,69	0,19	0,73	0,3	0,75	0,32	0,27
702	0,68	0,19	0,77	0,3	0,75	0,33	0,23

De façon générale, nous pouvons remarquer que, sous conditions similaires d’entraînement, à major nombre de données dans la catégorie non-alertes du *data set*, le modèle forêt aléatoire a plus du mal à lever des alertes. Ce fait a un impact intéressant sur la diminution du taux de fausses alertes mais au détriment de l’augmentation du nombre de non-détections. Ce comportement est surtout visible dans le Tableau 4.6. Nous constatons que les meilleures performances sont atteintes sous cette

configuration d’entraînement et qu’il y a bien un compromis entre le taux de fausses alertes et celui de non-détections.

### 4.9.2.2 Importance des caractéristiques pour la prise de décision

Nous avons utilisé la technique d’importance des caractéristiques par permutation proposée par Scikit-learn [Pedregosa et al., 2011] pour identifier la caractéristique ou les caractéristiques du *data set* sur lesquelles le modèle se concentre le plus pour prendre la décision de classification.

La méthode d’importance des caractéristiques par permutation est une technique d’inspection de modèle qui évalue l’impact de chaque caractéristique sur la performance d’un modèle ajusté à partir d’un ensemble de données tabulaires. Pour ce faire, la méthode implique la permutation aléatoire des valeurs d’une seule caractéristique à la fois. En observant comment cette permutation affecte le score du modèle, il est possible de déterminer l’importance relative de chaque caractéristique. Si la permutation d’une caractéristique entraîne une baisse significative de la performance du modèle, cela indique que le modèle dépend fortement de cette caractéristique pour ses prédictions. En revanche, si la permutation d’une caractéristique n’affecte pas de manière significative la performance du modèle, cela suggère que cette caractéristique n’est pas cruciale pour les prédictions du modèle.

Nous avons appliqué cette méthode pour les modèles forêt aléatoire et K plus proches voisins des Tableaux 4.3 et 4.4. Les Figures 4.18 et 4.19 présentent les résultats pour les deux modèles sous les deux configurations d’entraînement. A travers ces figures, nous pouvons constater que la caractéristique qui impacte le plus le score  $F_\beta$  des 4 modèles est : “NUM\_REF\_ETAGE\_00”. Cette caractéristique représente le nombre de pièces dans le kit qui proviennent de l’étage 0 des meubles de rangement des pièces de la zone de préparation des kits.

Ces résultats semblent cohérents selon notre expérience sur le terrain industriel. La prise d’une pièce de cette position n’est pas très pratique pour les opérateurs. Cette difficulté ergonomique peut donc entraîner des erreurs.

Une autre remarque intéressante est, contrairement à ce qui a été exprimé en Section 4.5.3, le nombre de supports physiques des pièces composant le kit (“NUM\_SUPPORT”) ne se révèle pas très importante pour la prise de décision.

## 4.9. STRATÉGIE DE MODÉLISATION

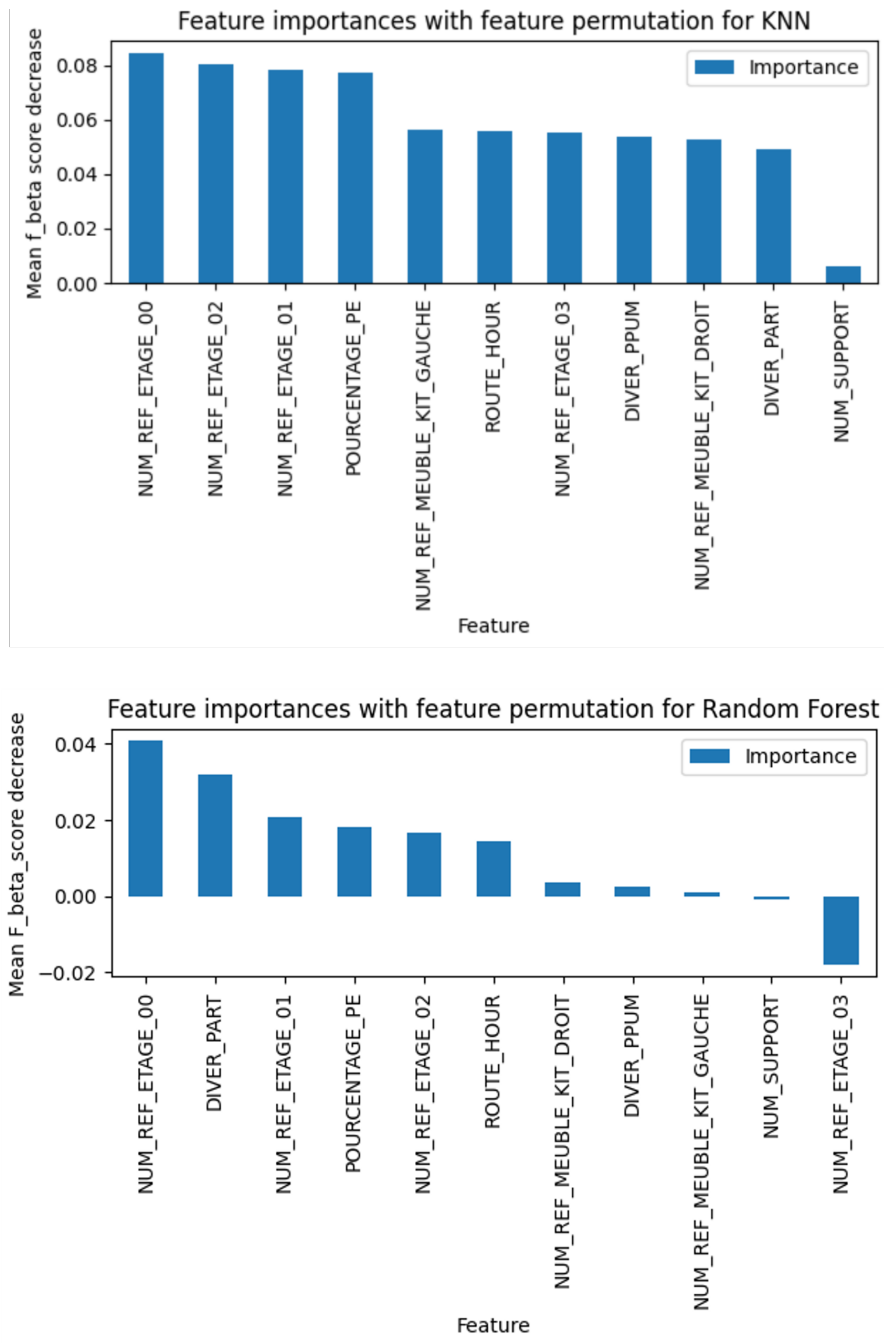


FIGURE 4.18 – Importance des caractéristiques pour les modèles K plus proches voisins et forêt aléatoire du Tableau 4.3

## 4.9. STRATÉGIE DE MODÉLISATION

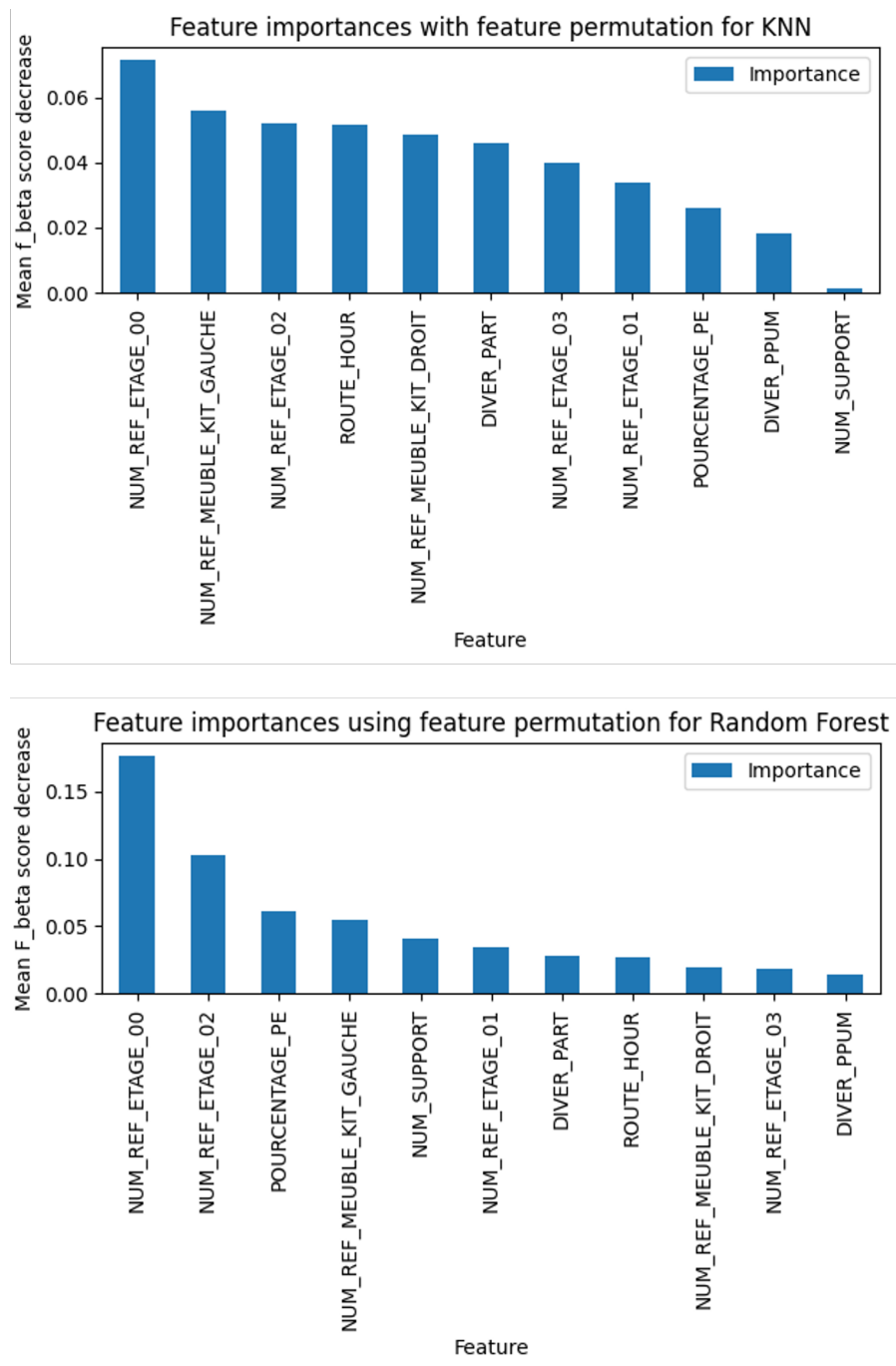


FIGURE 4.19 – Importance des caractéristiques pour les modèles K plus proches voisins et forêt aléatoire du Tableau 4.4

Pour résumer, nous pouvons dire qu’après l’application de techniques d’augmentation des données et d’équilibrage des distributions des classes, la performance des modèles développés est améliorée et ils arrivent à bien distinguer les 2 catégories.

Les résultats obtenus en termes de fausses alertes et de non-détections se révèlent convenables pour le cas d'étude industriel et cela encourage à la réalisation de tests avec un *data set* plus important et moins déséquilibré.

Ces résultats démontrent aussi que nous pouvons exploiter les informations provenant du système multi-agents en étude non seulement pour réagir aux perturbations de pièce manquante mais aussi pour les détecter au plus vite. Ce permettrait donc d'avoir plus de temps disponible pour corriger le problème, éviter des potentiels arrêts de la ligne de production et ainsi améliorer sa performance.

### 4.10 Conclusion

Dans ce chapitre nous présentons une démarche pour détecter de manière anticipée, juste à la sortie de la zone de kitting, la problématique de pièces manquantes dans les kits. La méthode proposée exploite des informations provenant du système multi-agents proposé dans le Chapitre 3. Ces informations sont collectées grâce aux différents systèmes d'information qui suivent le processus d'approvisionnement de pièces à la ligne d'assemblage.

Nous avons agrégé ces données afin de constituer un *data set* composé d'un ensemble d'11 colonnes qui caractérisent les kits et une colonne qui renseigne sur leur état de conformité. En termes de taille, ce *data set* comporte environ 5200 observations, dont 500 portent l'étiquette d'alerte.

Nous avons utilisé ce *data set* pour entraîner des modèles d'IA permettant de classifier les kits en deux catégories, conformes ou non conformes, en se basant sur l'11 caractéristiques définies. Plusieurs modèles de classification ont été entraînés sous des configurations différentes du *data set*, notamment (i) avant l'équilibrage des classes et (ii) après l'équilibrage des classes dans le *data set*.

La performance des modèles de classification entraînés sous la deuxième configuration, se révèle meilleure que celle des modèles entraînés sous la première. Les algorithmes K plus proches voisins et forêt aléatoire sont ceux qui présentent les meilleurs résultats avec des taux d'environ 20% et 30% en termes de fausses alertes et de non-détections.

