HAL
open science

Proposition d'un modèle générique de pilotage pour un système à flux guidés

Silviu Raileanu

To cite this version:

Silviu Raileanu. Proposition d'un modèle générique de pilotage pour un système à flux guidés : Application des concepts holoniques au transport intelligent (FMS/PRT). Sciences de l'ingénieur [physics]. Université de Valenciennes et du Hainaut-Cambrésis, 2011. Français. NNT : 2011VALE0011 . tel-03012597

HAL Id: tel-03012597

https://uphf.hal.science/tel-03012597

Submitted on 18 Nov 2020

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.

2011 VALE 0011

N° ORDRE : 11/11

Thèse

Présentée à l'Université Politehnica de Bucarest

En vue de l'obtention du titre de :

Docteur de l'Université de l'Université Politehnica de Bucarest, Roumanie

Docteur de l'Université de Valenciennes et du Hainaut Cambrésis, France

Par

Silviu RAILEANU

Ingénieur de l'Université Politehnica, Bucarest, Roumanie

Proposition d'un modèle générique de pilotage pour un système à flux guidés /Application des concepts holoniques au transport intelligent (FMS/PRT)

Soutenue publiquement le 29 juin 2011 devant le jury composé de :

Dumitru POPESCU	Président	Professeur, Université Politehnica, Bucarest, Roumanie
Radu DOBRESCU	Rapporteur	Professeur, Université Politehnica, Bucarest, Roumanie
Pierre CASTAGNA	Rapporteur	Professeur, Université de Nantes, France
Florin FILIP	Examinateur	Académicien, Université Politehnica, Bucarest, Roumanie
Theodor BORANGIU	Directeur de Thèse	Professeur, Université Politehnica, Bucarest, Roumanie
Khalid KOUISS	Examinateur	Maître de Conférences, IFMA Clermont-Ferrand, France
Christian TAHON	Directeur de Thèse	Professeur, Université de Valenciennes et du Hainaut-Cambrésis, France
Thierry BERGER	Co-encadrant	Maître de Conférences, Université de Valenciennes et du Hainaut-Cambrésis, France

N° ORDRE : 11/11

Thèse

Présentée à l'Université Politehnica de Bucarest

En vue de l'obtention du titre de :

Docteur de l'Université de l'Université Politehnica de Bucarest, Roumanie

Docteur de l'Université de Valenciennes et du Hainaut Cambrésis, France

Par

Silviu RAILEANU

Ingénieur de l'Université Politehnica, Bucarest, Roumanie

Proposition d'un modèle générique de pilotage pour un système à flux guidés /Application des concepts holoniques au transport intelligent (FMS/PRT)

Soutenue publiquement le 29 juin 2011 devant le jury composé de :

Dumitru POPESCU	Président	Professeur, Université Politehnica, Bucarest, Roumanie
Radu DOBRESCU	Rapporteur	Professeur, Université Politehnica, Bucarest, Roumanie
Pierre CASTAGNA	Rapporteur	Professeur, Université de Nantes, France
Florin FILIP	Examinateur	Académicien, Université Politehnica, Bucarest, Roumanie
Theodor BORANGIU	Directeur de Thèse	Professeur, Université Politehnica, Bucarest, Roumanie
Khalid KOUISS	Examinateur	Maître de Conférences, IFMA Clermont-Ferrand, France
Christian TAHON	Directeur de Thèse	Professeur, Université de Valenciennes et du Hainaut-Cambrésis, France
Thierry BERGER	Co-encadrant	Maître de Conférences, Université de Valenciennes et du Hainaut-Cambrésis, France

Sommaire

INTRODUCTION GENERALE 7

CHAPITRE 1 : PILOTAGE DES SYSTEMES A FLUX GUIDES (SFG) 9

1 SYSTEMES A FLUX GUIDE (SFG) 9

1.1 DESCRIPTION D'UN SYSTEME A FLUX GUIDE 9
1.2 SYSTEME DE FABRICATION FLEXIBLE (SFF) 10
1.2.1 LES SYSTEMES DE FABRICATION FLEXIBLES 10
1.2.2 CARACTERISTIQUES D'UN SFF 11
1.2.3 LA STRUCTURE DE TRANSPORT / ROUTAGE D'UN SFF 12
1.2.4 LA STRUCTURE DE FABRICATION D'UN SFF 13
1.3 SYSTEME DE TRANSPORT A LA DEMANDE (PRT) 13
1.3.1 LES SYSTEMES DE TRANSPORT A LA DEMANDE 13
1.3.3 LA STRUCTURE DE TRANSPORT 13
1.3.4 LES VEHICULES 14
1.4 PILOTAGE DES SFG 15

2 ARCHITECTURES DE PILOTAGE 16

2.1 PILOTAGE CENTRALISE 16
2.2 SYSTEMES HIERARCHIQUES 17
2.3 SYSTEMES HETERARCHIQUES 18
2.4 SYSTEMES HYBRIDES 19
2.4.1 OPEN-CONTROL (PILOTAGE OUVERT) 19
2.4.2 PILOTAGE HYBRIDE (SEMI HETERARCHIQUE) 20
2.5 MODELES DE PILOTAGE 21
2.5.1 SYSTEMES MULTI AGENT 21
2.5.2 SYSTEMES HOLONIQUES 23
2.6 SYNTHESE DES PARADIGMES DE PILOTAGE 24

3 MODELES HOLONIQUES DE PILOTAGE 25

3.1 PROSA 25
3.2 ADACOR 27
3.3 HCBA 29
3.4 SYNTHESE 29

4 CONCLUSION 30

CHAPITRE 2 : PROPOSITION D'UN MODELE DE PILOTAGE HOLONIQUE 31

D'UN SYSTEME A ENTITES ACTIVES (SEA) 31

1 LA NOTION D'ENTITE 31

1.1 ENTITES D'UN SYSTEME A FLUX GUIDE 31
1.2 CARACTERISATION D'UNE ENTITE 32
1.2.1 NATURE D'UNE ENTITE 32
1.2.2 MOBILITE D'UNE ENTITE 32
1.2.3 SERVICES ASSOCIES AUX ENTITES 32

2 MODELE ORGANISATIONNEL D'UN SYSTEME A ENTITES ACTIVES : SEA 33

2.1 SYSTEME PILOTE ET SYSTEME DE PILOTAGE D'UN SEA 33
2.2 ENTITE ACTIVE GENERALISEE (EAG) 34
2.3 ENTITE ACTIVE GENERALISEE COMPOSEE (EAG_C) 34
2.4 CARACTERISATION D'UNE ENTITE ACTIVE GENERALISEE 34
2.5 LES FLUX AU SEIN D'UN SEA 36

3 LE MODELE HOLONIQUE D'UN SEA (SEAH) 37

3.1 HOLONS ACTIF, NON ACTIF ET GENERALISE 37
3.1.1 LE HOLON ACTIF HA : 38
3.1.2 LE HOLON NON ACTIF HNA 38
3.2 LE HOLON ACTIF GENERALISE HAG : 39
3.3 MODELE DU HOLON ACTIF GENERALISE 39
3.5 HOLONS NON-COMPLETS 45
3.5.1 LE HOLON "HA RESTREINT" HA_R 45
3.5.2 LE HOLON OBJET RECEVEUR DES SERVICES, HNA_{ORS} : 46

4 MODELE STRUCTUREL DES HOLONS DU SEAH 46

4.1 MODELE GENERAL 46
4.2 STRUCTURE STATIQUE 48
4.2.1 HOLON COORDINATEUR (HC) 48
4.2.2 HOLON RESSOURCE (HR / HRA) 48
4.2.3 HOLON CONNAISSANCES SYSTEME (HCS) 48
4.3 STRUCTURE DYNAMIQUE 49
4.3.3 LE HOLON MOBILITE (HM) 49
4.3.4 HOLON CONDUITE (HCO) 50
4.4 LE HOLON FLUX (HF) 50

4.5 CYCLE DE VIE DU HOLON FLUX HF 50
4.5.1 EXEMPLE 1 : CAS D'UN FMS 51
4.5.2 EXEMPLE 2 : CAS DU PRT 53
4.6 SYNTHESE 53

5 LE MODELE DE COMPORTEMENT DU SEAH 56

5.1 MODELE DE COMPORTEMENT INTERNE D'UN HF 56
5.1.1 ARTICULATION ENTRE LES PROCESSUS D'ALLOCATION ET DE ROUTAGE 56
5.1.2 PROCESSUS D'ALLOCATION 57
5.1.3 PROCESSUS DE ROUTAGE 57
5.1.4 OBTENTION DU SERVICE 57
5.2 MODELE DE COMPORTEMENT ENTRE HOLONS HF ET HRA 58
5.2.1 MODÈLE GÉNÉRAL DE COMPORTEMENT 58
5.2.2 PILOTAGE EXPLICITE 59
5.2.3 PILOTAGE IMPLICITE 59
5.2.4 PILOTAGE IMPLICITE VIA MO SOCIETAL 60
5.2.5 PILOTAGE IMPLICITE VIA MO ENVIRONNEMENTAL 60
5.2.6 DIFFERENCE ENTRE PILOTAGE IMPLICITE ET PILOTAGE EXPLICITE 61

6 CONCLUSION 61

CHAPITRE 3: UTILISATION DU MODELE HOLONIQUE SEAH POUR LES SYSTEMES DE PRODUCTION DE TYPE "BATCH" 62

1 LE CONTEXTE 62

2 LA CELLULE FLEXIBLE CIMR 63

3 ARCHITECTURE DU SYSTEME DE PILOTAGE 69

4.1 HOLON ACTIF GENERALISE 69
4.2 ARCHITECTURE 71
4.3 STRUCTURE 74
4.4 MODELE DYNAMIQUE 90
4.5 LA PLANIFICATION ET L'ORDONNANCEMENT DES ORDRES DE PRODUCTION 92
4.6 L'ASSOCIATION DU PLAN DE PRODUCTION AUX HOLONS FLUX DANS UN FORMAT STANDARD POUR L'AUTOMATE PROGRAMMABLE 96
4.7 LE PROCESSUS DE REORDONNANCEMENT DES ORDRES DE PRODUCTION 99

4 CONCLUSION 107

CHAPITRE 4 : EXPERIMENTATION DU MODELE DE PILOTAGE HOLONIQUE CAS D'UN SYSTEME DE TYPE FMS 108

1 LA PLATEFORME DE FABRICATION CIMR 108

1.1 RESSOURCES, TYPES D'OPERATIONS ET TYPES DE PIECES 108

2 LE PROTOCOLE 130

3 RESULTATS 131

4 ANALYSE ET SYNTHESE 143

5 CONCLUSION 144

CHAPITRE 5 - PROPOSITION D'UN MODELE DE PILOTAGE HOLONIQUE POUR UN SYSTEME DE TRANSPORT DE TYPE PRT 146

1 MODELE STRUCTUREL HOLONIQUE D'UN SYSTEME DE TRANSPORT PRT 146

1.1 STRUCTURE STATIQUE D'UN SYSTEME DE TRANSPORT PRT 146
1.1.1 HOLON COORDINATEUR HC 146
1.1.2 HOLON RESSOURCE HR 147
1.1.3 HOLON CONNAISSANCE SYSTEME HCS 147
1.2 STRUCTURE DYNAMIQUE D'UN SYSTEME DE TRANSPORT PRT 148
1.2.1 LE HOLON FLUX HF 148
1.2.2 HOLON OBJET RECEVEUR DES SERVICES HORS 148
1.2.3 HOLON SERVICES A OBTENIR 148
1.2.4 HOLON MOBILITE 148
1.2.5 HOLON CONDUITE HCO 148
1.3 EXEMPLE DE MODELISATION HOLONIQUE D'UN SYSTEME PRT 149
1.3.1 LE SYSTEME PRT URBANAUT 149
1.4 MODELE HOLONIQUE DU SYSTEME URBANAUT. 152

2 MODELISATION DU COMPORTEMENT D'UN SYSTEME PRT 153

2.1 ARCHITECTURE DE PILOTAGE D'UNE FLOTTE DE PRT 153
2.2 PILOTAGE IMPLICITE VIA UN MO ENVIRONNEMENTAL 154
2.2.1 CONCEPT DE STIGMERGIE 155
2.2.2 MO ENVIRONNEMENTAL BASE SUR LA STIGMERGIE 155
2.2.3 FONCTIONNEMENT DU PILOTAGE IMPLICITE 157
2.2.4 ETUDE CRITIQUE DE LA STIGMERGIE POUR LE PILOTAGE D'UNE FLOTTE DE PRT 157

2.3 PILOTAGE IMPLICITE VIA UN MO SOCIETAL 159
2.3.1 ALGORITHME DE DIJKSTRA EMBARQUE POUR UN MO SOCIETAL 159
2.3.2 FONCTIONNEMENT DU PILOTAGE IMPLICITE 160

3 LES FLUX AU SEIN DU SYSTEME PRT. 161

4 CONCLUSION 163

CHAPITRE 6 – EXPERIMENTATION D'UN SYSTEME DE PILOTAGE HOLONIQUE D'UN SYSTEME DE TRANSPORT DE TYPE PRT 164

1 LE SITE EXPERIMENTAL 164

2 LE SYSTEME DE TRANSPORT 165

2.1 LES SHUTTLES 165
2.2 LE RESEAU DE TRANSPORT 165
2.3 SYSTEME DE COMMUNICATION ET DE TRAITEMENT DES DONNEES 166

3 MODELISATION HOLONIQUE DU SYSTEME PRT EXPERIMENTAL 169

3.1 MODELE STRUCTUREL DU SYSTEME PRT EXPERIMENTAL 169
3.1.1 PARTIE STATIQUE 169
3.1.2 PARTIE DYNAMIQUE 170
3.2 MODELE COMPORTEMENTAL DU SYSTEME PRT EXPERIMENTAL 171

4 SCENARIOS ET RESULTATS EXPERIMENTAUX 172

4.1 SCENARIO 1 : EVOLUTIVITE/"SCALABILITE" 172
4.2 SCENARIO 2 : GESTION DES PERTURBATIONS DE ROUTAGE PAR LE PILOTAGE DE NIVEAU INFERIEUR 173
4.3 SCENARIO 3: PILOTAGE IMPLICITE 175

5 CONCLUSION 176

CONCLUSION GENERALE 177

REFERENCES BIBLIOGRAPHIQUES 181

ANNEXE 1 : COMPOSANTS D'UN SYSTEME DE TRANSPORT PRT 190

Proposition d'un modèle générique de pilotage pour un système à flux guidés /Application des concepts holoniques au transport intelligent (FMS/PRT)

Introduction générale

Selon (Neubert, 1997), le pilotage d'un système de production, de biens ou de services, doit assurer l'organisation des ressources disponibles, dans le temps et dans l'espace, afin d'atteindre les objectifs assignés. L'autonomie dans la prise de décision au niveau opérationnel, la coopération entre les entités, entre elles et avec les composantes du système, l'agilité, la modularité, la tolérance aux perturbations apparaissent alors comme les qualités indispensables d'un système de pilotage.

Dans le contexte actuel, les responsables de systèmes de production de biens ou de services visent davantage l'adaptabilité de leurs systèmes de production aux changements rapides de l'environnement qu'un comportement optimisé (ex. ordonnancement optimal), (Sauer, 2008). En effet, la recherche de l'optimalité implique le plus souvent des hypothèses réductrices rendant le système de production très sensible aux changements de son environnement.

Aussi, de nouvelles architectures de pilotage de la production (Van Brussel, 1998 ; Leitao, 2004) et du transport (Fischer, 2005) sont apparues, fondées sur l'accroissement de l'autonomie de certaines entités du système de production (ressources, produits,...). Ces "entités actives", capables d'activités décisionnelles pour le pilotage des flux, présentent des capacités d'adaptation et de réactivité qui permettent de mettre en œuvre un système de pilotage réagissant à l'état réel du système de production pour maintenir le niveau de performance souhaité.

Par ailleurs, les technologies sans contact (réseaux sans fil, RFID, "smart cards", communication infrarouge...) permettent à l'heure actuelle d'envisager la mise en œuvre de ce type d'entités à des coûts acceptables.

Dans cette logique, les travaux présentés dans cette thèse proposent un modèle de pilotage générique pour des systèmes à flux guidés (SFG), s'appuyant sur les concepts holoniques et d'auto-adaptation. La thèse est structurée en six chapitres et une conclusion générale.

Organisation de la thèse

Dans le premier chapitre, une description des systèmes à flux guidés est d'abord proposée en s'appuyant sur deux types de système : système de production flexible (FMS, Groover, 1987; Leitão, 2004) et système de transport de type PRT (Personal Rapid Transport (PRT, Carnegie et al., 2007)). La problématique du pilotage d'un SFG (allocation et routage) est ensuite abordée.

Un l'état de l'art des différentes solutions "classiques" de pilotage (centralisé, hiérarchique, hétérarchique et semi hétérarchique), ainsi que les modèles sur lesquels elles reposent (systèmes multi agent, systèmes holoniques de fabrication, concept "*open control*") sont rappelées. La justification du choix d'une architecture de pilotage non centralisée, ouverte, inspirée du paradigme holonique apportant une plus grande autonomie par l'introduction d'entités de type holons actifs fait l'objet de la dernière partie.

Le deuxième chapitre précise tout d'abord le concept de système à entités actives (SEA) et décrit le modèle holonique qui lui est associé. Le modèle générique de pilotage d'un système caractérisé par des flux d'entités actives c'est-à-dire intervenant dans les décisions de pilotage et "non actives" (sans capacité décisionnelle de pilotage) est ensuite présenté. Le modèle structurel qui définit les entités composantes du système, leurs caractéristiques, leurs interactions internes et externes, et le modèle comportemental, s'appuyant sur l'"open-control", sont ensuite décrits. Le modèle générique de pilotage développé est fondé sur le dialogue entre les entités actives, et le routage dynamique qui utilise des informations captées en ligne par les entités actives.

Le troisième chapitre présente l'adaptation du modèle générique précédent aux systèmes flexibles de fabrication discrète (et parfois répétitive) de produits possédant un système de transport automatisé (en boucle fermée) reliant les postes de travail. La cellule flexible CIMR de l'Université Politehnica de Bucarest est décrite, puis le modèle holonique est détaillé en précisant la configuration du modèle d'accès aux services offerts par les ressources de la cellule, le suivi de la production avec mise à jour des états des ressources et les interventions mises en œuvre en cas de changement d'état des ressources.

Le quatrième chapitre décrit l'infrastructure du système de fabrication du Centre de recherche CIMR où le modèle du chapitre deux a été implémenté avec les adaptations apportées dans le chapitre trois. Les tests quantitatifs et qualitatifs réalisés sont présentés en comparant différents scénarios de planification et d'ordonnancement. Ce chapitre se termine par une analyse du comportement en situation de perturbation (ex. panne d'une ressource) qui justifie une architecture de pilotage décentralisée exploitant les capacités décisionnelles des entités actives.

Le cinquième chapitre expose l'adaptation du modèle générique pour le pilotage d'un système de transport de type PRT. Les composants essentielles (véhicules, stations, infrastructures) et les principales caractéristiques d'un tel système sont tout d'abord présentées. On précise ensuite le modèle holonique structurel (parties statique et dynamique) d'un système PRT qui est illustré par un exemple. Le modèle comportemental repose sur l'« open-control ». Chaque PRT comprend une carte de routage contenant le temps de parcours, réactualisé en temps-réel, et utilise l'algorithme de Dijkstra embarqué.

Le sixième chapitre présente l'expérimentation du modèle de pilotage décrit dans le chapitre 5. On donne tout d'abord une description de l'infrastructure de la cellule de l'atelier du Pôle AIP-PRIMECA de l'UVHC qui a été utilisé pour l'expérimentation puis le modèle holonique correspondant. L'expérimentation est ensuite proposée. Trois scénarios ont été utilisés pour évaluer le modèle. Les résultats obtenus sont analysés et montrent la capacité d'autonomie et d'auto-adaptation aux perturbations du système de pilotage proposé.

Chapitre 1 : Pilotage des Systèmes à Flux Guidés (SFG)

Introduction

Dans ce premier chapitre, une description des systèmes à flux guidés est d'abord proposée en considérant deux types de systèmes représentatifs : les systèmes de fabrication flexible (SFF ou FMS : Flexible Manufacturing System) et les systèmes de transport à la demande de type PRT (Personal Rapid Transport).

Dans la partie suivante, la problématique du pilotage d'un SFG qui comporte deux fonctions essentielles, l'allocation des ressources et le routage des éléments mobiles du système, est abordée. Un état de l'art des différentes solutions de pilotage (centralisé, hiérarchique, hétérarchique et semi hétérarchique) et des modèles (multiagent, holonique) sur lesquels elles s'appuient sont rappelées.

La dernière partie permet de justifier le choix, pour des SFG, d'une architecture de pilotage non centralisée, ouverte, inspirée du paradigme holonique.

1 Systèmes à flux guidé (SFG)

1.1 Description d'un système à flux guidé

Les SFG forment une classe de systèmes où certains éléments du système (produits, personnes) se déplacent au sein du système et reçoivent des services (usinage, assemblage, routage, embarquement/débarquement...). Ils utilisent pour ces déplacements des moyens de transport spécifiques (dans le cas des FMS : shuttle/navettes, AGV (Automated Guided Vehicles)... dans le cas des systèmes de type PRT : rames, bus-tramways...). Ces éléments et les moyens de transport constituent les entités mobiles du système. Les déplacements sont effectués au sein d'une infrastructure formant un réseau maillé comportant des voies de transport réservées ou non (convoyeurs, rails, systèmes de filoguidage...) et des points de réception de services (machines-outils, robots, stations...) où les éléments reçoivent les services.

Une représentation synthétique d'un SFG est illustrée dans la figure1.1, où le réseau (infrastructure du système) est représenté par un graphe orienté sur lequel les entités mobiles se déplacent. Les arcs représentent les liaisons entre les points de réception de services. Les entités mobiles se déplaçant d'un nœud à l'autre en suivant les liaisons directes (mono et/ou bidirectionnelles) soumises à des contraintes créent un flux guidé d'entités.

Les systèmes de fabrication flexible (SFF) et les systèmes de transport de type PRT sont deux exemples significatifs de tels systèmes. Nous en donnons, pour préciser le concept de système à flux guidé, une brève description dans les paragraphes suivants.

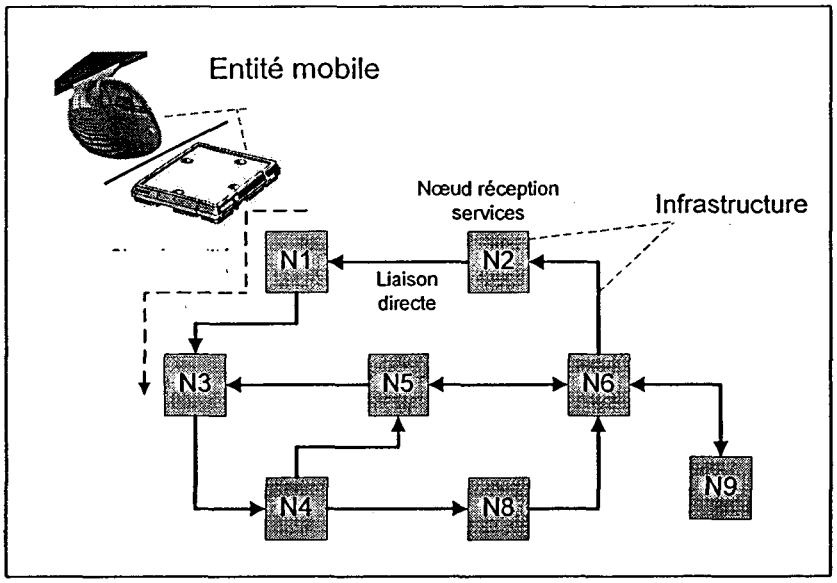


Figure 1.1 : principe de fonctionnement d'un système à flux guidé

1.2 Système de fabrication flexible (SFF)

1.2.1 Les systèmes de fabrication flexibles

La variété de plus en plus grande de produits, la fréquence des améliorations apportées, la dynamique de plus en plus élevée de la demande du marché, la demande d'une meilleure qualité, ainsi que la réduction de la taille des lots, des temps de production et de lancement assurant un niveau de stock réduit ont conduit à la nécessité de pouvoir fabriquer avec une plus grande flexibilité et efficacité. Pour répondre à ces défis, les systèmes de fabrication ont fortement évolué en termes d'architectures et de modes de pilotage (figure1.2).

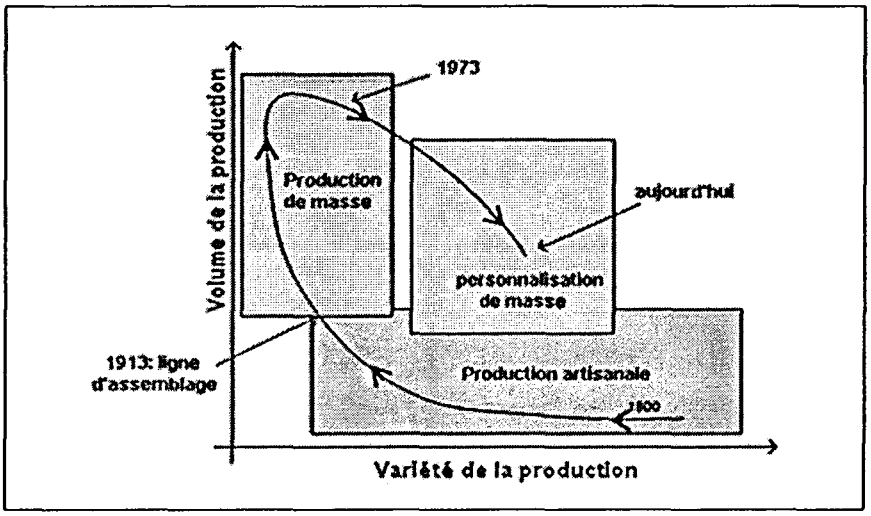


Figure 1.2 : évolution des modes de production (Leitao, 2004)

Cette évolution a entrainé la conception de systèmes flexibles de fabrication (SFF) combinant l'efficacité d'une infrastructure de transport avec la flexibilité du processus de fabrication, permettant la réduction du temps de transport, de l'investissement dans des équipements et la minimisation des temps d'installation (Groover 1987, Leitao 2004, Todd, 1997).

1.2.2 Caractéristiques d'un SFF

Un système de fabrication flexible (SFF) combine une structure de transport et une structure de fabrication (figure 1.3) lui apportant un niveau de flexibilité qui lui permet de réagir en cas de changements prévus ou imprévus du contexte de production (modification d'objectifs, modifications des ordres des clients, perturbations...). Cette capacité d'adaptation (concept d'agilité : Leitao, 2004, Groover, 1997) vise, en réponse à ces changements, à réduire le temps de reconfiguration du système pour adapter les caractéristiques du SFF (type de produits à réaliser, capacité de production...) à la nouvelle situation.

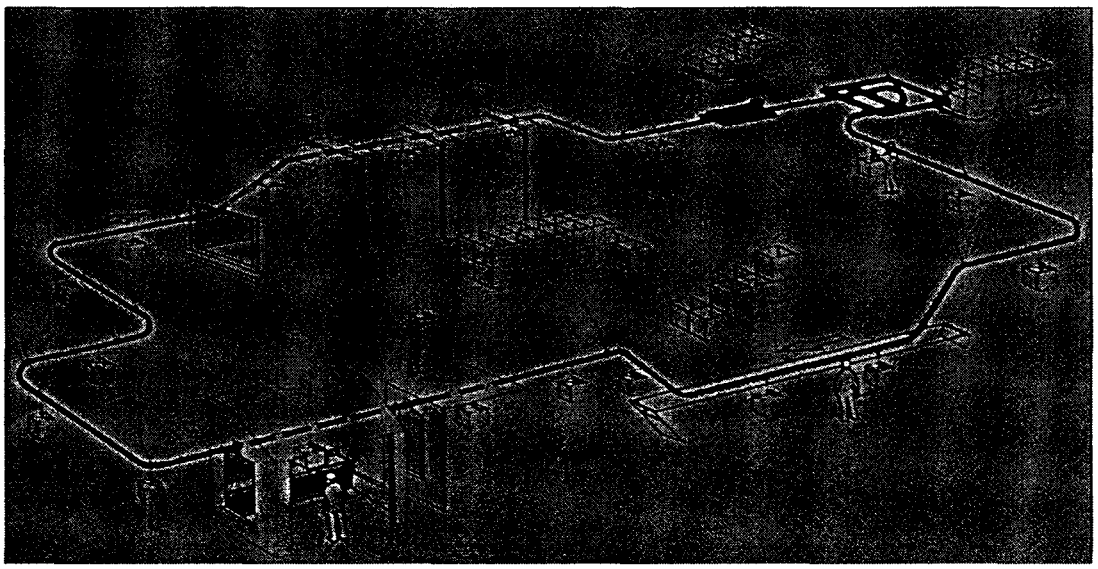


Figure 1.3 : système de fabrication flexible

Cette agilité (ou flexibilité d'obtention de services) est généralement déclinée selon deux catégories :

- flexibilité des ressources de traitement (machines outils, robots) : elle correspond à la capacité d'un système à se reconfigurer pour produire des nouveaux types de produits, à modifier l'ordre des opérations exécutées sur un produit, des machines multiples pour la même opération sur un produit et à s'adapter aux changements dans la demande des produits.

- flexibilité du routage / transport : elle représente la capacité du système à utiliser des trajets alternatifs pour acheminer, en général en utilisant des navettes, des AGV, qui circulent à l'aide de différents supports de guidage (convoyeurs, rails, filoguidage, système infrarouge...), les produits aux postes de travail pour recevoir des opérations (services) de traitement.

En fonction du nombre des ressources, des capacités de chaque ressource et des contraintes imposées pour la fabrication des produits selon les ordres des clients, un atelier de production est classé (http://www.speedylook.com/Scheduling_of_workshop.html) : selon l'un des types suivants :

1. Atelier *mono-ressource* : dans ce cas la planification consiste à ordonnancer les ordres de fabrication selon des règles de priorité (par exemple : priorité aux produits dont le temps d'exécution est le plus court, date de livraison la plus proche);

2. Atelier de type *flowshop* : l'atelier comporte N ressources et chaque produit doit passer par chacune (ou un certain nombre) de ces ressources. Le nombre d'opérations nécessaires pour réaliser le produit est généralement égal au nombre de ressources;
3. Atelier de type *jobshop* ou *flexible jobshop* : *m* ressources participent à la réalisation de *n* ordres de production (produits). La succession des opérations d'un ordre doit respecter des précédences imposées par la gamme de fabrication. Une partie des ressources est capable d'exécuter des opérations identiques;
4. Atelier de type *openshop* : *m* ressources (certaines polyvalentes) permettent de réaliser *n* ordres de production sans prédétermination de la ressource où les opérations sont effectuées;

C'est le cas 3 – le plus générique – qui sera considéré dans cette thèse (les autres sont des cas pouvant être considérés comme des cas particuliers d'un atelier de type jobshop).

1.2.3 La structure de transport / routage d'un SFF

La structure de transport comporte différents types d'éléments cités précédemment : convoyeurs, palettes robots de manutention, AGV. Le choix du mode de transport peut dépendre notamment du type de produits et du type de production réalisée. Par exemple les convoyeurs dans les systèmes de production à haute vitesse (embouteillage), les robots de manutention dans l'industrie automobile.

Parmi ces différents types de moyens de transport, les AGV constituent des systèmes bien adaptés pour répondre aux besoins de flexibilité de routage et de transport des FMS. La flexibilité des AGV résulte d'une part, du fait qu'ils peuvent se déplacer selon des circuits, fixes ou évolutifs, unidirectionnels (déplacement dans un seul sens sur toutes les liaisons) ou bidirectionnels (déplacement dans les deux sens sur une liaison) ; d'autre part qu'ils peuvent transporter différents types de charge.

De nombreux systèmes d'AGV existent actuellement dans des applications de transports, de stockage et de transbordement. Les principaux éléments de différentiation entre les systèmes d'AGV sont le nombre d'AGV utilisés, le nombre d'opérations de transport à réaliser, le taux de charge des AGV, les distances à parcourir entre chaque point de livraison/chargement et le nombre de ces points. On trouvera dans (Vis 2006) un état de l'art sur les systèmes AGV, en particulier ceux utilisés dans les FMS.

De nombreux travaux de recherche ont été réalisés sur le problème de la planification et du pilotage des réseaux de chariots automatisés. Plusieurs outils de simulation proposent également des modules spécifiques pour la modélisation et la simulation de tels réseaux (SIMAN, ARENA). Cependant, la plupart de ces travaux et outils concernent essentiellement les systèmes unidirectionnels. La bidirectionnalité d'un réseau, induit une plus grande complexité de la commande. Il est nécessaire de développer des algorithmes de commande assurant la gestion des conflits (Maza et Castagna, 2005), (Castagna et Maza, 2003), (Qiu et al., 2002).

1.2.4 La structure de fabrication d'un SFF

On peut distinguer, sans vouloir être exhaustif, trois grands types de ressources de fabrication :

- les machines outils (multi axes, multi groupes, à automate intégré...) : elles sont en général utilisées pour diverses opérations de fabrication (usinage, usinage à grande vitesse (UGV), découpe (laser, jet d'eau), rectification, électro-érosion...) et équipées de dispositifs de commandes numériques,

- les manipulateurs et robots : ils peuvent assurer des tâches de manipulation (déplacement entre postes de travail, positionnement de pièces...). Les robots sont également utilisés pour des tâches d'assemblage, seuls ou en coopération avec un ou plusieurs autres robots,

- les systèmes et machines de mesurage et de contrôle : ils utilisent différents types de technologie, en particulier les systèmes de vision (imagerie, lecture optique...).

1.3 Système de transport à la demande (PRT)

1.3.1 Les systèmes de transport à la demande

Le système de transport de type PRT (Personal Rapid Transit) est une forme de transport public conçue pour répondre aux inconvénients actuels du transport urbain (Carnegie et al., 2007), (Buchanan et al., 2005).

1.3.2 Caractéristiques d'un système de transport de type PRT

Un système PRT offre des services de transport même pour un petit nombre de passagers. Les véhicules sont appelés par une personne et partagés uniquement avec les personnes voulues. Les passagers montent et descendent à des arrêts prédéfinis.

La conduite et la coordination des différents véhicules, la gestion de la sécurité (évitement de collisions et d'obstacles) sont complètement automatisées, l'objectif étant d'optimiser les temps de trajet et d'attente.

La zone couverte par un système PRT peut correspondre à la dimension d'une ville, mais des zones plus petites (parc d'attraction) ou plus grandes (réseau interurbain par interconnexion de systèmes similaires) peuvent être envisagées.

1.3.3 La structure de transport

La structure de transport est essentiellement constituée de stations et de voies de transport.

- les stations : elles fournissent différents services, principalement l'embarquement/débarquement des passagers, le stockage de véhicules vides, l'intermodalité. Certaines stations permettent un déroulement sans interruption du voyage ("off-line station") (figure1.4). Les arrêts se font alors en dehors du réseau principal de voies, sans perturber le trafic.

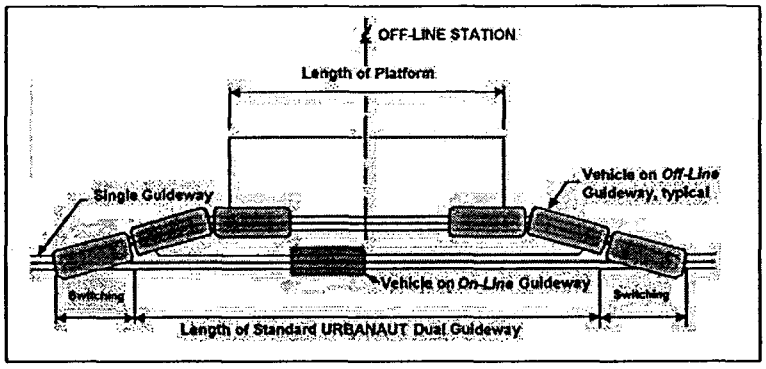


Figure 1.4 : station hors-ligne d'un système PRT (Système PRT Urbanaut)

- le réseau : les éléments du réseau (essentiellement des rails) assurent des fonctions de guidage, de communication et d'alimentation en énergie.

1.3.4 Les véhicules

Les véhicules sont de capacité variable (typiquement entre deux et six passagers, mais parfois plus) et sont (potentiellement) disponibles à toute heure (figure1.5). Il est possible de constituer un "train" de véhicules.

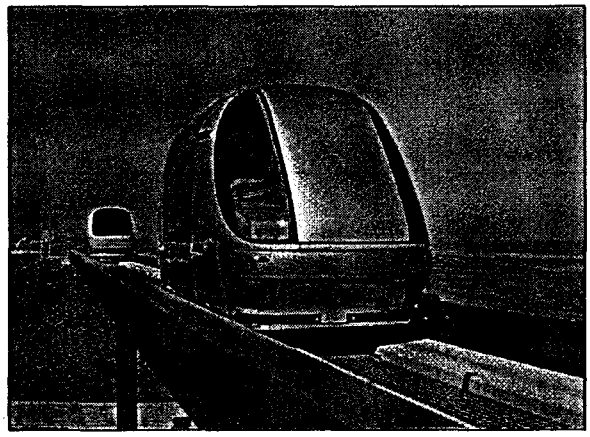


Figure 1.5 : véhicule d'un système de transport à la demande

Dans la littérature on fait référence à de nombreux types de systèmes PRT. La description et l'analyse de l'ensemble des systèmes PRT ne fait pas l'objet de cette thèse. On trouvera un état de l'art et une présentation des systèmes de transport PRT actuels et de leur niveau de développement dans (Berger et al. 2011) et sur les sites suivants :

- (http://faculty.washington.edu/jbs/itrans/prtquick.htm),

- (http://faculty.washington.edu/jbs/itrans/techtable.htm),

- (http://www.advancedtransit.org/news.aspx),

- (http://planningwiki.cyburbia.org/Personal_Rapid_Transit (Carnegie et al., 2007).

Les plus cités dans la littérature sont les systèmes SkyWeb Express (www.taxi2000.com, 2010, Anderson, 1998), Ultra (www.ultraprt.com, 2010, Lowson, 2003), Vectus (www.vectusprt.com, 2010) et FROG (www.2getthere.eu, 2010, Lohmann, 2007).

A titre d'exemple, le système SkyWeb Express, développé par Taxi 2000 Corporation, est un projet très avancé. Son principal concepteur, Edward J. Anderson (Anderson, 1998), a contribué au progrès du concept PRT depuis son début dans les années 1970. Son système de contrôle se concentre sur les problèmes de fonctionnement du véhicule (profils de vitesse, la poursuite du véhicule suivant, le confort et la sécurité du transport et également le fonctionnement en cas d'usage intense). L'architecture de pilotage est organisée sur trois niveaux: (1) un niveau central pour l'allocation et le routage, (2) la zone des contrôleurs latéraux qui gèrent le profil de la vitesse des véhicules et de la fusion des différents flux de véhicules et (3) le niveau véhicule en charge du positionnement sur les rails, du contrôle de la sécurité et des différentes fonctions relatives au confort des passagers. Jusqu'à présent, une plate-forme de test pour fournir des informations essentiellement sur les aspects mécaniques a été développée.

Le projet ULTRA (Urban Light TRAnsport) développé par Advanced Transportation Systems, a commencé au milieu des années '90 au Royaume Uni et, jusqu'à présent, est presque le seul projet déployé pour une application commerciale. Une implémentation réelle a été faite à l'aéroport d'Heathrow à Londres (www.ultraprt.com, 2010), qui relie la nouvelle aérogare avec le parking auto, et grâce aux bons résultats le projet sera ouvert aux utilisateurs.

1.4 Pilotage des SFG

Dans la littérature, le terme pilotage diffère d'un auteur à l'autre (Pujo et Kieffer, 2002). Le pilotage est généralement réparti en trois niveaux :

- *Niveau stratégique,* en charge de la planification à long terme du système, ex. : traiter et insérer les ordres de production dans le système réel (cas d'un FMS), définir une distribution proactive des navettes dans les stations (cas d'un PRT).

- *Niveau tactique,* en charge par exemple d'ordonnancer les opérations d'un ordre de fabrication, de définir les chemins des moyens de transport.

- *Niveau opérationnel,* qui doit assurer la commande directe des ressources.

Nous nous intéressons, dans le cadre de cette thèse, à deux fonctions essentielles du pilotage d'un SFG correspondant au niveau tactique : l'allocation et le routage. D'autres fonctions, du niveau opérationnel (commande des moyens de déplacements, gestion de la sécurité...) sont en général gérées localement au niveau des véhicules et ne seront pas abordées. Nous adoptons pour les fonctions d'allocation de ressources et de routage les interprétations suivantes :

- **Allocation :** le SFG est composé d'un ensemble de ressources (moyens de transport, points de réception de services) et d'un ensemble d'éléments (produits, personnes). L'allocation de ressources consiste à répartir, selon un certain nombre de critères, ces ressources entre les éléments.

- **Routage** : c'est le mécanisme par lequel des chemins sont sélectionnés au sein du réseau, selon un certain nombre de critères, pour acheminer les entités mobiles d'un point de départ jusqu'à un point de destination (intermédiaire ou final), (Sallez et al., 2010).

- *Remarque* : dans le cas des FMS, l'allocation de ressources est souvent couplée à l'ordonnancement des opérations de production (Borangiu et al., 2009). C'est pourquoi le problème d'ordonnancement sera également abordé dans le chapitre 3.

Différentes approches de pilotage existent :

- une *approche globale* qui adopte un concept de pilotage, qui a inspiré les systèmes CIM (Computer Integrated Manufacturing), où les entités composantes ont un rôle bien établi et où existent des relations strictes entre elles (ex. relation de type maître esclave). Ainsi, la planification est faite en premier, suivie par l'ordonnancement et enfin les ordres, planifiés et ordonnancés par une entité décisionnelle qui s'appuie sur une vue globale du système, sont exécutés. Cette politique convient pour le pilotage des systèmes à dynamique interne lente (ex. changement lent de l'état des ressources composantes, fluctuations rares dans la commande...).

- une *approche décentralisée* où les entités composantes du système de pilotage sont caractérisées par un degré d'autonomie donné et la possibilité de coopérer entre elles. Cette politique est mieux adaptée aux systèmes à dynamique rapide, les entités autonomes étant capables de contrôler en temps réel les processus du système. Dans ce cas, l'échange d'information se produit entre les entités autonomes pour se synchroniser et que le système global ait un comportement cohérent et convergent.

Dans le cas d'un système à flux guidé :

- un SFG est caractérisé par une dynamique rapide, nécessitant des temps de réaction brefs, indispensables par exemple pour assurer une bonne qualité de service,
- les entités d'un SFG peuvent actuellement disposer, grâce à l'évolution des technologies de l'information et de la communication, des capacités décisionnelles nécessaires pour disposer d'une certaine autonomie et reconfigurer le système en temps réel selon l'évolution de l'environnement et des objectifs imposés.

Dans ces conditions, l'approche décentralisée apparaît comme une alternative séduisante pour le pilotage de SFG. Dans la suite du chapitre, nous mettons en évidence l'intérêt d'un pilotage distribué pour un SFG.

2 Architectures de pilotage

On distingue classiquement quatre architectures de pilotage (Figure1.6 - a et 1.6 - b). Elles sont référencées dans la littérature en fonction de leur topologie (Dilts, 1991), (Trentesaux, 2007).

2.1 Pilotage centralisé

Dans ce cas il existe un seul point de pilotage qui prend toutes les décisions concernant les flux des matériels du système cible. Un système de contrôle centralisé a l'avantage d'une architecture simple (capteurs et actions connectés à un seul point décisionnel) et permet l'optimisation globale du fonctionnement.

Les désavantages de cette architecture sont :

- les temps de réponse de durée importante vis-à-vis de la dynamique du système lorsque le système de pilotage présente un nombre important d'entrées/sorties,

- la difficulté d'apporter des changements dans la structure physique du système de pilotage qui n'est pas modulaire,

- la faible tolérance aux défauts et aux fautes (si l'unité centrale tombe en panne tout le système devient inutilisable jusqu'au moment de sa remise en fonction).

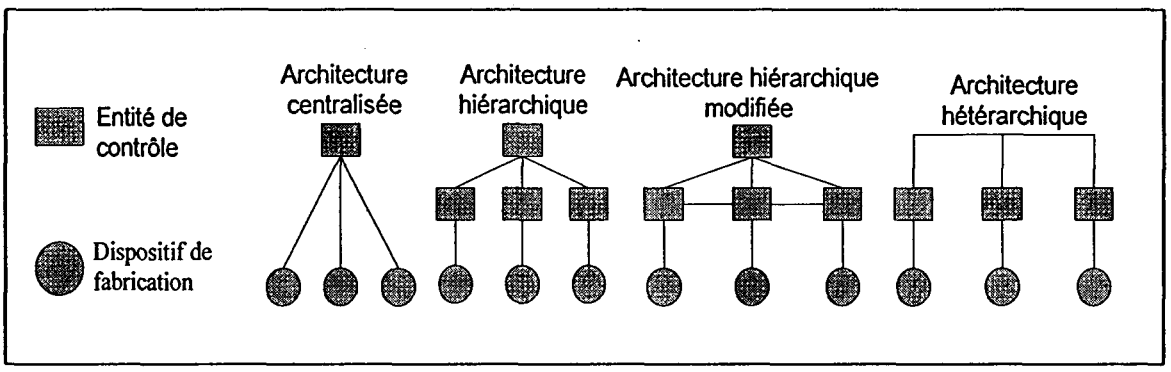


Figure 1.6 - a : architectures classiques de pilotage

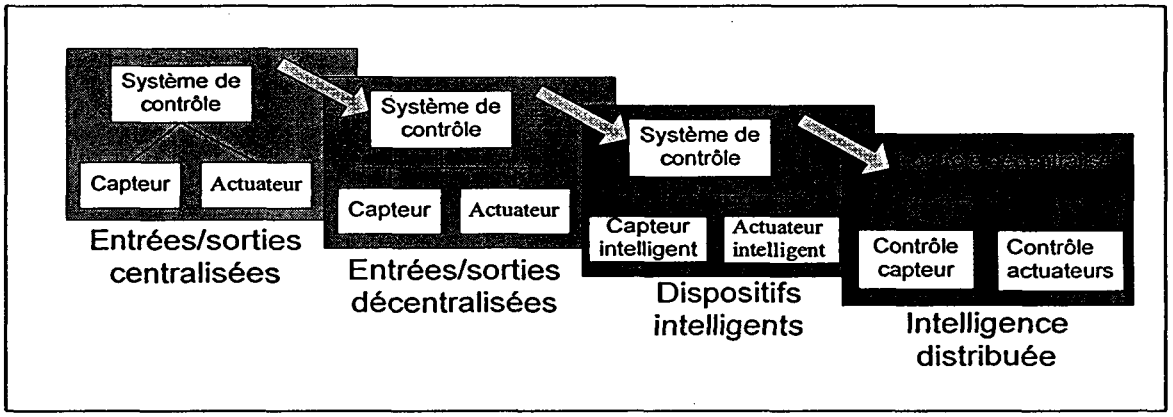


Figure 1.6 - b : évolution des architectures opérationnelles de pilotage

2.2 Systèmes hiérarchiques

Les systèmes de pilotage hiérarchiques sont le résultat de l'approche "de haut en bas" (*top down*), type maître esclave, où le flux des ordres est envoyé à partir des entités de pilotage de haut niveau vers les entités de pilotage de bas niveau. Cette approche est associée avec le concept CIM (*Computer Integrated Manufacturing*) et la pratique habituelle consiste à diviser le problème global (de complexité élevée) en plusieurs sous problèmes hiérarchiquement dépendants l'un de l'autre qui ont en charge le pilotage sur différents horizons de temps : planification, ordonnancement, pilotage direct des ressources et (ou) surveillance des ressources.

Les avantages de cette architecture sont :

- le temps de traitement d'une perturbation est plus faible que dans le cas centralisé grâce à la hiérarchisation des niveaux de traitement du problème,

- l'optimisation globale du fonctionnement du système,

Elle présente certains désavantages :

- un changement dans la structure du système (matériel et/ou logiciel) est une tâche très difficile qui ne peut être réalisée sans l'arrêt du système et la mise à jour de tous les niveaux hiérarchiques,

Cette approche de pilotage offre des solutions optimales mais sous réserve que des hypothèses strictes soient satisfaites (ex. absence d'ordres urgents, pas de pannes de ressources, stabilité des temps opératoires des ressource...). Mais la réalité est rarement aussi déterministe et cette approche devient rapidement inefficace lorsque le système doit faire face à un nombre d'aléas et de modifications importants.

2.3 Systèmes hétérarchiques

Cette architecture de pilotage, nommée également isoarchie par certains auteurs (Pujo et al., 2009), est composée d'un groupe d'entités indépendantes qui communiquent entre elles pour satisfaire les demandes. Dans ce cas il n'existe pas de relation figée comme dans le cas des architectures précédentes. Ce concept a été formalisé par Mc Culloch (Mc Culloch, 1945) en utilisant la théorie des graphes et, plus récemment, dans (Trentesaux, 2007) avec la notion de graphe d'influence. Un graphe d'influence est un graphe orienté, composé de nœuds qui représentent les entités décisionnelles et d'arcs orientés qui représentent une interaction directionnelle de type maître esclave : l'entité de départ de l'arc est l'entité maître et l'entité d'arrivée de l'arc est l'entité esclave. Dans le cas des systèmes de pilotage hétérarchique, l'ensemble des nœuds du graphe d'influence associé peuvent être considérés comme maître et esclave (Figure1.7).

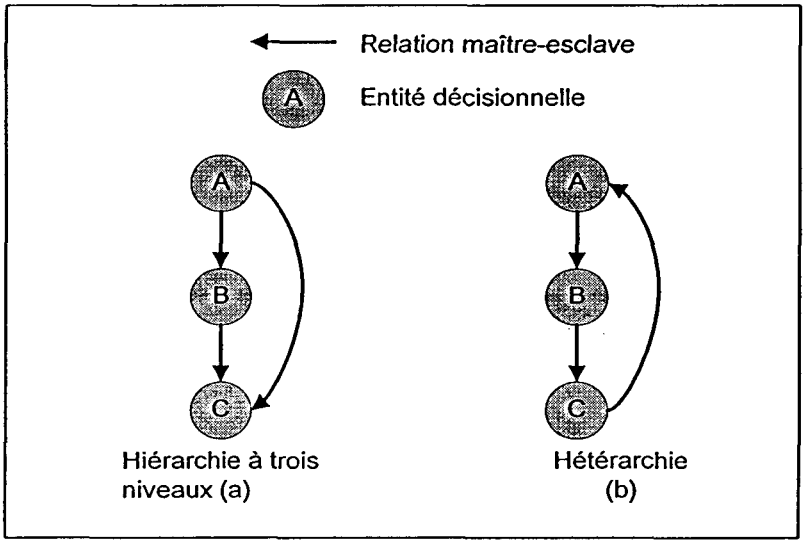


Figure 1.7 : différence entre une hiérarchie (a) et une hétérarchie (b) sur le graphe d'influence

Grâce à l'architecture décentralisée, les entités ont une autonomie totale et le système peut réagir rapidement aux types de changements prévus. Toutefois, parce que le comportement d'une entité dépend des comportements des autres entités, il est, dans certains cas, impossible

d'optimiser globalement le fonctionnement du système et ses performances sont difficiles à prévoir.

2.4 Systèmes hybrides

Les systèmes de pilotage hybride sont issues des architectures de type hétérarchique et hiérarchique, conçues en cherchant à bénéficier des avantages de chacune d'entre elles. Les architectures de pilotage hybride offrent la possibilité de coopération entre les entités de niveau inférieur pour résoudre le problème et, en même temps, elles gardent la notion de superviseur. En autres termes, les systèmes de pilotage hybrides ont la possibilité d'évoluer entre un fonctionnement purement hiérarchique et un fonctionnement purement hétérarchique.

Ainsi, un système peut commencer à fonctionner avec un pilotage de type hiérarchique, si les modifications des conditions de fonctionnement l'exigent, un pilotage hétérarchique peut être mis en œuvre (Borangiu et al., 2009). Les lignes directrices ci-dessus ont conduit les chercheurs à définir de nouvelles approches dans la conception des architectures de pilotage comme l' « open control »/« pilotage ouvert » (Sallez et al., 2009) et le pilotage semi hétérarchique (Rahimifard, 2004).

2.4.1 Open-control (Pilotage ouvert)

Développé à l'Université de Valenciennes, laboratoire TEMPO, (Sallez et al., 2009), le pilotage (contrôle) ouvert (figure1.8) est un cadre générique de contrôle dans lequel une entité peut atteindre son propre objectif en communiquant avec des entités similaires et, simultanément, en respectant les objectifs globaux du système.

Comme le montre la figure1.8 - a, ce cadre implique différents niveaux: le *niveau i* sert au *niveau i +1* en haut et est desservi par le *niveau i-1* en bas. Cette approche propose une organisation de la partie contrôle sur des horizons de temps avec les problèmes à long terme résolus en haut et les problèmes à court terme (temps réel) résolus en bas. Dans ce cadre, des entités autonomes peuvent exister dans le monde physique ainsi que dans le monde informationnel.

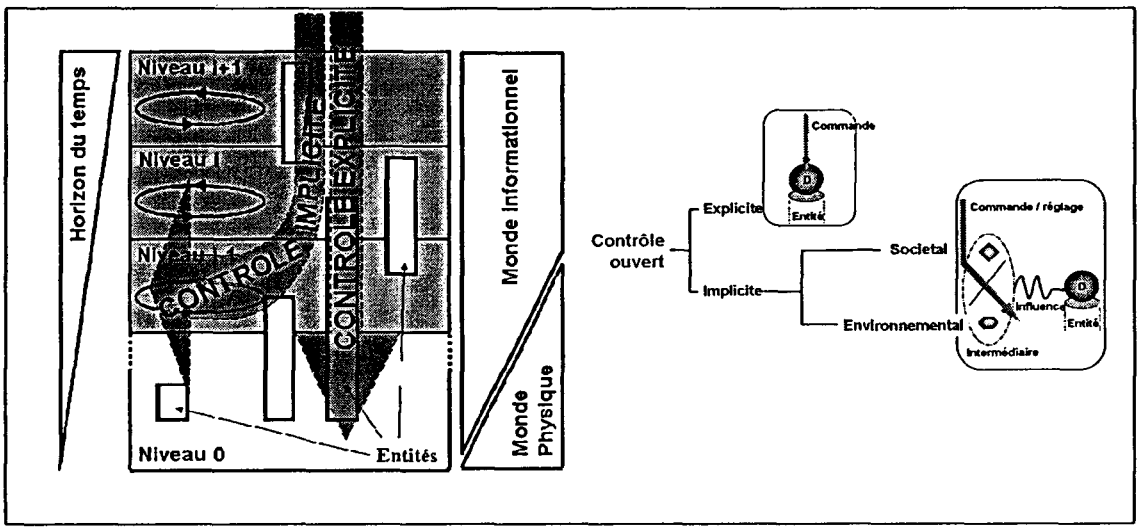


Figure 1.8 - a Figure 1.8 - b

Figure 1.8 : cadre général (a) et typologies de pilotage (b)

Ces entités peuvent être contrôlées selon deux modes (figure1.8 – b) :

- *pilotage explicite* : les entités de niveau bas sont directement pilotées par les entités de haut niveau suivant une relation stricte ;
- *pilotage implicite* : les entités de niveau bas sont *influencées* par un *mécanisme d'optimisation* (MO), mais elles ne sont pas nécessairement contrôlées. Ce dernier type de contrôle est encore divisé en deux types de contrôle implicite en fonction de la manière selon laquelle le niveau inférieur est influencé :
    - pilotage implicite avec un mécanisme d'optimisation sociétale (MOS) : dans ce cas le niveau haut est affiné de la vue partielle d'une propriété collective au sein d'une entité (après avoir modifié son comportement cette entité influence les autres par le dialogue), ou le niveau haut change la dynamique du dialogue dans le MOS en modifiant les paramètres du dialogue au sein de l'entité ;
    - pilotage implicite avec un *mécanisme d'optimisation environnementale* (MOE). Ce type de contrôle est mis en place en utilisant des informations récentes provenant de l'environnement (Borangiu et al., 2009). Cet environnement est caractérisé par un mécanisme de mémorisation, qui mémorise des informations ou des caractéristiques physiques, et un mécanisme d'optimisation qui agit sur les informations mémorisées pour les modifier.

2.4.2 Pilotage hybride (semi hétérarchique)

Un type de pilotage hybride appliqué dans le domaine des systèmes de fabrication flexibles, le pilotage semi hétérarchique propose une organisation de la partie pilotage sur deux niveaux (Rahimifard, 2004 ; Borangiu, 2009) : global et local. Le niveau global est en charge de la planification et de la coordination des activités au niveau de l'atelier (cellule / ligne d'assemblage / usine) et de la résolution des conflits entre les objectifs locaux. Le niveau local possède une certaine autonomie dans la planification et le pilotage des activités internes au sein d'un sous système (par exemple un poste de montage d'un robot).

Par contre, si les systèmes de pilotage sont analysés selon leurs fonctionnalités on met en évidence deux types de systèmes de pilotage classiques (Trentesaux, 2007) :

- *systèmes décentralisés* : chaque élément physique a d'associé sa propre fonctionnalité de pilotage (figure1.9 - a);
- *systèmes distribués* : la fonctionnalité de pilotage est divisée en plusieurs blocs et répartie sur plusieurs systèmes. Par exemple, la distribution d'une fonctionnalité F signifie diviser sa responsabilité entre plusieurs sous fonctions qui s'exécutent dans le même contexte, comme un programme divisé en procédures (figure1.9 b).

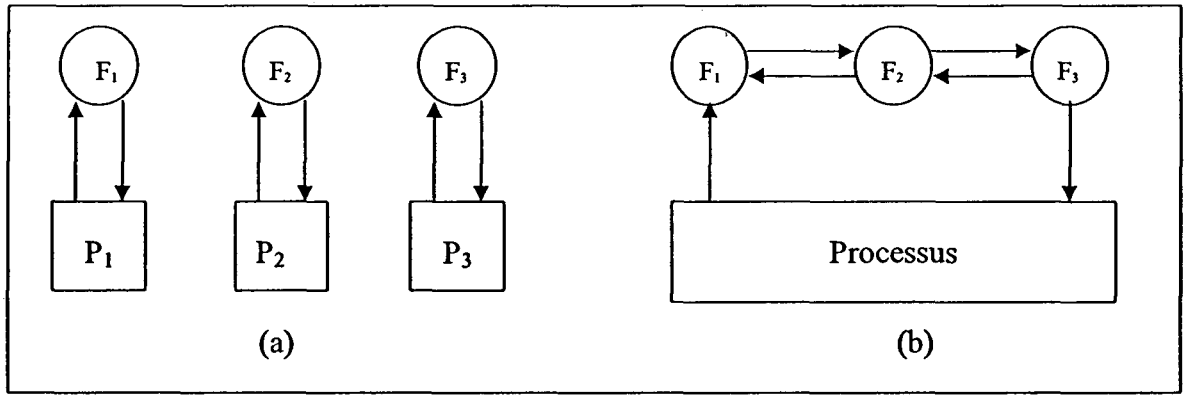


Figure 1.9 : systèmes décentralisés (a), distribués (b)

2.5 Modèles de pilotage

2.5.1 Systèmes multi agent

Les concepts d'*agent autonome* et de *système multiagent* constituent une approche pour l'analyse, la conception et l'implémentation des systèmes logiciels complexes. Cette approche offre un répertoire de techniques et d'outils performants qui ont le potentiel d'améliorer la mise en œuvre des logiciels. Les modèles agents sont utilisés dans diverses applications allant des systèmes de filtrage en messagerie électronique personnalisés jusqu'aux systèmes complexes tels que le contrôle du trafic aérien.

Plusieurs agents autonomes forment un système multi agent (SMA), concept très puissant, en particulier pour l'étude des systèmes complexes dont le comportement collectif est le résultat des interactions entre agents (Ferber, 1995). En fonction de la politique d'interaction entre les agents composants, le comportement collectif peut être cohérent ou non.

Grâce aux progrès techniques de ces dernières années, la majorité des ressources d'un système de fabrication est munie d'un système de contrôle intelligent (contrôleur industriel), ce que permet l'étude de l'ensemble en tant que système multi agent, caractérisé comme suit (Jennings et al., 1998) :

- chaque agent a une connaissance incomplète du système ou de ses capacités spéciales et distinctes,
- il n'existe pas un point de contrôle central,
- l'information est décentralisée,
- l'exécution des opérations est asynchrone.

Parmi les raisons qui suscitent l'intérêt dans l'utilisation des systèmes multi agent pour le pilotage on cite : la capacité de fournir robustesse et efficacité, la capacité de faciliter l'interopérabilité avec les systèmes existants et la capacité de résoudre les problèmes dans lesquels les informations, l'expérience et le contrôle sont distribués.

Les définitions générales n'étant pas suffisamment stabilisée et étant donné la diversité des domaines et des applications qui utilisent les SMA, la communauté scientifique a essayé de standardiser d'autres aspects concernant le concept d'intelligence distribuée qui utilise des agents comme images d'entités physiques (www.fipa.org, 2010).

- La FIPA (*Foundation for Intelligent Physical Agents*, www.fipa.org), qui développe ses travaux dans le domaine des agents autonomes, a apporté une contribution importante dans les domaines de la communication entre les agents et de la standardisation des messages. Les spécifications produites par la FIPA abordent en général les aspects de la conception du logiciel (diagrammes UML de classes et séquences), (figure 1.10)

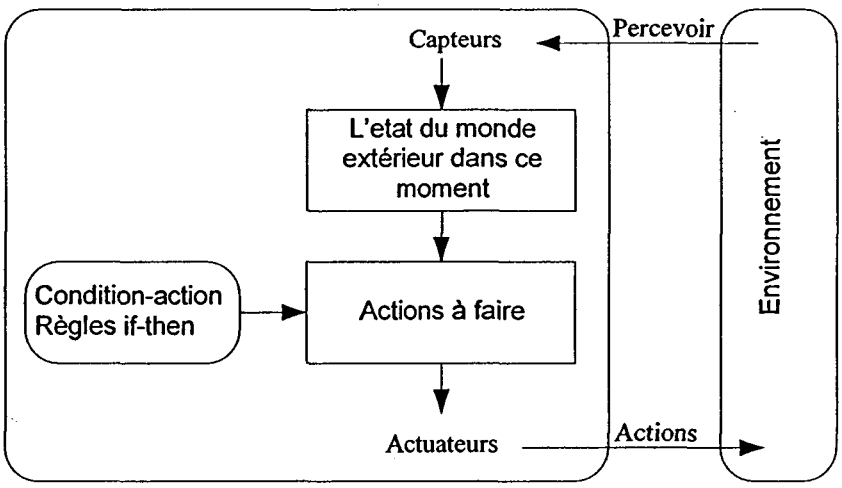


Figure 1.10 : agent réflexif simple (diagramme UML selon FIPA)

Ainsi, le protocole FIPA Contract Net (figure1.11) a été proposé et ensuite standardisé. Dans ce protocole un agent, (l'initiateur), a le rôle d'administrateur et peut demander la réalisation d'une tâche qui sera effectuée par un ou plusieurs agents participants. En outre, l'exploitation du système impliquant des coûts, l'optimisation d'une fonction de coût est prise en compte.

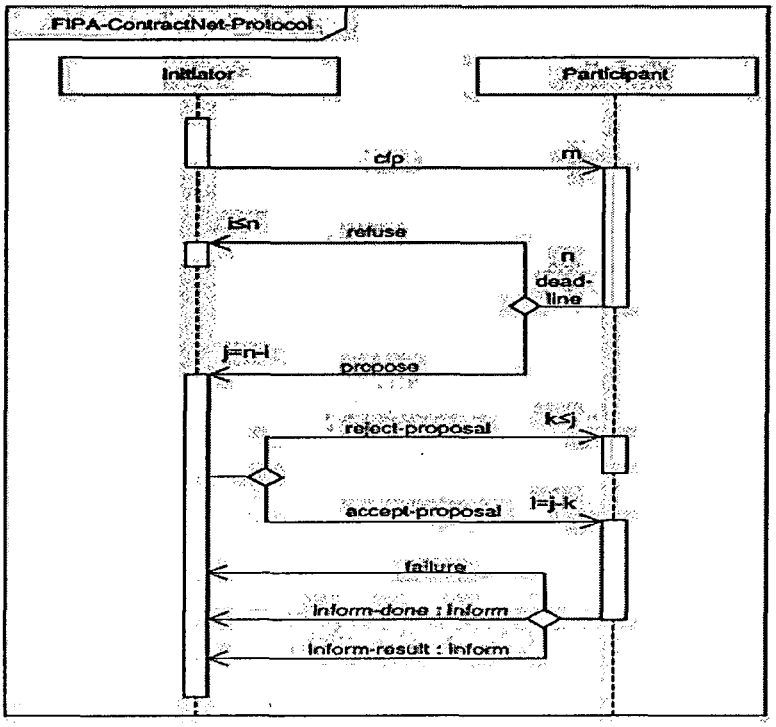


Figure 1.11 : protocole d'interaction Contract Net

Ce "coût de la tâche" est mesuré en fonction du domaine d'application du protocole (ex. le plus court temps d'exécution ou l'utilisation identique des ressources disponibles). Pour une tâche proposée, certains agents peuvent répondre en émettant une proposition, d'autres refusent la tâche. Des négociations sont ensuite menées pour le choix d'un agent parmi ceux ayant donné leur acceptation pour accomplir la tâche.

2.5.2 Systèmes holoniques

Un *holon,* selon Koestler (Koestler, 1967), est une partie identifiable d'un système. Un holon a une identité unique, bien qu'il soit composée d'un nombre de parties subordonnées et, à son tour, est partie d'un système plus vaste. Les holons peuvent être intégrés au monde réel et sont généralement liés à une entité physique.

Les holons sont des entités partiellement ou totalement autonomes, possédant un degré d'indépendance donné, et capables de traiter par elles-mêmes diverses situations. Cependant, ils peuvent recevoir des instructions d'autres holons appartenant à un niveau de décision supérieur. Ils peuvent également être agrégés et ainsi former d'autres holons globaux pour former des "systèmes de systèmes".

Les holons peuvent coopérer en vue d'élargir leur capacité de décision en formant une *holarchie.* Les scénarios de négociation entre holons sont en général très simples et n'impliquent que l'échange de messages.

Les *systèmes holoniques* utilisent la propriété de réactivité des agents pour reconfigurer le système et obtenir ainsi un comportement réactif (Schillo, 2004).

En raison de la nature "temps réel" des processus qui se déroulent dans les systèmes à flux guidés, des systèmes holoniques très simples sont appliqués avec succès aujourd'hui dans le contrôle de la fabrication discrète, répétitive (Schillo, 2004, Fischer, 2003).

Ainsi, une hiérarchie de holons qui fonctionnent comme suit :

1. ils constituent des *ensembles autonomes* en charge de leurs composantes,
2. ils sont considérés comme *partie dépendante* d'un niveau supérieur,
3. ils agissent en *coordination* avec leur environnement,

forme selon Koestler (Koestler, 1967) une *holarchie.*

L'introduction du concept de système holonique dans la fabrication vise à obtenir des avantages tels que : la stabilité contre les perturbations, l'adaptabilité, la flexibilité aux changements et l'utilisation efficace des ressources disponibles.

Pour adapter le concept de holon d'Arthur Koestler dans le domaine de la fabrication, le consortium Holonic Manufacturing Systems (HMS) a proposé les définitions suivantes (*Concepts for Holonic Manufacturing*, 2010) :

- *Holon* : un ensemble autonome et coopératif d'un système de fabrication pour transformer, transporter, stocker et/ou valider des informations et des objets physiques. Un holon est composé d'une partie informationnelle et souvent d'une partie physique. Un holon peut faire partie d'un autre holon.

- *Autonomie* : la capacité d'une entité (ici holon) de créer et contrôler l'exécution de ses propres plans et stratégies.
- *Coopération* : un processus dans lequel un ensemble d'entités développent des plans communs et exécutent ces plans.
- *Holarchie* : un ensemble de holons qui peuvent coopérer pour atteindre un objectif. La holarchie définit les règles de base pour la coopération entre les holons en limitant leur autonomie.
- *Système de fabrication holonique* : une holarchie qui intègre toutes les activités de fabrication à partir de la réception d'une commande, passant par la conception, la production et terminant avec la livraison.
- *Attributs holoniques* : attributs d'une entité qui lui confèrent le statut de holon. L'ensemble minimal d'attributs est l'*autonomie* et la *coopération*. Les autres attributs importants sont les algorithmes et les procédures pour négocier le partage des tâches et pour exécuter les actions établies en commun.

Parmi les attributs proposés par Koestler, l'adaptation dans le domaine manufacturier a été recherchée. Le groupe IMS (*Intelligent Manufacturing Systems*) a proposé un premier projet qui a utilisé les concepts ci-dessus et les a intégrés dans une nouvelle thématique de recherche : HMS (*Holonic Manufacturing Systems*). Ce projet avait les objectifs suivants (M. Höpf ,1992) :

- le développement des systèmes de fabrication utilisant des composants modulaires,
- le développement de modules intelligents qui se reconfigurent en fonction de la demande,
- le développement de systèmes adaptatifs ayant des capacités "*plug and play*",
- le développement de ressources ayant la capacité d'intégration dans des systèmes de production et des chaînes d'approvisionnement,
- permettre la migration de systèmes existants vers des systèmes purement distribués.

L'implémentation du concept de fabrication holonique est faite en utilisant la technologie informatique de gestion des systèmes multi agent qui offre modularité, décentralisation et réutilisation des structures complexes (Leitao, 2004), comme suggéré en figure1.13.

2.6 Synthèse des paradigmes de pilotage

Les deux paradigmes agent et holon ont été développés à partir des mêmes besoins : autonomie et coopération au sein de systèmes dynamiques et perturbés, en s'appuyant sur la décentralisation et la distribution des entités du système et des fonctions de pilotage. Toutefois, des différences existent entre les deux concepts. L'agent, concept et technologie, est apparu dans les domaines de la programmation et de l'intelligence artificielle. Le holon trouve son origine dans la fabrication assistée par ordinateur (Computer Integrated Manufacturing - CIM). C'est un concept initialement utilisé dans le développement de la partie pilotage d'un système manufacturier.

Les caractéristiques d'un SFG citées précédemment : dynamique rapide nécessitant une forte réactivité, possibilité, pour les entités du système, de disposer d'une autonomie de décision et de

communiquer entre elles pour reconfigurer le système en temps réel conduisent naturellement à envisager un modèle holonique de pilotage.

Nous décrivons dans le paragraphe suivant les principaux modèles d'architecture holonique.

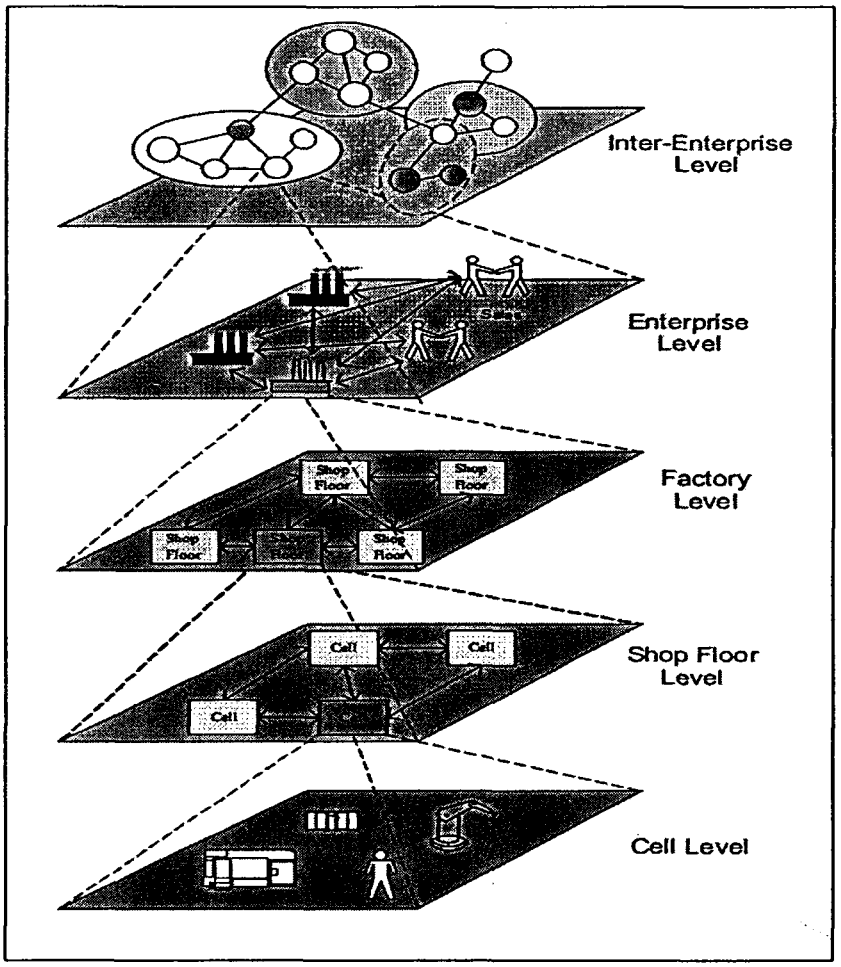


Figure 1.13 : niveaux de la fabrication holonique distribuée

3 Modèles holoniques de pilotage

3.1 PROSA

Developpée à l'Université Catholique de Leuwen l'architecture PROSA utilise les concepts d'agrégation et de spécialisation de la programmation orientée objet pour proposer une structure de contrôle composée de trois types de holons de base (Van Brussel et al., 1998) : holon ordre, holon produit et holon ressource.

Aux trois types de holons de base, des holons coordinateurs (Staff Holon) peuvent être utilisés pour l'aide à la décision dans les processus qui fournissent des conseils (ou recommandations) aux holons de base et résolvent d'une façon hiérarchique les conflits inter holons. L'architecture qui en résulte présente un degré élevé d'autosimilitude, ce qui réduit la difficulté d'intégrer de nouveaux composants et simplifie la configuration du système.