Les résultats obtenus en termes de fausses alertes et de non-détections se révèlent convenables pour le cas d'étude industriel et démontrent qu'il est possible d'exploiter les informations provenant du système multi-agents en étude non seulement pour réagir aux perturbations de pièce manquante mais aussi pour les détecter au plus vite. Cela permettrait d'avoir plus de temps disponible pour corriger le problème, éviter des potentiels arrêts de la ligne de montage et ainsi améliorer la performance du système de production.

Enfin, les modèles utilisés dans cette section représentent une boîte noire en termes de fonctionnement. Nous avons pu identifier que la caractéristique qui impacte le plus les décisions des modèles, c'est le nombre des pièces provenant de l'étage 0 des meubles de kitting. Néanmoins, pour envisager

#### 4.10. CONCLUSION

---

un possible déploiement sur le terrain industriel et une bonne adoption, des travaux futurs doivent se consacrer au développement d'une approche permettant d'expliquer leurs décisions de façon plus claire.

#### 4.10. CONCLUSION

---

## Chapitre 5

# Application de la méthodologie proposée sur un nouveau cas d'étude

### Contenu

---

5.1	Introduction . . . . .	140
5.2	Description du problème . . . . .	141
5.3	Méthodologie . . . . .	142
5.3.1	Architecture multi-agents du système d'étude . . . . .	142
5.3.2	Constitution du <i>data set</i> . . . . .	143
5.3.3	Modèle testé et métriques d'évaluation utilisées . . . . .	145
5.4	Résultats expérimentaux . . . . .	145
5.5	Conclusion . . . . .	148

---



### 5.1 Introduction

Dans les Chapitres 3 et 4, nous avons présenté une méthode multi-agents permettant de détecter et de réagir à des anomalies ayant lieu lors du processus d'assemblage d'une ligne de production automobile. Nous avons appliqué cette méthode au processus d'approvisionnement de pièces à la ligne de montage par kitting.

L'objectif de ce chapitre est d'évaluer la généralisation de la méthode proposée à travers son application sur un nouveau cas d'étude. Ce cas d'étude a lieu au sein du processus de la tôlerie. La tôlerie automobile est un processus de fabrication qui consiste à former et à assembler des pièces en tôle pour construire la carrosserie d'un véhicule automobile. Ce processus comprend plusieurs étapes, notamment le soudage et l'assemblage des pièces en tôle. Dans le contexte de notre étude, ces tâches sont hautement automatisées. Les pièces en tôle sont transportées et assemblées par des ilots robotisés implantés en mode flow shop.

Ces ilots robotisés peuvent présenter plusieurs types de perturbations qui affectent leur rendement opérationnel ainsi que le rendement opérationnel global du processus. L'une des causes qui peuvent arrêter la production d'un ilot robotisé est le désamorçage du flux, c'est-à-dire, l'ilot robotisé ne peut pas produire car il n'est pas alimenté en pièces. De plus, l'arrêt d'un ilot robotisé peut impacter les ilots en amont ou aval du flux de production.

La baisse en rendement opérationnel du système de production se traduit en nombre de carrosseries sensées être produites et qui, finalement, ne l'ont pas été. Pour rattraper ce retard et pouvoir alimenter les processus en aval, à savoir la peinture et le montage, des heures supplémentaires ainsi que des journées de production pendant les week-ends doivent être mises en place.

Dans ce contexte, la prédiction des arrêts, avec un horizon de temps suffisant, pourrait permettre de mettre en place des actions pour les éviter et ainsi, améliorer la performance du système de production et éviter l'engagement des coûts supplémentaires.

Basés sur les développements présentés dans les chapitres précédents, dans ce chapitre nous proposons une méthode multi-agents pour prédire les arrêts par manque de pièce d'une station de travail afin de pouvoir mettre en place des actions, au bon moment, pour les éviter.

Pour ce faire, nous exploitons des données provenant de chacun des agents composant le système d'étude. Nous utilisons des données historiques du système réel pour déterminer et modéliser son état toutes les minutes, sur une période d'environ 6 mois. Nous nous servons également de ces données historiques, et des états du système, pour constituer un *data set* tabulaire. Ce *data set* est ensuite utilisé pour entraîner un modèle de machine learning de classification.

Grâce à ce modèle, nous prédisons l'état (marche ou arrêt) d'une station de travail donnée, en fonction de l'état du système en amont de la station de travail étudiée. Nous entraînons le modèle de

prédiction sous différentes configurations permettant d'effectuer la prédiction à plusieurs horizons de temps différents. Nous évaluons les performances du modèle en réalisant des tests off-line, c'est-à-dire, en utilisant une partie des données historiques qui n'ont pas été utilisées pour l'entraînement.

### 5.2 Description du problème

Lors du processus de tôlerie, des pièces en tôle sont assemblées par des ilots robotisés pour construire la carrosserie des véhicules. Ces ilots robotisés sont implantés en série et travaillent en mode flow shop. Pendant une journée de production, ces ilots robotisés peuvent présenter des arrêts de production suite à plusieurs raisons, dont l'arrêt à cause d'un manque de pièces. L'arrêt d'un ilot peut impacter les ilots voisins. Néanmoins, afin d'éviter la propagation du problème sur l'ensemble du flux de production, des stocks intermédiaires sont placés entre deux ilots.

Actuellement, quand un risque de désamorçage du flux est identifié, c'est-à-dire, quand un stock intermédiaire atteint un niveau très faible, un opérateur doit prendre des pièces d'un stock de sécurité placé à côté des ilots, les porter et les charger manuellement dans l'ilot robotisé afin éviter l'arrêt de production.

Malgré les efforts des opérateurs, des arrêts de production par manque de pièces, de plusieurs minutes, peuvent avoir lieu pendant la journée. En effet, les opérateurs ne peuvent pas être suffisamment réactifs face à l'alerte.

L'objectif de ce travail est de modéliser le système d'étude en tant qu'un système multi-agents, permettant de prédire des arrêts par manque de pièce des ilot robotisés. De cette façon, les opérateurs pourraient s'organiser pour intervenir au bon moment et ainsi, trouver des solutions pour éviter les arrêts. Pour ce faire, nous exploitons des données provenant de chacun des agents composant le système d'étude.

Notre système d'étude considère un périmètre partiel de tout le processus de tôlerie, mais il pourrait être extrapolé à l'ensemble d'ilots robotisés impliqués dans ce processus. Le système d'étude est composé de 5 ilots robotisés : I0, I1, I2, I3 et I4. Chaque ilot robotisé peut être composé de plusieurs zones de travail, où des opérations d'assemblage sont réalisées.

La Figure 5.1 illustre le flux des pièces entre chaque zone de travail et entre les différents ilots robotisés. Les flèches à traits continus (S0, S1, S2 et S3), représentent des flux avec un stock intermédiaire entre la zone de départ et la zone d'arrivée. Les flèches à traits interrompus représentent des flux sans stocks intermédiaires. Cela veut dire que, par exemple, entre la zone 1 et la zone 2 de l'ilot I1 il n'y a maximum qu'une pièce, alors qu'entre l'ilot I1 et l'ilot I3, il peut y en avoir plusieurs.

Nous disposons des informations telles que le nombre de pièces dans les stocks intermédiaires, ainsi que l'état des machines (en marche ou à l'arrêt) toutes les minutes. En considérant un historique de ces

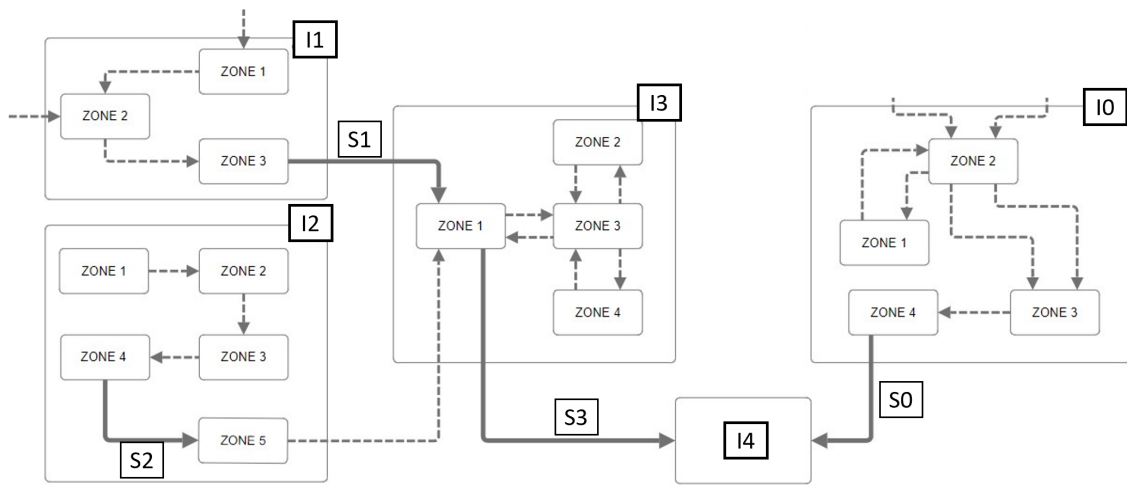


FIGURE 5.1 – Disposition du périmètre d'étude dans le processus de tôlerie automobile

informations, sur une fenêtre de temps, nous souhaitons prédire si le dernier ilot robotisé du système d'étude (I4) sera en état de marche normale ou en état d'arrêt par manque de pièce. Nous utilisons également des données historiques pour tester l'approche proposée.

## 5.3 Méthodologie

### 5.3.1 Architecture multi-agents du système d'étude

Nous modélisons le processus de tôlerie sous la forme d'un MAS. La Figure 5.2 présente une adaptation de l'architecture multi-agents proposée dans le Chapitre 2 pour le système d'étude actuel.

Le premier niveau, le niveau physique, est composé des 4 ilots robotisés en amont de I4 (I0, I1, I2 et I3), des 4 meubles de stockage intermédiaire (S0, S1, S2 et S3) et des leurs systèmes de contrôle (automates industriels). Le deuxième niveau est composé de la représentation virtuelle de chacun des agents physiques. Ils ont pour mission de collecter les données de ses agents physiques correspondants. Le troisième niveau est occupé par l'agent central. Cette fois-ci, cet agent collectera des informations provenant de tous les agents virtuels du deuxième niveau afin de prédire un éventuel arrêt par manque de pièces de l'ilot robotisé I4.

Une fois qu'un arrêt est détecté, l'information est transmise aux opérateurs pour mettre en place la marche dégradée sur l'ilot I4 au bon moment. Dans une future version plus développée sur ce cas d'étude, la prédiction d'un arrêt pourrait déclencher un processus de négociation entre les agents du deuxième niveau afin de proposer aux opérateurs des alternatives permettant de faire face à la perturbation, comme détaillé en Section 2.3.2.

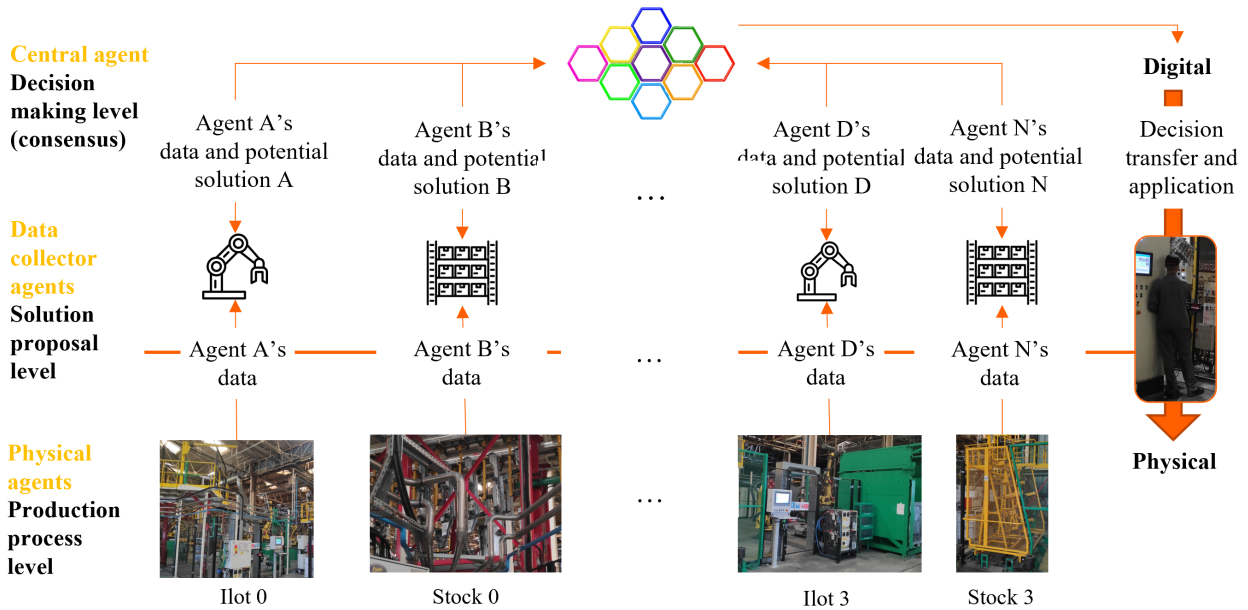


FIGURE 5.2 – Architecture multi-agents pour le périmètre d'étude du processus de tôlerie

### 5.3.2 Constitution du *data set*

De façon similaire à la démarche décrite dans la Section 4.5, afin d'entraîner le modèle de prédiction nous avons constitué un *data set* tabulaire avec des données historiques provenant du système d'étude.