Le système de pilotage final contiendra ainsi certains ou tous les éléments suivants :

- *Holon ressource* (*Resource Holon*) : une seule entité (figure1.14) composée d'un ou plusieurs processus physiques ou ressources, de leurs systèmes de contrôle et des opérations effectuées par l'opérateur humain (ex. d'un terminal, d'une interface).
- *Holon produit* (*Product Holon*) : un holon produit détient la connaissance sur les processus et les produits pour permettre d'assurer la fabrication effective de produits réels. Un holon produit contient des informations à jour sur, par exemple, le cycle de vie du produit, les exigences des utilisateurs, la qualité, la conception, les processus, les matériaux... Il contient le "modèle du produit " et non l'état des instances de ce produit en cours de fabrication. Le holon produit agit comme un serveur d'information pour les autres holons du HMS.
- *Holon ordre* (*Ordre Holon*) : ensemble composé par la totalité des informations nécessaires pour réaliser une commande spécifique demandée par le client. Il inclut des informations telles que la qualité du produit, les dates de livraison, les coûts et la priorité associée. Il peut aussi inclure les produits physiques finis ou en cours de fabrication et/ou des informations sur l'état actuel de la commande. Il est également responsable de la gestion de la réalisation du ou des produits de la commande qui lui est assignée.
- *Holon coordinateur* (*Staff Holon*) : entité optionnelle support (ordinateur, expert humain ou une combinaison des deux) qui assure la coordination entre les holons de base et la conformité avec les objectifs globaux de l'entreprise.

L'architecture générique PROSA intègre les aspects spécifiques aux topologies hiérarchiques et hétérarchiques en apportant des innovations significatives : la structure du système est séparée de l'algorithme de contrôle, les aspectes logiques sont découplés des aspectes techniques. Cette structure est ouverte pour le développement des algorithmes hybrides de contrôle / pilotage.

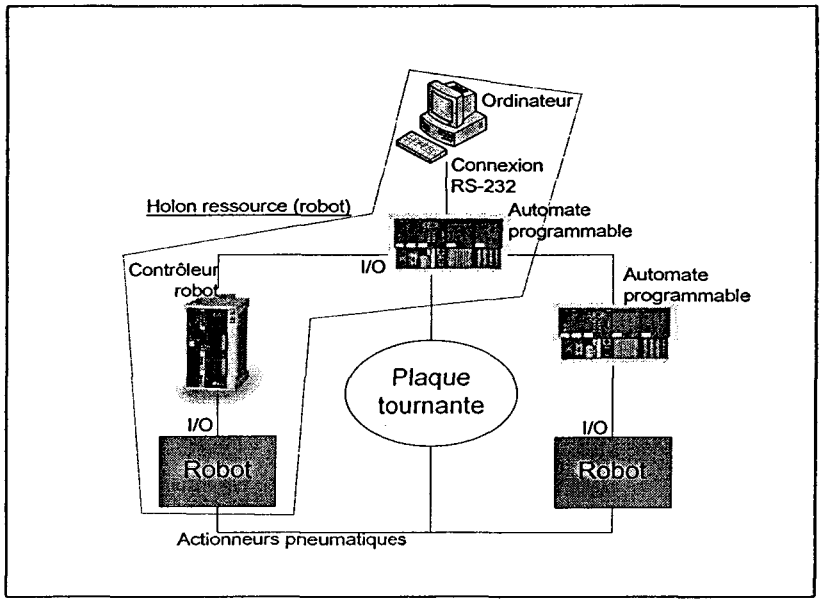
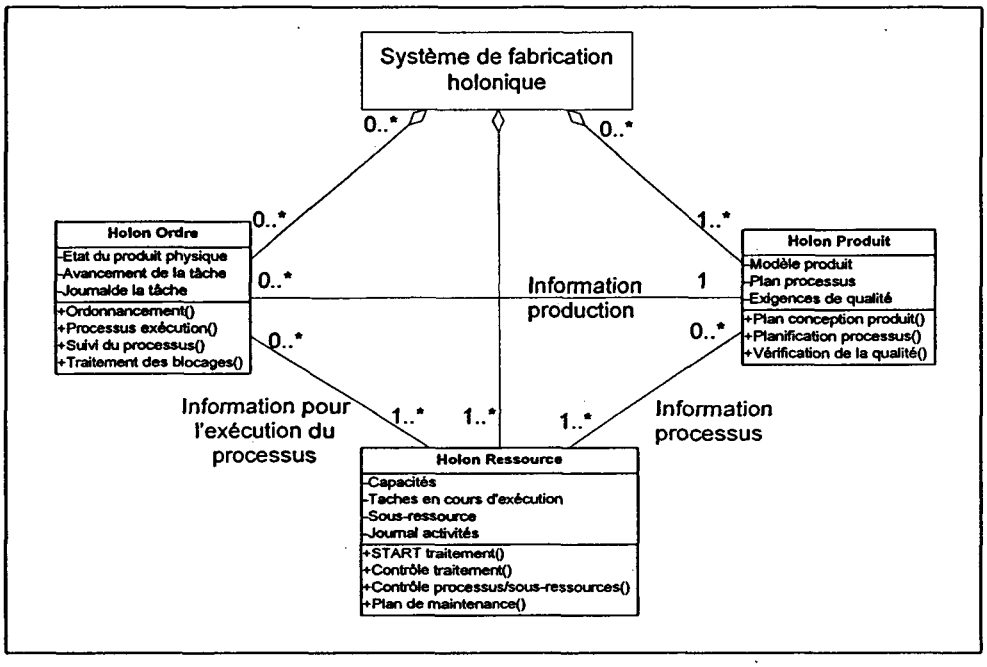


Figure 1.14 : exemple de holon ressource dans l'architecture PROSA

La figure1.15 illustre les trois types de holons de base et les relations entre eux ainsi que les types d'informations échangées dans une représentation utilisant le langage UML (Unified Modelling Language).

Chaque carré représente une entité dans le système, dans ce cas un type d'holon. Les lignes entre les carrés représentent les relations entre les holons (la relation d'*agrégation* est une ligne avec un diamant, la relation de *spécialisation* est une ligne avec des flèches et la relation d'*association* est représentée par une simple ligne). Aux deux extrémités de la ligne sont associés des numéros qui représentent le nombre des holons impliqués dans la relation.



Figigure 1.15 : fonctions et données associées aux trois types de holons de base

3.2 ADACOR

L'architecture ADACOR (Leitao, 2004) fait la décomposition des fonctions de production en un ensemble d'entités autonomes et coopératives, ce qui conduit aux avantages tels que : la modularité, la décentralisation, l'agilité, la flexibilité, la robustesse et l'évolutivité.

L'introduction des entités de surveillance crée une hiérarchie dans les systèmes décentralisés qui permet d'atteindre un optimum. Les capacités d'auto organisation des entités composantes permettent l'évolution et la reconfiguration dynamiques de la structure de contrôle distribué, en combinant l'optimalité avec une réponse rapide aux perturbations.

En ce qui concerne les types d'entités composantes, l'architecture ADACOR regroupe les holons de fabrication dans les catégories *produit*, *tâche*, entité *opérationnelle* et *entité superviseur* (figure1.16). Cette généralisation s'appuie sur le groupement des entités composantes du système de fabrication selon leurs fonctionnalités et objectifs.

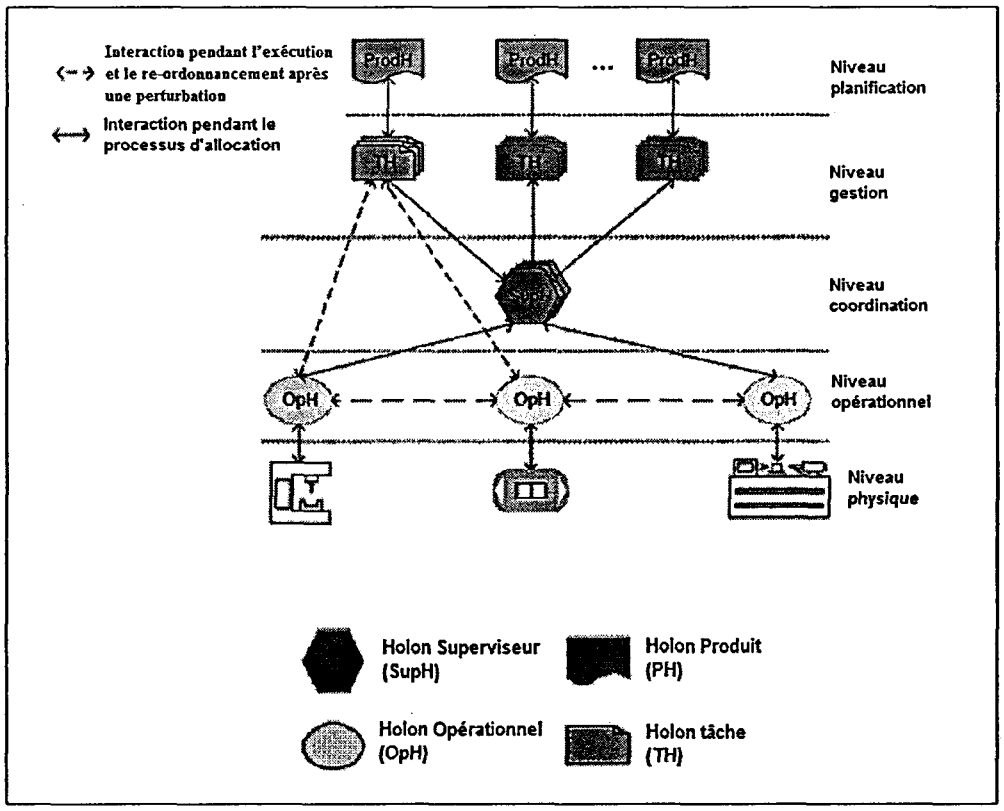


Figure 1.16 : types de holons ADACOR et leurs interactions

Le holon supervision est inspiré par les systèmes biologiques et présente des caractéristiques similaires à celles du holon coordinateur (*staff holon*) défini dans l'architecture PROSA, tandis que les holons tâche, opérationnel et produit sont similaires aux holons ordre, ressource et produit. Les holons ADACOR utilisent le concept *plug and produce*, étant capables d'être ajoutés sans la réinitialisation et reprogrammation du système (Figure1.17)

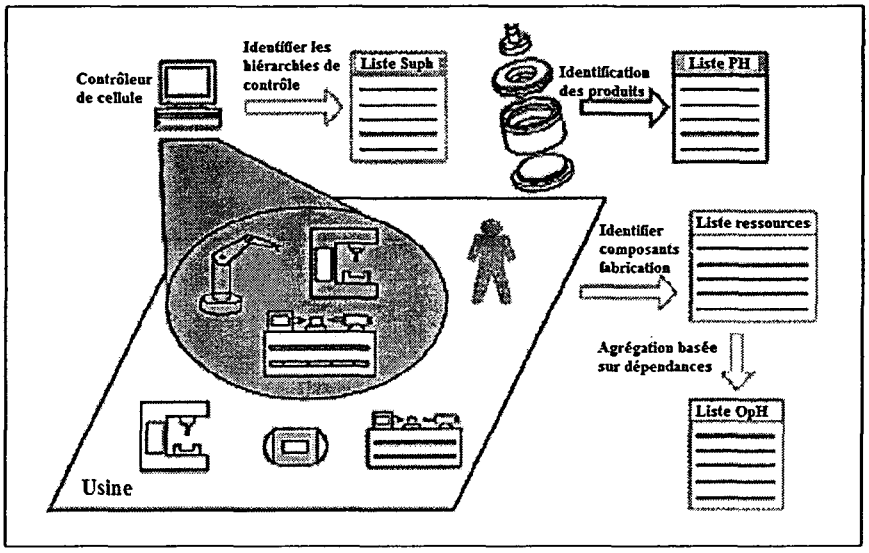


Figure 1.17 : méthodologie de formation d'un holon

Ainsi, il est possible d'obtenir une augmentation de la flexibilité par l'adaptation et la reconfiguration automatique du système. Chaque holon est autonome et peut entrer ou sortir du système à tout moment sans que le système globalement perçoive cette perturbation.

3.3 HCBA

Développée à Cambridge, l'architecture suit une approche ascendante (*bottom-up*) pour le développement. Comme exemple nous présentons ici les éléments de base d'un système de fabrication : produit et ressource.

Le holon ressource (figure1.18) est un élément indépendant du système qui intervient dans le processus de fabrication par : la transformation, l'assemblage, le transport et le test des produits. Des holons ressource typiques sont : les robots, les convoyeurs, les machines outils...

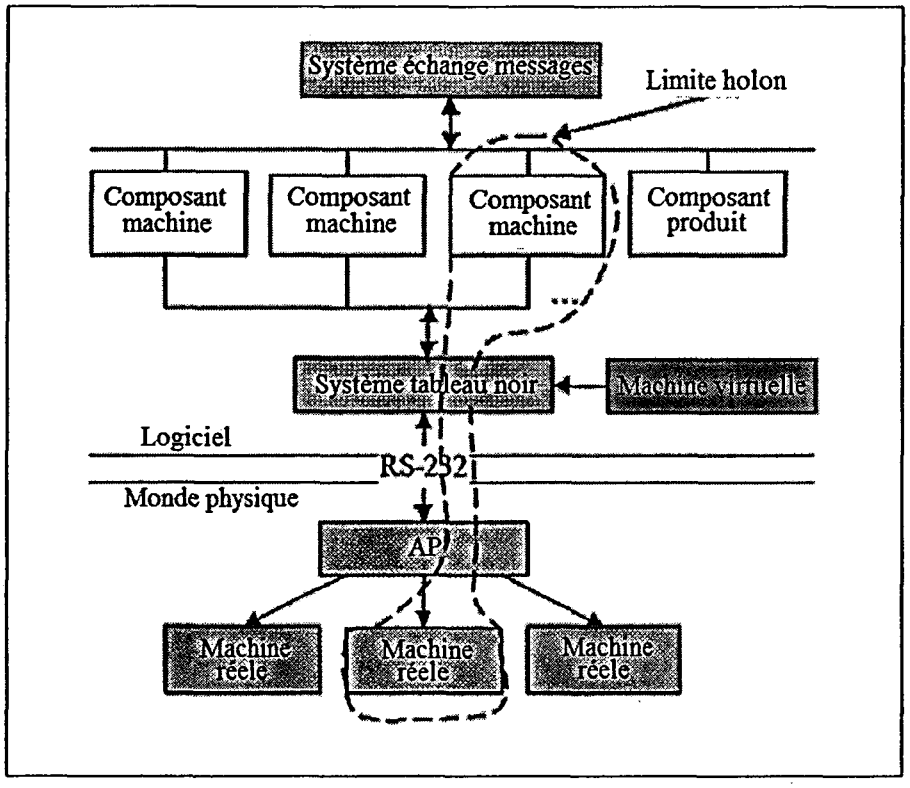


Figure 1.18 : architecture HCBA - Holon ressource

En ce qui concerne le holon produit il est représenté par une entité autonome qui est en charge de son exécution. Il contient les données de la production et décide à l'aide d'un mécanisme basé sur des négociations de l'ordre d'exécution des opérations. Il contient aussi les programmes (ou seulement les paramètres de configuration) qui contrôlent les ressources. Le point fort de l'architecture est la simplification du modèle.

3.4 Synthèse

Bien que plus efficaces en régime de fonctionnement stable, les architectures de pilotage centralisées ont évolué vers des architectures, plus réactives et en mesure de faire face aux perturbations et aux changements de leur environnement actuel de plus en plus incertain. Dans les architectures hétérarchiques et hybrides, une plus grande autonomie décisionnelle est laissée localement aux entités du système, notamment aux représentants des éléments physiques du système opératif. En revanche, la performance locale ne garantit pas nécessairement la performance globale (ce problème, non traité dans cette thèse, se réfère au phénomène de

"myopie" (Zambrano et al., 2011)). Cette baisse d'efficacité est cependant largement compensée par leur réactivité, leur adaptabilité et leur robustesse.

Les architectures hétérarchiques et hybrides sont généralement supportées par un système de traitement de l'information en retrait du système opératif. Ceci conduit à l'obligation de maintenir des liens logiques entre les entités réelles du système physique et leurs images (modèles) au sein du système de traitement de l'information. En cas de rupture de tout ou partie de ces liens, une désynchronisation entre le réel et le « virtuel » (modèles) conduit à un pilotage erroné, voir incohérent, et des dysfonctionnements.

Cependant, les progrès technologiques actuels (ex. optronique, RFID, "smart cards", systèmes embarqués, wifi, ZigBee...) permettent via une instrumentation adaptée de rendre, d'un point de vue décisionnelle, les entités du système, notamment physique, réellement actives. Ces entités, compte tenu d'un objectif, deviennent ainsi capables d'agir en conformité avec l'état réel du système de production. Cette nouvelle capacité décisionnelle peut être incorporée dans le produit ou la ressource elle-même (McFarlane et al., 2002 ; Meyer et al., 2009). Dans ce courant, peuvent être cités les travaux du projet exploratoire SCP "Système Contrôlé par le Produit" (JESA : numéro spécial « Systèmes contrôlés par le produit », 2009) visant à fédérer les efforts de la recherche en France afin d'analyser, d'évaluer la pertinence et la portée du paradigme "système contrôlé par le produit". Ce projet a abouti en 2009 à l'émergence du nouveau groupe de travail IMS2 "Intelligent Manufacturing & Services Systems" du GDR MACS (Groupement de Recherche – Modélisation, Analyse et Commande des Systèmes dynamiques) du CNRS.

Dans ce contexte, visant à rendre autonome les entités du système et directement, en limitant le recours à des représentations intermédiaires nécessitant le maintien logique de liens entre le réel et le virtuel, de nouveaux modèles pour le pilotage sont nécessaires. Cette thèse se positionne dans la proposition d'un modèle adapté à l'élaboration d'architecture de pilotage permettant et facilitant l'intégration de l'autonomie décisionnelle au sein du système, FMS ou PRT, en favorisant, via par exemple les technologies de « l'embarqué », un rapprochement « décision-matière ».

4 Conclusion

Dans ce premier chapitre nous avons tout d'abord présenté les systèmes à flux guidés (SFG) en décrivant deux exemples de tels systèmes : systèmes de type FMS et PRT. Nous avons ensuite précisé la notion de pilotage d'un SFG et plus particulièrement les fonctions d'allocation et de routage. Dans la troisième partie, nous avons montré les différentes architectures et modèles de pilotage existants. Enfin, les architectures holoniques ont été décrites plus en détail. Leur analyse nous conduit, compte tenu des caractéristiques d'un SFG, à proposer, dans le chapitre suivant un modèle holonique générique de pilotage.

Chapitre 2 : Proposition d'un modèle de pilotage holonique d'un Système à Entités Actives (SEA)

Introduction

Ce chapitre propose un modèle holonique d'un système à flux guidé. Nous introduisons tout d'abord la notion d'entité, élément constitutif des deux types de systèmes à flux guidé (FMS et PRT) qui concernent le travail de cette thèse. D'une part dans ces systèmes certaines entités constituent les différents flux physiques ; d'autre part, une entité peut participer au processus décisionnel de pilotage, allocation et routage, des différents flux. La structure générale d'un système à entités actives (SEA) est alors présentée et le concept d'entité active généralisée (EAG) est défini.

Pour élaborer le modèle holonique du SEA, l'holonification du SEA est réalisée en introduisant le holon actif générique (HAG). Le modèle holonique de pilotage d'un SEA (SEAH) est ensuite proposé en présentant dans un premier temps le modèle structurel du SEAH, puis le modèle comportemental.

1 La notion d'entité

1.1 Entités d'un système à flux guidé

Les systèmes à flux guidés que nous considérons sont constitués d'éléments que nous désignons par entité. Ainsi, dans le cas d'un atelier manufacturier flexible (FMS, Flexible Manufacturing System), une palette-support, un centre d'usinage, un convoyeur, un robot manipulateur sont des entités du FMS. Pour un système de transport de type PRT (Personal Rapid Transit), un véhicule, les passagers, la voie, les stations constituent une partie des entités du système. En termes de pilotage (allocation et routage), nous distinguons parmi ces entités, les entités actives et les entités non actives.

- **entité active EA**

Une EA prend de manière autonome des décisions de pilotage, liées aux différents flux du SEA (traitement décisionnel de pilotage) et ne comporte qu'une partie informationnelle. Une EA peut également apporter les supports et services informationnels non décisionnels participant indirectement au processus de pilotage comme le suivi et la traçabilité de son état, du (des) processus associé(s) aux services qu'elle fournit ou reçoit d'une autre entité. Les entités actives constituent le système de pilotage.

- **entité non active ENA**

Dans le cas contraire, pas de prise de décisions de pilotage, l'entité est définie comme entité non active (ENA). Une ENA reçoit des consignes de pilotage d'une EA. Elle possède une partie

physique et, en général, une partie informationnelle. Elle peut prendre de manière autonome des décisions concernant son fonctionnement propre (traitement décisionnel opératif).

Les entités actives et entités non actives constituent les éléments génériques d'un SEA.

Par exemple, dans le cas d'un FMS, une décision de routage doit être prise à un nœud divergent du système de convoyage qui constitue une EA, un robot manipulateur est une ENA. Dans le cas d'un système PRT, le système de guidage est une EA, une rame est une ENA.

1.2 Caractérisation d'une entité

Les différentes entités présentent une forte diversité, notamment par les différentes fonctions qu'elles assurent au sein du système. Pour les caractériser nous retenons trois paramètres : sa nature, sa mobilité et les services (offerts et/ou reçus) qui lui sont associés.

1.2.1 Nature d'une entité

La nature précise si l'entité comporte :

- une partie physique : l'entité est un élément de la structure physique du système (machines, convoyeur, rames PRT, voies...) ; l'entité est désignée comme *entité physique*,
- une partie informationnelle : l'entité est un élément du système informationnel (automates, réseaux de communications, capteurs, PC de commande, bases de données, calculateurs...) ; l'entité est définie comme *entité informationnelle.*

Ainsi, dans le cas général, une EA est de nature informationnelle, une ENA de nature physique et informationnelle.

1.2.2 Mobilité d'une entité

La mobilité caractérise le niveau de mobilité d'une entité. On distingue les entités :

- à mobilité autonome : l'entité peut se déplacer au sein du système par ses propres moyens,
- à mobilité non autonome : l'entité peut être déplacée au sein du système, mais elle ne peut se déplacer seule,
- fixe : l'entité occupe une position déterminée sur un horizon de temps donné sans pouvoir être déplacée.

La mobilité peut être valuée par les paramètres : position, vitesse, accélération.

1.2.3 Services associés aux entités

Les entités offrent et/ou reçoivent des services que nous classons en trois types (Le Moigne, 94) :

- transformation d'**espace** : transformation physique (ex : service de transport, manutention), transformation d'informations (ex : transmission),
- transformation de **temps** : la transformation physique (ex : service de stockage), transformation d'informations (ex : mémorisation de données),

- transformation de **forme** : transformation physique (ex : usinage, assemblage), transformation d'informations (ex : traitement de données).

Ces différentes caractéristiques seront précisées dans le paragraphe suivant.

2 Modèle organisationnel d'un Système à Entités Actives : SEA

2.1 Système piloté et système de pilotage d'un SEA

La structure et les flux d'un SEA sont représentés figure 2-1.

Un SEA comprend :

- l'ensemble P des Entités Actives (EA),
- l'ensemble S des Entités Non Actives (ENA).

P constitue le système de pilotage et S le système piloté.

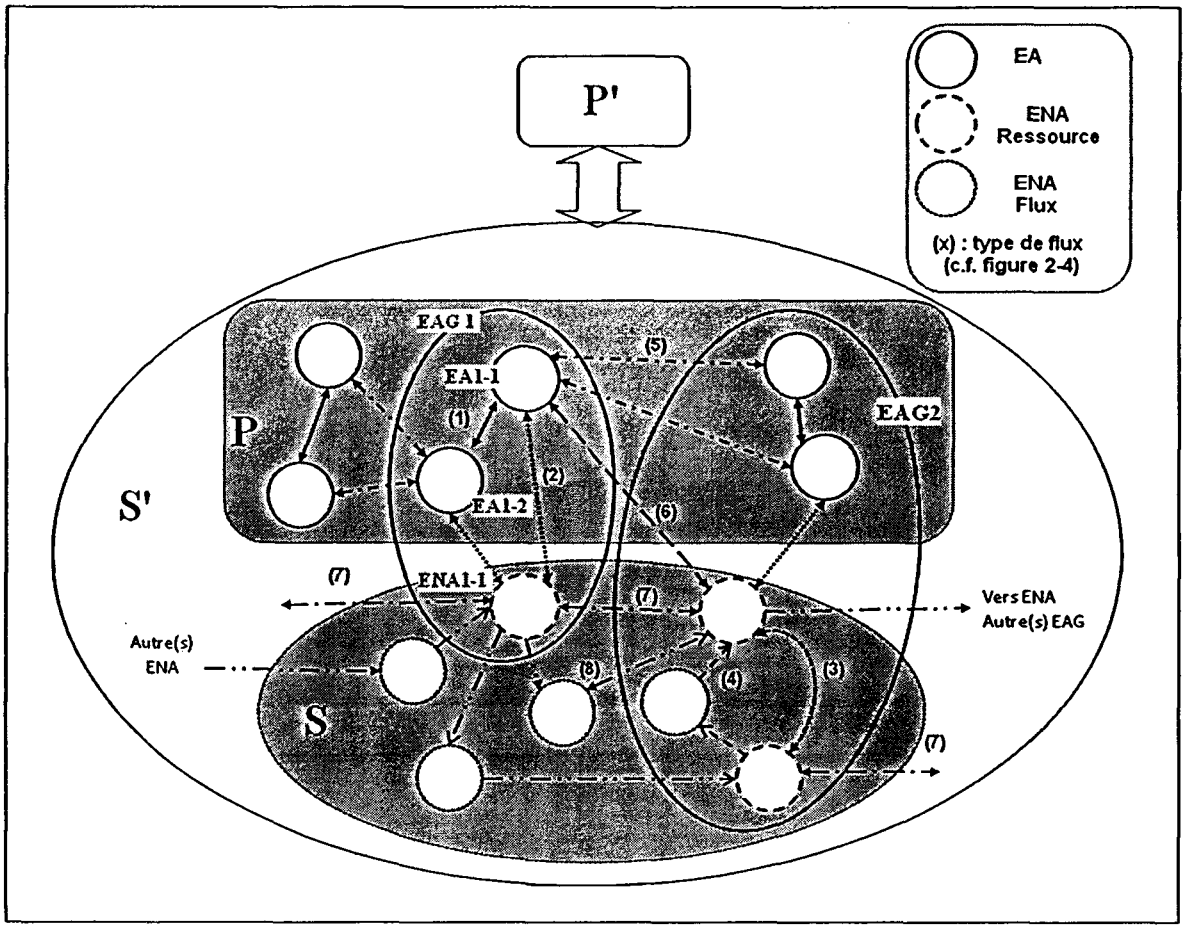


Figure 2.1 : structure et flux d'un SEA

Parmi l'ensemble S, nous distinguons les ENA Flux participant au flux physique et recevant des services fournis par des ENA Ressources. Une ENA Flux dans le domaine des FMS est par exemple un produit en cours de réalisation et, dans le domaine des PRT, une rame ou un véhicule. Cette structure est récursive : le système P' pilote le système S'. S', composition de P et S.

2.2 Entité Active Généralisée (EAG)

Une EA peut, dans le cas d'un SEA, être associée à une ENA pour représenter une entité capable de traitements décisionnels (de pilotage, opérationnels) et une partie physique. Nous définissons alors l'Entité Active Généralisée (EAG). La structure d'une EAG est représentée figure 2-2.

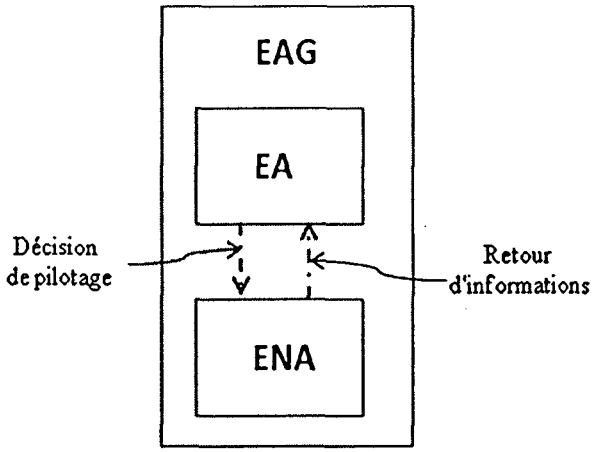


Figure 2.2 : EAG, entité active généralisée

La figure 2-1 illustre le cas d'une EAG 1 composée d'une EA (EA1, par exemple calculateur support de décision de pilotage) et d'une ENA Ressource (ENA1, par exemple un robot). Cette EAG1 permet de modéliser par composition un robot de base capable de prendre part aux décisions de pilotage.

L'entité active, l'entité non active et l'entité active généralisée constituent les éléments fondamentaux d'un SEA.

2.3 Entité Active Généralisée composée (EAG_C)

Dans le cas d'un SEA, plusieurs entités actives et entités non actives peuvent être regroupées. Nous définissons alors l'Entité Active Composée : EAG_C. Une EAG_C est une extension de l'EAG. Elle est constituée de plusieurs EA et ENA (exemples : EAG_C1 et EAG_C2, figure 2-1).

2.4 Caractérisation d'une entité active généralisée

Les notations retenues concernant les différentes caractéristiques d'une entité (nature, mobilité, services) sont décrites ci-dessous.

- Nature

Une EAG hérite des propriétés de ses constituants et une EAG, constituée d'une EA et d'une ENA, est donc toujours informationnelle et physique. La nature d'une EAG notée NAT, est physique, NAT = P et informationnelle, NAT = I. La partie informationnelle peut être décisionnelle et/ou non décisionnelle.

$$\text{soit : } \{NAT\} = \{P, I\}$$

- Mobilité

La mobilité d'une entité, notée MOB, peut être autonome MOB = MA, non autonome MOB = MNA, ou l'entité peut être fixe MOB = F.

soit : {MOB} = {MA, MNA, F}

- Services

Les différents services, notés SERV, cités précédemment sont notés comme suit :

- transformation d'espace : SERV = sep pour la transformation physique (ex : service de transport), sei pour la transformation d'informations (ex : communication),
- transformation de temps : SERV = stp pour la transformation physique (ex : service de stockage), SERV = sti pour la transformation d'informations (ex : mémorisation de données),
- transformations de forme : SERV = sfp pour la transformation physique (ex : usinage, assemblage), SERV = sfi pour la transformation d'informations.

Pour la transformation d'information, service sfi, nous distinguons :

- le sfi non décisionnel, SERV=sfi_T, correspondant au traitement de données (calculs, formatage..),
- le sfi décisionnel de pilotage du système, SERV = sfi_{DP}, c'est-à-dire l'élaboration de décisions de pilotage portant sur l'allocation et de routage, et élaborées de manière autonome par l'entité,
- le sfi décisionnel opérationnel, SERV = sfi_{DO} : les décisions prises, contrairement aux précédentes, ne concernent pas le pilotage, mais assurent l'exécution des consignes de pilotage par la partie physique d'une ENA (choix d'un mode de commande, choix d'un programme de robot...).

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique	sep	stp	sfp
Information	sei	sti	sfi_T, sfi_{DP}, sfi_{DO}

Tableau 2.1 des services offerts / reçus par les entités

- Les services (offerts et/ou reçus) associés désignés par SERV sont ainsi notés :

{SERV} = {sep, sei, stp, sti, sfp, (sfi_T, sfi_{DP}, sfi_{DO})}

- Une entité est donc caractérisée par le triplet : {E} = {NAT, MOB, SERV}

Exemples :

- Un robot de manutention fixe {Rob_Man} sera caractérisé par : {Rob_Man} = {sep, sti, (sfi_T, sfi_{DO})}

- Un calculateur de bord utilisé pour le pilotage d'un véhicule PRT {Cal_PRT} sera caractérisé par :
  {Cal_PRT} = { sei, sti, (sfi_T, sfi_{DP})}

Une EAG peut donc être représentée selon le schéma figure 2.3.

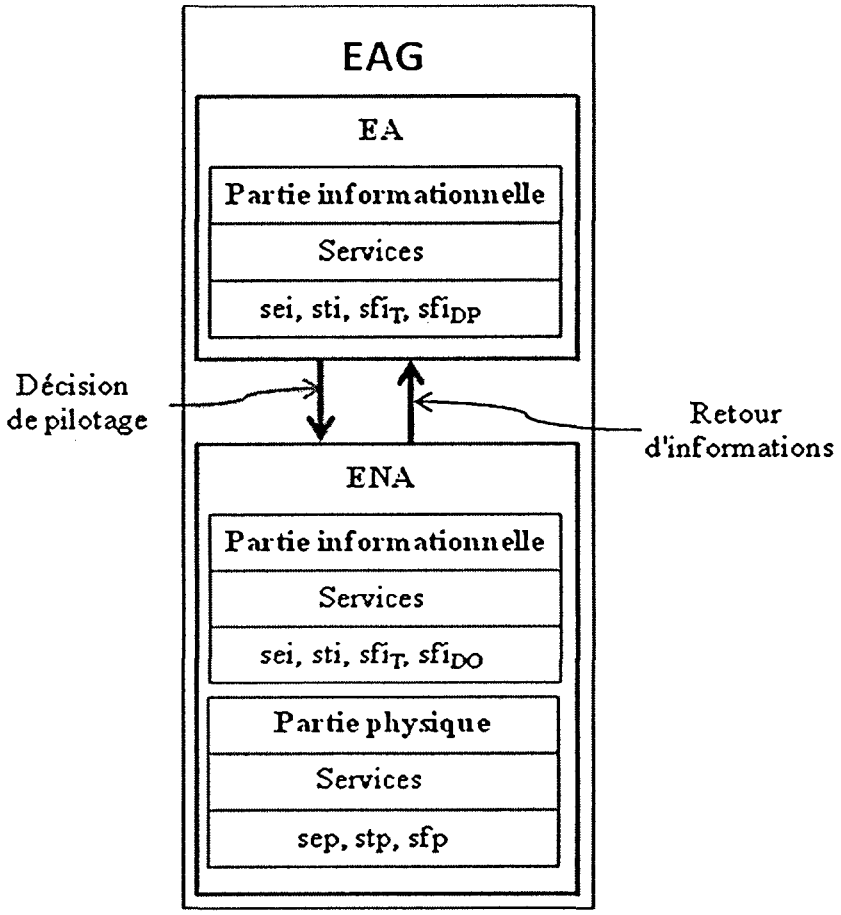


Figure 2.3 EAG - Services offerts/reçus

2.5 Les flux au sein d'un SEA

2.5.1 Nature des flux

Dans un système FMS ou PRT, les flux existant dans le système sont physiquement constitués par certaines entités : les palettes-supports assurent les déplacements des produits d'un poste de travail à un autre, un véhicule PRT se déplace pour réaliser le transport de passagers.

Selon les cas, le flux sera constitué d'entités actives et/ou non actives, le pilotage des flux étant réalisé par les entités actives du système. Ceci s'applique notamment dans le cas d'un flux physique constitué d'éléments prenant part au pilotage.

Un exemple, dans le domaine des FMS, est celui d'un produit en cours de fabrication qui possède des capacités décisionnelles et prend part à son propre pilotage (Berger et al., 2009).

2.5.2 Les différents types de flux

Nous distinguons (figure 2.4) :

- les flux d'informations et les flux physiques,
- les flux internes au sein d'une EAG et les flux externes entre EAG.

Ces différents flux sont représentés figure 2.1.

		Référence du flux	Type de flux
Flux internes	Flux informations	1	Flux entre EA d'une même EAG (ou EAGc)
		2	Flux entre une EA et ENA d'une même EAG (ou EAGc)
		3	Flux entre ENA d'une même EAG (ou EAGc)
	Flux physique	4	Flux entre ENA d'une même EAG (ou EAGc)
Flux externes	Flux informations	5	Flux entre EA appartenant à des EAG (ou EAGc) différentes
		6	Flux entre une EA d'une EAG (ou EAGc) et une ENA d'une EAG (ou EAGc) différente
		7	Flux entre ENA d'EAG (ou EAGc) différentes
	Flux physique	8	Flux entre ENA d'EAG (ou EAGc) différentes

Figure 2.4 : les flux au sein d'un SEA

3 Le modèle holonique d'un SEA (SEAH)

Le modèle holonique d'un SEA proposé (SEAH) est construit en combinant le modèle entité présenté précédemment, la proposition de Christensen (Christensen, 1994) et le modèle holonique de référence PROSA (van Brussel et al., 1998).

La différence majeure entre le modèle PROSA et le modèle holonique SEAH réside dans l'adoption d'un principe systémique de faible couplage et de forte cohésion qui conduit au concept de holon flux.

3.1 Holons actif, non actif et généralisé

L'élaboration du modèle "Holon"ou holonification, s'effectue en associant un holon à une entité (active, non active) du modèle "Entité". Les holons héritent des caractéristiques des entités auxquelles ils sont associés. Nous définissons ainsi le Holon Actif (HA) et le Holon Non Actif (HNA), (Figure 2-5).