Toutes les données utilisées ont été collectées à travers les automates contrôlant les ilots robotisés. Ces données ont été remontées et stockées dans un lac de données hébergé sur GCP. Nous avons donc réalisé l'extraction, le nettoyage et l'agrégation de ces données directement sur cette plateforme. Etant donné que les données ont été générées par des événements qui ont eu lieu pendant le fonctionnement du système réel, nous avons utilisé leur horodatage pour effectuer l'agrégation.

Nous disposons des informations telles que le nombre de pièces dans les stocks intermédiaires, ainsi que l'état des machines (en marche ou à l'arrêt) toutes les minutes. A titre d'exemple, la Figure 5.3 montre l'évolution dans le temps du nombre de pièces dans le meuble de stockage intermédiaire S1, dans une fenêtre de 4 heures de production. Cela est représenté par les points bleus sur la figure. Cette figure montre également, l'état de l'ilot I4 à chaque minute sur la même période. Les périodes du graphique colorées en jaune représentent les périodes où l'ilot I4 était à l'arrêt par manque de pièces. Le reste des périodes, l'ilot fonctionnait normalement.

Grâce à ces informations, nous pouvons identifier l'état du système d'étude, toutes les minutes. Ainsi, l'état du système d'étude à l'instant  $t$ , serait défini par l'état de marche ou d'arrêt (1 ou 0, respectivement) des ilots robotisés, ainsi que le nombre de pièces dans les stocks intermédiaires

### 5.3. MÉTHODOLOGIE

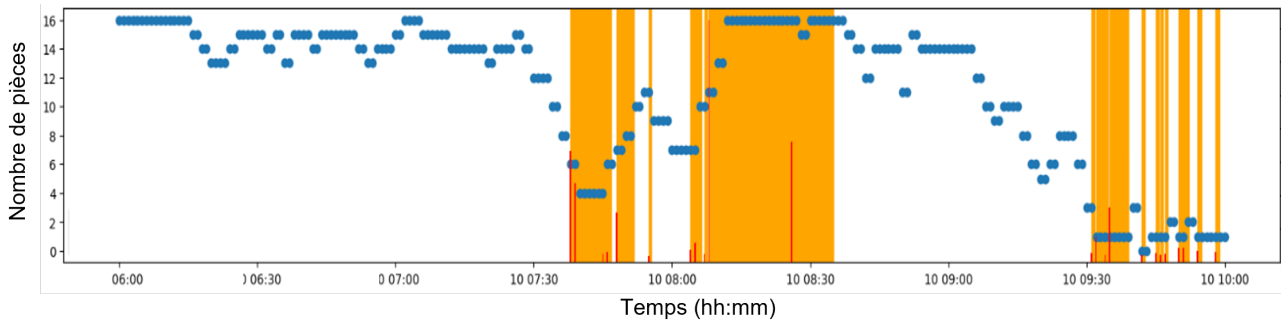


FIGURE 5.3 – Evolution dans le temps du nombre de pièces dans le meuble de stockage intermédiaire S1

composant le système. Le Tableau 5.1 présente un exemple de l'état du système à l'instant  $t$ . Veuillez remarquer que les colonnes représentent l'état de chaque agent du système d'étude.

TABLEAU 5.1 – Exemple d'état du système à l'instant  $t$

Temps	Etat I0	Etat I1	Etat I2	Etat I3	Nombre pièces en S0	Nombre pièces en S1	Nombre pièces en S2	Nombre pièces en S3	Etat I4 instant $t$
$t$	1	0	1	1	3	6	7	5	1

L'objectif est de prédire l'état de l'ilot I4 en fonction de l'état du système en amont de ce poste de travail. Nous nous servons des données historiques pour constituer un *data set* tabulaire, permettant d'entraîner un modèle de classification. La variable cible de ce *data set* correspond à l'état de l'ilot I4 et les caractéristiques correspondent à l'état des agents en amont de I4.

En complément, pour la constitution des caractéristiques, nous considérons deux éléments importants : l'horizon de prédiction ( $h$ ) et la fenêtre de temps analysée ( $p$ ) pour prendre une décision. En d'autres termes, nous utilisons un ensemble d'états du système en amont de I4, compris dans la période  $[t - (h + p), t - h]$ , pour prédire l'état de I4.

Cela nous permet (i) de connaître l'état de I4  $h$  minutes avant qu'un éventuel arrêt ait lieu, et d'entreprendre les actions nécessaires pour l'éviter, et (ii) de prendre en compte la composante historique ou temporelle de l'état du système pour faire la prédiction (l'état du système à l'instant  $t$  dépend de son état à l'instant  $t-1$ ). Le Tableau 5.2 illustre l'allure du *data set* constitué.

Le *data set* final est composé de 3649 observations, dont 2000 présentent un état de marche pour I4 et 1649 un état d'arrêt par manque de pièces. Pour la réalisation de tests, nous avons utilisé 2919 lignes pour l'entraînement des modèles et 730 pour les tester. Également, nous avons fait varier les valeurs de  $h$  et de  $p$  pour entraîner les modèles sous différentes configurations.

## 5.4. RÉSULTATS EXPÉRIMENTAUX

---

TABLEAU 5.2 – Illustration du *data set* constitué pour le système d'étude

Temps	Etat système t-(h+p)	Etat système t-(h+...)	Etat système t-(h+1)	Etat système t-h	Etat I4
t	[1,0,1,1,3,6,7,5]	...	[1,1,1,1,3,6,7,5]	[1,1,1,0,3,3,7,4]	1
t+1	[0,1,1,1,4,5,8,5]	...	[1,1,1,0,3,3,7,4]	[1,1,1,1,3,12,7,4]	1
t+2	[1,1,1,0,3,6,9,3]	...	[1,1,1,1,3,12,7,4]	[1,1,0,1,3,10,8,5]	0
t+ ...	...	...	...	...	...

### 5.3.3 Modèle testé et métriques d'évaluation utilisées

Basé sur les performances obtenues avec le modèle forêt aléatoire dans les tests présentés à la Section 4.9.2, nous pouvons constater que ce type de modèle est pertinent pour la classification de *data sets* tabulaires.

De plus, selon la littérature les modèles basés sur des arbres des décisions sont les plus efficaces pour exploiter des *data sets* tabulaires. [Grinsztajn et al., 2022] présente un benchmark où plusieurs algorithmes basés sur des arbres de décisions et plusieurs algorithmes d'apprentissage profond sont comparés. L'algorithme XGBoost présente les meilleurs résultats, juste devant l'algorithme forêt aléatoire (les deux étant des algorithmes basés sur des arbres de décisions). Pour le présent cas d'étude, nous utilisons donc l'algorithme XGBoost [Chen et Guestrin, 2016] pour effectuer des prédictions sur notre *data set* tabulaire.

Pour évaluer les modèles de classification développés, nous utilisons les métriques *accuracy*, précision, rappel et score  $F_1$ , qui sont fréquemment employées dans des projets de machine learning de façon générale. Cependant, nous utilisons aussi les métriques taux de fausses alertes et taux de non-détections, car elles sont plus parlantes pour nos interlocuteurs du terrain industriel.

La Figure 4.13 présente la matrice de confusion type qui sert de base de calcul pour les métriques d'évaluation. Cette matrice regroupe le nombre de vrais positifs (TP), de faux positifs (FP), de vrais négatifs (TN) et de faux négatifs (FN) résultants de la classification effectuée par le modèle. Elle permet également de donner un aperçu des performances d'un modèle de classification. Le calcul de ces métriques est effectué avec les équations (4.1) à (4.6).

## 5.4 Résultats expérimentaux

Nous avons entraîné le modèle XGBoost sous différentes configurations du *data set*. Nous avons fait varier l'horizon de prédiction, il a pris les valeurs de 7, 10 et 15 minutes. Également, nous avons fait varier la taille de la fenêtre de temps prise en compte pour constituer les caractéristiques de chaque observation. Nous avons fait varier cette fenêtre de 30 à 60 minutes. La Figure 5.4 présente l'*accuracy*

## 5.4. RÉSULTATS EXPÉRIMENTAUX

---

obtenu par les modèles.

En observant cette figure, nous pouvons remarquer que, de façon générale, l'*accuracy* pour tous les modèles est compris entre 0.71 et 0.78. Ces valeurs d'*accuracy* sont assez similaires aux résultats obtenus par les modèles présentés dans la Section 4.9.2 pour le cas d'étude de la pièce manquante. Nous pouvons remarquer également, que plus la taille de la fenêtre d'analyse est importante, plus l'*accuracy* des modèles augmente. Les meilleures valeurs d'*accuracy* sont atteintes par les modèles quand la fenêtre d'analyse est de 60 minutes. En complément, plus l'horizon de prédiction est loin, plus l'*accuracy* diminue. Ainsi, le modèle qui présente le meilleur résultat, c'est le modèle entraîné avec une fenêtre de 60 minutes et avec un horizon de prédiction de 7 minutes.

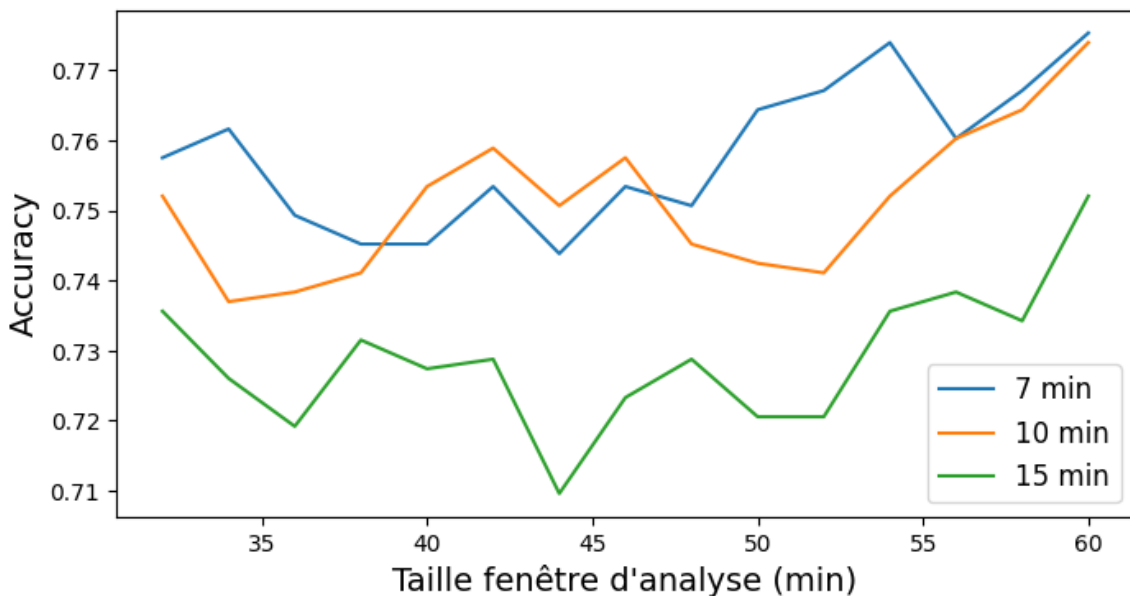


FIGURE 5.4 – *Accuracy* des modèles entraînés en fonction de la taille de la fenêtre d'analyse

Le Tableau 5.3 présente non seulement l'*accuracy* mais aussi les autres métriques d'évaluation calculées pour les modèles. Nous y retrouvons les modèles entraînés avec une fenêtre d'analyse de 60 minutes et avec un horizon de prédiction de 7, 10 et 15 minutes.

Ce tableau permet de mettre en évidence que, en plus d'avoir des résultats satisfaisants en termes d'*accuracy*, les modèles présentent des bons résultats en termes de précision, rappel et Score F1. En outre, les taux de fausses alertes et de non-détections sont tout à fait acceptables pour l'application.

Nous pouvons remarquer que les résultats des modèles avec  $h=7$  et  $h=10$  minutes sont très similaires, présentant tous les deux des valeurs de précision et de rappel de 0.76 et de 0.79, respectivement. Les résultats en termes de fausses alertes présentent une hausse quand on augmente l'horizon de prédiction. Néanmoins, le taux de non-détections peut rester stable ou peut même présenter une faible

## 5.4. RÉSULTATS EXPÉRIMENTAUX

décroissance avec l'augmentation de l'horizon de prédiction. Les résultats présentés par le Tableau 5.3 sont également illustrés par les matrices de confusion de la Figure 5.5.