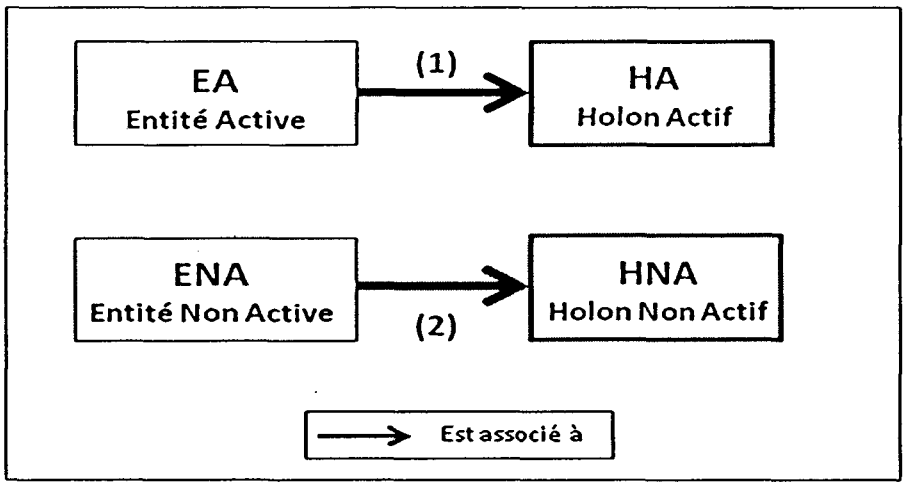


Figure 2.5 : modèle entité – modèle holonique

Remarque : au regard de la définition d'un holon donnée par le consortium Holonic Manufacturing Systems (chapitre 1, §2.5.2), les holons HA et HNA peuvent être considérés comme des "holons complets". Le holon HA dispose d'autonomie (décisionnel de pilotage, par exemple choix d'un trajet pour un véhicule PRT) et peut coopérer avec d'autres holons HA. Le holon HNA dispose d'autonomie (décisionnel opérationnel, par exemple choix d'un programme de commande numérique ou d'outillage pour un centre d'usinage), il peut coopérer avec d'autres holons HNA et il comporte une partie physique.

3.1.1 Le holon actif HA :

Le holon HA (ou holon pilotage) est associé à une entité active EA (association de type (1), figure 2-5). Il possède, dans le cas général, des capacités décisionnelles de pilotage correspondant au service de transformation de forme d'information de type sfi_{DP}.

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique			
Information	sei	sti	sfi_T, sfi_{DP}

3.1.2 Le holon non actif HNA

Le holon HNA (ou holon opérationnel) est associé à une entité non active ENA (association de type (2), figure 2-5) et ne participe pas directement au pilotage. Il comporte toujours une partie physique et possède des capacités de décision opérationnelle correspondant au service de transformation de forme d'information de type sfi_{DO}.

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique	sep	stp	sfp
Information	sei	sti	sfi_T, sfi_{DO}

3.2 Le holon actif généralisé HAG :

Un holon HAG comporte un holon actif HA et un holon non actif HNA piloté par le HA. (Figure 2-6).

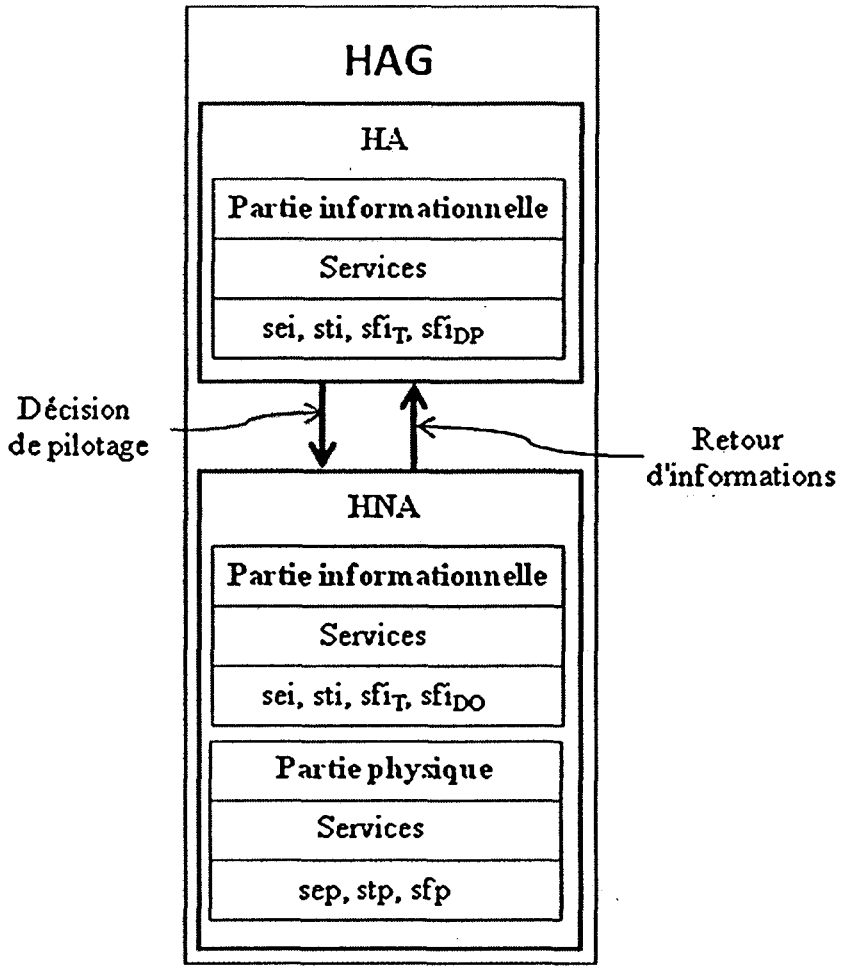


Figure 2.6 : Holon Actif Généralisé

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique	sep	stp	sfp
Information	sei	sti	sfi$_T$, sfi$_{DP}$, sfi$_{DO}$

Les Holons Actif (HA), Non Actif (HNA) et généralisé HAG constituent les éléments génériques du SEAH. Le modèle proposé pour le HAG est décrit dans la partie suivante.

3.3 Modèle du holon actif généralisé

Le modèle décrit figure 2-7 fait apparaître les services et les flux associés à un holon HAG. Les services de type Sti sont désignés par le terme générique "Mémorisation".

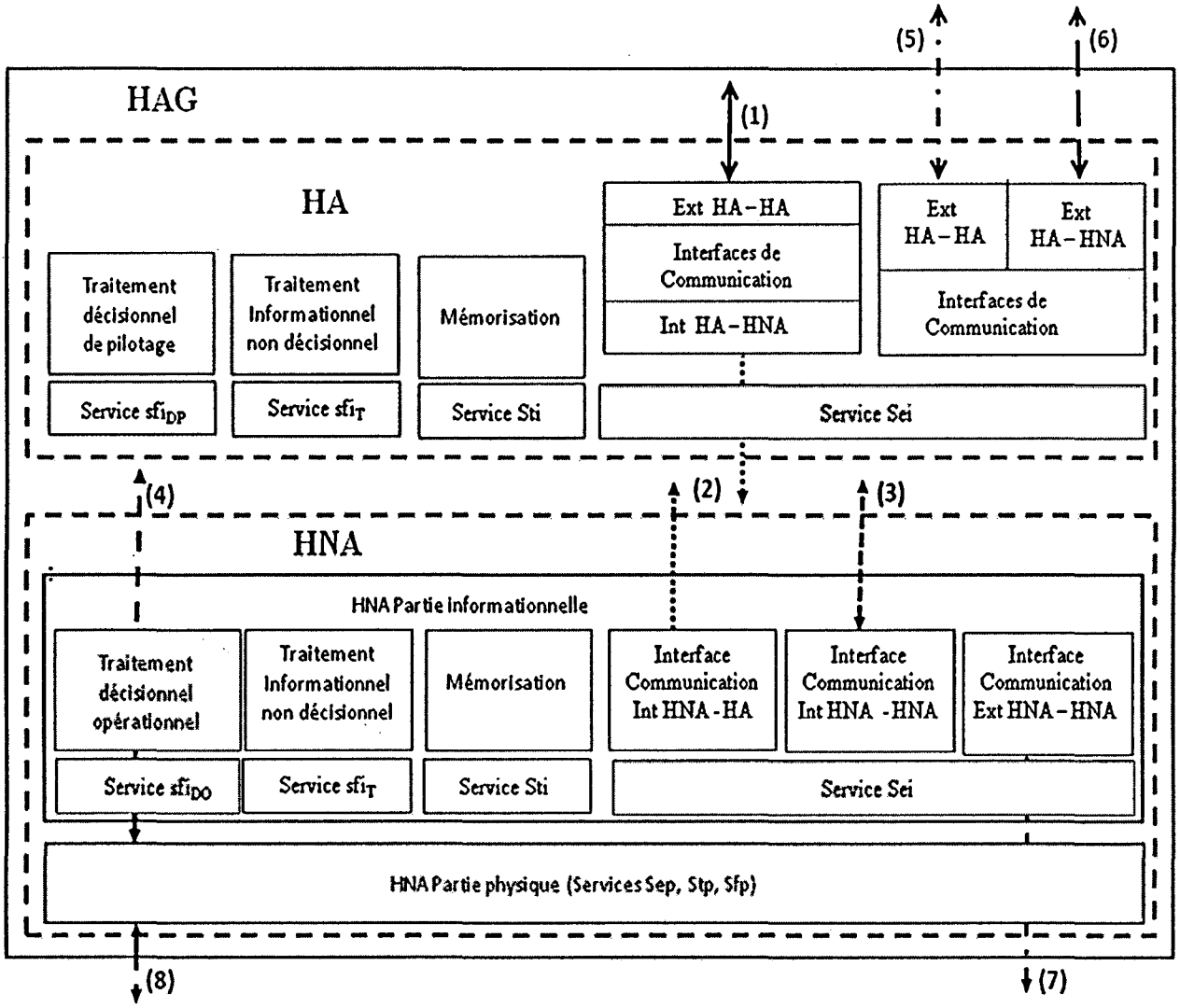


Figure 2.7 : modèle du HAG

Les flux associés aux holons correspondent aux différents flux définis dans le tableau 2-2.

3.4 Le holon HAG composite

Le HAG composite, noté HAG_C, est une extension du HAG. En se plaçant dans la perspective récursive holonique (un holon pouvant être constitué de plusieurs holons) le HAG_C est constitué de HAG et/ou de HAG composite(s).

Le HAG est un cas particulier du HAG composé HAG_C.

		Référence flux	Type de flux	Exemple de flux
Flux internes	Flux informations	1	Flux entre HA d'un même HAG (ou HAGc)	Collaboration entre l'allocation et le routage d'un produit
		2	Flux entre un HA et HNA d'un même HAG (ou HAGc)	PC de pilotage d'un nœud d'aiguillage divergent d'un convoyeur
		3	Flux entre HNA d'un même HAG (ou HAGc)	Collaboration de deux robots partageant un même espace de travail
	Flux physique	4	Flux entre HNA d'un même HAG (ou HAGc)	Flux de pièces convoyées d'une machine à une autre d'un même îlot
Flux externes	Flux informations	5	Flux entre HA appartenant à des HAG (ou HAGc) différents	Communications de coordination entre systèmes de conduite de deux zones de trafic d'un PRT
		6	Flux entre un HA d'un HAG (ou HAGc) et un HNA d'un HAG (ou HAGc) différent	Produit actif qui demande à un aiguillage d'établir un chemin particulier
		7	Flux entre HNA d'HAG (ou HAGc) différents	Synchronisation entre deux palettes pour éviter de rentrer en collision
	Flux physique	8	Flux entre HNA d'HAG (ou HAGc) différents	Flux de pièces convoyées d'une machine à une autre d'îlots différents

Tableau 2.2 : flux et exemples au sein d'un SEAH

- **Exemples de HAG et HAG_C :**

  - **Exemple 1 :**

La figure 2.8 - a représente une cellule flexible comportant trois centres d'usinage notés CUi (i = 1, 2, 3),

  - Les centre d'usinage, possèdent chacun une zone de stockage en entrée ZSei et en sortie ZSsi. Les centres CU1 et CU2 réalisent le même groupe d'opérations sur un ensemble de produits.

- Les produits à usiner, placés sur des palettes, sont des produits actifs, c'est-à-dire capables de prendre des décisions concernant leur trajectoire en certains points de la cellule (ici, NAe et NAs).

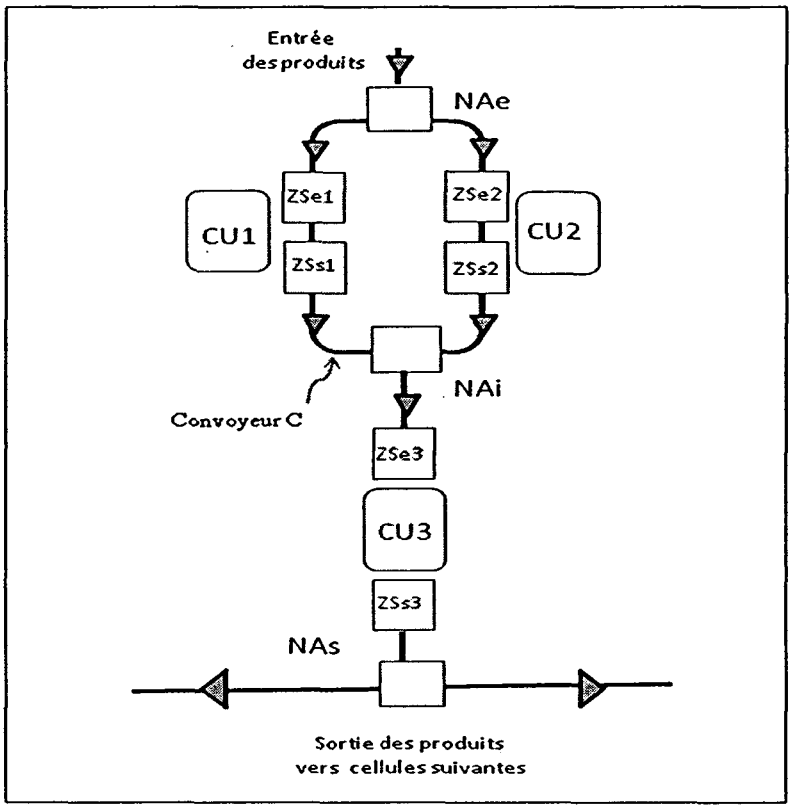


Figure 2.8 - a : cellule flexible

- Circulation des produits :

- Les produits pénètrent dans la cellule par le nœud d'aiguillage NAe qui les dirige vers CU1 ou CU2 selon le taux de remplissage des zones de stockage en entrée ZSe1 et ZSe2.

- Ils sont acheminés à travers la cellule par le convoyeur C.

- A la sortie de CU1 ou CU2 les produits se dirigent vers la zone de stockage en entrée ZSe3 de CU3, le nœud convergent intermédiaire NAi assure le transit des palettes.

- Le centre CU3 réalise la finition sur ces pièces.

- En sortie de CU3, les produits s'orientent vers l'une des deux cellules de sortie. Le nœud NAs assurent le transfert des palettes du convoyeur C sur le convoyeur de sortie Cs.

Le modèle holonique se construit comme suit, figure 2.8 - b :

- l'ensemble de la cellule constitue un holon actif généralisé nommé HAG0, elle comporte :

- le holon HAGe correspond au nœud d'aiguillage des produits vers CU1 ou CU2, il comporte le holon de pilotage de l'aiguillage HAe qui va décider de l'affectation des produits à CU1 ou CU2 et le HNAe, holon d'aiguillage des produits,

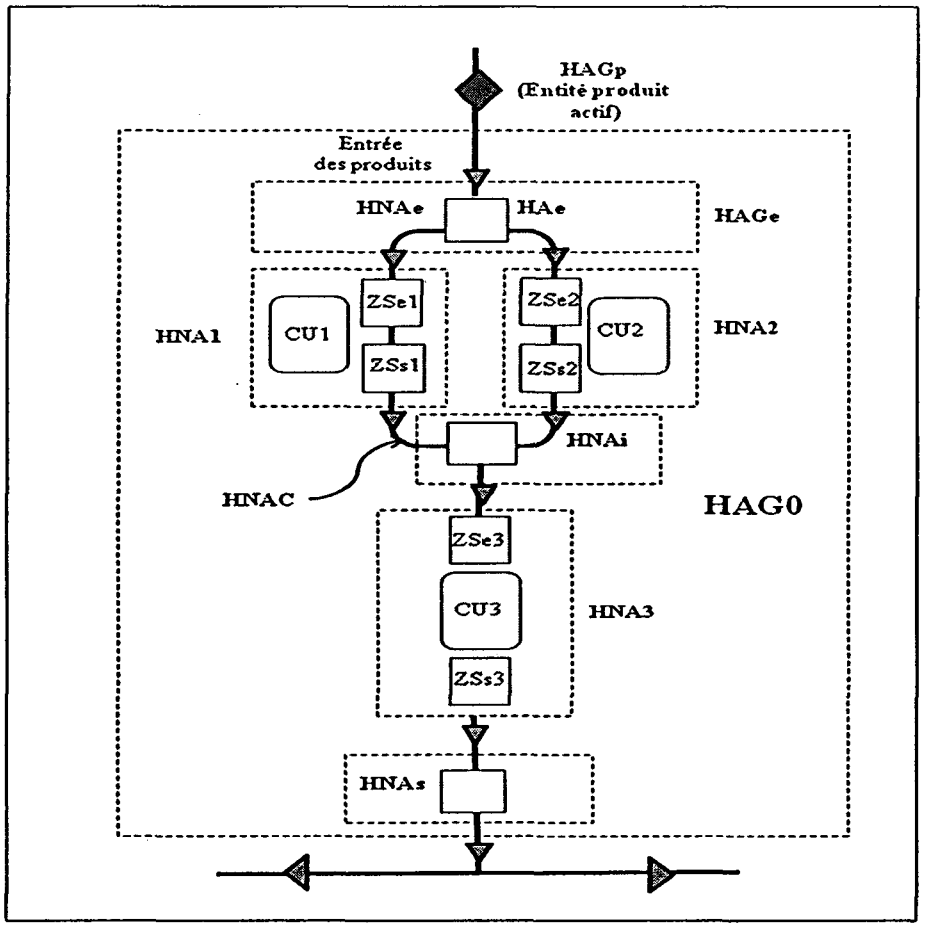


Figure 2.8 – b : modèle holonique de la cellule flexible exemple

- 5 holons HNA :

  - HNA1, HNA2, HNA3 : holons associés aux centres d'usinage,

  - HNAi : holons associé au nœud convergent NAi chargé de déplacer les produits vers la zone de stockage en entrée ZSe3 de CU3,

  - HNAs : holon associé au système de transfert des produits sur la zone ZSe3.

  Les convoyeurs assurant le transport des pièces sont modélisés par des HNA (HNAC)

  - Les produits actifs sont des holons actifs généralisés notées HAGp sur la figure 2.8 - b. Ils pilotent les nœuds NAe et NAs.

- **Exemple 2 :**

En considérant qu'un réseau PRT est constitué de secteurs (géographiques, fonctionnels...), la figure 2.8 - c représente un réseau PRT constitué de 4 secteurs. Le système de pilotage est un système hybride.

Chaque secteur comporte :

- un (ou plusieurs) terminal (T) correspondant à une extrémité d'une ligne du réseau,

- un (ou plusieurs) pôle d'échange (PE), point de correspondance entre plusieurs lignes,

- un (ou plusieurs) aiguillages (Aig).

- Un poste de pilotage centralisé du réseau (PPCR) : il détermine notamment l'allocation des PRT. Il est ici localisé au sein du pôle d'échange PE1-1 du secteur 1.

Les tronçons de voie constituant l'infrastructure du réseau sont des holons HNA. Pour des raisons de lisibilité, ils ne sont pas représentés sur la figure 2.8 - d.

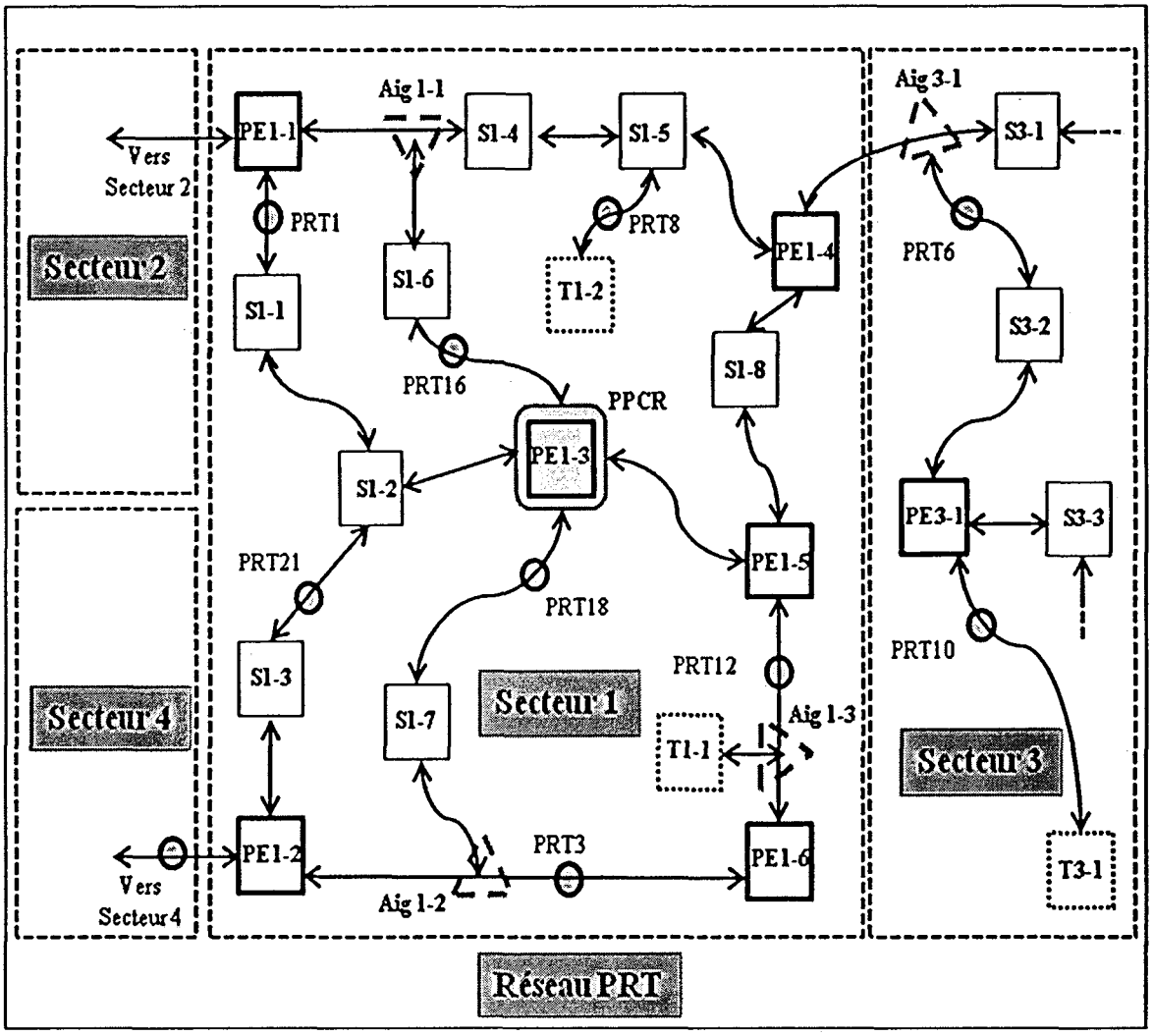


Figure 2.8 - c : réseau PRT

Le terme PRT désigne ici le véhicule physique et l'ensemble des systèmes de traitement décisionnel et non décisionnel embarqués sur le véhicule. Les PRT peuvent, suivant leur allocation, circuler sur l'ensemble du réseau. Ils assurent le pilotage dynamique pour, par exemple, optimiser les correspondances.

Le modèle holonique correspondant est représenté figure 2.8 – d.

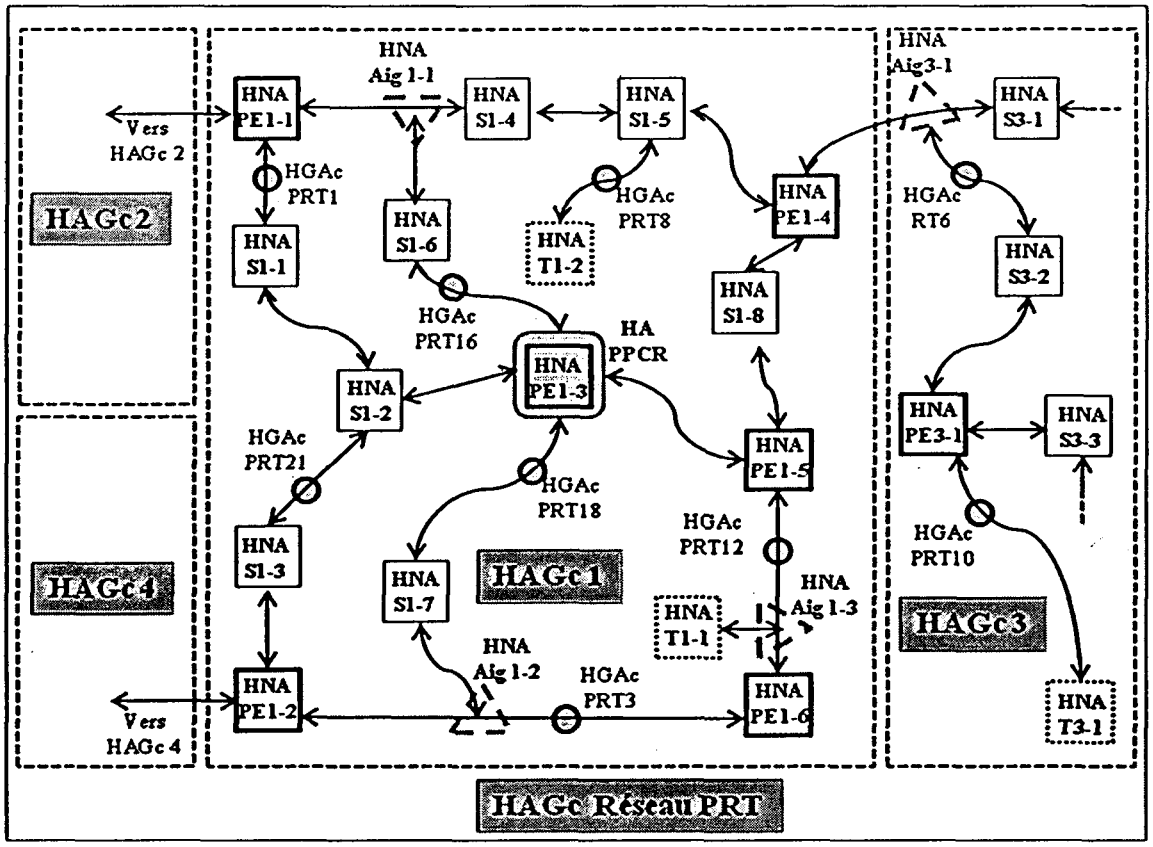


Figure 2.8 - d : modèle holonique du réseau PRT

3.5 Holons non-complets

Nous désignons par "holon non-complet" un élément dérivé d'un holon complet, holon HA ou holon HNA, mais ne possédant pas l'ensemble de leurs caractéristiques propres. Il ne constitue donc pas à un holon complet. Cependant, on trouve dans la littérature d'autres exemples ou le terme holon est utilisé pour désigner un système privé en partie ou totalement d'autonomie et de coopération. Par exemple le holon "Product" de PROSA (Brussel et al., 1998) qui ne représente que de l'information ou de la connaissance.

3.5.1 Le holon "HA restreint" HA_R

Le holon HA_R, figure 2.9 - a, assure les services de transformation d'information sei, sti, et sfi_T mais pas le service sfi_{DP}. Il ne participe pas au pilotage proprement dit, mais il constitue un soutien informationnel utilisé par un ou plusieurs HA (ex : serveur de données).

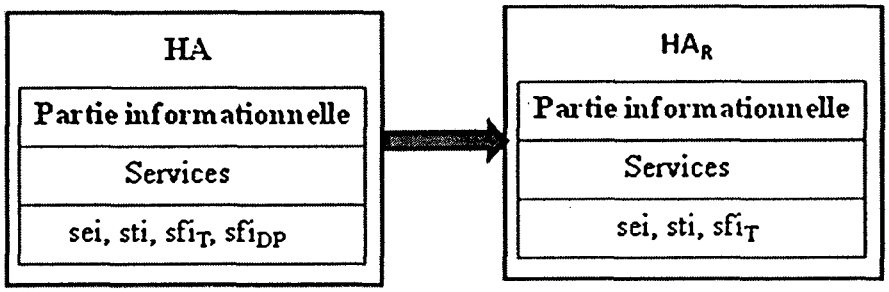


Figure 2.9 - a : holon HA_R

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique			
Information	sei	sti	sfi_T

3.5.2 Le holon Objet Receveur des Services, HNA_{ORS} :

Le holon HNA_{ORS}, figure 2.9 – b, est un holon HNA qui ne comporte qu'une partie physique. Il est considéré comme passif et ne fournit aucun service.

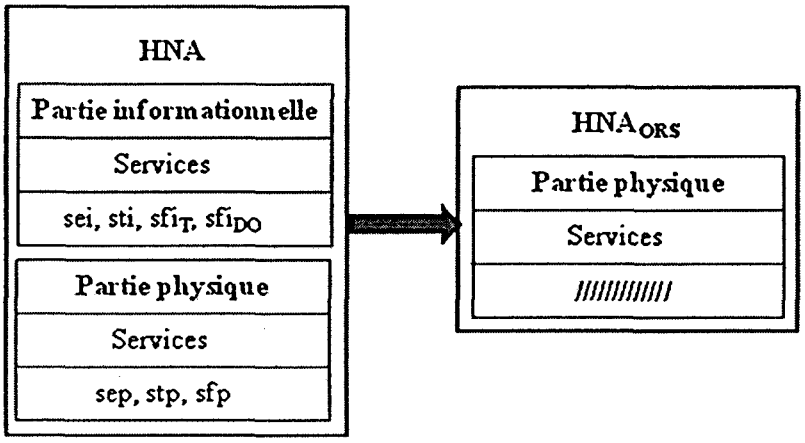


Figure 2.9 - b : holon HNA_{ORS}

L'Objet Receveur des Services peut être une pièce usinée, un passager d'un PRT.

Remarque : cependant, dans certains cas, un objet ne subissant aucune transformation de forme peut intervenir sur le pilotage des flux. Par exemple, dans le cas d'un FMS, des containers peuvent, pour des raisons de sécurité (ex : température trop élevée), alerter pour que la disposition du stock soit modifiée. Dans le cas du PRT un passager peut actionner un arrêt d'urgence et stopper la rame, bloquant ainsi le trafic. Ces cas particuliers ne sont pas abordés dans le travail présenté ici.

4 Modèle structurel des holons du SEAH

4.1 Modèle général

La structure du modèle holonique a été élaborée avec pour objectifs :

- de pouvoir modéliser explicitement les flux du SEA et leur caractère dynamique. Le "Holon Flux" est ainsi caractérisé par son cycle de vie correspondant à un processus de construction dynamique activé en fonction des besoins à satisfaire en termes de services,

- de préserver une cohérence sémantique avec les modèles holoniques existant (notamment le modèle PROSA),

- d'obtenir l'exhaustivité nécessaire pour décrire les différents types de systèmes à flux guidés,

- d'assurer la transparence du modèle obtenu pouvant être compris par un large éventail d'utilisateurs,

- de garantir la souplesse et la maniabilité nécessaires à une utilisation simple et facilitant les modifications.

Le modèle holonique du SEA comporte ainsi (figure 2.10) :

- *une structure statique (1)* composée d'un holon coordinateur HC, d'un (ou plusieurs) holon Ressource HR et d'un holon Connaissances Système (ex : produits et services),
- *une structure dynamique (2)* comportant les holons Objet Receveur des Services HORS, Services à obtenir HSO, Mobilité HM et Conduite HCO qui constituent le holon Flux HF. Le holon flux est capable, en toute autonomie, de prendre en charge la fabrication d'un produit au sein d'un FMS, ou, dans le domaine des PRT, d'assurer le transport de bout en bout de la course d'un usager.

Que ce soit dans le cas d'un FMS ou d'un PRT, le flux, habituellement passif, est ici explicite et actif au regard du pilotage.

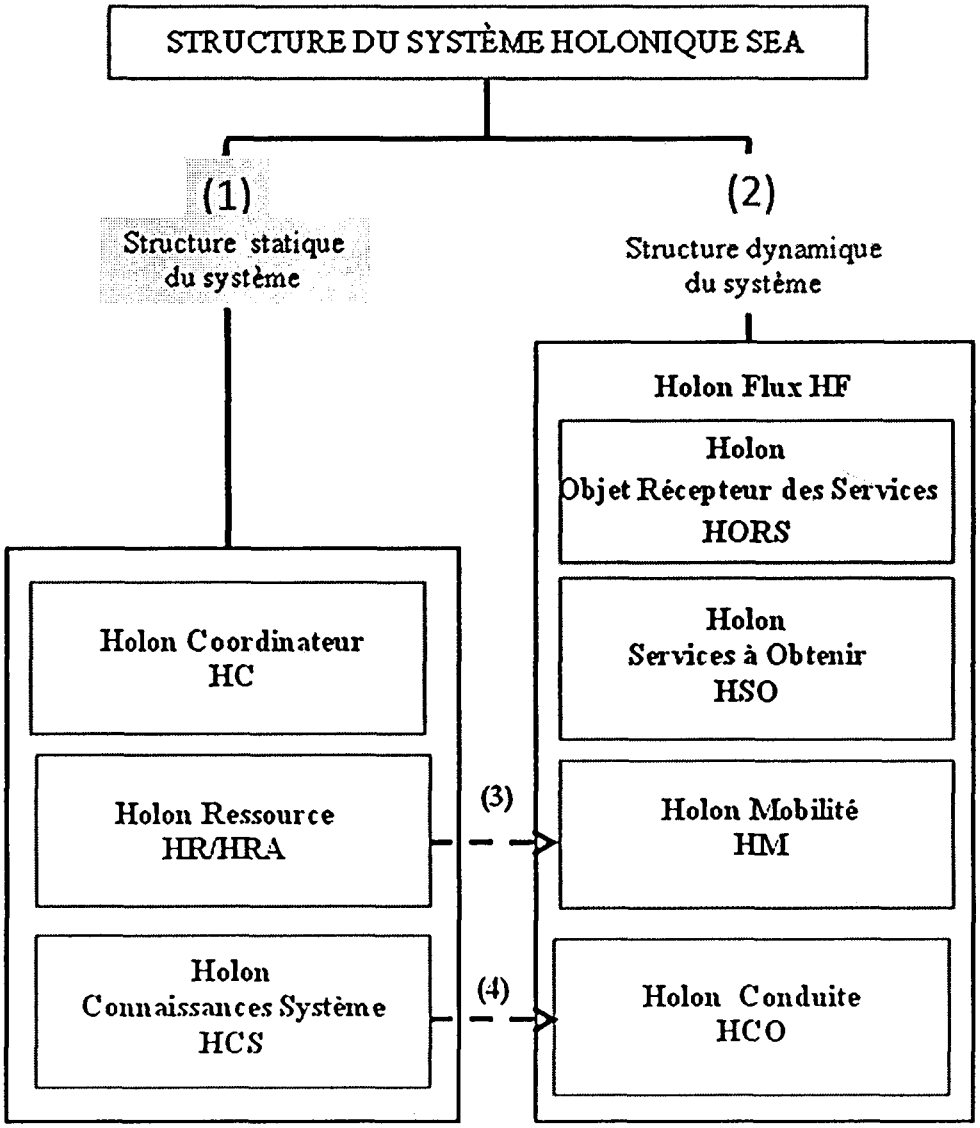


Figure 2.10 : SEAH - Modèle structurel holonique d'un SEA

Les différents types de holons sont décrits dans les paragraphes suivants.

4.2 Structure statique

4.2.1 Holon Coordinateur (HC)

Le holon coordinateur HC est un holon de type HA. Il a en charge le maintien de la cohérence de l'ensemble des services fournis au sein du SEA. Il optimise globalement l'obtention des services (lots de production en cours dans un FMS, ensemble des tâches de transport dans un PRT). Il fournit des services de type sei, sti, sfi_T, sfi_{DP}, Dans le modèle PROSA, il correspond au holon "staff".

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique			
Information	sei	sti	sfi_T, sfi_{DP}

4.2.2 Holon Ressource (HR / HRA)

Le holon Ressource est, selon le type de système de pilotage, centralisé ou hybride, un holon de type HNA (HR) ou HAG (HRA : Holon Ressource Actif). Il intervient dans le pilotage dans le cas d'un holon HRA. Nous considérons par ailleurs qu'un constituant d'une infrastructure (convoyeur, rail de guidage, poteau support de voie de circulation,....) est un holon HR capable de fournir des informations (sur son propre état ou sur la présence d'éléments étrangers au système...).

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique	sep	stp	sfp
Information	sei	sti	sfi_T, sfi^*

- Pilotage centralisé (Holon HNA- HR) : sfi* : sfi_{DO}

- Pilotage hybride (Holon HAG- HRA) : sfi* : sfi_{DP}, sfi_{DO}

Les informations caractérisant le holon HR comprennent par exemple l'identificateur de la ressource, les services disponibles et les informations associées à chacun de ces services offerts par le holon HR, l'historique des services offerts au holon HF par le holon HR considéré, le modèle du système, mis à jour en permanence, l'état des ressources fournissant les services (libre, occupée, réservée, hors-ligne, taux de charge).

4.2.3 Holon Connaissances Système (HCS)

Le holon HCS est un holon de type HA_R. C'est une extension du holon "Product" du modèle PROSA. Il contient l'ensemble des éléments caractérisant les différents éléments du système de production de type FMS (machines d'usinage, robots, convoyeurs, shuttles...) ou de transport de type PRT (véhicules, stations, voies...).

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique			
Information	sei	sti	sfi_T

4.3 Structure dynamique

Le holon flux est un holon composite HAG_C constitué dynamiquement. Il est constitué des holons HORS, HSO, HM et HCO. Nous décrivons ces 4 types de holons et donnons en synthèse la description du holon HF (figure 2-11).

4.3.1 Holon Objet Receveur des Service (HORS)

Le holon HORS est un holon de type HNA_{ORS}. Il modélise une entité passive (pièce usinée, passager d'une rame...) et constitue un des éléments du flux physique du système. Il est uniquement consommateur des services offerts par les holons HR.

4.3.2 Holon Services à Obtenir (HSO)

Le holon HSO est un holon de type HA_R. Il contient la liste des services à obtenir par le holon flux successivement auprès des holons HR.

Le holon HSO est associé aux spécifications des processus techniques Il contient l'ensemble des informations qui concernent les services nécessaires (contraintes techniques, indices de qualité des processus, paramétrage des ressources...).

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique			
Information	sei	sti	sfi_T

Les informations attachées au holon HSO comportent notamment les services qui doivent être offerts pour satisfaire la demande du client (réaliser un produit, effectuer un transport), les contraintes liées à ces services (contraintes de précédence, d'exclusion...), l'historique de l'obtention des services et de l'utilisation des ressources qui offrent ces services. Ces informations peuvent être exploitées pour l'évaluation des performances.

4.3.3 Le holon mobilité (HM)

Le holon mobilité HM est un holon de type HR (lien 3 la figure 2-10) qui offre des services de transport au holon flux. Il est, comme le holon HORS, un des éléments du flux physique du système.

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique	sep	stp	
Information	sei	sti	sfi_T, sfi*

(Pilotage centralisé (Holon HNA- HR) : sfi* : sfi_{DO}, - Pilotage hybride (Holon HAG- HRA) : sfi* : sfi_{DP}, sfi_{DO})

4.3.4 Holon conduite (HCO)

Le holon HCO est un holon de type HA. Il assure les tâches de : suivi de l'état d'avancement des services, choix des ressources, lancement, interruption, reprise de service. Il s'apparente au holon "order" du modèle PROSA. Il exploite (lien 4 la figure 2-10) certaines informations du HCS (caractéristiques des machines d'usinage, robots, convoyeurs, shuttles, véhicules, caractéristiques voies...).

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique			
Information	sei	sti	sfi_T, sfi_{DP}

Les quatre holons précédents sont les composants du holon Flux HF que nous décrivons ci-dessous.

4.4 Le holon Flux (HF)

Comme illustré figure 2-10, le holon flux HF est constitué :
- du Holon Conduite : HCO,
- du Holon Mobilité : HM,
- du Holon Services à Obtenir : HSO,
- du Holon Objet Receveur des Services : HORS.

Le holon HF est un holon composite et donc, comprend potentiellement la totalité des services.

	Transformation d'espace	Transformation de temps	Transformation de forme
Physique	sep	stp	sfp
Information	sei	sti	sfi_T, sfi_{DP}, sfi_{DO}

4.5 Cycle de vie du holon flux HF

Le Holon Flux HF peut être initialement composé de tout ou partie des holons HORS, HSO, HM et HCO. La figure 2.11 met en évidence la composition du holon HF, correspondant à l'agrégation, et son "éclatement "qui correspond à la désagrégation du holon HF.

La figure 2.11 montre également que, suite à l'agrégation, le holon HF se déplace (grâce au holon HM) auprès des différents holons HR nécessaires (dans la figure 2.9 de HR_1 pour le premier service à HR_N pour le dernier service) pour que le holon HORS ("embarqué" sur le holon HF) bénéficie des services souhaités listés et contenus dans le holon HSO. Le contrôle du processus est assuré par le holon HCO en charge du holon HF.

L'agrégation et la désagrégation du holon HF se manifeste physiquement par l'agrégation et la désagrégation partielles ou totales des supports des holons composant un holon HF. Deux exemples illustratifs sont présentés ci-dessous, l'un relatif au domaine des FMS, l'autre au domaine des PRT.

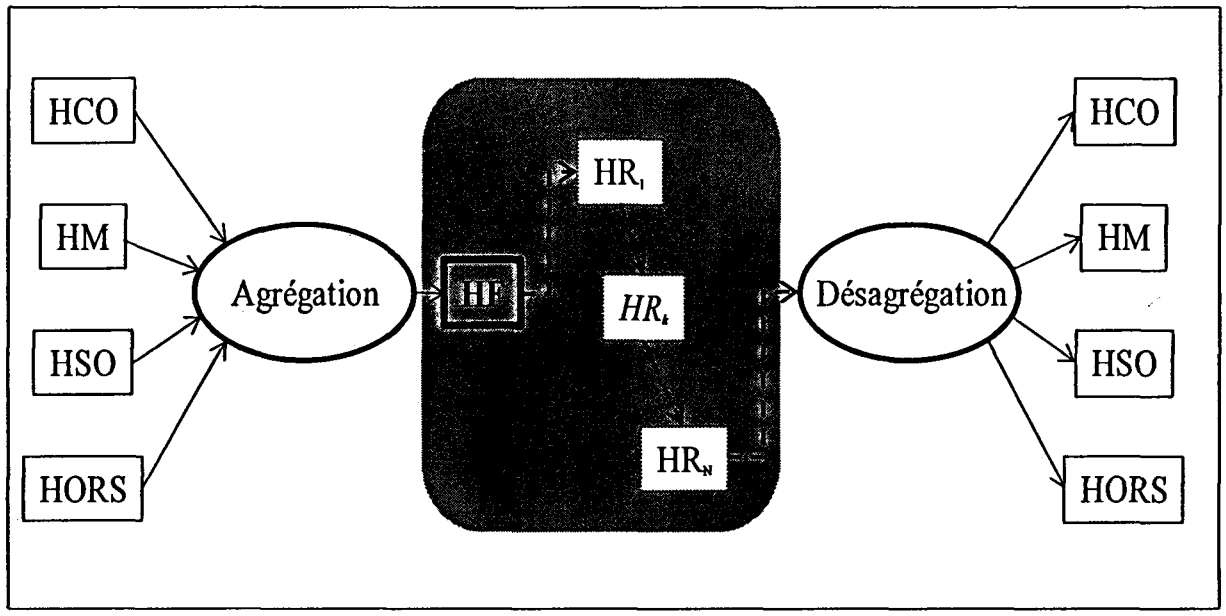


Figure 2.11 : Cycle de vie du holon HF

4.5.1 Exemple 1 : cas d'un FMS

La figure 2.12 présente une réalisation expérimentale d'un holon HF dans le domaine des FMS (processus d'assemblage).

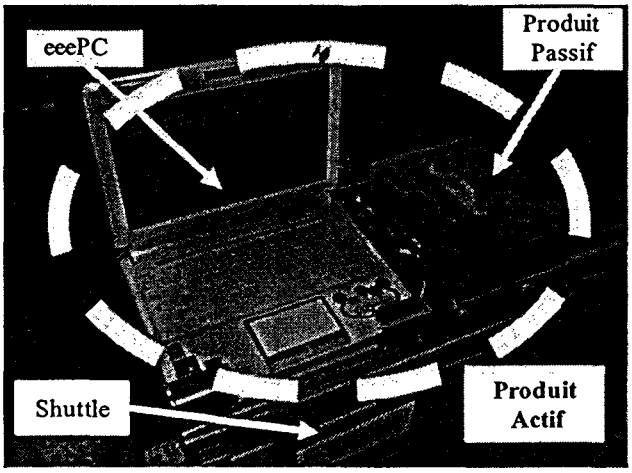


Figure 2.12 : Exemple de réalisation d'un holon HF (FMS)

Dans cet exemple un "eeePC" (netbook) permet de traiter, mémoriser et communiquer de l'information. Le "Produit Passif "correspond au produit à fabriquer ou en cours de fabrication. Le "Shuttle" permet la mobilité de l'ensemble ("eeePC"+"Produit Passif").

L'ensemble ("eeePC"+"Shuttle" +"Produit Passif") est vu comme un "Produit Actif" assurant le pilotage de la fabrication du "Produit Passif "et correspond précisément à une mise en œuvre du holon HF dans le domaine des FMS.

Des détails relatifs à la notion de "Produit Actif" peuvent être consultés dans les publications de l'équipe Tempo-PSI, par exemple (Sallez et al., 2010).

Dans cette vue, figure 2.13, le "Produit Passif" désigne la base-support du produit qui au fil du processus d'assemblage accueille les différents composants.

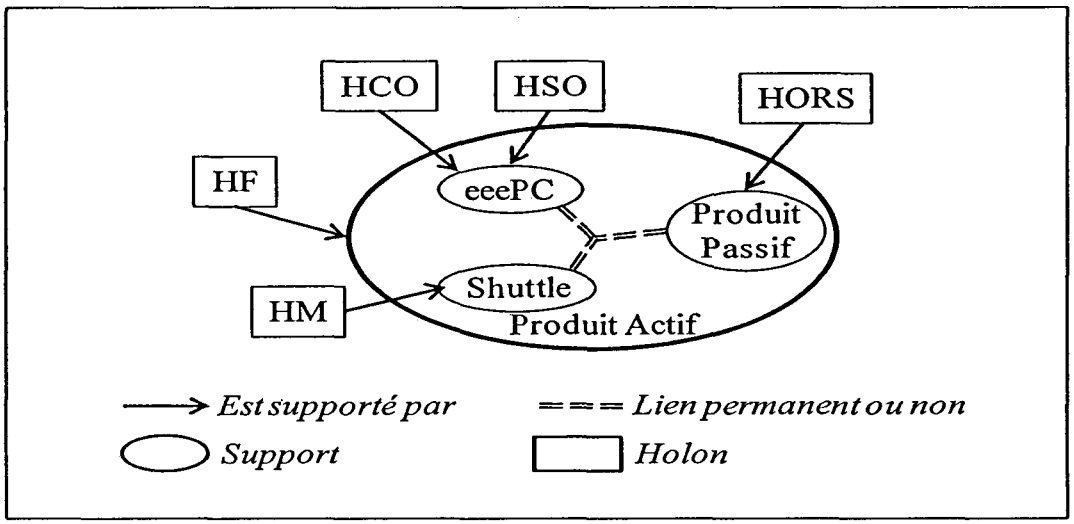


Figure 2.13 : holons et supports associés

La figure 2.14 illustre le cycle de vie du holon HF via son support composé, i.e. le "Produit Actif".

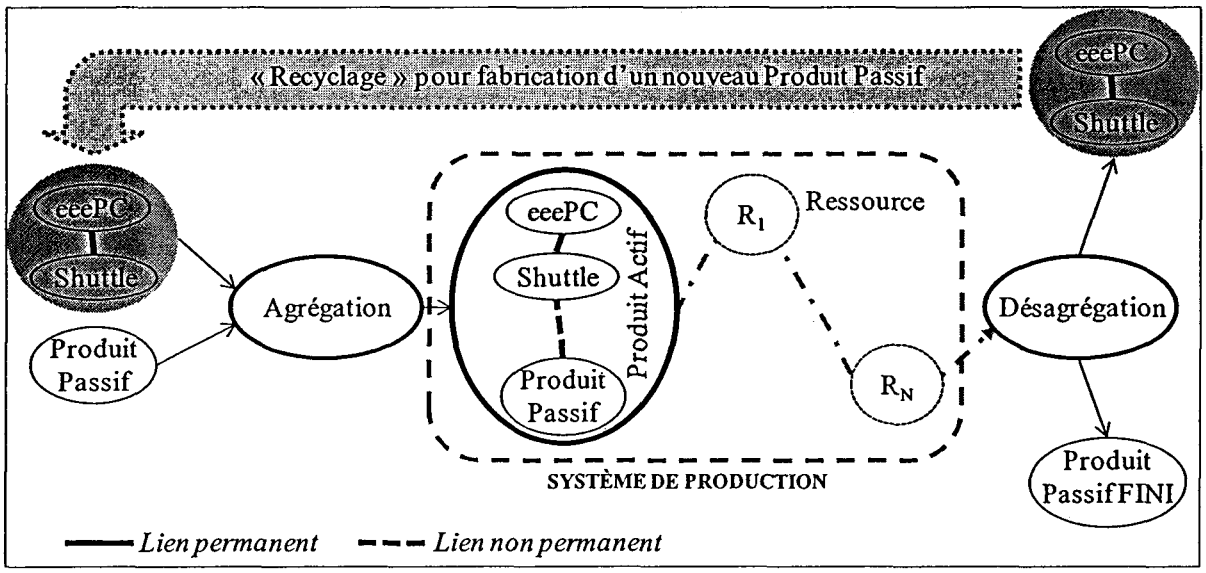


Figure 2.14 : exemple de cycle de vie du "Produit Actif "support du holon HF

Dans l'exemple de la figure 2.14, R_1 à R_N correspondent aux ressources (ex : d'assemblage, d'usinage) "traversées "par le "Produit Actif" pour obtenir finalement le "Produit Passif FINI". Suite à la désagrégation du "Produit Actif", le "Produit Passif FINI" correspond au produit

fabriqué souhaité, et l'ensemble ("eeePC"+"Shuttle") peut être "Recyclé" pour fabriquer un nouveau produit.

4.5.2 Exemple 2 : cas du PRT

La figure 2.15 présente, dans le domaine des PRT, un exemple de cycle de vie du holon HF via les supports qui permettent de le mettre en œuvre.

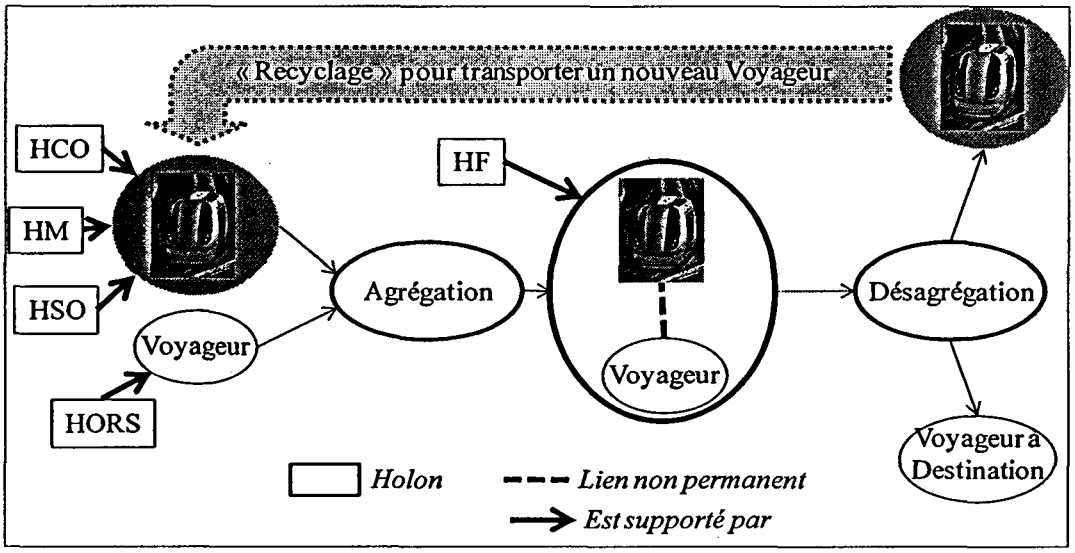


Figure 2.15 : exemple de réalisation d'un holon HF (PRT)

Dans cet exemple, le "Voyageur" (ou groupe limité de voyageurs) correspond au holon HORS. Le "Voyageur" bénéficie d'un service de transport. Les holons HCO, HSO et HM sont supportés par le véhicule PRT, son électronique et ses calculateurs embarqués. HSO contient la destination souhaitée par le "Voyageur". L'agrégation est faite au niveau d'une ressource (station) permettant au "Voyageur" de monter dans le PRT. La désagrégation est faite également au niveau d'une ressource (station), mais celle-ci permet au "Voyageur" de descendre du PRT.

4.6 Synthèse

Les tableaux 2.4 et 2.5 présentent une synthèse des différents holons du modèle holonique d'un SEAH.

	Holons du SEAH											
Nom	Type	Caractéristiques	Nature([1])	MOB([1])	Services							
					Sep	Sei	Stp	Sti	Sfp	Sfi_T	Sfi_{DP}	Sfi_{DO}
Holons de base												
Holon Actif	HA	Holon générique	I	MNA, F		X		X		X	X	
Holon Non Actif	HNA	Holon générique	P, I	MA, MNA, F	X	X	X	X	X	X		X
Holon Actif Généralisé	HAG	Composé d'un HA et d'un HNA	P, I	MA, MNA, F	X	X	X	X	X	X	X	X
Holon Généralisé Composé	HAG_C	Peut comporter plusieurs HAG et/ou HAG_C	P, I	MA, MNA, F	X	X	X	X	X	X	X	X
Holons particuliers												
Holon Actif Restreint	HA_R	HA qui ne fournit pas de service Sfi_{DP}	I	MNA, F		X		X		X		
Holon HNA physique	HNA_{ORS}	HNA qui ne comporte qu'une partie physique	P	MNA								

Tableau 2.4 : holons fondamentaux et holons non-complets

Nom	Type	Intitulé	Nature[1]	MOB[1]		Services						
					Sep	Sei	Stp	Sti	Sfp	Sfi_T	Sfi_{DP}	Sfi_{DO}
Structure statique												
Holon Coordinateur	HA	HC	I	F		X		X		X	X	
Holon Ressource	HNA / HAG[2]	HR / HRA[2]	P, I	MA, MNA, F	X	X	X	X	X	X	X[2]	X
Holon Connaissances Système	HA_R	HCS	I	MNA, F		X		X		X		
Structure dynamique												
Holon HNA physique	HNA_{ORS}	HORS	P	MNA								
Holon Service à Obtenir	HA_R	HSO	I	MNA		X		X		X		
Holon Mobilité	HNA / HAG[2]	HM	P, I	MA	X	X	X	X	X	X	X[2]	X
Holon Conduite	HA	HCO	I	F		X		X		X	X	
Holon Flux	HAG_C	HF	P, I	MA	X	X	X	X	X	X	X	X

Tableau 2.5: tableau de synthèse des holons d'un SEAH

(1) : la nature et la mobilité sont héritées des caractéristiques des entités auxquelles sont associés les holons. Elles peuvent dépendre de l'application (FMS, PRT).

(2) : le traitement Sfi_{DP} est assuré dans le cas d'un pilotage hybride.

5 Le modèle de comportement du SEAH

Le modèle de comportement est basé sur l'"open-control"(Sallez et al., 2009) et se concentre, dans la présente thèse, sur le niveau des holons HF (le cas échéant étendu aux HRA). Ce niveau constitue un système S' piloté par un pilote de niveau supérieur P' (cf. § 2.1, figure 2.1). Cependant, compte tenu de la récursivité, le raisonnement peut être reconduit pour un pilote de niveau quelconque P^{i+1} pour le système piloté S^{i+1}. S^{i+1} est lui-même composé de P^i et S^i.

Le modèle est constitué de deux parties. La première est relative au comportement interne du holon HF, la seconde concerne le comportement inter holons HF et HRA.

5.1 Modèle de comportement interne d'un HF

Le comportement propre d'un holon HF concerne, dans le contexte de la thèse, les deux processus : d'allocation et de routage.

Les processus d'allocation et de routage sont décrits dans la partie suivante. Ces processus sont basés sur les hypothèses suivantes (Sallez et al., 2010) :

- La topologie du système de transport est un graphe orienté fortement connexe composé de nœuds convergents/divergents connectés par des arcs, sur lesquels se déplacent les holons HF,
- Un chemin est une succession d'arc à parcourir pour se rendre d'un nœud de départ à un nœud destination,
- Un nœud du graphe peut accueillir une ou plusieurs ressources (modélisées par des holons ressources HR/HRA) capables de rendre un ou plusieurs services à un holon HF,
- Une décision de routage doit être prise par le holon HF au niveau de chaque nœud divergent de façon à atteindre, de proche en proche, un nœud souhaité où se trouve une ressource nécessaire fournissant au moins un service au holon HF ou bien le nœud permettant au holon HF de sortir du système,
- Sur un arc un holon HF peut seulement se déplacer du nœud d'entrée de cet arc à son nœud d'extrémité de sortie, aucune décision de routage n'est nécessaire sur un arc.

5.1.1 Articulation entre les processus d'allocation et de routage

Chaque holon HF possède une liste de services à obtenir successivement auprès des holons HR/HRA localisés sur certains nœuds. Cette liste est modélisée par le holon HSO composant du holon HF. Le holon HF se déplace de nœud en nœud sur le système de transport jusqu'à atteindre un nœud voulu pour obtenir au moins un service souhaité. Le holon HF situé sur un nœud peut dialoguer avec ce dernier pour acquérir des informations sur, par exemple, l'état des ressources du système, les temps de parcours sur les arcs…

5.1.2 Processus d'allocation

Lorsque que le holon HSO d'un holon HF est à jour, le holon HF doit successivement trouver des ressources et se déplacer auprès d'elles pour obtenir les services souhaités. Le processus d'allocation est hétérarchique. Le holon HF sollicite les holons HR/HRA pour identifier un (ou plusieurs) holon HR/HRA capable de satisfaire le service souhaité courant. Le holon HR/HRA est sélectionné selon un ou plusieurs critères, par exemple sur le temps de parcours du nœud courant au nœud destination, le nombre de holons HF dans la file d'attente du holon HR/HRA potentiel, les différents services souhaités que peut rendre ce holon HR/HRA.

5.1.3 Processus de routage

Lorsque le holon HR/HRA a été sélectionné, c'est le processus de routage qui permet d'atteindre la destination choisie. Ce processus est également hétérarchique. Le holon HF choisit le chemin, constitué d'une succession d'arcs, le plus court en temps entre sa position actuelle et le nœud destination visé et ceci en tenant compte de l'état réel du système de transport (temps de parcours actualisé) et des holons ressources HR/HRA (état de fonctionnement actualisé) remis à jour au passage du holon HF à chaque nœud.

5.1.4 Obtention du service

La figure 2.16 ci-dessous résume l'enchaînement des processus d'allocation et de routage avant l'obtention d'au moins un service.

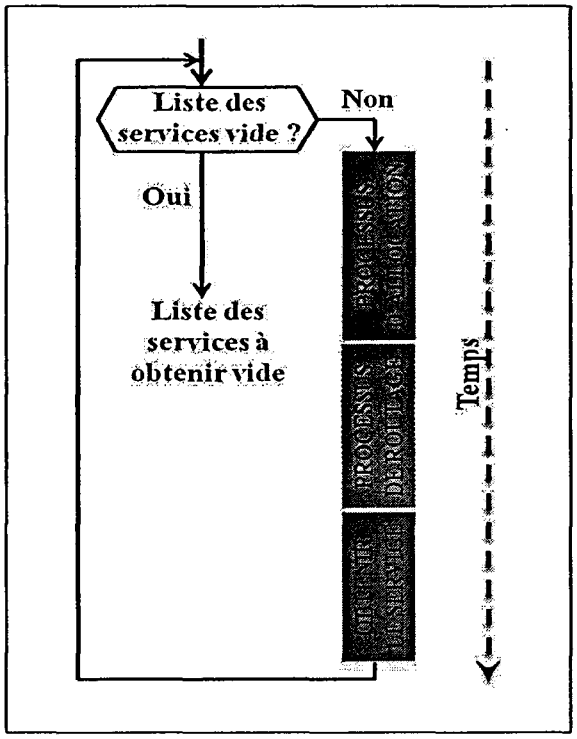


Figure 2.16 : processus d'allocation et de routage.

Sur le nœud destination le holon HF obtient au moins un service souhaité en fonction de la liste des services à obtenir successivement. Après qu'un service soit obtenu, le holon HF

réitère les processus d'allocation puis de routage jusqu'à ce que la liste des services à obtenir contenue dans le holon HSO soit épuisée.

5.2 Modèle de comportement entre holons HF et HRA

5.2.1 Modèle général de comportement

Le modèle est présenté figure 2.17. Dans ce modèle, les holons HF (i.e., PRT, produit intelligent) ou les holons HRA (ressources prenant part aux décisions de pilotage) possèdent tous une capacité décisionnelle (représentée par D) et jouent un rôle actif dans le déroulement des processus d'allocation et de routage. Les holons HF/HRA, système S', sont pilotés par un pilote du niveau supérieur P' conformément au modèle d'un SEA (cf. § 2.1, figure 2.1). Nous nous intéressons ici plus particulièrement aux HF mais des holons HR/HRA, apportant leurs services aux holons HF, sont également présents au sein du système S'.

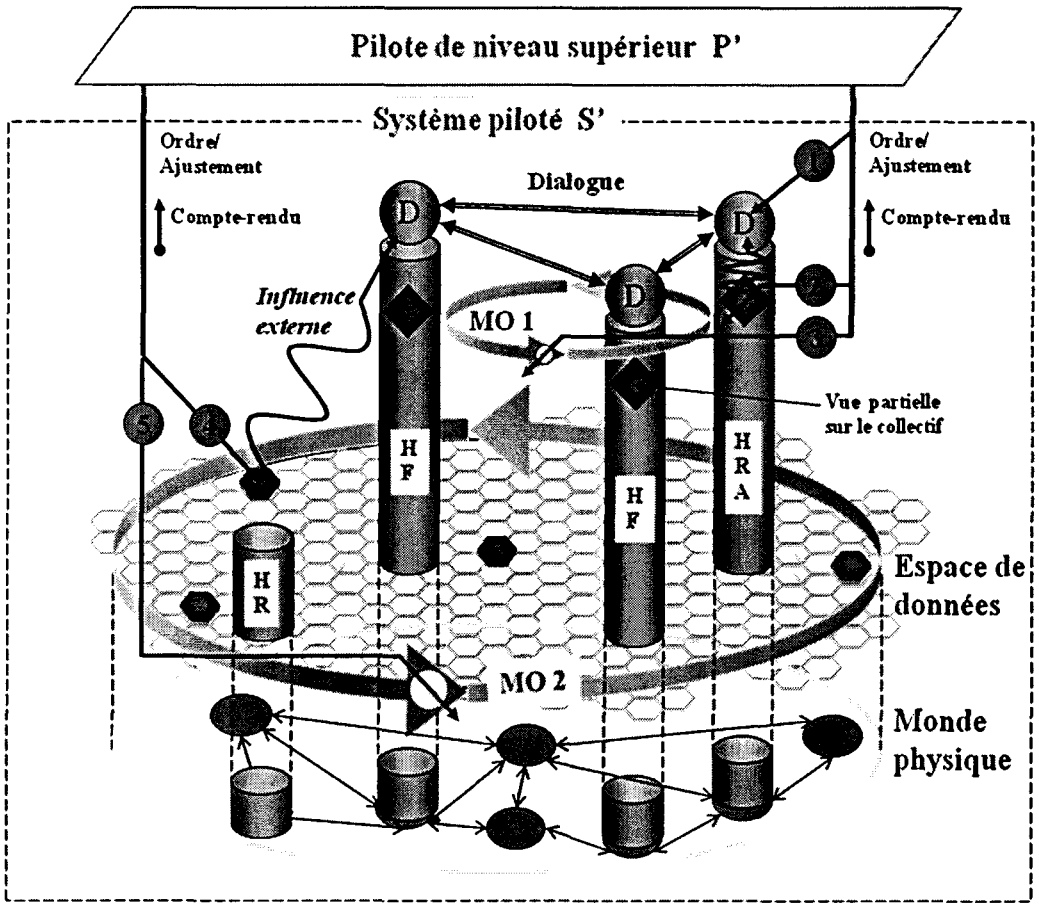


Figure 2.17 : modèle de comportement inter HF et HRA

Chaque holon HF cherche à atteindre ses propres objectifs, en terme de performance globale pour le collectif, et possède une vue partielle sur ce collectif (représentée par un losange dans la figure 2.17). Cette vue est mise à jour par un dialogue entre holons HF et HRA. Ce dialogue instaure le premier type de Mécanisme d'Optimisation (MO1 dans la

figure 2.17), appelé sociétal car il ne concerne que les holons HF et HRA et ne fait pas intervenir l'environnement.

Les holons HF ont aussi accès à de l'information disponible dans l'environnement (représentée par des hexagones dans la figure 2.17). Les holons HF peuvent ainsi accéder à de l'information dans leur proximité et l'intégrer pour prendre leurs décisions. Les holons HF peuvent également enrichir l'information disponible dans l'environnement en y déposant des données. Ces données permettent alors de faire profiter d'autres holons HF de l'expérience/connaissance individuelle d'un holon HF. Par exemple, dans les applications de routage, des marques disposées dans l'environnement au niveau des nœuds d'un graphe peuvent être utilisées pour fournir une vue sur la fluidité au sein du graphe et permettre l'adaptation des déplacements de chaque holon HF aux circonstances du trafic. L'expérience individuelle de chaque entité, engrangée lors d'un déplacement, peut être exploitée pour mettre à jour le marquage du graphe représentant par exemple des temps de parcours et mis à profit par le collectif.

L'environnement informationnel (nommé "espace de données" dans la figure 2.17) est dynamique et gouverné par des lois de mise à jour. Ces lois instaurent le second type de Mécanisme d'Optimisation (MO2 dans la figure 2.17), appelé environnemental. Par exemple, l'approche "Ant Colony Optimization" (Dorigo and Stützle, 2004), inspirée de fourmis en quête de nourriture et basée sur l'utilisation de lois de diffusion et d'évaporation pour résoudre des problèmes d'optimisation discrets (ex : routage dans un graphe), peut être utilisée. Les lois de diffusion utilisent une rétroaction positive pour répandre l'information dans l'environnement, tandis que les lois d'évaporation utilisent une rétroaction négative pour mettre à jour les données dans l'environnement.

5.2.2 Pilotage explicite

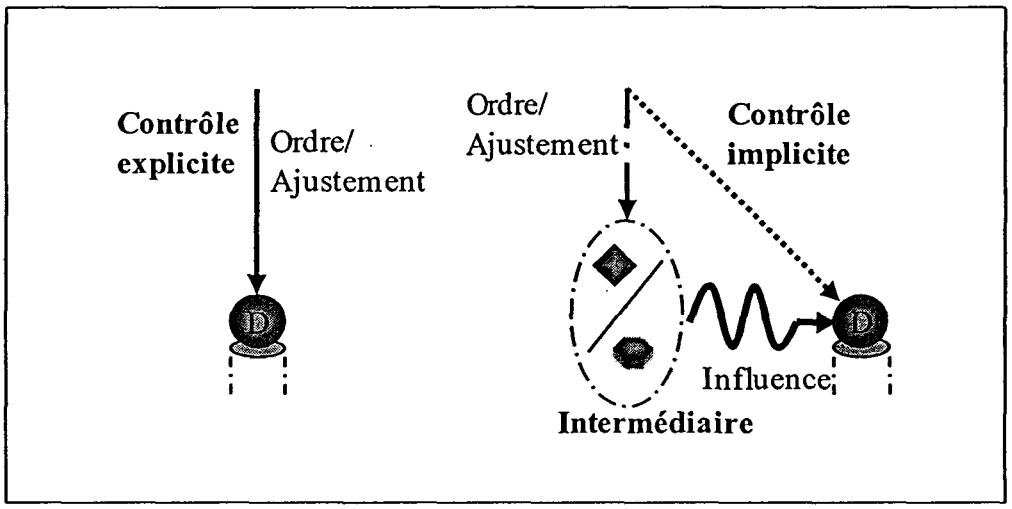
Le pilotage explicite induit un pilotage "du niveau supérieur vers le niveau inférieur" avec une approche maitre-esclave classique : un pilote envoie ses ordres à un subordonné clairement identifié (flèche 1 dans la figure 2.17 et figure 2.18 (a)). Le subordonné retourne des comptes-rendus d'exécution aux ordres du pilote. Comme en contrôle-commande classique, le pilotage explicite d'un holon HF/HRA peut être réalisé de deux façons : en modifiant des entrées et/ou en ajustant des paramètres. Par exemple, un pilote de niveau supérieur peut décider d'arrêter l'exploitation d'un PRT afin de réaliser sa maintenance ou encore changer sa destination initiale.

5.2.3 Pilotage implicite

Le pilotage implicite est relatif à l'influence du comportement d'un holon HF/HRA par des ajustements des paramètres associés aux mécanismes d'optimisations. Ce type de pilotage s'effectue en deux temps (figure 2.18 (b)) :

- dans un premier temps, au travers d'un ordre ou d'un ajustement, le pilote supérieur agit directement sur un intermédiaire qui joue un rôle dans un mécanisme d'optimisation sociétal ou environnemental,

- puis, un échange d'informations (ex : par dialogue ou diffusion) influence le comportement d'autres holons HF/HRA. Par exemple, dans le contexte FMS, de l'information peut être diffusée dans le voisinage de tronçons de convoyage subissant des engorgements pour influencer les shuttles afin qu'ils empruntent d'autres tronçons. Ainsi les shuttles potentiellement concernés pourront éviter d'être ralentis dans leurs progressions en utilisant des tronçons alternatifs.



(a) (b)

Figure 2.18 : pilotages explicite (a) et implicite (b)

5.2.4 Pilotage implicite via MO sociétal

Ce type de pilotage implicite peut être effectué de deux façons. La première façon induit l'ajustement de la vue partielle, liée au collectif, dans un holon HF/HRA (flèche 2 dans la figure 2.17). Cette modification peut être vue comme une influence interne qui altère le comportement du holon HF/HRA. Cette modification de comportement influence alors les autres holons HF/HRA via le mécanisme d'optimisation sociétal qui est instauré par dialogue entre les holons HF/HRA. Par exemple, des PRT peuvent dialoguer entre eux pour déterminer en négociant (ex : par appel d'offre) le PRT le plus approprié pour satisfaire une demande de déplacement. La seconde façon induit d'altérer le protocole d'échange impliqué dans le mécanisme d'optimisation sociétal (flèche 3 dans la figure 2.17). Par exemple, dans un processus d'appel d'offre, un holon HF pourrait être incité à communiquer qu'avec tous, certains HF ou encore ceux dans la proximité.

5.2.5 Pilotage implicite via MO environnemental

Ce type de pilotage implicite est réalisé, via de l'information localisée dans l'environnement, de deux façons : la première (flèche 4 dans la figure 2.17) implique d'agir sur les données directement (ex : création, modification, effacement), tandis que la seconde (flèche 5 dans la figure 2.17) implique un réglage des paramètres utilisés par le

mécanisme d'optimisation environnemental. Ces deux façons de faire induisent une influence qui affecte tous les holons HF capables d'accéder à l'information située dans l'environnement. Pour ce type de pilotage implicite, aucune communication directe entre les HF n'est nécessaire. Les applications basées sur la stigmergie sont une bonne illustration. Des phéromones virtuelles sont alors localisées sur les nœuds d'un graphe dans l'environnement. Ces phéromones indiquent le meilleur chemin vers une destination. Le pilote peut alors modifier directement les phéromones virtuelles sur un nœud spécifique ou encore ajuster globalement les taux de diffusion et d'évaporation des phéromones.

5.2.6 Différence entre pilotage implicite et pilotage explicite

La principale différence entre le pilotage implicite et explicite est que dans le pilotage implicite le holon HF/HRA concerné n'est pas directement visé par le pilote. Le pilotage implicite est particulièrement intéressant pour des applications impliquant un grand nombre de holons HF/HRA qui ne peuvent pas être pilotés individuellement directement. Le pilotage implicite peut être mis en œuvre en ayant recours à des méthodes inspirées par des mécanismes d'interactions biologiques ou sociétaux. Par exemple, des phéromones virtuelles (Parunak et al., 2002), des hormones numériques (Shen et al., 2004) ou encore les champs de potentiel (Ferber, 1995) peuvent être utilisés pour piloter des holons HF/HRA via l'environnement.

6 Conclusion

Dans ce chapitre nous avons défini un modèle de système à entités actives (SEA) adapté aux systèmes à flux guidés dans les domaines des systèmes de production flexible (FMS) et des transports de personnes guidés (PRT) et présenté son processus d'holonification. Ceci a permis d'élaborer le modèle holonique SEAH du SEA mettant en évidence le holon Flux HF. Celui-ci permet de modéliser les flux du SEA et leur caractère dynamique.

Afin d'évaluer la validité du modèle SEAH, ce modèle a été appliqué et expérimenté sur deux plateformes réelles :

- pour le FMS : la cellule de fabrication robotisée CIMR de l'Université "Politehnica"de Bucarest (Roumanie). Ce premier cas fait l'objet des chapitres 3 et 4,
- pour le PRT : la plateforme de l'AIP - PRIMECA de l'Université de Valenciennes et du Hainaut - Cambrésis (France). Ce second cas fait l'objet des chapitres 5 et 6.