TABLEAU 5.3 – Métriques d'évaluation pour les modèles entraînés avec une fenêtre d'analyse de 60 minutes, sous différentes valeurs de h

Modèle	<i>Accuracy</i>	Précision	Rappel	Score F1	Taux fausses alertes	Taux non détections
h = 7 minutes	0,78	0,76	0,79	0,77	0,23	0,21
h = 10 minutes	0,77	0,76	0,79	0,78	0,25	0,21
h = 15 minutes	0,75	0,72	0,81	0,76	0,30	0,19

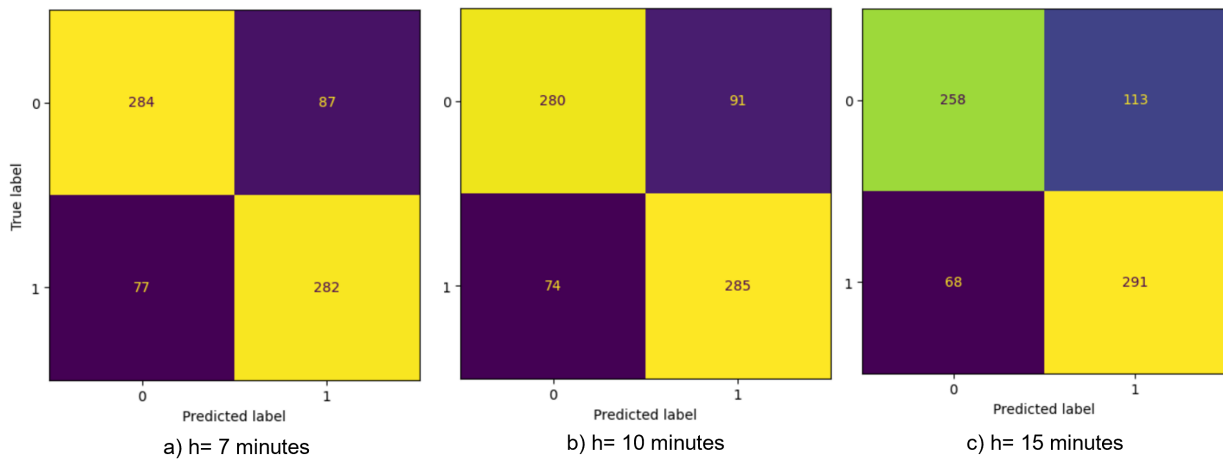


FIGURE 5.5 – Matrices de confusion pour les modèles entraînés avec une fenêtre d'analyse de 60 minutes, sous différentes valeurs de h

En comparant ces résultats avec les résultats présentés en Section 4.9.2, nous pouvons constater que nous obtenons des résultats similaires dans les deux cas, surtout en termes de taux des fausses alertes et de non-détections. D'un point de vue "métier", nos interlocuteurs du terrain industriel se montrent très satisfaits par rapport à ces taux. De leur point de vue, plus l'horizon de prédiction est grand, plus ce serait pertinent pour eux. Néanmoins, un horizon de prédiction de 10 minutes pourrait répondre à leurs besoins. Ce délai permettrait de s'organiser pour mettre en place les actions nécessaires pour éviter des arrêts.

Actuellement, les arrêts par manque de pièces peuvent représenter jusqu'à 10% du temps de travail de l'ilot robotisé étudié. Ainsi, malgré la présence des fausses alertes, la détection d'environ 79% de ces arrêts pourrait permettre d'améliorer d'environ 8% le rendement opérationnel de cet ilot de production.

Afin de tester la pertinence de la méthode proposée, nous avons appliqué exactement le même principe sur un autre ilot robotisé présentant la même disposition du flux de production. Les résultats



obtenus ont été très similaires et ils ont été satisfaisants pour les acteurs du terrain industriel.

A ce jour, uniquement deux ilots robotisés ont été étudiés. Néanmoins, grâce aux bons résultats obtenus sur le deuxième ilot, un déploiement sur d'autres ilots du processus de tôlerie est envisageable. Ce déploiement permettrait donc la création d'un agent capable de détecter les arrêts par manque de pièce de chaque ilot robotisé. Nous pourrions donc envisager de faire interagir ces agents non seulement lors de la phase de détection mais aussi lors de la phase de recherche de solutions pour répondre à l'alerte. Des processus de négociations pourraient être mis en place pour prendre des décisions ou pour orienter la prise de décisions par les opérateurs. Par exemple, si un arrêt se révèle complètement imminent, l'ilot robotisé en question pourrait communiquer cette information aux ilots robotisés en amont pour qu'ils réduisent leurs vitesses de production afin d'éviter la saturation du flux et surtout pour diminuer la consommation énergétique de la chaîne. Également, si deux ilots robotisés semblent présenter des arrêts dans les prochaines minutes, la négociation entre ilots pourrait permettre d'orienter les opérateurs sur le choix de l'ilot où intervenir en premier lieu.

## 5.5 Conclusion

Dans ce chapitre, nous avons présenté l'application de la méthode multi-agents proposée dans les chapitres précédents, sur un cas d'étude du processus de tôlerie automobile. Nous avons utilisé la méthode multi-agents pour prédire l'apparition d'arrêts par manque de pièces d'un ilot robotisé composant ce processus de production.

Pour ce faire, nous avons collecté des données historiques provenant des différents agents composant le système d'étude. Nous avons agrégé et traité ces données réelles afin de reconstituer l'état du système de production toutes les minutes, sur une période d'environ 6 mois.

Cela nous a permis de constituer un *data set* tabulaire où la colonne cible est l'état de marche ou d'arrêt de l'ilot robotisé sur lequel nous voulons effectuer la prédiction et les caractéristiques sont l'état du système en amont de cet ilot, sur une période de 30 à 60 minutes précédentes. Ce *data set* a servi à entraîner un modèle ML de classification sous différentes configurations. Nous avons fait varier l'horizon de prédiction ainsi que la fenêtre d'analyse prise en compte pour la prise de décision des modèles.

Les résultats obtenus montrent que les meilleures performances sont atteintes quand la fenêtre d'analyse est de 60 minutes. Sous cette configuration, les modèles présentent des niveaux d'*accuracy* compris entre 75 et 78% avec des taux de fausses alertes et de non-détections inférieurs à 30%.

Les résultats obtenus en termes de fausses alertes et de non-détections se révèlent convenables pour le cas d'étude industriel et répondent aux besoins des acteurs du terrain industriel. En outre, les résultats avec un horizon de prédiction de 10 minutes permettent de constater qu'une amélioration

## 5.5. CONCLUSION

---

jusqu'à 8% du rendement opérationnel de l'îlot robotisé étudié pourrait être possible.

A présent, la méthode a été testée sur deux îlots robotisés composant le système de production de la tôlerie. Grâce aux résultats obtenus, un déploiement sur d'autres îlots est envisageable. Ce déploiement pourrait permettre de mettre en place une interaction entre les agents des îlots robotisés non seulement pour la détection des arrêts mais aussi pour le processus de recherche des solutions pour éviter les arrêts.

Les travaux futurs se consacraient sur le déploiement de cette méthode sur d'autres îlots robotisés et sur la mise en place d'un protocole de négociation entre les îlots robotisés pour proposer des solutions ou orienter les opérateurs lors de la prise de décisions. En outre, cela pourrait être accompagné de la réalisation des simulations en temps réel afin de déterminer l'impact que pourrait avoir chaque arrêt et ainsi être capable de prioriser de manière pertinente. Également, pour rendre possible un déploiement sur le terrain industriel, il faut prévoir la mise en place de tests on-line. Il serait donc nécessaire d'avoir accès aux données en temps réel.

## 5.5. CONCLUSION

---

# Conclusion générale et perspectives

## Conclusion générale

Dans ce document, nous avons présenté une méthode basée sur les systèmes multi-agents pour optimiser la performance globale d'un processus de production, en gérant les interactions des différents agents impliqués. Le processus de logistique interne d'une usine de montage automobile a été utilisé comme cas d'étude principal dans ces travaux de recherche. Également, une étude supplémentaire portant sur le périmètre de la tôlerie automobile a été réalisée.

La première partie de ce document a introduit le cadre industriel et scientifique de ces travaux, mettant en évidence la pertinence du sujet d'étude. Le travail sur le terrain dans des sites industriels de Renault et les nombreuses discussions avec les personnes en charge des métiers logistiques ont permis de comprendre le contexte de la thèse et de clarifier la problématique ainsi que ses objectifs. Ensuite, le processus d'approvisionnement de pièces à la ligne d'assemblage d'une usine automobile a été décrit.

Dans un deuxième temps, un état de l'art sur l'application des systèmes multi-agents dans des contextes industriels a permis d'identifier plusieurs initiatives dans divers secteurs d'activité, utilisant différents outils de modélisation ou de développement. Néanmoins, il a été constaté que très peu de ces initiatives ont été mises en œuvre dans des environnements industriels réels. La plupart des développements ont été réalisés dans des environnements contrôlés, tels que des laboratoires. De plus, l'application des systèmes multi-agents dans le domaine de la logistique interne a été peu explorée.

En s'appuyant sur la revue de la littérature, ce document a présenté la modélisation du processus de logistique interne d'une usine automobile sous la forme d'un système multi-agents. L'un des problèmes les plus récurrents rencontrés lors de ce processus de logistique interne, utilisant la méthode de kitting, est la problématique de pièce manquante dans un kit. Ce chapitre a présenté également une méthode basée sur les systèmes multi-agents pour réduire le temps de réaction à une alerte de pièce manquante déclenchée par un opérateur sur une ligne d'assemblage.

Nous avons utilisé des données historiques provenant d'une usine automobile pour modéliser l'état du système d'étude aux moments où un opérateur a déclenché l'alerte pour une pièce manquante dans l'un des kits. Nous avons testé la méthode proposée dans un scénario ponctuel ainsi que sur l'ensemble

## CONCLUSION GÉNÉRALE ET PERSPECTIVES

---

de scénarios d'alerte qui ont eu lieu au cours d'une journée de production. Les résultats obtenus ont démontré l'efficacité de l'approche proposée pour fournir des options pertinentes aux opérateurs de la ligne d'assemblage afin de réduire les temps de réaction lorsqu'un kit est défectueux et ainsi éviter des arrêts de ligne de production inutiles. Pour le scénario spécifique, la meilleure option suggérée a permis de réduire le temps de réaction de plus de 70% par rapport à la méthodologie utilisée actuellement dans l'usine pour répondre à ce type d'anomalie. En étendant l'analyse à un ensemble de 85 scénarios, l'approche proposée réduit le temps de réaction de 46% par rapport à la méthodologie existante. La méthode proposée permettrait également de diminuer de 72% le temps d'arrêt de production lié aux alertes de pièce manquante.

Après avoir constaté la pertinence des approches multi-agents pour des situations de réaction face à une perturbation, nous avons ensuite exploré l'utilisation des données du système multi-agents pour détecter et anticiper les anomalies grâce à l'IA. Nous présentons une démarche pour détecter de manière anticipée, juste à la sortie de la zone de kitting, la problématique de pièces manquantes dans les kits.

Nous collectons des informations grâce aux différents systèmes d'information qui suivent le processus d'approvisionnement de pièces à la ligne d'assemblage. Avec ces données, nous avons constitué un *data set* composé de onze colonnes caractérisant les kits et une colonne qui renseigne sur leur état de conformité. Nous avons entraîné plusieurs modèles ML de classification avec différentes configurations de cet ensemble de données. Les meilleurs résultats obtenus présentent des taux d'environ 20% et 30% en termes de fausses alertes et de non-détections.

Ces résultats sont jugés satisfaisants pour l'étude industrielle et démontrent qu'il est possible d'exploiter les informations provenant du système multi-agents non seulement pour réagir aux perturbations de pièces manquantes, mais aussi pour les détecter rapidement. De plus, la détection anticipée permettrait d'avoir plus de temps disponible pour corriger le problème, éviter d'éventuels arrêts de la ligne de montage et ainsi améliorer la performance du système de production.

La dernière partie de ces travaux se concentre sur l'application des méthodes multi-agents proposées à un cas d'étude ayant lieu dans le processus de la tôlerie. Nous avons utilisé la méthode multi-agents pour prédire l'apparition d'arrêts par manque de pièces dans un îlot robotisé faisant partie de ce processus de production.

Nous avons constitué un *data set* tabulaire avec des données historiques provenant des différents îlots robotisés composant le système d'étude. Avec ce *data set*, nous avons entraîné des modèles ML de classification afin de réaliser la prédiction de l'état d'un îlot robotisé à différents horizons temporels. Les résultats obtenus présentent des niveaux d'accuracy compris entre 75 et 78% avec des taux de fausses alertes et de non-détections inférieurs à 30%. Cela permettrait d'améliorer le rendement opérationnel de l'îlot robotisé étudié jusqu'à 8%.

Ce dernier chapitre a permis de prouver la possibilité de généralisation de la méthode proposée à travers son application sur un nouveau cas d'étude. Également, les résultats obtenus ont permis de convaincre nos interlocuteurs industriels des avantages que l'application de la méthode proposée pourrait fournir. Par conséquent, un déploiement sur d'autres îlots robotisés et sur d'autres applications du terrain industriel se révèle envisageable et prometteur.