Chapitre 3: utilisation du modèle holonique SEAH pour les systèmes de production de type "batch"

Introduction

L'objectif de ce chapitre est l'élaboration d'un modèle holonique d'un système de type FMS en adaptant le modèle holonique de pilotage d'un SEAH, présenté dans le chapitre 2.

La première partie du chapitre décrit la cellule flexible du CIMR. L'ensemble des éléments de fabrication et de transport sont présentés en détail. La partie suivante est consacrée à la définition du modèle holonique : les différents holons, puis le système de pilotage sont décrits en détail. L'architecture du système de pilotage et les mécanismes d'allocation de ressources et de routage sont précisés.

Les modèles de planification, d'ordonnancement et de réordonnancement sont également décrits ainsi que l'outil de simulation qui a été développé.

1 Le contexte

Pour être compétitif, les systèmes de fabrication doivent s'adapter à l'évolution et aux changements imposés par le marché économique. Une diversité des produits qui augmente, des fluctuations importantes dans les demandes des clients, des cycles de vie des produits de plus en plus réduits et des attentes plus élevées des clients en termes de qualité et de temps de livraison représentent des challenges auxquelles les entreprises manufacturières doivent répondre pour rester compétitives. En plus aux problèmes imposés par le marché, les entreprises doivent faire la preuve de flexibilité et d'adaptation aux nouvelles technologies développées et aux changements permanents des normes de protection de l'environnement (Nylund et al, 2008).

Bien que l'optimisation des processus de production reste un point important dans le domaine des systèmes de fabrication, "l'entreprise agile" gagne de plus en plus de terrain (Barata, 2007 ; Sauer, 2008). Les systèmes de fabrication flexible doivent être capables de s'adapter rapidement aux perturbations / situations non prévues comme les défaillances physiques des ressources, l'insertion de nouvelles ressources dans la chaîne de production, la rupture des stocks de matériaux, et la prise en compte rapide des ordres urgents des clients ("*rush orders*") (Borangiu et al, 2008).

Dans les dernières décennies, les développements scientifiques ont défini de nouvelles architectures décentralisées de pilotage de type hétérarchique (ex : un groupe d'entités indépendantes dénommées agents négocient l'attribution des ordres de production basé sur leur état et charge future) – alternatives aux architectures hiérarchiques dans le domaine des structures flexibles de fabrication (FMS) (Rahimifard, 2004).

Pour les structures de fabrication avec des ressources connectées en réseau (robots, machines-outils, vision) et contrôle de qualité en ligne, une architecture de commande semi-hétérarchique est proposée (Raileanu et al., 2009). Dans cette architecture, le contrôle est disposé sur deux niveaux, appelés global et local.

Le niveau *global* assume la responsabilité de la planification et la coordination des activités au niveau de la cellule/atelier/usine (niveau *shop-floor*) et la résolution des problèmes entre les entités qui ont des objectifs locaux, non compatibles, qui entrent en conflit.

Le niveau *local* par contre, possède l'autonomie pour modifier un ordonnancement créé hors ligne et contrôler les activités internes des sous-systèmes (par exemple, d'une équipe de robots qui effectue un montage).

Le paradigme de "fabrication collaborative" basé sur plusieurs critères – similarité des activités, localisation, regroupement des ressources en fonction des tâches – qui décrivent les processus de planification et le contrôle (Leitao, 2006, Barata, 2005), a été adopté pour assurer la flexibilité, l'agilité et la réactivité nécessaires pour faire face aux demandes « volatiles » des clients.

2 La cellule flexible CIMR

Cette architecture a été conçue pour minimiser le temps total de production par une planification globale et un ordonnancement global des ordres de production. Ainsi, le mode de fonctionnement hiérarchique qui assure l'optimalité globale est prioritaire, et ceci tant que la fréquence des pannes des ressources est réduite. Aussi, le système de fabrication cible est un système de type holonique avec une partie contrôle robuste, réactive et distribuée qui est activée en mode hétérarchique lors de l'apparition de perturbations. Ce type d'organisation permet aux ressources de fonctionner individuellement et indépendamment, suivant des ordres précis issus du système de contrôle global ; ordres vus comme des recommandations qui optimisent globalement le processus de production. Dès qu'une perturbation est détectée (panne d'une ressource, rupture d'un stock de composants) le système de contrôle passe en mode hétérarchique, abandonne les recommandations et utilise sa partie distribuée pour refaire en ligne l'ordonnancement des produits en exécution ; les représentants (correspondants informatiques) des ressources autonomes communiquent entre eux en ce but dans un processus de négociation basé sur contrat qui décrit de modèle d'accès aux services (MdAaS).

Le système de pilotage sera divisé en deux niveaux principaux : un niveau <u>haut</u>, qui est en charge de la planification et l'ordonnancement global des ordres, et un niveau <u>bas</u> en charge avec le ré-ordonnancement des ordres de fabrication en cas de perturbation et l'exécution des opérations assignées. Le niveau bas est à son tour divisé en deux à cause de la relation maître esclave entre la partie contrôle du système de transport (en charge avec le transport intelligent des produits) et celle des postes de travail (en charge avec le traitement des produits). Ainsi, le sous niveau "contrôleurs des ressources" représente le niveau automatisation des processus technologiques et le niveau "automate programmable" (partie

contrôle du système de transport) deviendra un sous niveau intermédiaire entre le niveau *planification et ordonnancement* et le niveau *automatisation des processus* (routage, manipulation, usinage).

Dans cette structure, les ordres clients qui sont associés aux lots de production avec une date limite de livraison (établie en accord commun entre le client et l'entreprise qui gère le système de fabrication) disposent d'entités qui enrichissent leurs capacités décisionnelles. Un ordre est le représentant d'un produit et pour sa fabrication il doit subir une série d'opérations de traitement (usinage, assemblage, vérification en ligne). Dans le cas le plus général, entre ces opérations peuvent exister des précédences *a priori* connues.

Parce qu'en théorie la façon de résoudre le problème de la planification et de l'ordonnancement des lots de produits dans les systèmes de fabrication ne tient pas compte des caractéristiques réelles du système de transport – files d'attente des ressources (considérées infinies (Montana et al., 2007, Lopez et al, 2001)), temps de transport (considérés négligeables (Kusiak, 1990)), la méthode d'allocation des opérations à l'aide des règles de production (Kusiak, 1990) a été adaptée aux besoins courants : ressources de fabrication interconnectées par un système de transport automatisé de type convoyeur en boucle fermée et disposition spatiale des postes de travail robotisés (Figure 3.1).

Cette méthode d'allocation des opérations utilise un algorithme de planification heuristique avec règles de priorité pour générer la succession des ordres de production. L'objet de la planification est de proposer une succession pour les ordres qui seront introduits dans le système de fabrication. Ensuite l'ordonnancement des opérations associées à un ordre de production et l'allocation des opérations aux ressources sont faits en utilisant un simulateur à base de temps variable. Ce simulateur s'occupe du suivi de l'évolution temporelle, dans le système de fabrication, des ordres déjà planifiés pour qu'ils n'entrent pas en conflit avec d'autres ordres et minimisent aussi le temps de fabrication au niveau global des lots de produits. Une opération déjà planifiée et qui a été introduite dans le simulateur est considérée ordonnancée et l'ordre auquel elle appartient ne peut plus être retiré du planificateur. Les deux étapes, *planification* (avec ordonnancement) et *simulation*, dépendent l'une de l'autre parce que l'insertion de nouveaux ordres (allocation d'une première opération) est faite en fonction du degré d'occupation du système, en particulier de l'occupation des ressources de traitement. Pour les ordres déjà insérés, le planificateur fait l'ordonnancement des opérations à suivre. Dans le cas des systèmes de fabrication avec un système de transport classique il y a d'autres contraintes en plus par rapport à l'occupation d'une ressource du processus : le représentant physique d'un ordre – la palette avec le produit en cours de fabrication – doit attendre si le tronçon de transport en amont est occupé par une autre palette.

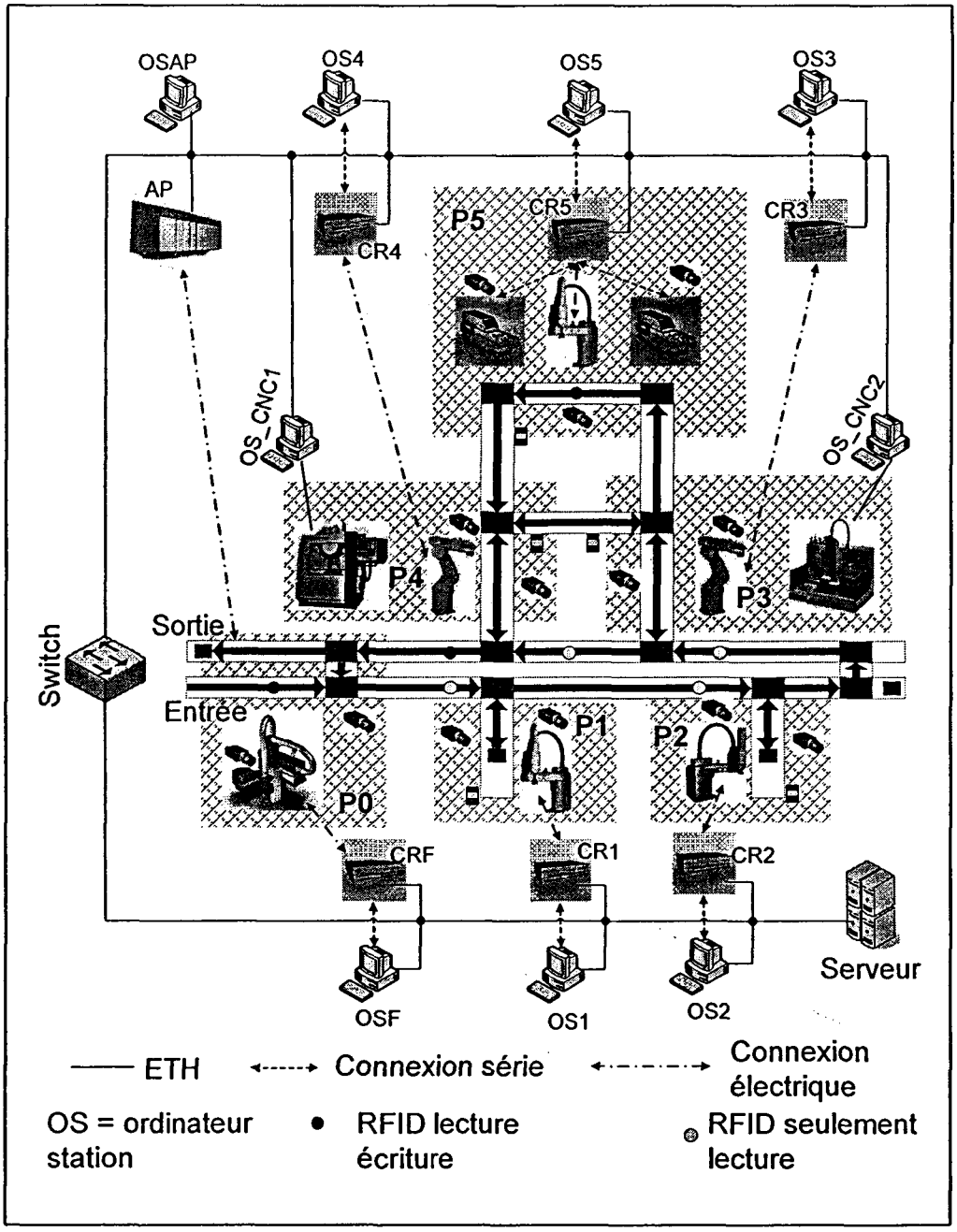


Figure 3.1 : structure de la cellule CIMR

Dans la figure précédente les ressources sont organisées en quatre postes de travail (P1, P2, P3 et P4), un poste d'entrée/sortie des palettes (P0) et un poste d'alimentation (P5). Chaque poste contient une ou plusieurs ressources (ex : le poste P3 contient un robot industriel avec 6 degrés de liberté, deux cameras vidéo – une fixe et une mobile, et une machine à commande numérique). Les parties informationnelles de chaque ressource (CR*i* – contrôleur robot *i*, OS*i* – ordinateur station *i*, AP – automate programmable) sont interconnectées entre elles par deux types de réseaux de communication (ETHERNET et série) et avec le serveur de la cellule en réseau ETHERNET.

La traçabilité des produits dans le système de fabrication est faite avec des dispositifs RFID. Il y a deux types de dispositifs RFID :

- *RFID lecture écriture* (RW – *read-write*), à l'entrée et à la sortie du système ;
- *RFID lecture seulement* (*read only*), avant les déviations vers les postes de travail.

En ce qui concerne l'échange d'information entre les ressources, plusieurs interconnexions sont utilisées (Figure 3.2), ce qui assure un niveau élevé de tolérance aux défauts de type échec de communication.

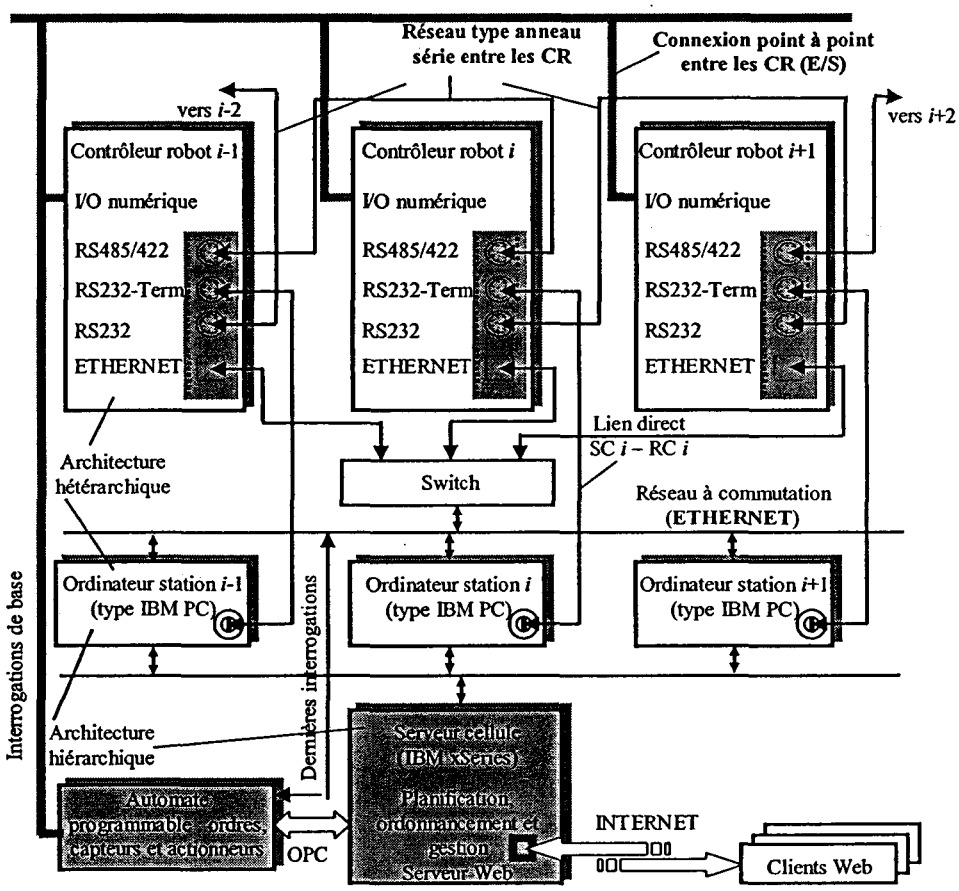


Figure 3.2 : architecture distribuée de contrôle et communication, tolérante aux défauts

Ainsi, le système multi réseau de communication comprend trois types d'interconnexions : (1) le réseau de type *ETHERNET*; (2) le réseau qui utilise la *connexion série* directe entre deux ressources – PC station et contrôleur équipement industriel (robot, machine outil) pour créer un réseau de communication de type anneau, et (3) l'ensemble d'*entrées/sorties numériques* des ressources.

Un point original du système de contrôle est le simulateur qui fait l'allocation des opérations sur les ressources. Pour la partie « planification et ordonnancement des ordres de production », effectuée séparément du calcul de l'alimentation des postes de travail

(sans tenir compte du poste P5), le simulateur utilise le partitionnement en tronçons (T_i, i = 1, ..., 12) suivant la topologie du système de transport (Figure 3.3) :

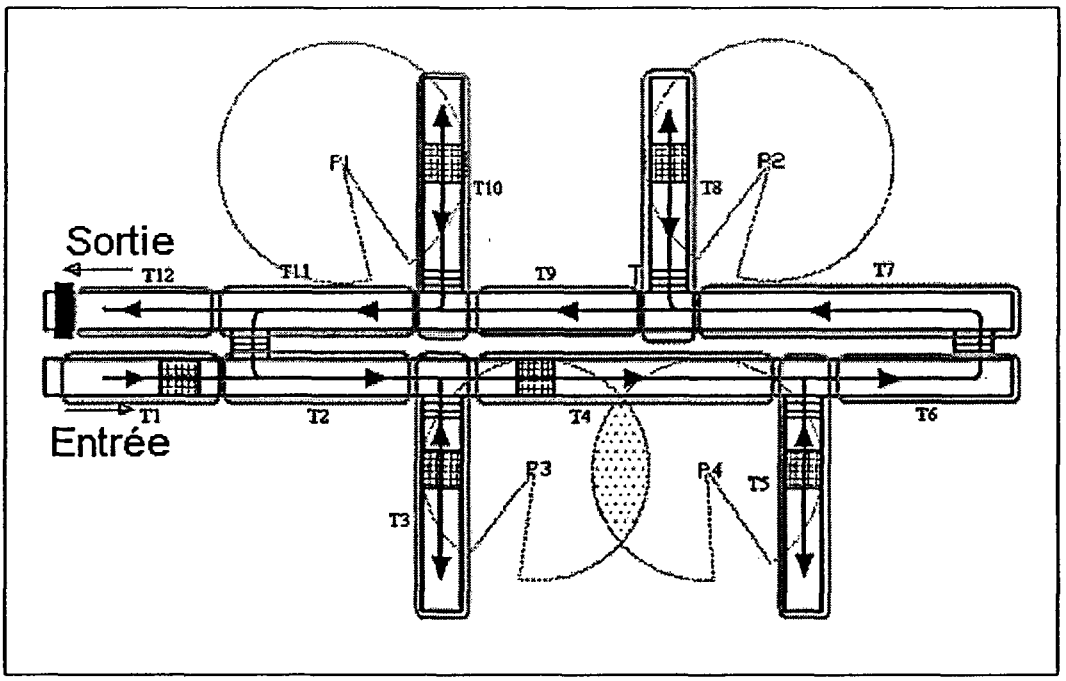


Figure 3.3 : partitionnement du système de transport en tronçons

En tenant compte du partitionnement précédent, le modèle du système peut être synthétisé avec un graphe orienté dans lequel les nœuds représentent les ressources (de traitement robotisé (R0-R3) et de transport (R4-R14)) et les arcs orientés représentent les liaisons directes entre les ressources. Cette procédure conduit au modèle ci-dessous (Figure 3.4) du convoyeur, qui sera utilisé pour l'implémentation de la partie planification et ordonnancement.

La signification des éléments composants du graphe est la suivante :

- *les nœuds* représentent les ressources des processus d'usinage/montage/vérification (R0, R1, R2, R3) ou de transport (R4, ..., R14). Le convoyeur est divisé en sous ressources, chaque ressource étant capable d'offrir une seule opération de transport à un moment donné (cette contrainte est le résultat du type de contrôle des systèmes à événements discrets pour éviter les temporisations). Chaque nœud est caractérisé par une capacité qui, dans le cas de ce type de système de fabrication est 1 pour tous les nœuds, et par un temps minimal de stockage des palettes qui représente le temps pour lequel chaque palette reçoit le service de traitement ou de transport requis. Une palette passe au nœud prochain après obtention du service dans le nœud courrant à condition que le nœud prochain soit libre.

Ce type de fonctionnement pourra générer des blocages (exemple : si plusieurs palettes s'agglomèrent sur la boucle de transport principale, il y a le risque que le système de

fabrication se bloque en totalité), ce qui réclame un système de contrôle performant pour le système de fabrication.

- *les arcs* représentent les liaisons directes entre les ressources.

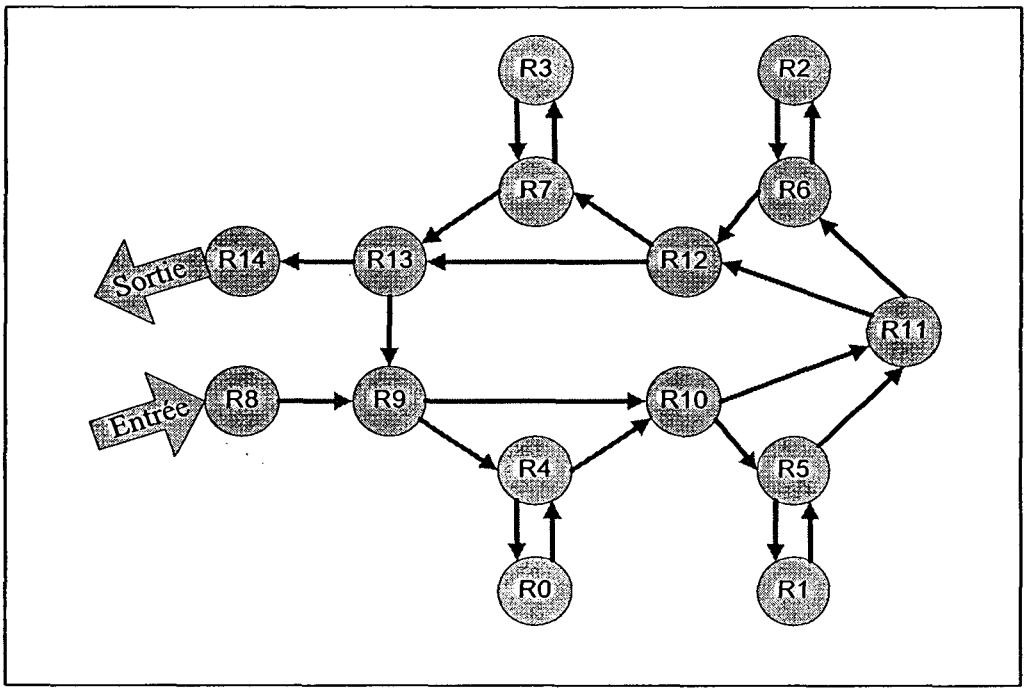


Figure 3.4 : modélisation du système de transport avec un graphe orienté

L'utilisation d'un graphe pour modéliser le système de fabrication ne limite pas la généricité de la solution. Si le problème change (une autre infrastructure), alors le graphe doit être refait, mais les algorithmes d'allocation restent les mêmes.

En ce qui concerne les produits que le système est capable de fabriquer, ils seront aussi modélisés avec des graphes à cause des précédences qui peuvent exister entre les opérations. Les nœuds représentent les opérations qui doivent être exécutées et les arcs directs représentent les précédences entre elles. Pour cela, le graphe n'est pas nécessairement connexe : s'il y a des opérations sans précédences, il n'y a pas de chemins jusqu'aux nœuds associés à ces opérations. Pour ce type de système de fabrication les types de précédences suivantes ont été considérés (Figure 3.5) :

- *Convergences* : pour qu'une opération soit exécutée, plusieurs opérations doivent être exécutées en avance et
- *Divergences*, après l'exécution d'une opération, plusieurs opérations peuvent être exécutées. Par mesure de sécurité le graphe doit être vérifié et ne doit pas comporter de circuits parce que si c'est le cas, alors le produit ne peut pas être exécuté.

Ces précédences proviennent des demandes de produits particuliers que le système de production réel doit exécuter : ainsi, il y a des opérations qui réclament que le produit présente certaines caractéristiques avant qu'elles puissent être débutées (exemples : le montage par insertion d'un composant suppose l'existence / l'exécution préalable d'un trou dans le corps du sous ensemble, une opération d'inspection visuelle d'un montage nécessite que tous les composants soient d'abord mis en place).

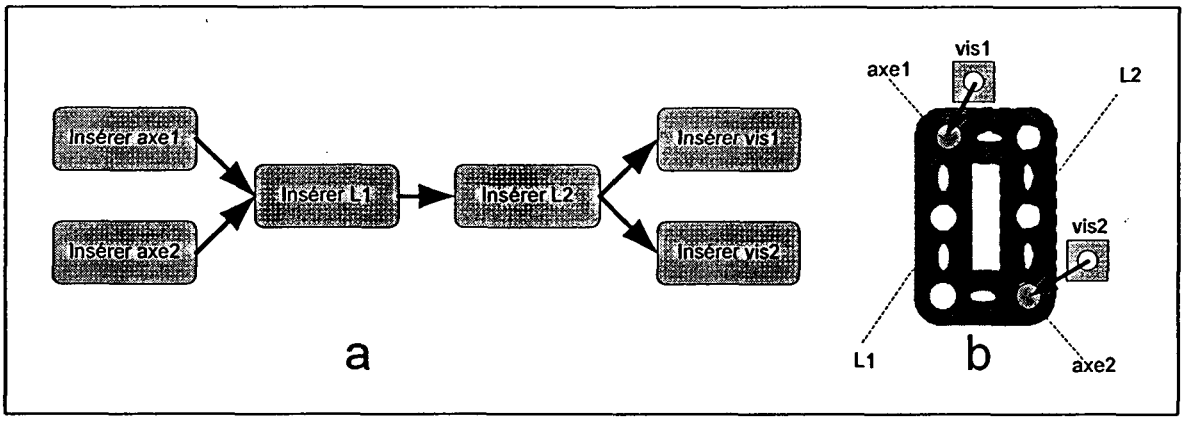


Figure 3.5 : exemple de modèle : (a) pour la fabrication d'un produit - (b) le produit fini

En conclusion, la solution de contrôle proposée rend le système de fabrication *flexible* au niveau des postes de travail (types des produits, technologies alternatives, nouveaux programmes de contrôle des ressources) et des ordres clients (dimension du lot de produits, ordres rapides), mais aussi *robuste* en cas de pannes temporaires ou permanentes des ressources. Ces fonctionnalités sont une conséquence des caractéristiques suivantes du système de pilotage :

- Commutation automatique entre les modes de contrôle hiérarchique pour l'utilisation et planification optimale des ressources, et hétérarchique pour un fonctionnement agile grâce aux centres décisionnels proches des endroits qu'ils contrôlent ;
- Détection automatique des stocks vides avec génération, planification et exécution automatique des séquences d'alimentation ou réalimentation des postes de travail ;
- Contrôle de la qualité des produits en ligne (par systèmes de vision) à différentes étapes de la production ;
- Traitement physique automatisé des produits en cours avec des robots industriels et contrôle visuel des produits finis.

3 Architecture du système de pilotage

1.1 Holon actif généralisé

Le modèle générique de pilotage s'appuie sur le système holonique SEAH du chapitre II, ses composants représentent les entités physiques du FMS fonctionnant ensemble en groupements reconfigurables pour offrir et réaliser des services. Ces entités sont gérées par

un système de pilotage distribué et caractérisé par autonomie et coopération. Le système de pilotage est composé de systèmes individuels de pilotage embarqués sur les entités physiques du système cible qui communiquent entre elles et avec d'autres entités physiques passives du point de vue pilotage.

Même si les entités du système ont des rôles différents (ex : traitement, stockage ou transport) pour que l'architecture soit "scalable" il est préférable qu'elle ait la même structure générique (Nylund, 2008) et les mêmes interfaces. Ainsi, la structure suivante est proposée pour assurer les fonctionnalités et les interactions des holons, basée sur un rapprochement information matière (Figure 3.6).

L'entité informationnelle est composée de trois parties qui permettent au holon actif généralisé d'être autonome et coopérant/communicant. Pour cela, elle contient un **module de mémorisation** pour stocker des informations nécessaires à l'obtention des services : son propre modèle, le modèle du système et aussi des informations supplémentaires comme les paramètres des ressources ou les programmes et procédures à exécuter. Les informations peuvent être « seulement lecture » *(read only)* ou « lecture écriture » *(read-write)* en fonction des besoins du système de contrôle. Le deuxième module, qui assure l'autonomie décisionnelle, est le **module de traitement** qui gère les informations transférées entre différents types de holons et aussi le contrôle direct des ressources physiques. Le troisième module est le **module de communication** qui gère l'accès aux réseaux informationnels et l'échange des données entre entités.

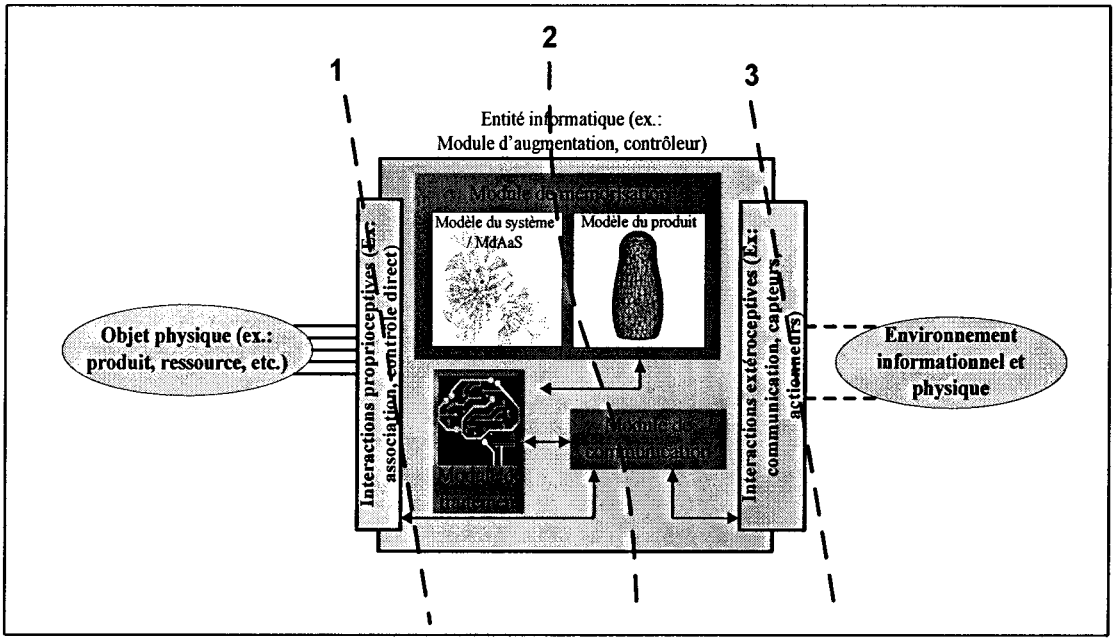


Figure 3.6 : la structure d'un holon actif généralisé

En fonction des capacités décisionnelles embarquées sur les entités et des moyens de communication/synchronisation avec l'environnent extérieur, la partie décisionnelle peut être localisée dans un des trois modes suivants :

1) *à distance*, sur une machine ou sur un groupe de machines connectées au réseau, capables de dialoguer avec les holons associés à d'autres entités. Pour que la partie informationnelle soit au courant de l'état réel de la partie physique, une synchronisation entre elles doit exister. D'habitude pour ce type d'entité intelligente la technologie RFID est utilisée (MacFarlane, 2002; Cea Ramirez, 2006) et tous les traitements sont faits à distance. Sur le modèle générique, ce type de rapprochement information - matière correspond à la ligne 1 en Fig.3.6. Les avantages d'un système de contrôle de ce type sont : la disponibilité des ressources pour implémenter l'architecture et le couplage fort entre les entités informatiques, ce qui conduit à un système plus centralisé qui a un comportement optimal dans les situations normales mais de faibles performances dans un environnement caractérisé par des changements fréquents ;

2) *local*, directement sur l'entité active sous la forme d'un module qui augmente ses capacités décisionnelles et de communication. Dans ce cas, qui correspond à la ligne 3 en Figure 3.6, l'entité devient plus autonome, ce qui fait que le système de pilotage soit plus décentralisé et les décisions sont prises plus proches de l'endroit où elles sont nécessaires. Ces caractéristiques permettent au système de contrôle de devenir plus tolérant aux défauts et plus réactif ;

3) *hybride*, avec un agent principal qui s'exécute à distance sur une machine dédiée, qui a le rôle de prendre des décisions et de communiquer avec les autres entités qui participent au processus et un agent secondaire embarqué directement sur l'entité de contrôle qui surveille son état (ex : mesure la température, la distance de sécurité, etc.), prend des décisions locales (fait la localisation dans le système, calcule le chemin le plus court, etc.) et informe l'agent maître en cas de problèmes. Ce type d'entité correspond à la ligne 2 de la Figure 3.6 qui découpe la partie informationnelle en deux. Dans ce cas il est désirable de synchroniser l'information avec la matière par un réseau de communication pour que la partie physique avec la partie décisionnelle embarquée puisse recevoir et envoyer des messages complexes vers l'agent maître.

1.2 Architecture

Pour le système de pilotage holonique l'architecture est non centralisée, sans liaisons maître- esclave ; par conséquent la structure proposée diffère d'une structure purement centralisée :

- il y a seulement deux niveaux de gestion des tâches,
- la fonctionnalité *ordonnancement* est déplacée à l'intérieur du niveau automatisation qui devient ainsi plus réactif et autonome. Sauf les demandes d'optimisation globale des lots de production réalisées hors ligne, l'ordonnancement est maintenant fait en temps réel. L'avantage de cette approche est que pour toutes situations non prévue il n'est pas nécessaire de remonter le

problème au niveau supérieur car le niveau automatisation est capable de prendre tout seul les décisions appropriées.

Pour que le système de pilotage respecte le concept d' »open-control » avec influence implicite sociétale (décrit dans le Chapitre 1, état de l'art), les entités du système sont réparties dans une architecture à deux niveaux : un *niveau décisionnel bas* qui est en charge du contrôle direct des processus et de l'optimisation de l'allocation des ordres sur un horizon limité – faits en ligne, et un *niveau décisionnel supérieur* qui est en charge de la planification des ordres et de l'interface du client avec le système cible considéré.

Le niveau décisionnel bas (**automatisation**, niveau 1) est composé d'entités de type HNA (holon ressource) et HF (holon flux) qui dialoguent entre eux (HF – HR et HF – HF, Figure 3.7) pour optimiser l'obtention des services sur un horizon de temps limité (exemple : minimiser le temps total d'obtention des services pour les palettes introduites dans le système, maximiser l'utilisation des ressources, etc.). Le niveau décisionnel supérieur (**planification**, niveau 2) est composé du (des) holon(s) coordinateur(s) et d'une base de connaissances qui contient les informations sur le processus et sur les services que le système offre. La tâche du holon coordinateur est d'influencer directement ou indirectement les entités du premier niveau et aussi d'intégrer le système de pilotage avec les couches supérieures de l'entreprise (gestion des appels d'offre et des ordres client, association des ordres client aux HFs, suivi du processus et traçabilité du produit, calcul des paramètres qui décrivent le comportement global du système, etc.).

La conception de l'architecture de pilotage s'inspire des structures hiérarchiques et hétérarchiques de pilotage en ne conservant que les fonctionnalités et les propriétés jugées comme nécessaires pour un nouveau modèle de pilotage agile, reconfigurable selon les besoins et influences de l'environnement. Les fonctionnalités des systèmes hiérarchiques (planification, ordonnancement et automatisation - du routage, des traitements) sont conservées, mais l'ordonnancement (fait hors ligne au niveau opérationnel de planification) est maintenant déplacé vers le niveau automatisation qui devient plus autonome et réactif grâce aux *capacités décisionnelles et de communication* (inter HF et HF-HR) et aussi grâce à la *capacité de reconfiguration dynamique de l'accès aux services* fournis par les ressources. Ainsi, pour piloter le système cible, l'architecture générique de contrôle ouvert, hybride, décrite en Figure 3.7 est proposée.

Le choix du type de pilotage, qui peut être fait automatiquement par un système expert connecté à une base des connaissances ou manuellement par un opérateur humain, permet de décliner cette architecture générique vers un système plutôt :

- *hiérarchique*, si le système reçoit les ordres de production déjà planifiées (l'ordre dans lequel les demandes du client seront traitées par le système) et ordonnancées (la succession imposée des services à obtenir pour chaque demande du client) – le cas 2 dans la Figure 3.7,

- *hétérarchique*, si le système reçoit les ordres de production sans aucune information sur l'ordre de réception des services – le cas 1 dans la Figure 3.7 ;

- *semi hétérarchique*; si on considère le système cible au niveau global, un pilotage hétérarchique n'est pas pratique à cause de certaines fonctionnalités qui sont d'une façon inhérente centralisées. Cela s'applique à la réception des demandes client et au lancement des ordres de production pour satisfaire les demandes, qu'on devrait mieux maintenir centralisés (le cas de la planification des ordres que le système devra réaliser). Un système semi hétérarchique choisit le meilleur mode de pilotage en fonction des événements externes, de la dynamique des demandes des clients, des caractéristiques du système cible et de son état courant. Le pilotage hiérarchique peut alterner avec le pilotage hétérarchique, ou peut coexister avec lui. Il s'agit du cas 3 figure 3.7, où les ordres client arrivent partiellement planifiés, mais c'est la tâche du système de faire l'allocation des services en temps réel.