### Perspectives

Les résultats présentés dans cette thèse ouvrent la voie à plusieurs perspectives intéressantes pour approfondir la recherche dans diverses directions. La première direction concerne la généralisation et l'amélioration de la méthode proposée. L'utilisation des MAS pour le développement de la méthode présente plusieurs avantages en termes de modularité, flexibilité, réactivité et scalabilité.

Concernant la modularité et la scalabilité, l'ajout ou le retrait d'agents au sein du système physique n'entrave pas les fonctions globales du MAS. Par exemple, que des kits soient ajoutés ou retirés du circuit, la solution reste opérationnelle, grâce à la capacité d'instancier les agents kits nécessaires. Grâce à ces caractéristiques, la solution pourrait être testée dans un tronçon de la ligne de montage, comportant un nombre d'agents différent.

Vis-à-vis de la flexibilité, seule la problématique des pièces manquantes dans les kits est actuellement adressée. De même, seuls les agents kits et les agents opérateurs sont considérés. Néanmoins, les caractéristiques des MAS nous permettront d'intégrer progressivement d'autres agents qui sont aussi impliqués dans le processus d'approvisionnement de pièces à la ligne d'assemblage. L'ajout de nouveaux agents permettrait d'envisager également l'intégration de nouveaux services, et donc de répondre à d'autres types de problématiques. Par exemple, l'intégration des agents opérateurs caristes pourrait permettre de répondre à une problématique de type emballage de pièce manquant au sein de la zone de kitting.

Dans la méthode proposée, des ensembles d'agents intelligents sont capables d'identifier des perturbations dans un processus de production et de proposer des solutions pour y faire face, en fonction des capacités des agents physiques. En considérant cette méthode comme un moyen d'exploiter des informations et des services provenant de différentes entités distribuées, son champ d'application s'élargit considérablement. Les deux applications présentées dans ce manuscrit, le processus de montage de portes et de tôlerie, se caractérisent par avoir un fonctionnement en mode flow shop. En extrapolant cette idée, la méthode proposée pourrait être étendue à d'autres processus de production fonctionnant en mode flow shop et composés de plusieurs sous-systèmes de nature distribuée, tels que l'industrie agroalimentaire ou l'industrie de production d'électroménagers.

Une piste d'amélioration qui s'avère pertinente est le couplage entre les systèmes multi-agents et

la simulation en ligne. Dans le protocole d'interaction utilisé, les agents participants doivent estimer le coût de la solution qu'ils proposent. Par exemple, dans le cas de l'anomalie pièce manquante dans un kit, chaque agent participant estime le temps de déplacement de l'opérateur. Cette estimation pourrait être réalisée à l'aide d'une simulation. De façon plus générale, une fois qu'une perturbation a été détectée et que la recherche de solutions est déclenchée, chacun des agents composant le système pourrait lancer une simulation, en considérant l'état actuel du système de production, afin de valider les hypothèses de solution qu'il propose et d'évaluer leur pertinence.

Le démarrage d'une simulation en considérant l'état actuel du système, c'est-à-dire démarrer une simulation dans un état différent de l'état initial, n'est pas un problème trivial. Néanmoins, pouvoir le faire représenterait plusieurs bénéfices. D'une part, dans ces travaux de thèse, nous avons validé les propositions effectuées en utilisant des données historiques provenant des systèmes réels étudiés. Cela nous a permis de réaliser des tests off-line basés sur des scénarios réels. Il serait néanmoins intéressant de pouvoir initialiser une simulation dans un état de perturbation réel, afin de tester le comportement de la méthode proposée, en tenant compte de l'évolution du système réel pendant la phase de calculs ou de recherche des solutions. Nous pourrions donc tester la méthode proposée de manière on-line, sur la base des scénarios réels.

D'autre part, avoir un simulateur capable de démarrer sur la base de données historiques réelles pourrait faciliter la comparaison des différentes solutions ou méthodologies multi-agents. D'un point de vue scientifique, cela permettrait de créer et alimenter les benchmarks dans le domaine des systèmes multi-agents et, d'un point de vue industriel, identifier la solution la plus adaptée pour une problématique donnée.

Dans la méthode proposée, les opérateurs gardent une place importante en tant que décideurs finaux. Ils ont la possibilité de choisir la solution la plus appropriée parmi celles proposées par le MAS et de valider les décisions prises par le système de détection. Aujourd'hui, ces interactions entre les opérateurs et notre système ne sont pas exploitées. Néanmoins, les retours effectués par les opérateurs pourraient être utilisés pour réentraîner et améliorer continuellement le système.

Une deuxième direction importante, pour les perspectives de recherche, concerne l'industrialisation et le déploiement sur le terrain industriel de la méthode proposée. Ce sujet soulève plusieurs questions qui n'ont pas encore été résolues mais sur lesquelles des travaux sont en cours au sein du Groupe Renault. D'abord, pour tester la solution sur le terrain industriel, notamment pour l'application de la pièce manquante dans un kit, il faut avoir accès aux données du processus de production en temps réel. Actuellement, la mise à disposition des données de production dans le data lake de GCP peut prendre plusieurs minutes. Il existe des configurations permettant d'avoir accès à ces données plus rapidement. Néanmoins, cela soulève la question de la structure d'hébergement des agents virtuels : faut-il déployer les agents virtuels dans des devices sur le terrain industriel pour être au plus proche

du processus ? Faut-il les déployer dans le cloud pour avoir accès à plus de données ? Ou faut-il les déployer dans des serveurs se trouvant entre ces deux extrêmes ?

D'autre part, pour imaginer un déploiement à plus grande échelle, sur d'autres applications et sur d'autres sites de fabrication, la maîtrise des interactions entre le système et les opérateurs est primordiale. L'adoption du système par les acteurs du terrain industriel est essentielle. Il est donc nécessaire d'assurer la formation des opérateurs, de prendre en compte leurs retours, et de mettre en place des stratégies d'explicabilité pour les algorithmes de détection d'anomalies. Pour aller plus loin, la question de la démocratisation de cette technologie constitue un axe de développement crucial. Il serait intéressant de réfléchir aux outils nécessaires à fournir et au processus à mettre en place pour permettre à d'autres personnes, non spécialistes du sujet, d'adapter la méthode proposée à leurs besoins.

Enfin, une troisième direction à prendre en compte concerne la durabilité et l'empreinte carbone numérique générée par ces types d'applications. Nous sommes conscients que les technologies utilisées ou évoquées dans ce manuscrit, telles que le stockage des données, la réalisation des simulations et l'entraînement des modèles d'IA, peuvent être très énergivores. Ainsi, la mise en place d'actions pour réduire cette consommation est nécessaire. L'une des pistes de développement sur ce sujet, porte sur l'utilisation des techniques d'apprentissage fédéré afin d'éviter le transfert, la duplication et donc le stockage des jeux de données de grande taille pour l'entraînement et le réentraînement des modèles d'IA.





# Bibliographie

- [Adam, 2015] Adam, E. (2015). A volatile knowledge approach to improve the autonomy of holons : Application to a flexible job shop manufacturing system. In Mařík, V., Schirrmann, A., Trentesaux, D., et Vrba, P., editors, *Industrial Applications of Holonic and Multi-Agent Systems*, pages 84–95, Cham. Springer International Publishing.
- [Alonso, 2002] Alonso, E. (2002). AI and agents : State of the art. *AI Magazine*, 23(3) :25–29.
- [Amblard et al., 2021] Amblard, F., Fallah-Seghrouchni, A. E., Gaudo, B., et Hanachi, C. (2021). Systèmes multi-agents. *Techniques de l'ingénieur Technologies logicielles Architectures des systèmes*, base documentaire : TIP402WEB.(ref. article : h5020). Publisher : Editions T.I.
- [Baker et al., 2020] Baker, B., Kanitscheider, I., Markov, T., Wu, Y., Powell, G., McGrew, B., et Mordatch, I. (2020). Emergent Tool Use From Multi-Agent Autocurricula. *arXiv :1909.07528*.
- [Balbo et al., 2016] Balbo, F., Adam, E., et Mandiau, R. (2016). Positionnement des systèmes multi-agents pour les systèmes de transport intelligents. *Revue des Sciences et Technologies de l'Information-Série RIA : Revue d'Intelligence Artificielle*, 30(3) :299–327.
- [Banks et al., 2007] Banks, A., Vincent, J., et Anyakoha, C. (2007). A review of particle swarm optimization. Part I : background and development. *Natural Computing*, 6(4) :467–484.
- [Banks et al., 1996] Banks, J., Carson, J. S., et Nelson, B. L. (1996). *Discrete-event System Simulation*. Prentice Hall.
- [Barbosa et al., 2015] Barbosa, J., Leitão, P., Adam, E., et Trentesaux, D. (2015). Dynamic self-organization in holonic multi-agent manufacturing systems : The adacor evolution. *Computers in Industry*, 66 :99–111.
- [Beauville dit Eynaud et al., 2019] Beauville dit Eynaud, A., Klement, N., Gibaru, O., Roucoules, L., et Durville, L. (2019). Identification of reconfigurability enablers and weighting of reconfigurability characteristics based on a case study. *Procedia Manufacturing*, 28 :96–101.
- [Bellifemine et al., 2007] Bellifemine, F. L., Caire, G., et Greenwood, D. (2007). *Developing Multi-Agent Systems with JADE*. John Wiley & Sons.
- [Block et al., 2016] Block, C., Morlock, F., Dorka, T., et Kuhlenkötter, B. (2016). A Human Centered Multi-Agent-System for Production Planning and Control. *Applied Mechanics and Materials*,

## BIBLIOGRAPHIE

---

- 840 :132–139. Conference Name : Robotics and Automated Production Lines ISBN : 9783038357599  
Publisher : Trans Tech Publications Ltd.
- [Bonabeau et al., 1999] Bonabeau, E., Dorigo, M., et Theraulaz, G. (1999). *From Natural to Artificial Swarm Intelligence*. Oxford University Press, Inc., USA.
- [Brynzér et Johansson, 1995] Brynzér, H. et Johansson, M. I. (1995). Design and performance of kitting and order picking systems. *International Journal of Production Economics*, 41(1) :115–125.
- [Bychkov et al., 2017] Bychkov, I., Oparin, G., Tchernykh, A., Feoktistov, A., Bogdanova, V., Dyadkin, Y., Andrukhova, V., et Basharina, O. (2017). Simulation modeling in heterogeneous distributed computing environments to support decisions making in warehouse logistics. *Procedia Engineering*, 201 :524–533.
- [Cardin, 2007] Cardin, O. (2007). *Apport de la simulation en ligne dans l'aide à la décision pour le pilotage des systèmes de production : application à un système flexible de production*. These de doctorat, Nantes.
- [Cardin et Castagna, 2006] Cardin, O. et Castagna, P. (2006). Handling uncertainty in production activity control using proactive simulation. *IFAC Proceedings Volumes*, 39(3) :579–584.
- [Cardin et L'Anton, 2018] Cardin, O. et L'Anton, A. (2018). Proposition of an implementation framework enabling benchmarking of Holonic Manufacturing Systems. page 267.
- [Catterson et al., 2012] Catterson, V. M., Davidson, E. M., et McArthur, S. D. J. (2012). Practical applications of multi-agent systems in electric power systems. *European Transactions on Electrical Power*, 22(2) :235–252.
- [Chawla et al., 2002] Chawla, N. V., Bowyer, K. W., Hall, L. O., et Kegelmeyer, W. P. (2002). SMOTE : Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16 :321–357.
- [Chen et Guestrin, 2016] Chen, T. et Guestrin, C. (2016). XGBoost : A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. arXiv :1603.02754.
- [Daoud et al., 2021] Daoud, A., Balbo, F., Gianessi, P., et Picard, G. (2021). A Generic Multi-Agent Model for Resource Allocation Strategies in Online On-Demand Transport with Autonomous Vehicles. In Endriss, U., Nowé, A., Dignum, F., et Lomuscio, A., editors, *Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021)*, pages 1489–1491. International Foundation for Autonomous Agents and Multiagent Systems. Extended abstract.
- [de Lima et al., 2020] de Lima, O. P., Santiago, S. B., Taboada, C. M. R., Rodríguez, J. L. M., Rodríguez, M. B. R., Maduro, M. R., Araújo, P. C. D. d., Junior, N. J. d. O., Zogahib, A. L. N., et Lima, J. C. d. S. (2020). *Conceptualization, Definition and Assessment of Internal Logistics through Different*

- Approaches Using Artificial Intelligence*. IntechOpen. Publication Title : Operations Management - Emerging Trend in the Digital Era.
- [Dias et al., 2016] Dias, L. M. S., Vieira, A. A. C., Pereira, G. A. B., et Oliveira, J. A. (2016). Discrete simulation software ranking — A top list of the worldwide most popular and used tools. In *2016 Winter Simulation Conference (WSC)*, pages 1060–1071. ISSN : 1558-4305.
- [Dogan et Birant, 2021] Dogan, A. et Birant, D. (2021). Machine learning and data mining in manufacturing. *Expert Systems with Applications*, 166 :114060.
- [Dorigo et al., 1999] Dorigo, M., Caro, G. D., et Gambardella, L. M. (1999). Ant Algorithms for Discrete Optimization. *Artificial Life*, 5(2) :137–172. Conference Name : Artificial Life.
- [Dorri et al., 2018] Dorri, A., Kanhere, S. S., et Jurdak, R. (2018). Multi-Agent Systems : A Survey. *IEEE Access*, 6 :28573–28593. Conference Name : IEEE Access.
- [Durfee et Rosenschein, 1994] Durfee, E. H. et Rosenschein, J. S. (1994). Distributed problem solving and multi-agent systems : Comparisons and examples. In *Proceedings of the Thirteenth International Distributed Artificial Intelligence Workshop*, pages 94–104.
- [Fadil et al., 2019] Fadil, A., Trentesaux, D., et Branger, G. (2019). Event management architecture for the monitoring and diagnosis of a fleet of trains : a case study. *Journal of Modern Transportation*, 27(3) :169–187.
- [Falco et Robiolo, 2019] Falco, M. et Robiolo, G. (2019). A Systematic Literature Review in Multi-Agent Systems : Patterns and Trends. In *2019 XLV Latin American Computing Conference (CLEI)*, pages 1–10.
- [Fauadi et al., 2018] Fauadi, M. H. F. M., Azimi, N. Z., Anuar, N. I., Ali, M. M., Sued, M. K., et Ramlan, S. (2018). Simulation Approach for Logistical Planning in a Warehouse : A Review. *Oriental Journal of Computer Science and Technology*, 11(4) :201–208.
- [Firme et al., 2023] Firme, B., Figueiredo, J., Sousa, J. M. C., et Vieira, S. M. (2023). Agent-based hybrid tabu-search heuristic for dynamic scheduling. *Engineering Applications of Artificial Intelligence*, 126 :107146.
- [Gamboa Quintanilla et al., 2016] Gamboa Quintanilla, F., Cardin, O., L’Anton, A., et Castagna, P. (2016). A modeling framework for manufacturing services in Service-oriented Holonic Manufacturing Systems. *Engineering Applications of Artificial Intelligence*, 55 :26–36.
- [Garnelo et Shanahan, 2019] Garnelo, M. et Shanahan, M. (2019). Reconciling deep learning with symbolic artificial intelligence : representing objects and relations. *Current Opinion in Behavioral Sciences*, 29 :17–23.
- [Georgeff et Lansky, 1987] Georgeff, M. P. et Lansky, A. L. (1987). Reactive reasoning and planning. In *AAAI*, volume 87, pages 677–682.

- [Giret et al., 2017] Giret, A., Trentesaux, D., Salido, M. A., Garcia, E., et Adam, E. (2017). A holonic multi-agent methodology to design sustainable intelligent manufacturing control systems. *Journal of cleaner production*, 167 :1370–1386.
- [Grinsztajn et al., 2022] Grinsztajn, L., Oyallon, E., et Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on tabular data? arXiv :2207.08815.
- [Gunasekaran et al., 2017] Gunasekaran, S., Ahmad, M., Tang, A., et Mostafa, S. (2017). The Collective Intelligence concept : A literature review from the behavioral and cognitive perspective. pages 154–159.
- [Ingalls, 2008] Ingalls, R. G. (2008). Introduction to simulation. In *2008 Winter Simulation Conference*, pages 17–26. ISSN : 1558-4305.
- [Jaderberg et al., 2019] Jaderberg, M., Czarnecki, W. M., Dunning, I., Marris, L., Lever, G., Castañeda, A. G., Beattie, C., Rabinowitz, N. C., Morcos, A. S., Ruderman, A., Sonnerat, N., Green, T., Deason, L., Leibo, J. Z., Silver, D., Hassabis, D., Kavukcuoglu, K., et Graepel, T. (2019). Human-level performance in 3D multiplayer games with population-based reinforcement learning. *Science*, 364(6443) :859–865. Publisher : American Association for the Advancement of Science.
- [Kagermann et al., 2013] Kagermann, H., Wahlster, W., Helbig, J., et al. (2013). Recommendations for implementing the strategic initiative industrie 4.0. *Final report of the Industrie 4.0*, 4(0) :82.
- [Karaboga et Basturk, 2008] Karaboga, D. et Basturk, B. (2008). On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing*, 8(1) :687–697.
- [Karaduman et al., 2023] Karaduman, B., Tezel, B. T., et Challenger, M. (2023). Rational software agents with the BDI reasoning model for Cyber-Physical Systems. *Engineering Applications of Artificial Intelligence*, 123 :106478.
- [Klement et Silva, 2020] Klement, N. et Silva, C. (2020). A Generic Decision Support Tool to Planning and Assignment Problems : Industrial Applications and Industry 4.0. In *Scheduling in Industry 4.0 and Cloud Manufacturing*, pages 167–192. Springer International Publishing.
- [Klement et al., 2017] Klement, N., Silva, C., et Gibaru, O. (2017). A generic decision support tool to planning and assignment problems : Industrial application & industry 4.0. *Procedia Manufacturing*, 11 :1684–1691. 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy.
- [Lafou et al., 2016] Lafou, M., Mathieu, L., Pois, S., et Alochet, M. (2016). Manufacturing system flexibility : Sequence flexibility assessment. *Procedia CIRP*, 57 :229–234. Factories of the Future in the digital environment - Proceedings of the 49th CIRP Conference on Manufacturing Systems.
- [Le Mortellec et al., 2013] Le Mortellec, A., Clarhaut, J., Sallez, Y., Berger, T., et Trentesaux, D. (2013). Embedded holonic fault diagnosis of complex transportation systems. *Engineering Applications of Artificial Intelligence*, 26(1) :227–240.

- [Leitão et al., 2022] Leitão, P., Queiroz, J., et Sakurada, L. (2022). Collective Intelligence in Self-Organized Industrial Cyber-Physical Systems. *Electronics*, 11(19) :3213. Number : 19 Publisher : Multidisciplinary Digital Publishing Institute.
- [Leitão et Restivo, 2006] Leitão, P. et Restivo, F. (2006). Adacor : A holonic architecture for agile and adaptive manufacturing control. *Computers in Industry*, 57(2) :121–130.
- [Leitão et Rodrigues, 2011] Leitão, P. et Rodrigues, N. (2011). Multi-Agent System for On-demand Production Integrating Production and Quality Control. In Mařík, V., Vrba, P., et Leitão, P., editors, *Holonic and Multi-Agent Systems for Manufacturing*, Lecture Notes in Computer Science, pages 84–93, Berlin, Heidelberg. Springer.
- [Littman, 1994] Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In Cohen, W. W. et Hirsh, H., editors, *Machine Learning Proceedings 1994*, pages 157–163. Morgan Kaufmann, San Francisco (CA).
- [McArthur et al., 2007a] McArthur, S. D. J., Davidson, E. M., Catterson, V. M., Dimeas, A. L., Hatziargyriou, N. D., Ponci, F., et Funabashi, T. (2007a). Multi-Agent Systems for Power Engineering Applications Part I : Concepts, Approaches, and Technical Challenges. *IEEE Transactions on Power Systems*, 22(4) :1743–1752. Conference Name : IEEE Transactions on Power Systems.
- [McArthur et al., 2007b] McArthur, S. D. J., Davidson, E. M., Catterson, V. M., Dimeas, A. L., Hatziargyriou, N. D., Ponci, F., et Funabashi, T. (2007b). Multi-Agent Systems for Power Engineering Applications Part II : Technologies, Standards, and Tools for Building Multi-agent Systems. *IEEE Transactions on Power Systems*, 22(4) :1753–1759. Conference Name : IEEE Transactions on Power Systems.
- [Mitchell, 1997] Mitchell, T. M. (1997). *Machine learning*, volume 1. McGraw-hill New York.
- [Montoya Zapata et al., 2023] Montoya Zapata, S., Klement, N., Silva, C., Gibaru, O., et Lafou, M. (2023). Collective Intelligence Application in a Kitting Picking Zone of the Automotive Industry. In *Advances on Mechanics, Design Engineering and Manufacturing IV*, Lecture Notes in Mechanical Engineering, pages 410–420, Cham. Springer International Publishing.
- [Montoya Zapata et al., 2024a] Montoya Zapata, S., Klement, N., Silva, C., Gibaru, O., et Lafou, M. (2024a). Multi-agent system for perturbations in the kitting process of an automotive assembly line. *Submitted to the journal Engineering Applications of Artificial Intelligence (Second revision round)*.
- [Montoya Zapata et al., 2024b] Montoya Zapata, S., Klement, N., Silva, C., Gibaru, O., et Lafou, M. (2024b). Multi-agent system model for kitting process of an automotive assembly line. *Submitted to INCOM 2024*.
- [Montoya Zapata et al., 2024c] Montoya Zapata, S., Klement, N., Silva, C., Gibaru, O., et Lafou, M. (2024c). Simulation of a Kitting System for the Replenishment of an Automotive Assembly Line. In Borangiu, T., Trentesaux, D., Leitão, P., Berrah, L., et Jimenez, J.-F., editors, *Service*

- Oriented, Holonic and Multi-Agent Manufacturing Systems for Industry of the Future*, Studies in Computational Intelligence, pages 177–188, Cham. Springer Nature Switzerland.
- [Movahedi et Bastanfard, 2021] Movahedi, Z. et Bastanfard, A. (2021). Toward competitive multi-agents in Polo game based on reinforcement learning. *Multimedia Tools and Applications*, 80(17) :26773–26793.
- [Müller et Fischer, 2014] Müller, J. P. et Fischer, K. (2014). Application Impact of Multi-agent Systems and Technologies : A Survey. In Shehory, O. et Sturm, A., editors, *Agent-Oriented Software Engineering : Reflections on Architectures, Methodologies, Languages, and Frameworks*, pages 27–53. Springer, Berlin, Heidelberg.
- [Negahban et Smith, 2014] Negahban, A. et Smith, J. S. (2014). Simulation for manufacturing system design and operation : Literature review and analysis. *Journal of Manufacturing Systems*, 33(2) :241–261.
- [Nguyen et al., 2020] Nguyen, T. T., Nguyen, N. D., et Nahavandi, S. (2020). Deep Reinforcement Learning for Multiagent Systems : A Review of Challenges, Solutions, and Applications. *IEEE Transactions on Cybernetics*, 50(9) :3826–3839. Conference Name : IEEE Transactions on Cybernetics.
- [Papadopoulos et al., 2021] Papadopoulos, G., Antona, M., et Stephanidis, C. (2021). Towards Open and Expandable Cognitive AI Architectures for Large-Scale Multi-Agent Human-Robot Collaborative Learning. *IEEE Access*, 9 :73890–73909.
- [Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., et Duchesnay, E. (2011). Scikit-learn : Machine learning in Python. *Journal of Machine Learning Research*, 12 :2825–2830.
- [Piardi et al., 2022] Piardi, L., Costa, P., Oliveira, A., et Leitão, P. (2022). Collaborative Fault Detection and Diagnosis Architecture for Industrial Cyber-Physical Systems. In *2022 IEEE International Conference on Industrial Technology (ICIT)*, pages 1–6.
- [Piardi et al., 2023] Piardi, L., Costa, P., Oliveira, A., et Leitão, P. (2023). MAS-based Distributed Cyber-physical System in Smart Warehouse. *IFAC-PapersOnLine*, 56(2) :6376–6381.
- [Picard, 2022] Picard, G. (2022). Trajectory coordination based on distributed constraint optimization techniques in unmanned air traffic management. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS-22)*, pages 1065–1073. IFAAMAS.
- [Picard, 2023] Picard, G. (2023). Multi-agent consensus-based bundle allocation for multi-mode composite tasks. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS-23)*, pages 504–512. IFAAMAS.
- [Pinquié et al., 2022] Pinquié, R., Duigou, J. L., Grimal, L., et Roucoules, L. (2022). An open science platform for benchmarking engineering design researches. *Procedia CIRP*, 109 :472–477.