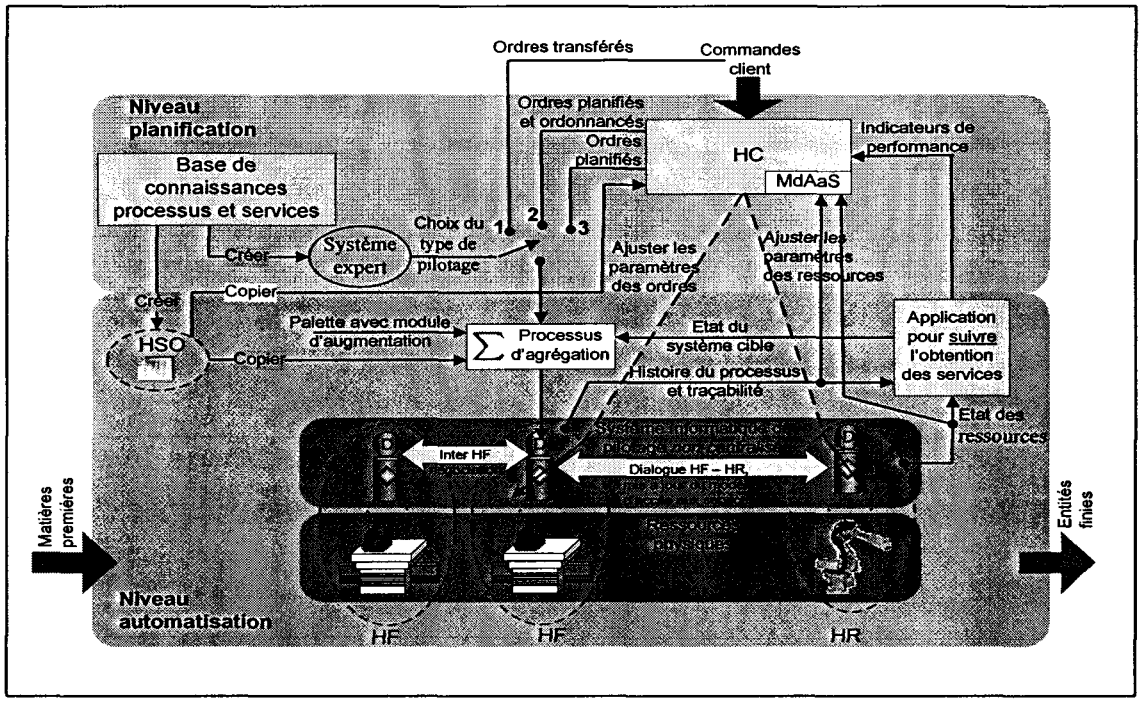


Figure 3.7 : architecture générique de contrôle ouvert du système cible et les flux des commandes (adaptée de Trentesaux, 2007)

On propose donc une architecture de type « open-control » qui peut être reconfigurée d'une manière dynamique ; le choix du mode de pilotage se fait en fonction des particularités techniques du système cible piloté et aussi en fonction des performances requises.

Les arcs en pointillés entre le niveau planification (le holon coordinateur) et le niveau automatisation (un HF et les HRs qui offrent des services) représentent une influence indirecte des ressources (un pilotage implicite par l'influence d'un mécanisme d'optimisation sociétale). Les autres arcs orientés entre les entités représentent des flux d'information ou de matière. Les entrées et les sorties physiques du système sont représentées avec des grandes flèches.

Le pilotage implicite par optimisation environnementale consiste à influencer le comportement des ressources par des capteurs intelligents (caméras vidéo, détecteurs de

présence) qui extraient des caractéristiques d'intérêt sur les flux de matériaux / des routes à suivre et ajustent ou même changent les actions des robots et navettes.

1.3 Structure

Dans cette architecture de pilotage une entité de niveau décisionnel supérieur, le **Planificateur Global de la Production** (PGP), optimise globalement la production, c'est-à-dire l'exécution de l'ensemble des ordres clients. La planification et l'ordonnancement des ordres clients calculés par le PGP sont ensuite envoyés à l'automate programmable (placé à un niveau intermédiaire) qui est en charge du routage des produits et qui communique avec les entités placées au le niveau inférieur – contrôleurs robots, machines outils, postes vision, etc. – pour accomplir les ordres.

Dans ce système de fabrication holonique la planification (l'ordre dans lequel les ordres clients entrent dans le système de fabrication) et l'ordonnancement (*l'ordre d'exécution des opérations* qui composent un ordre client (qui sont nécessaires pour obtenir un produit) et *l'allocation des ressources* pour les opérations fournis par le PGP ne sont pas imposés, mais traités plutôt comme des *recommandations* par les entités actives de la production (ordres). Ces recommandations seront suivies tant qu'il n'y a pas de changements dans l'état des ressources (Borangiu et al., 2009; www.rvholon.cimr.pub.ro). En cas de changements non prévus, les recommandations produites hors ligne seront ignorées et une nouvelle planification et un nouvel ordonnancement seront calculés avec la participation active des ressources du système de contrôle.

En adaptant les éléments composants la structure générique SEAH (chapitre II) aux besoins présentés ci-dessus (sous chapitre II.2), la structure de pilotage dérivée est composée des holons suivants.

- Le Holon Flux

Le Holon Flux (HF), partie centrale du système de fabrication holonique cible, est composé d'un porte palette, d'une palette porteuse des matériaux et des informations (un identificateur unique sur le dispositif RFID attaché, qui permet de localiser la palette avec le produit en fabrication relativement aux ressources – machines, robots, etc.) et d'un système informationnel associé qui est localisé à distance (l'ensemble des PC connectés en réseau – *Ordinateurs Station* OS en Figure 3.1, qui représentent les ressources – *représentants/images* des ressources).

Dans le cas du système de fabrication (Figure 3.1), parce que la partie contrôle des processus dans le niveau "automatisation" est fixe (elle est toujours développée en fonction de l'infrastructure matérielle existante), la solution *i* (*décision à distance, cf. § 2.1*) a été adoptée pour la partie informationnelle d'un HF. Ainsi, la partie informationnelle (le module d'augmentation) est distribuée sur deux niveaux ("automate programmable" pour les exécutions des ordres et "Ordinateurs Stations" pour le ré-ordonnancement) comme décrit en Figure 3.8 et la synchronisation information matière se fait avec des dispositifs RFID (ces derniers sont placés dans les points stratégiques du système de transport, comme les aiguillages vers les ressources physiques) qui localisent les palettes. La partie

informationnelle du niveau haut de contrôle (le serveur PGP) est en charge de la planification et de l'ordonnancement des ordres de fabrication.

Une fois les informations de ce type obtenues dans le niveau haut, elles seront transférées et utilisées dans le niveau informationnel bas par l'automate programmable de la cellule pour le transport les palettes et l'obtention des services de traitement de la part des ressources pour l'exécution des ordres de production.

Une palette se déplace dans le système de fabrication sur une infrastructure de transport dédiée – convoyeur en boucle fermé avec des ramifications vers les ressources ; quand elle arrive à un point important (divergence, convergence du système de transport ou poste de travail) elle génère des événements qui sont reçus par son représentant informationnel (qui réside dans l'automate programmable) et qui est en charge de sa partie décisionnelle.

Contrairement au plan de réalisation d'un produit (HSO), qui est une description d'un certain type de produit et qui ne change pas, le holon HF contient aussi la réalisation physique optimisé d'un produit pouvant subir de nombreux changements pendant le processus de fabrication.

Un holon HF (Figure 3.8) est composé de :

- *une partie logiciel* (le Holon Conduite, partie niveau haut) située sur le serveur de la cellule qui fait l'allocation en PGP. Ici la partie informationnelle du holon HF se trouve dans un représentant du monde réel et possède des capacités décisionnelles (le symbole D dans le cercle jaune) ;
- *un programme de gestion* situé sur l'automate programmable (le Holon Conduite, partie niveau bas). L'information qui concerne la planification et l'ordonnancement des HFs est mémorisée sur ce niveau, permettant à l'automate programmable de déplacer la palette vers les ressources attribuées (le symbole diamant dans le cercle jaune) ;
- *une palette* avec *un identificateur RFID unique* (Holon Mobilité). Une palette est associée uniquement à un produit auquel elle offre des services de transport pendant le processus de fabrication (le symbole ID dans le cercle jaune) ;
- *produit* ou *composantes de montage transportées* (le Holon Objet Récepteur des Service). Un produit demandé par le client ou des matériaux nécessaires est/sont transporté(s) dans la cellule sur le tapis du convoyeur.
- En fonction des processus réalisés par le holon HF, celui-ci est de deux types : HF de production et HF d'alimentation.

- Le Holon Flux de Production

- Un HF de production est le représentant d'un ordre client et directement associé avec un produit en cours de fabrication. Il contient l'information nécessaire pour fabriquer ce type de produit, les informations calculées hors ligne qui concernent l'ordonnancement des opérations et aussi l'information temps réel qui concerne

son état : s'il est entré dans le système de production ou s'il est encore en attente, l'opération courante, la ressource courante et l'histoire des opérations réalisées (les temps d'exécution et les ressources qu'il a "visité").

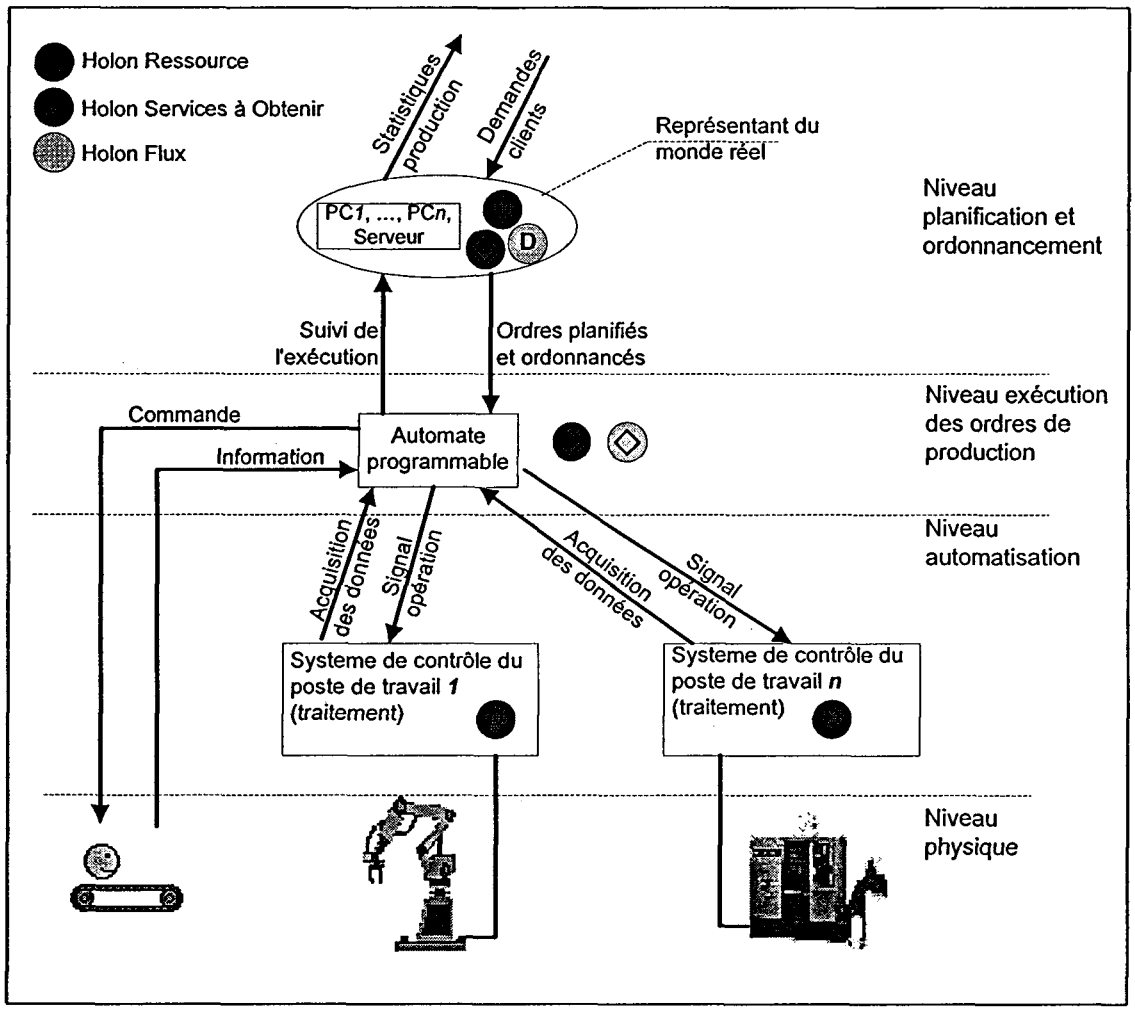


Figure 3.8 : association entre holons (ordre, ressource et produit) et l'architecture physique du système

Pendant la fabrication chaque HF de production passe par trois étapes :

- *en attente* : un ordre qui sera en charge de la fabrication d'un produit a été créé, planifié dans un lot et le lot a été aussi planifié dans un ensemble d'autres lots ;
- *en production* : une palette est associée avec l'ordre client, elle est automatiquement routée par l'automate programmable vers les ressources choisies pour exécuter les opérations nécessaires ; l'exécution successive des opérations crée le produit ;

- *fini* : après avoir quitté le système de fabrication, le produit a été séparé de la palette, et le HF passe dans l'état produit fini et la palette avec sa partie décisionnelle sera réutilisée pour un autre HF de production.

La structure (informations, données et méthodes) d'un HF est présentée en détails ci-dessous :

Informations et méthodes de haut niveau pour l'identification, association des opérations nécessaires, planification et ordonnancement du produit attaché au HF courant. Ces informations, localisées au niveau planification et ordonnancement, ont la structure suivante :

- *Nom* : Chaîne de caractères qui donnent le nom du produit représenté (sti) ;
- *Index* : Identificateur numérique unique dans le système de fabrication, utilisé pour faire référence à ce holon (sti) ;
- *Nombre d'opérations* : Le nombre d'opérations nécessaires pour fabriquer le produit représenté par le holon courant (sti) ;
- *Opérations* : Vecteur qui contient les noms génériques des opérations nécessaires pour fabriquer le produit représenté par le holon courant (sti) ;
- *Précédences* : Matrice qui contient les précédences entre les opérations à exécuter pour fabriquer le produit représenté par le holon courant (sti) ;
- *Informations sur les ressources composantes du système de fabrication* : cette partie contient le modèle du système de fabrication (ressources de transport, traitement, contrôle) qui a été utilisé pour la planification et ordonnancement de cet HF de production. Le modèle peut être représenté par un graphe comme celui de la Figure 3.4 (nombre de ressources, liaisons entre les ressources et types des ressources) avec les opérations exécutées par chaque ressource et leurs temps d'exécution (sti) ;
- *Informations utilisées dans le processus d'allocation* (sti) :
    - *Localisation courante* de la partie informationnelle du HF dans le simulateur du système de fabrication ;
    - *L'état courant du holon* dans la planification : planifié, non planifié et en cours de planification;
    - *L'état courant de chaque opération* : en cours d'allocation ou non allouée ;
    - La *ressource courante* sur laquelle se trouve la partie informationnelle du HF dans le simulateur du système de fabrication ;
    - L'*opération courante* ;

- *Le temps qui reste pour que l'opération courante soit terminée.*

- *Informations qui résultent des processus de planification et ordonnancement* :
  - *Les opérations ordonnées* selon leur date d'exécution (sti) ;
  - *Les ressources choisies* pour offrir les opérations précédentes (sti) ;
  - *Les dates d'exécution des opérations* – les temps (planifiés) auxquels le HF doit visiter les ressources (sti).

- *Méthodes et fonctions pour le traitement des informations embarquées sur le HF courant*:
  - *Validation recette HF* : Méthode qui vérifie si la recette du produit associé peut être utilisé dans l'ordonnancement (pas de circuits dans le graphe d'exécution) (sfi_{ND}) ;
  - *Test transport* : Méthode qui teste si le représentant physique du holon a besoin d'une opération de transport pour arriver à la ressource choisie (sfi_{ND}) ;
  - *Insérer opération* : Fonction qui fait l'insertion d'une opération pour être ordonnancée (sfi_{D}) ;
  - *Insérer opération planifiée* : Fonction qui fait l'insertion d'une opération déjà ordonnancée (sfi_{ND}) ;
  - *Durée* : Fonction qui retourne le temps d'exécution d'une opération spécifiée par son nom (sfi_{ND}) ;
  - *Transfert automate programmable* : méthode en charge de communiquer les données du niveau planification et ordonnancement vers le niveau exécution (sei).

Informations et méthodes de bas niveau, localisées dans le niveau exécution des ordres de production ; elles seront utilisées pour la gestion intelligente du système de fabrication :

- *Localisation courante* du HF dans le système de fabrication : un service qui offre une information qui concerne la localisation du HF dans le temps (sti) et l'espace (sei) ;
- *L'état courant du holon* dans la planification : en attente (planifié et ordonnancé), en production et produit fini (sfi_{ND}) ;
- *L'état courant de chaque opération* : en attente qu'un poste de travail reçoit le HF, en production et produit fini (sfi_{ND}) ;
- La *ressource courante* sur laquelle se trouve en réalité la partie physique du HF (sei) ;
- L'*opération courante* (sti) ;

- *Lancement en exécution* de l'opération courante (sep ou sfp en fonction du type d'opération qui est lancée en exécution : transport ou usinage).

Avant que la production commence pour une séquence d'ordres associés à un produit (**Liste Agrégée des Ordres Produits**, LAOP) l'ordre client existe seulement dans un format digital ; pendant la fabrication, chaque ordre est progressivement fabriqué sur une palette sous la forme d'un HF ; après la fabrication le produit est séparé de la palette et il existera dans une forme physique. Les HF sont crées a partir des ordres bruts (entités de la liste LAOP) qui contiennent au début seulement l'information nécessaire à l'exécution des produits. Si un certain produit doit être fabriqué en *n* exemplaires, alors un nombre de *n* ordres bruts sera créé dans un premier temps.

Pour avoir une synchronisation entre la partie décisionnelle de haut niveau qui fait l'allocation des opérations et la partie automatisation qui met en œuvre l'allocation, les indexes écrits sur les étiquettes RFID de chaque HF doivent être uniques au niveau global et respecter une structure standard.

- Le Holon Flux d'Alimentation

Un cas spécial de HF, le **HF d'Alimentation** a la tâche d'apporter des matières premières aux postes de travail en début et pendant le processus de fabrication. Contrairement à un HF de production qui a les opérations ordonnancées a priori, le HF d'alimentation calcule en ligne les opérations à réaliser en fonction du degré d'utilisation des ressources (les ressources plus utilisées seront réalimentées plus fréquemment). En dehors de cela les deux types de holons ont la même structure (fonctionnalités et propriétés). Pour ne pas bloquer le système de fabrication, le nombre des HF d'alimentation doit être le plus bas possible pendant la production.

La partie physique du holon d'alimentation est représentée par la palette qui transporte les pièces dans le système et la partie informationnelle, partagée en deux, est représentée par l'ensemble des applications qui s'exécutent sur le serveur de la cellule et sur l'automate programmable :

- Applications qui s'exécutent sur le serveur de la cellule : elles calculent les chemins des palettes ;
- Applications qui s'exécutent sur l'automate programmable de la cellule : elles contrôlent directement les ressources de transport (moteurs des tapis, aiguillages) ; elles sont aussi en charge avec les messages de synchronisation entre les postes de travail et le poste de (ré) alimentation.

Entre les deux parties il y a une relation 1-1 : pour chaque palette avec un produit en cours de fabrication il y a un seul agent qui la contrôle et la synchronisation entre les deux applications se fait avec un code unique écrit sur une étiquette RFID de la palette.

En fonction de la phase d'alimentation des postes de travail deux types des holons existent : le premier assure l'alimentation au début du processus de production et le deuxième assure la réalimentation pendant le processus de production quand il y a un

manque de matières premières dans un poste de travail (épuisement d'un stock local). La différence entre les deux holons est faite par la procédure de transport de la palette : au début du processus de fabrication il y a seulement des HFs d'alimentation, tandis que pendant le processus de fabrication les HFs d'alimentation qui apparaissent coexistent avec les HFs de production en cours d'exécution.

Aussi, la vie du premier type de holon d'alimentation s'arrête au début de la production, après que les postes de travail ont été alimentés, tandis que le deuxième type existe en principe jusqu'à la fin de la production.

- Le Holon Ressource

Comme dans le cas d'un HF, la partie informationnelle d'un Holon Ressource (HR) est divisée entre les trois niveaux du système de contrôle (Figure 3.8) :

(a) la partie décisionnelle de la ressource (symbole D en cercle rouge au niveau des PC Stations de travail figure 3.8) prend part active au processus de planification et d'ordonnancement des ordres de production en mode hétérarchique, pour remplacer les recommandations produites hors ligne par le PGP ;

(b) le niveau intermédiaire a un rôle de mémorisation (symbolisé par le diamant dans le cercle rouge de la figure 3.8) et d'interface entre le niveau planification et ordonnancement et le niveau automatisation ;

(c) le niveau automatisation est en charge du contrôle directe des ressources physiques (symbole D dans le cercle rouge au niveau du contrôleur de la ressource), y compris la gestion en temps réel des postes de travail (ex : nombre des pièces dans les stocks) et la synchronisation avec le niveau exécution des ordres de production.

L'échange permanent de données entre le PGP et l'automate programmable qui contrôle le système de transport de la cellule assure que la partie décisionnelle du niveau haut utilise pour la planification des informations qui sont en conformité avec la réalité.

Les propriétés et fonctionnalités d'un HR sont dérivées des services génériques présentées dans le chapitre II et se partagent en :

- Informations et méthodes de haut niveau pour l'identification, l'accessibilité des opérations que la ressource est capable d'exécuter et la participation au processus de planification des produits associés aux HFs. Ces informations se trouvent au niveau planification et ordonnancement et concernent seulement les processus qui se déroulent à ce niveau :

- *Nom* : Le nom et l'identificateur de la ressource courante (sti) ;
- *Opérations* : Structure indexée qui contient toutes les opérations que la ressource associée est capable d'exécuter (sti) ;
- *Nombre d'opérations* : Le nombre total des opérations disponibles sur la ressource courante (sti) ;

- *Etat* : L'état de la ressource courante dans le processus de planification et ordonnancement des ordres de production (sti) ;
- *Temps ressource en ligne* : L'intervalle de temps que la ressource est en ligne (sti) ;
- *Fin réservation ressource* : L'intervalle de temps dans lequel la ressource peut accepter un autre ordre (sti) ;
- *Index holon ordre* : L'identificateur unique de l'ordre avec lequel la ressource est occupé au moment actuel (sti) ;
- *Opération courante* : L'identificateur de l'opération en cours de traitement (sti) ;
- *Temps restant* : L'intervalle de temps qui reste jusqu'à la fin de l'opération courante (sti) ;
- *Fonctions* :
  - *Définir une ressource* : Définit les opérations/services qu'une ressource est capable d'accomplir (sti) ;
  - *Transfert ordre* : Transfère l'opération à une ressource et commence son exécution (sei) ;
  - *Opération finie* : Libère la ressource et informe le HF associé que l'opération a été finie (sfi_{ND}) ;
  - *Allouer le temps* : Alloue le temps de traitement d'une opération sur une ressource (sti) ;
  - *Opération échouée* : Marque l'opération spécifiée comme non disponible (sfi_{ND}) ;
  - *Opération rétablie* : Marque l'opération spécifiée comme disponible (sfi_{ND}).

- Informations et méthodes de bas niveau, localisées dans les niveaux "exécution des ordres de production" et "automatisation", et qui seront utilisées en temps réel pour la gestion intelligente du système de fabrication. Ces informations qui concernent le processus de fabrication réel sont :

- *Etat* : L'état de la ressource courante (sei) ;
- *Opérations* : Structure indexée qui contient toutes les opérations que la ressource associée est capable d'exécuter (sei) ;
- *Fonctions* :
  - *Contrôle de la ressource attachée* : Fonction en temps réel du HR qui assure le contrôle directe de la ressource physique (ex : le contrôle de la trajectoire d'un robot industriel) (sfi_{ND}) ;
  - *Interface avec l'extérieur* : Cette partie est en charge avec la standardisation de l'interface de communication (interconnexion physique, protocole de communication et protocole d'interaction) et de l'échange des données avec

l'extérieur (ex : transfert des ordres (un HF demande à un HR de recevoir une opération), signalisation état opération – opération finie, opération échouée, etc.) (sei) ;

- *Gestion des stocks* : Fonction qui s'occupe de l'alimentation et réalimentation des postes de travail en matières premières. Ainsi, cette fonction est en charge avec la mise à jour des stocks (nombre et emplacement des pièces) et avec la signalisation entre la ressource courante et le poste de réalimentation (sfi_{ND}).

Il y a trois catégories de HR selon le type de service de transformation de forme (Figure 3.1) qu'ils offrent aux holons HF :

i. Usinage (sfp)

Ce type d'opération est exécuté par une machine à commande numérique (MCN), pour un seul produit/pièce à la fois. Elle nécessite un ou plusieurs outils et est nécessairement précédée par une opération de type "alimentation MCN", "changement d'outil" et suivie par une opération de type "déchargement MCN". Les opérations alimentation et déchargement MCN sont effectuées par le robot industriel associé au poste de travail.

Une opération d'usinage est de deux types : traitement extérieur d'une pièce (l'aspect) et traitement intérieur d'une pièce (perçage, filetage) pour permettre de monter sur elle d'autres pièces. En dehors du type d'opération la MCN a besoin des informations suivantes :

- Le type du produit sur lequel elle intervient ;
- Le matériel du produit sur lequel elle intervient ;
- La séquence d'opérations atomiques que la MCN doit exécuter pour offrir le service requis.

ii. Manutention des pièces ("material handling") (sfp)

L'opération de manutention des pièces avec le robot est utilisée pour plusieurs tâches, comme décrit ci-dessous :

- *Alimentation MCN* : opération exécutée automatiquement par un robot quand une pièce qui devra être assemblée sur le produit nécessite d'être usinée sur la MCN. Après l'alimentation de la MCN, les deux ressources (MCN, robot) fonctionnent indépendamment jusqu'au déchargement de la pièce ;
- *Déchargement MCN* : opération exécutée automatiquement par un robot quand la MCN a fini le traitement d'une pièce. La pièce est déposée soit dans un dépôt, soit directement sur le produit à réaliser ;
- *Alimentation dépôt du poste de travail* : opération exécutée par un robot pour remplir son propre dépôt ou le dépôt d'une autre ressource (ex : celui de la MCN)

- *Assemblage/désassemblage produit* : opération de base exécutée par un robot industriel – ici utilisée pour l'assemblage et le désassemblage des produits ;
- *Changement outil MCN* : similaire à l'opération d'alimentation, mais cette fois au lieu de remplir le dépôt, le robot est utilisé pour automatiser le processus de changement d'outil ;
- *Traitement physique robotisé d'un produit* : en montant un outil sur le robot, il peut exécuter des traitements physiques sur un produit (ex : peinture, finition, etc.).

La description générique de l'opération d'assemblage d'un produit avec le robot, décrite ci-dessous, met en évidence les services qu'un HR de type robot peut offrir et doit être complétée lors de la création du holon ordre correspondant avec toutes les valeurs numériques des paramètres associés (spécifiques pour le robot assigné par l'ordonnancement y compris son environnement de travail et pour la séquence d'opérations de montage à réaliser sur le produit).

- **Fixer un sous ensemble** (FIX(*pr*, [*sub*], [*el.fix*], [*tool*], *rob*)): fixer avec le robot un ou plusieurs composants / un ou plusieurs sous ensembles [*sub*] qui ont été antérieurement montés sur le produit *pr*, éventuellement à l'aide d'un ou de plusieurs éléments de fixation [*el.fix*] (vis, couvercle, etc.) et éventuellement e n utilisant un ou plusieurs outils [*tool*] (tourne vis, axe, etc.) ; l'opération nécessite 1 ou plusieurs programmes spécifiés en #IDprg( ), avec 1 ou plusieurs routines de type *pick and place* (p_p):

  FIX(*pr*, [*sub*], [*el.fix*], [*tool*], *rob*)):

  [pr(#IDpr, loc.pr); [sub(t, loc.sub, [mod.fix, el.fix, tool])]; ...

  ... [el.fix(t, stor.fix, loc.fix)]; [tool(t, tool.hld, loc.tool)]; ...

  ... robot(tip.rob, tip.grip, TOOL); mesaj()]

  - pr(#IDpr, loc.pr): code du produit transporté par la palette qui sera amenée par le HF en exécution dans le poste de travail du robot assigné; un nombre sous ensembles déjà montés sur le produit, et qui sont décrites en [sub( )], doivent être fixés à l'aide des éléments de fixation décrites en [el.fix( )] par le robot de type tip *tip.rob* assigné en (#IDrob), qui utilise les instruments de fixation décrites en [tool( )]
  - [sub(t, loc.sub, [mod.fix, el.fix, tool])]: caractérise les sous ensembles qui seront fixés :

    [...; (tip_*i*.sub, loc_*i*.sub, [mod_*ij*.fix, loc_*ij*.fix, [tr.fix_*ij*], el_*ij*.fix, tool_*ij*]); ...

    ... | $1 \le i \le n.s$, $1 \le j \le n.fix$]

    - tip_*i*.sub, $1 \le i \le n.s$: le type du sous ensemble qui sera fixé ;
    - loc_*i*.sub, $1 \le i \le n.s$: l'emplacement robot de montage du sous ensemble no. *i*

sur le produit pr(#IDpr) ;

- [mod_*ij*.fix, loc_*ij*.fix, [tr.fix_*ij*], el_*ij*.fix, tool_*ij*]: description du mode de fixation du sous ensemble no. *i* :
  - mod_*ij*.fix, $1 \le j \le u_i$: les modes de fixation du sous ensemble no. *i* (ex presser, visser, monter couvercle et tourner, etc.);
  - loc_*ij*.fix, $1 \le j \le u_i$: emplacement de l'outil / de la pince robot pour la fixation no. *j* du sous ensemble no. *i* ;
  - [tr.fix_*ij*] | = $\{P_{0j}\}(\{P_{1j}\}\text{pat.type}_1),\ldots,(\{P_{u_j}\}\text{pat.type}_{u_j})$], $1 \le j \le u_i$

    où :
    - $\{P_{0j}\}, 1 \le j \le u_i$: point initial / final respectivement sur les trajectoires d'approche (*appro*) / de départ (*departs*) lors de la fixation no. *j* du composant no. *i* ;
    - $\{P_{1j}\}\{P_{2j}\},\ldots,\{P_{u_j}\}, 1 \le j \le u_i$: points robot définis sur les trajectoires *appro/departs* lors de l'opération *j* de fixation ;
    - pat.type_k, $1 \le k \le u_j$: type du segment de trajectoire *appro* / *departs* $\text{tr}(P_{(k\ 1)i}P_{ki})$ lors de la procédure de fixation no. *j* du composant no. *i* ;
  - el_*ij*.fix, $1 \le j \le u_i$, $1 \le i \le n.s$: type de l'élément de fixation utilisé pendant l'opération de fixation no. *j* du sous ensemble no. *i* ; si aucun élément de fixation n'est utilisé alors #IDel_*ij*.fix = 0 ;
  - tool_*ij*, $1 \le j \le u_i$, $1 \le i \le n.s$: type de l'outil utilisé pour fixer le sous ensemble *i* pendant l'opération *j* avec l'élément el_*ij*.fix ; si aucun élément de fixation n'est utilisé en opération *j*, alors #IDtool_*ij* = 0.

- [el.fix(t, stor.fix, loc.fix)] : vecteur de données qui décrit les *f* types d'éléments de fixation qui sont nécessaires pour les *ns* sous ensembles du produit *pr* :

  [...; (tip_*k*_el.fix, stor_*k*.fix, loc_*k*.stor); ...], $1 \le k \le f$

  - tip_*k*_el.fix, $1 \le k \le f$: le type de l'élément de fixation ;
  - tip_*k*.stor, $1 \le k \le f$: le type de magasin qui contient des composants de fixation de type tip_*k*_el.fix;
  - loc_*k*.stor, $1 \le k \le f$: l'emplacement d'où le composant de fixation de type tip_*k*_el.fix sera extrait du magasin défini par tip_*k*.stor.

- [tool(t, tool.hld, loc.tool)]: vecteur de données qui caractérise les *nt* types d'outils de fixation qui sont nécessaires aux *ns* sous ensembles du produit *pr* :

  [...; (tip_*m*.tool, tool_*m*.hld, loc_*m*.tool, *m*_TOOL); ...], $1 \le m \le nt$

  - tip_*m*.tool, $1 \le m \le nt$: le type d'instrument utilisé pour la fixation ;

- tool_m.hld, $1 \leq m \leq nt$: le type de magasin d'outils robot pour tâches de fixation ;
- loc_m.tool, $1 \leq m \leq nt$: emplacement de prélèvement de l'outil pour la fixation d'un élément de fixation *el.fix* ou directement du sous ensemble ;
- m_TOOL, $1 \leq m \leq nt$: transformation associée à l'outil de fixation de type tip_m.tool.

❖ robot(tip.rob, tip.grip, TOOL) : description du robot utilisé pour l'opération FIX( ):

- tip.rob: type du robot qui est capable de fixer les sous ensembles du type tip_i.sub,

  $1 \leq i \leq n.s$ directement ou par éléments de fixation du type tip_k_el.fix, $1 \leq k \leq f$ directement ou avec des outils du type tip_m.tool, $1 \leq m \leq nt$ (COBRA, VIPER);
- tip.grip: type de la pince pour l'opération FIX( ) exécutée avec un robot du type *tip.rob* (à mors / à ventouse à vide, mouvement des mors : axial / parallèle, forme symétrique / asymétrique des mors, no. des mors, forme et empreinte des mors ;
- TOOL: transformation associé *tip.grip* au décalage TMF (flanche de fixation de la pince – *tool mounting flange*) – pince.

❖ message(STATUS (FIX), timp.r) où :

- STATUS (FIX) :
  - ♣ *Terminé* (ok / nok(cod.er)):
    - □ ok : fin normale de l'opération FIX(*pr*, [*sub*], [*el.fix*], [*tool*], *rob*);
    - □ nok : fin anormale (abandon), avec transmission du code d'erreur *cod.er* ;
  - ♣ *En exécution* ;
  - ♣ *Suspendue* (IN(adr_AP)) : dans l'attente :
    - □ de la fin d'une opération de changement d'outil SCT(*old.tool, new.tool, rob*), qui doit être insérée chaque fois quand deux opérations successives de fixation $j, j+1$ s'exécutent avec des outils différents, c'est-à-dire tip_j.tool $\neq$tip_($j+1$).tool; SCT( ) est exécutée et confirmée par le même robot du type *tip.rob*(#IDrob) qui exécute aussi l'opération de fixation discutée, FIX( );
    - □ d'une approvisionnement avec éléments de fixation du type tip_k_el.fix (un HF d'alimentation sera crée et exécuté quand un épuisement de composants *el.fix* est détecté, c'est-à-dire quand le magasin *stor.fix* du type tip_k.stor associé *el.fix* se vide), situation confirmée par l'automate programmable AP qui pilote la cellule, par

un signal qui est transféré à une ligne d'entrée du contrôleur robot assigné (#IDrob) de la ligne adr_AP de l'automate programmable ;

Note:

A. Quand une opération FIX( ) est suspendue dû au manque d'éléments de fixation du type tip_*k*_el.fix dans le magasin tip_*k*.stor(#IDstor) associé, le contrôleur du robot *tip.rob* assigné (#IDrob) exécutera l'opération ASTO( ) – alimentation d'un magasin vide en s'approvisionnant de la palette *supply pallet* arrivée sur le tapis convoyeur dans le poste robot – à la réception d'un signal active en provenant de l'AP sur la ligne IN(adr_AP), après quoi il reprend l'opération suspendue.

B. L'opération FIX( ) doit être suivie automatiquement d'une opération SCT( ) pour déposer l'outil de fixation dans le magasin d'instruments du robot, si la dernière opération de fixation a été réalisée avec un outil du type tip_*m*.tool(#IDtool $\neq 0$).

- timp.r: temps effectif d'exécution de l'opération FIX( ), y compris des opérations de changement d'outils SCT( ) insérées et des éventuelles interruptions ASTO( ) pour l'approvisionnement ; tous ces délais sont cumulés.

Pour toutes les opérations de manutention avec le robot les aspects suivants doivent être définis : les points et le style de saisie des objets, la structure des dépôts (ex piles multi dimension) contenant des pièces : nombre total de pièces, distribution sur les axes des piles, distances entre les pièces et index de la pièce courante, et les programmes robot en détail.

iii. Inspection vidéo (sfi_{ND})

L'opération est exécutée par les ressources de type caméra vidéo avec système de traitement d'images, et sert pour valider qu'un produit a bien été exécuté (mesure de sa géométrie, finition de sa surface) et aussi pour guider le mouvement des robots industriels (suivi d'un gabarit de montage, d'un cordon de soudure etc.).