## BIBLIOGRAPHIE

---

- [Pooch et Wall, 1992] Pooch, U. W. et Wall, J. A. (1992). *Discrete Event Simulation : A Practical Approach*. CRC Press.
- [Prytz, 2014] Prytz, R. (2014). *Machine learning methods for vehicle predictive maintenance using off-board and on-board data*. PhD thesis.
- [Rao et Georgeff, 1991] Rao, A. S. et Georgeff, M. P. (1991). Modeling Rational Agents within a BDI-Architecture.
- [Robinson, 2004] Robinson, S. (2004). *Simulation : The Practice of Model Development and Use*. Wiley.
- [Rodrigues et al., 2018] Rodrigues, N., Oliveira, E., et Leitão, P. (2018). Decentralized and on-the-fly agent-based service reconfiguration in manufacturing systems. *Computers in Industry*, 101 :81–90.
- [Rodrigues et al., 2013] Rodrigues, N., Pereira, A., et Leitão, P. (2013). Adaptive Multi-Agent System for a Washing Machine Production Line. In Mařík, V., Lastra, J. L. M., et Skobelev, P., editors, *Industrial Applications of Holonic and Multi-Agent Systems*, Lecture Notes in Computer Science, pages 212–223, Berlin, Heidelberg. Springer.
- [Russell, 1999] Russell, R. (1999). Ant trails - an example for robots to follow? In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, volume 4, pages 2698–2703 vol.4. ISSN : 1050-4729.
- [Sanogo et al., 2023] Sanogo, K., Mekhalef Benhafssa, A., Sahnoun, M., Bettayeb, B., Abderrahim, M., et Bekrar, A. (2023). A multi-agent system simulation based approach for collision avoidance in integrated Job-Shop Scheduling Problem with transportation tasks. *Journal of Manufacturing Systems*, 68 :209–226.
- [Shannon, 1975] Shannon, R. (1975). *Systems Simulation : The Art and Science*. Prentice-Hall, Englewood Cliffs N.J.
- [Singh et Gupta, 2009] Singh, V. et Gupta, A. (2009). From artificial to collective intelligence : Perspectives and implications. pages 545–549.
- [Smith, 1980] Smith (1980). The Contract Net Protocol : High-Level Communication and Control in a Distributed Problem Solver. *IEEE Transactions on Computers*, C-29(12) :1104–1113. Conference Name : IEEE Transactions on Computers.
- [Stone, 2007] Stone, P. (2007). Multiagent learning is not the answer. It is the question. *Artificial Intelligence*, 171(7) :402–405.
- [Tchappi et al., 2022] Tchappi, I., Mualla, Y., Galland, S., Bottaro, A., Kamla, V. C., et Kamgang, J. C. (2022). Multilevel and holonic model for dynamic holarchy management : Application to large-scale road traffic. *Engineering Applications of Artificial Intelligence*, 109 :104622.
- [Tigli et Thomas, 1994] Tigli, J. et Thomas, M. (1994). Use of multi agent systems for mobile robotics control. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, volume 1, pages 588–592 vol.1.



- [Tonelli et al., 2016] Tonelli, F., Bruzzone, A. A. G., Paolucci, M., Carpanzano, E., Nicolò, G., Giret, A., Salido, M. A., et Trentesaux, D. (2016). Assessment of mathematical programming and agent-based modelling for off-line scheduling : Application to energy aware manufacturing. *CIRP Annals*, 65(1) :405–408.
- [Torbali et Alpan, 2023] Torbali, B. et Alpan, G. (2023). A multi-agent-based real-time truck scheduling model for cross-docking problems with single inbound and outbound doors. *Supply Chain Analytics*, 3 :100028.
- [Trentesaux et al., 2013] Trentesaux, D., Pach, C., Bekrar, A., Sallez, Y., Berger, T., Bonte, T., Leitão, P., et Barbosa, J. (2013). Benchmarking flexible job-shop scheduling and control systems. *Control Engineering Practice*, 21(9) :1204–1225.
- [Tuyls et Weiss, 2012] Tuyls, K. et Weiss, G. (2012). Multiagent learning : Basics, challenges, and prospects. *AI Magazine*, 33(3) :41–52.
- [Van Brussel et al., 1998] Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L., et Peeters, P. (1998). Reference architecture for holonic manufacturing systems : Prosa. *Computers in Industry*, 37(3) :255–274.
- [Verbraeken et al., 2020] Verbraeken, J., Wolting, M., Katzy, J., Kloppenburg, J., Verbelen, T., et Rellermeyer, J. S. (2020). A Survey on Distributed Machine Learning. *ACM Computing Surveys*, 53(2) :1–33.
- [Vogel-Heuser et al., 2021] Vogel-Heuser, B., Ocker, F., et Scheuer, T. (2021). An approach for leveraging Digital Twins in agent-based production systems. *at - Automatisierungstechnik*, 69(12) :1026–1039. Publisher : Oldenbourg Wissenschaftsverlag.
- [Weiss, 2013] Weiss, G. (2013). *Multiagent Systems*. MIT Press, second edition edition.
- [Wooldridge, 2009] Wooldridge, M. (2009). *An Introduction to MultiAgent Systems*. John Wiley & Sons.
- [Xie et Liu, 2017] Xie, J. et Liu, C.-C. (2017). Multi-agent systems and their applications. *Journal of International Council on Electrical Engineering*, 7(1) :188–197.
- [Zenisek et al., 2022] Zenisek, J., Gröning, H., Wild, N., Huskic, A., et Affenzeller, M. (2022). Machine Learning based Data Stream Merging in Additive Manufacturing. *Procedia Computer Science*, 200 :1422–1431.
- [Zhou et al., 2021] Zhou, T., Tang, D., Zhu, H., et Zhang, Z. (2021). Multi-agent reinforcement learning for online scheduling in smart factories. *Robotics and Computer-Integrated Manufacturing*, 72 :102202.
- [Zhu et al., 2021] Zhu, X., Luo, Y., Liu, A., Bhuiyan, M., et Zhang, S. (2021). Multiagent Deep Reinforcement Learning for Vehicular Computation Offloading in IoT. *IEEE Internet of Things Journal*, 8(12) :9763–9773.

# Annexes

## — Agrégation Flux rang 0 et flux rang 1

AGV	DAT_ENT_TAG_CHRG	DAT_DEP_TAG_CHRG	DAT_ENT_TAG_DCHRG	DAT_DEP_TAG_DCHRG
3243	2022-04-25 07:42:28 UTC	2022-04-25 08:58:09 UTC	2022-04-25 09:16:42 UTC	2022-04-25 09:27:43 UTC
3243	2022-04-25 10:07:52 UTC	2022-04-25 10:18:56 UTC	2022-04-25 10:50:29 UTC	2022-04-25 11:29:22 UTC
3243	2022-04-25 11:57:10 UTC	2022-04-25 12:11:09 UTC	2022-04-25 12:55:35 UTC	2022-04-25 13:27:31 UTC
3243	2022-04-25 13:53:50 UTC	2022-04-25 14:25:14 UTC	2022-04-25 14:29:16 UTC	2022-04-26 07:18:22 UTC
3243	2022-04-26 07:32:11 UTC	2022-04-26 08:20:19 UTC	2022-04-26 08:26:46 UTC	2022-04-26 08:52:09 UTC
3243	2022-04-26 09:08:04 UTC	2022-04-26 09:57:31 UTC	2022-04-26 10:01:32 UTC	2022-04-26 10:13:49 UTC
3243	2022-04-26 10:50:53 UTC	2022-04-26 12:00:39 UTC	2022-04-26 12:04:49 UTC	2022-04-26 12:10:02 UTC
3243	2022-04-26 12:22:18 UTC	2022-04-26 12:33:16 UTC	2022-04-26 12:49:45 UTC	2022-04-26 13:22:28 UTC
3244	2022-04-22 14:17:16 UTC	2022-04-25 07:39:55 UTC	<i>null</i>	<i>null</i>
3244	2022-04-25 07:39:55 UTC	2022-04-25 07:40:27 UTC	2022-04-25 07:45:37 UTC	2022-04-25 08:04:50 UTC
3244	2022-04-25 09:16:56 UTC	2022-04-25 09:31:58 UTC	2022-04-25 10:04:12 UTC	2022-04-25 10:31:21 UTC
3244	2022-04-25 10:54:07 UTC	2022-04-25 11:03:01 UTC	2022-04-25 11:59:17 UTC	2022-04-25 12:14:18 UTC
3244	2022-04-25 12:54:25 UTC	2022-04-25 13:52:16 UTC	2022-04-25 13:56:18 UTC	2022-04-25 14:02:01 UTC
3244	2022-04-25 14:26:51 UTC	2022-04-25 14:32:49 UTC	2022-04-26 07:33:08 UTC	2022-04-26 07:51:29 UTC
3244	2022-04-26 08:21:51 UTC	2022-04-26 08:56:36 UTC	2022-04-26 09:15:51 UTC	2022-04-26 09:31:43 UTC
3244	2022-04-26 09:59:02 UTC	2022-04-26 10:12:56 UTC	2022-04-26 10:50:48 UTC	2022-04-26 11:29:54 UTC
3244	2022-04-26 12:07:09 UTC	2022-04-26 12:20:44 UTC	2022-04-26 12:24:45 UTC	2022-04-26 12:34:12 UTC

FIGURE A.1 – Prétraitement des données AMH

ARTICLE	SUPPORT	AMH	DAT_CHRG_AMH	DAT_ARR_ZONE_TAMP
806428493R	52040000000146000	3243	2022-04-25 08:39:33 UTC	2022-04-25 09:16:42 UTC
826421924R	52040000000146000	3243	2022-04-25 08:39:33 UTC	2022-04-25 09:16:42 UTC
826426105R	52040000000146000	3243	2022-04-25 08:39:33 UTC	2022-04-25 09:16:42 UTC
826422248R	52040000000146000	3243	2022-04-25 08:39:33 UTC	2022-04-25 09:16:42 UTC
806425792R	52040000000146000	3243	2022-04-25 08:39:33 UTC	2022-04-25 09:16:42 UTC
802219946R	52040000000145504	3243	2022-04-25 08:42:07 UTC	2022-04-25 09:16:42 UTC
802170110R	52040000000145504	3243	2022-04-25 08:42:09 UTC	2022-04-25 09:16:42 UTC
802161221R	52040000000145504	3243	2022-04-25 08:42:10 UTC	2022-04-25 09:16:42 UTC
822172065R	73041014721046	3243	2022-04-25 08:42:12 UTC	2022-04-25 09:16:42 UTC
824304630R	77040383773892	3243	2022-04-25 08:42:13 UTC	2022-04-25 09:16:42 UTC
806066696R	52040000000145296	3243	2022-04-25 08:42:15 UTC	2022-04-25 09:16:42 UTC
806423401R	52040000000146496	3243	2022-04-25 08:46:30 UTC	2022-04-25 09:16:42 UTC
806428493R	52040000000146400	3243	2022-04-25 08:46:32 UTC	2022-04-25 09:16:42 UTC
826428933R	52040000000146400	3243	2022-04-25 08:46:34 UTC	2022-04-25 09:16:42 UTC
826428691R	52040000000146496	3243	2022-04-25 08:46:35 UTC	2022-04-25 09:16:42 UTC
601998806R	52040000000145200	3243	2022-04-25 08:46:36 UTC	2022-04-25 09:16:42 UTC
601982587R	52040000000145200	3243	2022-04-25 08:46:41 UTC	2022-04-25 09:16:42 UTC

FIGURE A.2 – Agrégation flux rang 0

## ANNEXES

ARTICLE	SUPPORT	DATE_DEP_ZONE_TAMP	DATE_MSK	EMP_MSK	DATE_DEP_MSK	DATE_MPI	EMP_PICK
806829981R	S05205000...	2023-05-09 17:21:19 UTC	2023-05-09 17:21:43 UTC	F001 1 120 10 C	2023-05-25 14:33:08 UTC	2023-05-25 14:34:34 UTC	D005 2 096 00 A
806829981R	S05205000...	2023-05-26 17:22:24 UTC	2023-05-26 17:23:02 UTC	F001 1 141 10 C	2023-06-07 16:29:41 UTC	2023-06-07 16:29:44 UTC	D005 2 096 00 A
806834756R	S05205000...	2023-05-26 17:25:24 UTC	2023-05-26 17:26:42 UTC	D005 2 129 10 B	2023-06-02 08:10:52 UTC	2023-06-02 08:10:53 UTC	D005 2 099 00 A
822200597R	S05315105...	2023-04-05 10:32:44 UTC	2023-04-05 10:36:11 UTC	D005 1 140 20 B	2023-04-07 16:58:03 UTC	2023-04-07 16:58:34 UTC	D005 2 102 00 A
822200597R	S05315105...	2023-04-11 13:03:36 UTC	2023-04-11 13:36:56 UTC	D005 1 143 20 C	2023-04-13 16:08:24 UTC	2023-04-13 16:08:25 UTC	D005 2 102 00 A
823304578R	S25445200...	2023-05-04 12:24:57 UTC	2023-05-04 12:27:20 UTC	F001 2 128 40 A	2023-05-10 18:32:24 UTC	2023-05-10 18:34:10 UTC	D005 2 102 00 A
823004499R	S05277705...	2023-04-05 08:47:28 UTC	2023-04-05 08:49:25 UTC	F001 2 139 00 C	2023-04-07 08:26:21 UTC	2023-04-07 08:26:28 UTC	D005 2 104 00 A
823004499R	S05277705...	2023-04-12 04:40:39 UTC	2023-04-12 04:43:36 UTC	F001 2 139 00 C	2023-04-12 13:43:34 UTC	2023-04-12 13:45:25 UTC	D005 2 104 00 A
803305696R	S25445200...	2023-05-04 12:14:52 UTC	2023-05-04 12:16:47 UTC	F001 2 124 10 A	2023-05-10 15:22:58 UTC	2023-05-10 15:24:06 UTC	D005 2 106 00 A
823001543R	S24164505...	2023-04-11 18:10:41 UTC	2023-04-11 18:13:47 UTC	F001 2 128 00 A	2023-05-11 18:19:22 UTC	2023-05-11 18:25:44 UTC	D005 2 108 00 A
823004131R	S24164505...	2023-04-14 17:41:23 UTC	2023-04-14 17:43:04 UTC	F001 2 128 00 C	2023-05-03 12:50:58 UTC	2023-05-03 12:52:46 UTC	D005 2 109 00 A
823004131R	S24164505...	2023-05-04 10:08:08 UTC	2023-05-04 10:13:34 UTC	F001 2 128 00 C	2023-05-11 20:47:37 UTC	2023-05-11 20:50:39 UTC	D005 2 109 00 A
807206850R	S11869205...	2023-04-14 16:27:44 UTC	2023-04-14 16:30:02 UTC	F001 2 151 30 C	2023-05-04 09:34:40 UTC	2023-05-04 09:34:42 UTC	D005 2 110 00 A
807206850R	S11869205...	2023-05-04 12:28:36 UTC	2023-05-04 12:33:41 UTC	F001 2 160 10 C	2023-05-09 16:07:18 UTC	2023-05-09 16:08:33 UTC	D005 2 110 00 A
807206850R	S11869205...	2023-05-09 19:23:57 UTC	2023-05-09 19:24:46 UTC	F001 2 160 10 B	2023-05-10 13:58:14 UTC	2023-05-10 13:58:17 UTC	D005 2 110 00 A
807206850R	S11869205...	2023-05-10 15:41:16 UTC	2023-05-10 15:42:15 UTC	F001 2 160 10 B	2023-05-10 18:00:43 UTC	2023-05-10 18:06:00 UTC	D005 2 110 00 A
807206850R	S11869205...	2023-04-14 06:50:35 UTC	2023-04-14 06:54:09 UTC	F001 2 142 60 B	2023-05-11 14:46:23 UTC	2023-05-11 14:47:11 UTC	D005 2 110 00 A

FIGURE A.3 – Agrégation flux rang 1

ARTICLE	SUPPORT	AMH	DAT_CHRG_AMH	EMP_MSK	EMP_PICK
806425792R	520400000...	3243	2022-04-25 08:39:33 UTC	null	F001 1 153 03 C
826426105R	520400000...	3243	2022-04-25 08:39:33 UTC	null	F001 1 153 02 H
826422248R	520400000...	3243	2022-04-25 08:39:33 UTC	null	F001 1 153 04 F
826421924R	520400000...	3243	2022-04-25 08:39:33 UTC	F001 1 153 10 B	F001 1 153 04 G
806428493R	520400000...	3243	2022-04-25 08:39:33 UTC	null	F001 1 150 04 H
802219946R	520400000...	3243	2022-04-25 08:42:07 UTC	F001 2 128 10 A	D005 2 137 03 B
802170110R	520400000...	3243	2022-04-25 08:42:09 UTC	C004 1 147 10 A	D005 2 137 03 A
802161221R	520400000...	3243	2022-04-25 08:42:10 UTC	null	F001 1 132 03 F
822172065R	730410147...	3243	2022-04-25 08:42:12 UTC	D005 1 129 00 A	null
824304630R	770403837...	3243	2022-04-25 08:42:13 UTC	F001 1 138 10 A	D005 2 141 02 C
806066696R	520400000...	3243	2022-04-25 08:42:15 UTC	D005 1 135 00 B	null
806423401R	520400000...	3243	2022-04-25 08:46:30 UTC	null	F001 1 153 04 A
806428493R	520400000...	3243	2022-04-25 08:46:32 UTC	null	F001 1 150 04 H
826428933R	520400000...	3243	2022-04-25 08:46:34 UTC	F001 1 150 10 A	null
826428691R	520400000...	3243	2022-04-25 08:46:35 UTC	null	F001 1 153 03 H
601998806R	520400000...	3243	2022-04-25 08:46:36 UTC	F001 1 117 30 C	null
601982587R	520400000...	3243	2022-04-25 08:46:41 UTC	F001 1 117 10 C	null

FIGURE A.4 – Agrégation flux rang 0 et flux rang 1

### — Composition et passage des véhicule dans la zone du montage des portes

PLANT_CODE	PJI	PPUM_FAMILY_CODE	PART_NUMBER	UC	COMPONENT_COEF	ROUTE_DATE
039	1430181	1432	769219758R	UM	1	2023-04-05 04:37:50 UTC
039	1430181	1496	822176862R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	1422	802192435R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	2232	828317825R	UM	1	2023-04-05 04:37:50 UTC
039	1430181	2273	828475871R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	1421	802189736R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	2234	828306261R	UM	1	2023-04-05 04:37:50 UTC
039	1430181	2274	828467412R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	2276	805620902R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	2275	805622909R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	2266	806829981R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	2267	806063997R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	1495	822161345R	UC	1	2023-04-05 04:37:50 UTC
039	1430181	1441	769227075R	UM	1	2023-04-05 04:37:50 UTC

FIGURE A.5 – Agrégation du composition et passage des véhicules

— Extraction des états d’alerte à partir des appels DWS

id	pji	date_create	type	id_workstation
5047032	0510671	2023-01-31 09:11:46.084	MEANS	1762
6335934	2520008	2023-06-20 06:35:15.514	MEANS	57059
5634161	1330007	2023-04-03 09:58:00.057	MEANS	57063
5671279	1410291	2023-04-06 09:14:23.742	MEANS	1777
5758349	1440348	2023-04-17 11:30:05.627	MEANS	1014
5285265	0930030	2023-02-23 13:37:47.715	MEANS	32511
6100481	2020009	2023-05-25 10:15:05.403	MEANS	32511
6176208	2330016	2023-06-02 06:13:26.023	MEANS	32511
6084230	1910019	2023-05-24 07:31:28.008	LOGISTICS	32513
5357675	0830060	2023-03-02 21:43:08.048	LOGISTICS	8962
5357677	0830060	2023-03-02 21:43:16.424	LOGISTICS	8962
5357679	0830060	2023-03-02 21:43:29.413	LOGISTICS	8962
5357683	0920108	2023-03-02 21:44:07.331	LOGISTICS	12034

FIGURE A.6 – Agrégation des appels d’alerte avec les PJI

— Liaison PJI-Support

Liste détaillée

Code site	Code activité	Code article	Numéro du support	Quantité du mouvement de GEI en VL de base	Code dépôt physique destination transfert	Code dépôt physique
UGB	001	285487381R	024062051019635617	476,00		041
UGB	001	285487381R	024062051019635617	476,00		041
UGB	001	285487381R	024062051019635617	476,00		041
UGB	001	285487381R	024062050019644333	595,00		041
UGB	001	285487381R	024062050019644333	476,00		041
UGB	001	285487381R	024062051019637778	238,00		041
UGB	001	285487381R	05205000000373685	5950,00		041
UGB	001	285487381R	024062051019637778	357,00		041
UGB	001	285487381R	024062051019637778	357,00		041
UGB	001	285487381R	024062051019637732	476,00		041
UGB	001	285487381R	024062051019637742	357,00		041
UGB	001	285487381R	024062051019637742	357,00		041
UGB	001	285487381R	024062051019637802	357,00		041
UGB	001	285487381R	024062051019637802	476,00		041
UGB	001	285487381R	024062051019638558	476,00		041
UGB	001	285487381R	024062051019638528	476,00		041
UGB	001	285487381R	024062051019638528	714,00		041

FIGURE A.7 – Mouvements de GEI pour un seul article : on constate que les supports physiques changent à chaque fois

## ANNEXES

---

PJI	PPUM_FAMILY_CODE	PART_NUMBER	SUPPORT	EMP_PICK	UC	COMPONENT_COEF	ROUTE_DATE
1710096	1421	802189736R	S000573051780903765	F001 1 138 01 B	UC	1	2023-04-06T04:46:32
1710096	1496	822176862R	S000573051780903687	D005 2 135 01 A	UC	1	2023-04-06T04:46:32
1710096	2274	828467412R	S002394050357680375	D005 2 130 02 B	UC	1	2023-04-06T04:46:32
1710096	2232	828317825R	S286813050220450344	F001 1 058 00 A	UM	1	2023-04-06T04:46:32
1710096	2266	806829981R	S002394050358057591	F001 1 129 03 F	UC	1	2023-04-06T04:46:32
1710096	1441	769227075R	S286813050220450837	F001 1 077 00 A	UM	1	2023-04-06T04:46:32
1710096	2271	803274890R	S254452050233088958	F001 1 129 03 H	UC	2	2023-04-06T04:46:32
1710096	2271	803274890R	S254452050233088958	F001 1 129 03 H	UC	2	2023-04-06T04:46:32
1710096	2273	828475871R	S002394050357652996	F001 1 123 02 B	UC	1	2023-04-06T04:46:32
1710096	2265	806834756R	S002394050358057461	D005 2 135 03 A	UC	1	2023-04-06T04:46:32
1710096	1495	822161345R	S000573051781273013	F001 1 129 01 G	UC	1	2023-04-06T04:46:32
1710096	1432	769219758R	S286813050220450971	<i>null</i>	UM	1	2023-04-06T04:46:32
1710096	2267	806064184R	S029289050814386510	D005 2 147 03 B	UC	1	2023-04-06T04:46:32
1710096	2268	806073626R	S029289050814807577	F001 1 147 03 B	UC	1	2023-04-06T04:46:32

FIGURE A.8 – Agrégation des compositions avec les flux rang 0 et rang 1



**Résumé :** Les environnements de production et les processus de fabrication deviennent de plus en plus complexes, dynamiques et incertains. Ils font face à des changements constants, avec une tendance actuelle marquée vers la personnalisation de masse et une variété croissante de produits. Dans le contexte du secteur automobile, les processus de fabrication se caractérisent par une grande diversité de produits, un volume de production élevé et fluctuant, ainsi que de nombreux composants de véhicules. Ces caractéristiques rendent la logistique interne extrêmement complexe et sujette à des perturbations. Les systèmes de contrôle traditionnels pour les processus de fabrication évoluent vers des systèmes plus modulaires, intelligents et distribués. La littérature existante indique que les systèmes multi-agents (MAS) sont une approche adaptée pour mettre en œuvre ces nouveaux systèmes de contrôle. Cependant, le nombre d'implémentations réelles rapportées dans le domaine industriel est limité. Dans ces travaux de thèse, nous avons présenté une méthodologie basée sur les systèmes multi-agents pour optimiser la performance globale d'un processus de production, en gérant les interactions des différents agents impliqués. Ces travaux sont menés principalement dans les processus d'approvisionnement de pièces à une ligne d'assemblage automobile utilisant le processus de kitting. Nous proposons une méthodologie multi-agents pour détecter et répondre aux alertes de pièce manquante dans un kit, ayant lieu lors de ce processus. Dans le but de mettre en œuvre la méthodologie proposée sur le terrain industriel, les opérateurs jouent un rôle important dans la phase de prise de décision. Nous adaptons et appliquons également cette méthodologie dans le processus de tôlerie. Nous avons testé la méthodologie proposée dans des scénarios réels, reconstitués avec des données historiques provenant d'une usine du Groupe Renault. Les résultats obtenus ont montré la pertinence de la méthodologie pour répondre aux problématiques industrielles étudiées et permettent d'envisager un déploiement sur le terrain industriel.

**Mots clés :** Systèmes multi-agents, Intelligence artificielle, Logistique interne, Humain dans la boucle, Industrie 4.0, Industrie automobile.

**Abstract :** Production environments and manufacturing processes are becoming increasingly complex, dynamic, and uncertain. They face constant changes, with a current trend towards mass customization and a growing variety of products. In the context of the automotive sector, manufacturing processes are characterized by a wide variety of products, high and fluctuating production volume, and numerous vehicle components. These characteristics make internal logistics extremely complex and prone to disruptions. Traditional control systems for manufacturing processes are evolving towards more modular, intelligent, and distributed systems. Existing literature indicates that multi-agent systems (MAS) are a suitable approach to implement these new control systems. However, the number of reported real-world implementations in the industrial domain is limited. In this thesis, we have presented a methodology based on multi-agent systems to optimize the overall performance of a production process by managing the interactions of different involved agents. These works are mainly conducted in the parts supply processes to an automotive assembly line using the kitting process. We propose a multi-agent methodology to detect and respond to alerts of missing parts in a kit occurring during this process. Operators play an important role in the decision-making phase to implement the proposed methodology in the industrial field. We also adapt and apply this methodology in the welding workshops. We tested the proposed methodology in real scenarios, reconstructed with historical data from a Renault Group factory. The results obtained have shown the relevance of the methodology to address the studied industrial issues and allow us to consider deployment in the industrial field.

**Keywords :** Multi-agent systems, Artificial intelligence, Internal logistics, Human in the loop, Industry 4.0, Automotive industry.