- Le Holon Services à Obtenir

Ce type de holon contient la recette qui décrit comment un certain type de produit sera fabriqué. En plus des informations qui concernent les opérations requises et la succession de ces opérations, le HSO contient des compléments techniques qui décrivent le produit fini, ses sous-ensembles et les paramètres que les ressources doivent utiliser pour offrir les services requis. L'existence d'un HSO n'implique pas que le produit sera nécessairement fabriqué, mais s'il y a une commande pour un produit de ce type, le HSO sera copié dans un HF qui sera associé à la demande du client. Le HSO est une entité qui :

- intervient seulement au niveau de la planification et de l'ordonnancement ;
- contient des informations statiques (une fois établie la modalité de fabrication elle ne change pas) ;

- contient des méthodes qui sont utilisées seulement hors ligne pour la gestion des produits (ex : créer un nouveau produit, modifier un produit existant et copier le produit dans un HF).

Ainsi, le HSO offre vers l'extérieur les fonctionnalités et les propriétés suivantes sous la forme de services informationnelles :

- Informations et méthodes de haut niveau pour l'identification, pour accéder aux opérations de base et pour manipuler le HF :

- *Attributs publics* :
  - *Plan de fabrication* : Entité informationnelle qui contient la méthode (succession des services) utilisée pour la réalisation d'un produit (sti) ;
- *Propriétés* :
  - *Index holon* : Identificateur unique de l'ordre (sti) ;
  - *Le nom* du produit (sti) ;
  - Le *nombre total* d'opérations à réaliser pour que le produit soit fabriqué (sti) ;
  - *La quantité* dans laquelle le produit sera fabriqué pour un client (sti) ;
  - *La priorité* d'un produit demandé par le client (sti) ;
  - *Le nombre des produits* du même type qui ont été déjà *fabriqués* (sti) ;
  - *Le nombre de produits* du même type qui ont été *détruits* pendant la fabrication (sti).
- *Fonctions publiques* :
  - *Définir produit* : Définit un produit avec son nom, la quantité nécessaire, la priorité et la structure qui décrit comment les services de base seront obtenus (sfi_{ND}) ;
  - *Vérifier plan fabrication* : Vérifie si le produit courant peut être exécuté (les services requis sont disponibles) (sfi_{ND}) ;
  - *Accès à l'information* (sei) : fonction utilisée dans le processus d'agrégation permettant d'obtenir le holon HF.

- Le Holon Coordinateur

Le holon coordinateur HC est l'entité qui s'occupe de l'optimisation globale du système (de la planification et de l'ordonnancement au niveau complet des lots de fabrication), de l'accès rapide aux services offerts par les ressources, de la configuration de l'accès aux ressources et de l'interface avec d'autres systèmes qui peuvent être: clients, fournisseurs, autres ateliers, etc. Entre le holon coordinateur, situé sur le niveau planification (Figure 3.7), et le niveau qui contient les holons flux et ressource deux types d'interactions peuvent exister: fortes et faibles, en fonction de la manière dans laquelle les ordres influencent les entités cible. Selon le concept de « **open-control** », les deux types d'interactions sont

associées au *contrôle explicite* pour les interactions fortes quand le contrôle des entités est exercé directement, respectivement au *contrôle implicite* pour les interactions faibles quand le contrôle se fait par influence indirecte ou par des entités intermédiaires.

La *configuration rapide de l'accès aux services* et *sa mise à jour permanente*, ainsi que *l'accès aux services en temps réel* sont réalisées par l'intégration d'un *modèle d'accès aux services* (**MdAaS**) (Borangiu et al., 2010) dans le holon HC (qui le configure et le met à jour, voir la Figure 3.7) et dans les HFs (qui accèdent aux services et informent le holon HC de la nécessité de mettre à jour ce modèle, voir la Figure 3.6). MdAaS assure la réactivité et agilité du système de contrôle relativement à des services prompts et de qualité. Le MdAaS est une représentation complète et réelle de l'état des ressources [en fonction / fonctionnement dégradé / en panne], de leur degré d'occupation, du coût de leurs services, et des bonifications / pénalités accumulées suite aux services offerts jusqu'au temps courant.

Deux fonctionnalités importantes du HC sont introduites par rapport aux ressources valides :

- la création initiale de groupes de ressources capables d'assurer les services requis par les clients à travers des ordres (la configuration orientée ordre du MdAaS) ;
- la mise à jour des groupes (reconfiguration du MdAaS : ajout / élimination d'une ressource, suite à sa remise après une panne / à la dégradation de ses performances).

Les décisions de reconfiguration sont prises automatiquement grâce au suivi en temps réel, réalisé par les holons HF, de :

- *l'histoire du processus* et la *traçabilité des produits* (basées sur les informations provenant des contrôleurs des ressources et les résultats de l'inspection de qualité) ;
- *l'état des ressources* (temps d'exécution des services, résultats des programmes de diagnostic, alarmes, etc.)

Ces deux catégories d'information influencent aussi la décision de commuter entre différents modes de pilotage : hiérarchique, hétérarchique et semi hétérarchique. Dans le cas général représenté en Fig.3.7, le holon HC offre les fonctionnalités suivantes :

- *planification et ordonnancement hors ligne* (recommandations pour le fonctionnement normal du système, sans pannes des ressources) (sfi_D) ;
- *conversion en fichiers standard pour l'automate programmable et transfert* (suivi du mode d'exécution d'une planification valide) (sei) ;
- *replanification en temps réel* (en cas de défaut la planification est refaite basée sur les dernières informations provenant de la cellule réelle) (sfi_D).

Un HC a la structure suivante :

- Un *Planificateur Global de la Production* (PGP) qui est en charge de générer les plans de fabrication associés aux demandes clients sous la forme des informations de production qui seront embarquées sur les HFs. Le PGP contient plusieurs *algorithmes pour la planification* de la production qui sont groupés dans une base de connaissances (Figure 3.10). Aussi, il contient un *simulateur dynamique*, avec période de temps d'échantillonnage variable, qui aide à la visualisation et à la validation de la planification et de l'ordonnancement. Le simulateur comprend un représentant du monde réel en Figure 3.8.
- Les *parties décisionnelles des holons HFs et HRs* (de traitement et transport) qui interagissent entre elles et le MdAaS mis à jour ;
- La base de données avec les *produits* que le système de fabrication est capable d'exécuter (opérations nécessaires, documentation technique, mesures, tolérances, programmes des ressources, modèles de vision, etc.) ;
- *Interfaces* avec l'utilisateur et pour communiquer avec l'infrastructure physique du système de fabrication.

Le holon HC est localisé sur le serveur de la cellule (Figure 3.1), et réparti sur l'ensemble des PC (ordinateurs des postes de travail) interconnectés dans un réseau local de communication ; aussi plusieurs holons HC peuvent exister en même temps mais seulement un seul d'entre eux peut être actif (prendre des décisions) tandis que les autres suivent le processus. La figure 3.9 présente les interactions entre les entités décisionnelles associées aux processus de planification et ordonnancement des produits, ainsi qu'à la gestion des stocks :

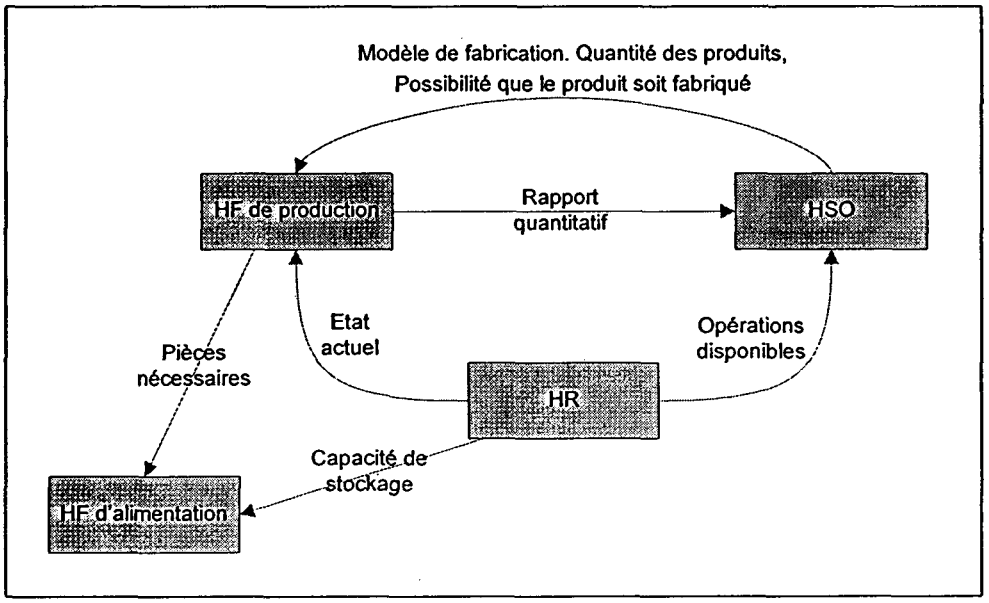


Figure 3.9 : interactions entre les parties informationnelles composantes du système pour la planification, l'ordonnancement et le calcul des stocks

1.4 Modèle dynamique

Comme présenté ci-dessus, le système de pilotage utilise quatre types de holons de base : **le Holon Ressource (HR), le Holon Services à Obtenir (HSO) et le Holon Flux (HF), et un** holon responsable de la gestion générale de la fabrication, le Holon Coordinateur (HC). La communication entre ces différents holons est représentée figure 3.10.

Le processus de production tient compte dans un premier temps des commandes du client, en fonction desquelles sont crées les informations associées aux produits à réaliser. Le client spécifie aussi le nombre de produits nécessaires et la date limite de leur livraison. Ces informations représentent les données d'entrée pour l'entité qui est en charge de planifier et d'ordonnancer la production (HC). Cette entité travaille directement avec les parties informationnelles des ressources, produits et ordres (ordres de production simples ou ordres de réalimentation) : elle reçoit les plans des produits, les états actuels et les capacités des ressources et utilise un algorithme centralisé pour calculer :

- l'allocation des opérations des ordres de production sur les ressources de la cellule et
- le mode d'alimentation des ressources avec des matières premières (si nécessaire).

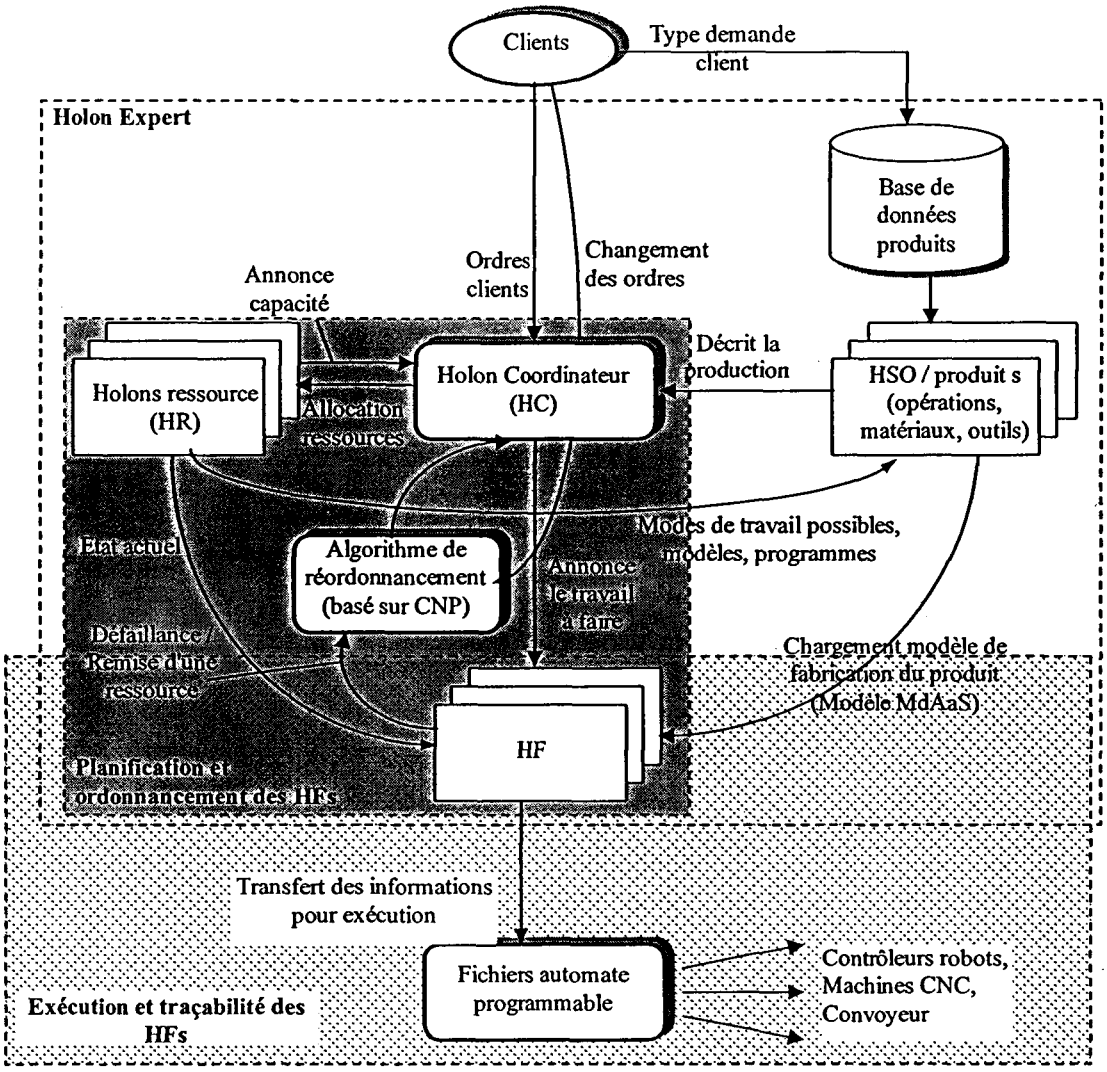


Figure 3.10 : structure de communication entre les holons qui composent l'architecture de contrôle

Le résultat de ce processus est l'ensemble des ordres planifiés (la séquence dans laquelle les ordres de production entrent dans le système) et ordonnancés (l'allocation des ressources aux opérations qui composent les ordres de production). Ces informations sont transférées vers la partie opérationnelle (l'ensemble des automates programmables qui pilotent le routage des palettes avec des produits) sous la forme de fichiers texte standard qui contiennent l'identificateur d'un holon HF de Production et les opérations ordonnancées. Ensuite, c'est l'automate programmable qui contrôle la production selon les ordres reçus et qui informe le HC de l'état des opérations exécutées (réussies, abandonnées) et des ordres (en cours, terminés) pour maintenir la traçabilité des produits.

1.5 La planification et l'ordonnancement des ordres de production

- Algorithmes qui minimisent le temps total de fabrication

Parce que la problématique d'ordonnancement des opérations dans un atelier de type *job shop* est NP complexe, c'est-à-dire l'espace solution augmente d'une façon exponentielle avec la dimension du problème (Garrido et al. 2000), l'utilisation d'un algorithme à base d'heuristique, qui effectuera l'allocation des opérations sur les ressources et supporté par un un simulateur du système de production, a été préférée. Des algorithmes heuristiques sont proposés pour minimiser le temps total de production en utilisant deux stratégies :

- la première stratégie fait une allocation pour que les ressources soient utilisées d'une façon équitable et
- la deuxième stratégie fait une allocation pour que chaque ordre pris individuellement soit exécuté le plus vite possible.

Les deux types de stratégies ont des avantages mais aussi des inconvénients. La première stratégie offre l'avantage d'augmenter la probabilité qu'une ressource critique (une ressource qui est la seule qui peut exécuter un type d'opération) soit libre et que les autres ordres ne forment pas une file d'attente devant cette ressource en bloquant le système de transport. De plus, le fait que les ressources sont utilisées équitablement réduit leur usure. D'autre part, cette méthode utilise d'une façon plus intense la partie transport qui elle-même est une ressource critique, et si les temps de transport sont comparables avec les temps d'exécution des opérations alors le temps total d'exécution du produit augmente beaucoup. La deuxième stratégie a l'avantage de ne pas utiliser de façon intensive le système de transport et permet ainsi aux ordres de circuler plus facilement ce qui réduit le temps total de fabrication.

Dans les deux cas l'algorithme de Dijkstra est utilisé pour calculer le chemin le plus court entre un nœud de départ et un nœud destination en utilisant des données en temps réel, mesurées par les holons HF qui se déplacent dans le système de fabrication (Raileanu et al., 2009).

Trois procédures de planification ont été réalisées en utilisant les caractéristiques des deux types d'algorithmes présentés.

A. Une procédure **StepScheduler** qui fait la planification des ordres de production avec la même date limite de livraison, les opérations impliquées par chaque ordre de production étant ordonnancées pour que chaque ressource soit chargée d'une manière approximativement égale par rapport aux autres. Cette procédure a été complétée avec une version améliorée qui exécute la planification et l'ordonnancement plusieurs fois et garde la meilleure allocation (**StepScheduler Best**). Le nombre de répétitions du StepScheduler influence directement la qualité de la solution (un temps total de fabrication plus bas signifie une qualité supérieure). Les deux paramètres, temps total de fabrication et nombre des répétitions, varient selon une fonction similaire à celle présentée figure 3.11 ci-dessous qui a été obtenue pour un lot de 100 produits. Le

temps d'exécution de l'algorithme est direct proportionnel avec le nombre des répétitions de l'algorithme StepScheduler (Figure 3.11).

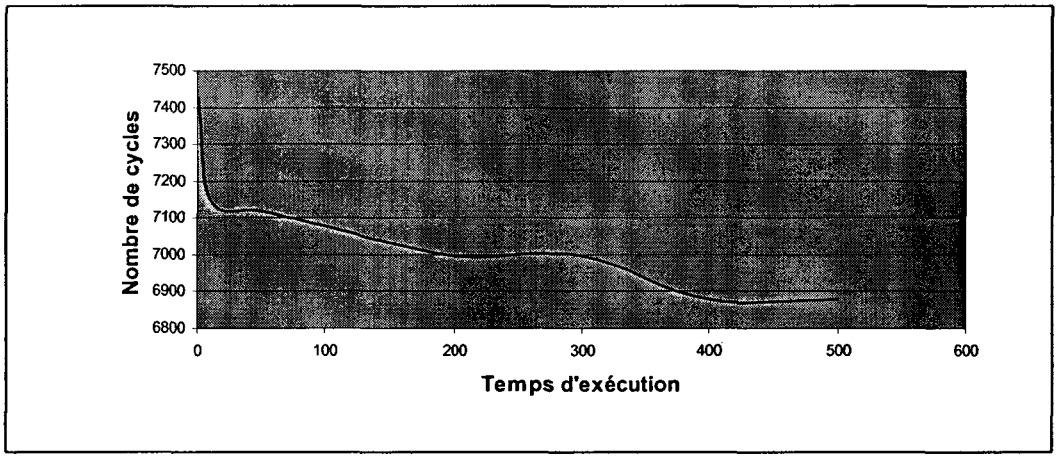


Figure 3.11 : diminution du temps d'exécution (l'exécution planifiée) en fonction du nombre des cycles d'exécution de l'algorithme

Les étapes suivies par la procédure **StepScheduler** pour planifier et ordonnancer les ordres de production sont décrites ci-dessous :

1. Vérifier combien de palettes existent dans le simulateur ; si il y a moins de deux palettes, aller au pas 2, sinon aller au pas 3 ;

2. Choisir un ordre pour la queue d'attente, en tenant compte de la date limite de livraison, et ensuite un ordre aléatoire s'il reste plusieurs ordres ayant la même date limite de livraison ;

   2.1. Pour l'ordre choisi dans le pas 2 générer une liste avec les opérations possibles en tenant compte des priorités et des contraintes entre les opérations ;

   2.2. Pour chaque opération possible, chercher quels sont les postes de travail capables de l'exécuter et calculer les temps d'attente pour que l'opération soit reçue ;

   2.3. Choisir l'opération avec le plus court temps et attacher l'ordre à un représentant informationnel d'un HF. La première destination de cet HF est la ressource qui exécutera l'opération choisie. Mémoriser le temps quand le HF a été introduit dans le simulateur ;

3. Exécuter un pas ayant une unité de temps : *temps* = *temps* + *intervalle_fixe* avec le simulateur du convoyeur et des postes de travail. Si un poste de travail termine une opération aller au pas 4. Si un HF arrive à un poste de travail, aller au pas 6. Si aucune des deux situations n'apparaît, aller au pas 5 ;

4. Pour le HF qui vient de finir une opération mémoriser le temps courant de la simulation comme le temps de fin de cette opération, marquer le poste de travail libre et continuer avec les pas suivants :

4.1. Vérifier si le produit associé au holon HF courant a été terminé. Si c'est le cas, envoyer le HF vers la sortie du système de fabrication et aller au pas 5 ;

4.2. Pour l'ordre attaché au holon HF courant générer une liste avec les opérations possibles en tenant compte des priorités et des contraintes entre les opérations ;

4.3. Pour chaque opération possible, chercher quels sont les postes de travail capables de l'exécuter et calculer les temps d'attente pour que l'opération soit reçue ;

4.4. Choisir l'opération avec la ressource qui peut l'exécuter le plus vite possible et envoyer le HF vers cette ressource ;

5. S'il y a encore des ordres à planifier ou des holons HF dans la simulation aller au pas 1, sinon quitter l'algorithme ;

6. Pour le holon HF courant mémoriser le temps de la simulation, qui sera le temps de démarrage de l'opération ordonnancée, et ensuite transférer le holon HF courant au poste de travail. Aller au pas 1.

B. Une procédure (**StepSchedulerV2**) qui exécute l'ensemble de la planification et de l'ordonnancement des ordres de production en utilisant des règles de priorité. Les règles suivantes ont été utilisées dans l'ordre écrit ci-dessous (si deux ordres ne peuvent pas être différenciés par une règle, alors la règle suivante est testée pour établir la priorité d'un ordre) :

- les ordres déjà insérés dans le système sont prioritaires ;
- les ordres avec le plus grand nombre de successeurs sont prioritaires ;
- les ordres avec le plus petit nombre d'opérations qui peuvent être ordonnancées sont prioritaires ;
- les ordres avec le plus grand nombre d'opérations qui ont des successeurs immédiats sont prioritaires ;
- les ordres avec le plus grand nombre d'opérations qui ne sont pas encore effectuées sont prioritaires ;
- les ordres avec le plus court temps de fabrication sont prioritaires ;
- la dernière règle (si toutes les autres ne peuvent pas être utilisées) est de choisir de façon aléatoire entre les opérations des ordres disponibles (Kusiak, 1990).

Les étapes suivies par la procédure **StepSchedulerV2** pour planifier et ordonnancer les ordres de production sont décrites ci-dessous :

1. Mettre à zéro le temps de la simulation, collecter les opérations que les postes de travail sont capables d'exécuter, collecter les recettes de chaque type de produit demandé et mémoriser toutes les opérations possibles dans une liste S des opérations possibles ;
2. Appliquer les règles de production précédentes dans l'ordre dans leur apparition ;
3. Allouer une opération à une ressource utilisant les règles ci-dessus et modifier son état ;
4. Mettre à jour la liste S (effacer les opérations choisies au pas 3 de la liste S). Si la liste devient vide aller au pas 5, sinon aller au pas 2 ;
5. Mettre à jour la simulation (planification et ordonnancement des holons HF) et incrémenter le temps de la simulation ;
6. Si un poste de travail a terminé une opération, mémoriser l'opération et le temps de fin dans la structure informationnelle du holon HF en cours de traitement ;
7. Mettre à jour les états des postes de travail et de la liste S. Si la liste S est vide aller au pas 5, sinon aller au pas 1.

- L'outil de simulation

Le composant central de l'architecture de contrôle est un simulateur conçu comme *outil multi usage*, utilisé dans les processus suivants :

- *Configuration de la production* : il permet à l'exploitant du système de configurer les lots de production, les plans de production (la méthode de fabrication de chaque produit en termes d'opérations et précédences) et aussi de configurer les ressources ;
- *Planification de la production* : il exécute la planification de la production et l'ordonnancement des ordres de production ;
- *Replanification de la production et réordonnancement des ordres de production* en cas de pannes des ressources : les ordres en exécution sont réordonnancés et les ordres qui attendent d'entrer dans le système sont replanifiés et réordonnancés ;
- *Suivi de la production* : assure la traçabilité des palettes en mouvement dans le système de transport et des activités qui s'effectuent dans les postes de travail.

Ce simulateur dynamique et interactif, contrôlable par l'exploitant, a été adapté au système particulier de fabrication présenté dans la figure 3.1 et intégré dans le programme de pilotage hétérarchique (figure 3.12).

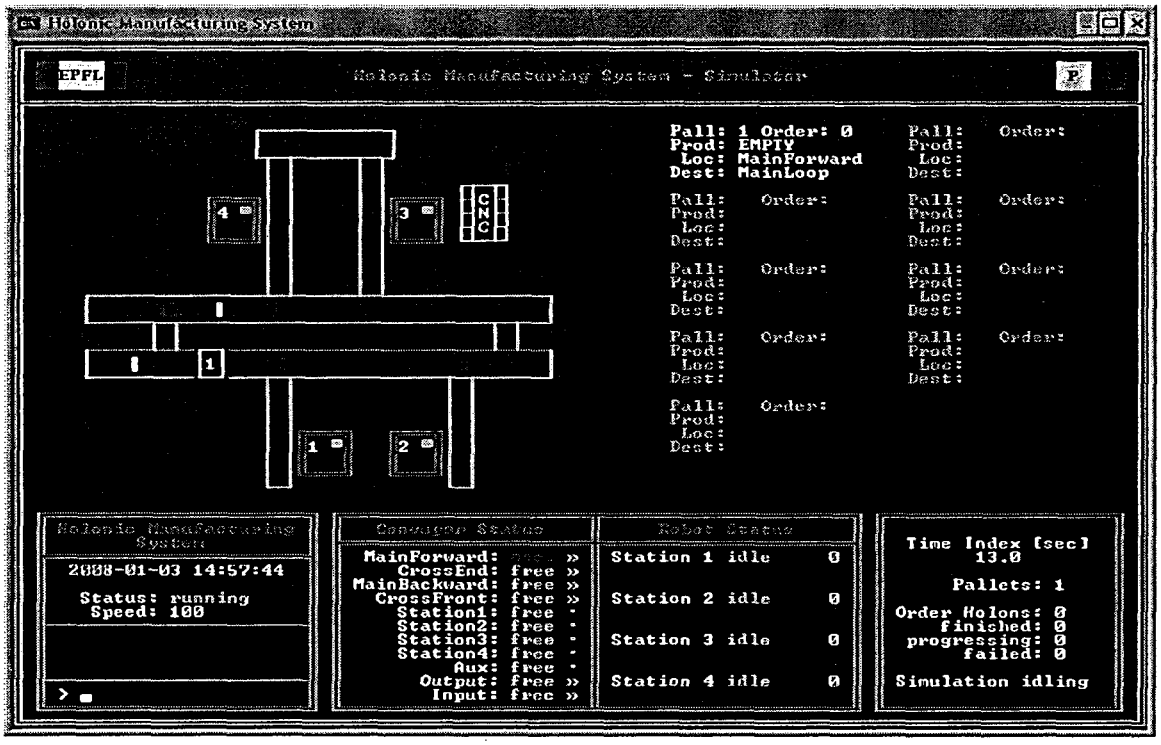


Figure 3.12 : l'interface graphique du simulateur dynamique de la production

Tous les éléments importants du système de transport (ascenseurs, capteurs de présence, mécanismes d'arrêt, moteurs) sont présents dans le simulateur et fonctionnent exactement comme en réalité. Avec ces éléments, le simulateur est utilisé pour valider en ligne la planification du système de fabrication pour la simulation visuelle et pour résoudre le problème de planification et ordonnancement.

Le simulateur fonctionne selon le principe suivant : le programme tourne dans une boucle jusqu'au moment où la production est terminée et à chaque itération, qui a une durée fixe, les palettes sont déplacées dans le système de transport pour l'équivalent de 0.5 secondes en fonction des éléments physiques de contrôle (ex : si le tapis convoyeur marche les palettes sont aussi déplacées, ou si il y a un obstacle devant une palette elle s'arrête).

Dans le cas de la simulation visuelle l'intervalle de temps auquel la boucle se répète est plus grand pour que le fonctionnement du système soit visible à l'exploitant. Par contre, quand le simulateur est utilisé (hors ligne) pour la planification des commandes clients l'intervalle de temps est minimal (de l'ordre de quelques millisecondes, correspondant au temps de calcul du serveur) pour que le processus se déroule le plus vite possible, mais les valeurs réelles du temps sont maintenues et elles sont enregistrées dans la liste des HFs de Production.

1.6 L'association du plan de production aux holons flux dans un format standard pour l'Automate Programmable

Comme décrit en figure 3.8, la partie informationnelle d'un holon HF est distribuée sur plusieurs machines : le modèle de fabrication (qui correspond au modèle d'accès aux

services des ressources) et l'intelligence sont localisés sur le serveur de la cellule sur le HC qui est en charge, de façon optimale, de la planification et de l'ordonnancement, et le résultat est envoyé au niveau « Exécution ordres de production ». Ici, le résultat est mémorisé par l'automate programmable qui est en charge de mettre en œuvre le routage des parties physiques des HFs selon leurs planifications et ordonnancements. Cette information, mémorisée dans un format standard dans l'automate programmable, a la structure suivante :

- Le poste de travail où l'opération sera effectuée ;
- L'opération que doit être effectuée dans le poste de travail spécifié ;
- Le temps de début de l'opération ;
- Le temps dans lequel l'opération doit être effectuée ;
- Le rapport sur l'exécution de l'opération.

Pour simplifier l'échange des données entre la partie informationnelle d'un HF (qui est une partie du HC et qui s'exécute sur un ordinateur OS) et l'automate programmable (qui contient les HFs déjà planifiés et ordonnancés) le protocole OPC (*OLE for Process Communication*) a été choisi grâce à sa facilité d'échange d'informations au niveau des variables. Ainsi, les groupes et les variables suivantes ont été définis :

Variable OPC	Type	Description
.writeOPC	bool	bit écrit par l'automate qui indique que le PGP peut lire les données
.readOPC	bool	bit écrit par le PGP qui indique que l'automate peut lire les données
.stand_by	bool	bit qui arrête l'automate programmable (pause)
.resume	bool	bit qui indique que l'automate programmable peut reprendre le fonctionnement normal
.time_index	ulong	le temps réel du système de fabrication

Tableau 3.1 – Groupe des variables de contrôle :

Variable OPC	Type	Description
.robot1status	byte	état du robot 1
.robot2status	byte	état du robot 2
.robot3status	byte	état du robot 3
.robot4status	byte	état du robot 4
.response_stock_1	byte	le code de la pièce qui manque du stock du robot 1
.response_stock_2	byte	le code de la pièce qui manque du stock du robot 2
.response_stock_3	byte	le code de la pièce qui manque du stock du robot 3
.response_stock_4	byte	le code de la pièce qui manque du stock du robot 4
.production_finished	bool	variable qui indique la fin de la production

Tableau 3.2 – Groupe pour la synchronisation d'état entre les ressources physiques et leurs parties informationnelles

Variable OPC	Type	Description
.order	byte	l'ordre courant qui est écrit/lu
.time_insertion	string("T#xxxms")	le temps (par rapport au début de la production) quand l'ordre associé sera inséré
.failed	bool	variable qui indique que l'ordre associé a été endommagé
.op_index	byte	l'opération courante

Tableau 3.3 – Groupe pour le contrôle des ordres de production échangés

Variable OPC	Type	Description
.poste1	byte	station robot (1-4, 255 pour la sortie du système)
.operation1	byte	l'index de l'opération à exécuter
.temps1	byte	le temps d'exécution pour l'opération associée
.poste2	byte	station robot (1-4, 255 pour la sortie du système)
.operation2	byte	l'index de l'opération à exécuter
.temps2	byte	le temps d'exécution pour l'opération associée
.poste3	byte	station robot (1-4, 255 pour la sortie du système)
.operation3	byte	l'index de l'opération à exécuter
.temps3	byte	le temps d'exécution pour l'opération associée
...		
.poste15	byte	station robot (1-4, 255 pour la sortie du système)
.operation15	byte	l'index de l'opération à exécuter
.temps15	byte	le temps d'exécution pour l'opération associée
.poste16	byte	station robot (1-4, 255 pour la sortie du système)
.operation16	byte	l'index de l'opération à exécuter
.temps16	byte	le temps d'exécution pour l'opération associée

Tableau 3.4 – Le groupe pour écrire/lire un seul ordre de production (note : un nombre maximal de seize opérations a été établi a priori)

Variable OPC	Type	Description
.index_pal_1	byte	l'index de l'ordre associe à la première palette
.index_pal_2	byte	l'index de l'ordre associe à la deuxième palette
.index_pal_3	byte	l'index de l'ordre associe à la troisième palette
.index_pal_4	byte	l'index de l'ordre associe à la quatrième palette
.location_pal_1	byte	emplacement de la première palette
.location_pal_2	byte	emplacement de la deuxième palette
.location_pal_3	byte	emplacement de la troisième palette
.location_pal_4	byte	emplacement de la quatrième palette

Tableau 3.5 – Le groupe pour synchroniser l'état réel du système de fabrication avec la partie informationnelle

1.7 Le processus de réordonnancement des ordres de production

Le holon expertise HC distribué sur les PC des stations de travail, qui surveille en permanence l'état des ressources et la disponibilité des opérations qu'elles peuvent réaliser, est en charge de refaire la planification et l'ordonnancement des holons ordres en cas de perturbation. Cette caractéristique permet au système de fabrication de continuer à fonctionner même en présence de défauts. En fonction de l'état courant du système de fabrication (MdAaS) et de l'événement qui le perturbe (une ressource tombe en panne / reprend son fonctionnement / ne dispose plus de stock), le HC distribué a le comportement représenté en Figure 3.13.

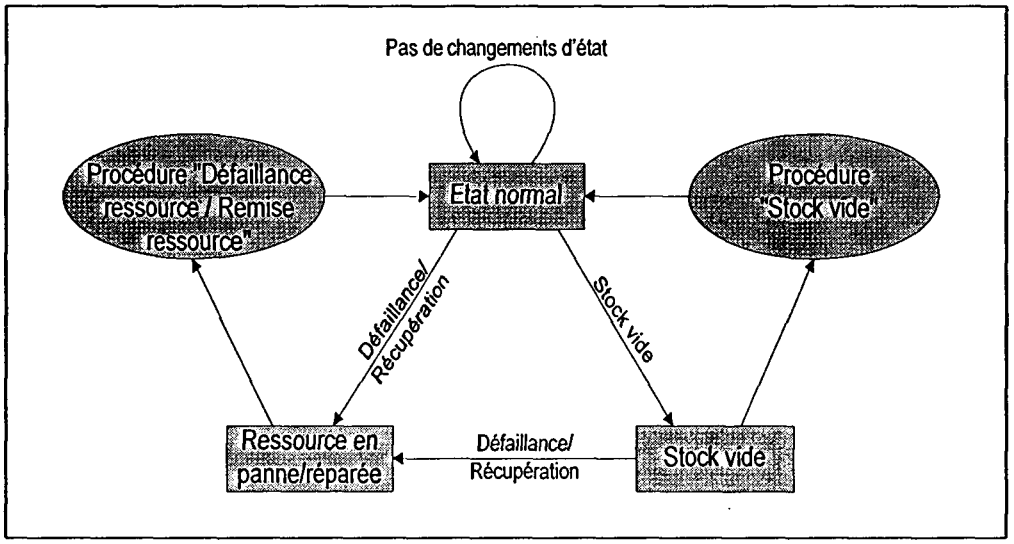


Figure 3.13 : comportement en cas de changement d'état des ressources

Une ressource peut être dans un des états suivants : fonctionnelle avec matières premières, fonctionnelle sans matières premières et en panne. En fonction de cet état une ressource sera utilisée ou pas dans le réordonnancement fait par le HC distribué pour les

ordres déjà introduits en fabrication. Les perturbations évoquées sont détectées et traitées automatiquement sans que l'exploitant du système intervienne.

- Le réordonnancement pour réagir aux changements d'état des ressources de traitement

La gestion des ressources en panne

La procédure utilisée pour la gestion des défauts des ressources (figure 3.14) de traitement (machines CNC, robots, systèmes vision) est la même pour les deux types de perturbations : *défaillance d'une ressource* respectivement *reprise de ses fonctions*.

L'état des ressources est vérifié en permanence et si un comportement incohérent est détecté alors la planification est refaite. Les actions qui se déroulent au niveau *planning* sont :

1. Arrêt de l'introduction de palettes (parties physiques des holons HF) dans le système de fabrication ; les opérations de transport et de traitement en cours de déroulement sur les ressources en ligne continuent ;

2. Mise à jour des parties informationnelles des holons HR avec les nouveaux états des ressources (mis à jour dans le MdAaS) ;

3. Lecture des holons HR en cours d'exécution dans le système (le lot courant des holons HF) ;

4. Evaluation de la possibilité de fabrication de chaque holon HF du lot en testant son état réel dans le système de fabrication :

   4.1. Si le holon HF est dans un poste de travail qui est tombé en panne, alors il sera marqué comme endommagé et ensuite évacué ;

   4.2. Si le holon HF est en cours de fabrication, mais qu'il ne peut pas recevoir certains services (opérations d'usinage, montage, vérification) à cause d'une ressource critique (une ressource qui peut réaliser des opérations qu'aucune autre ressource ne peut fournir) qui est tombée en panne, il sera marqué comme endommagé et évacué ;

   4.3. Si le holon HF est en attente d'entrer dans le système de fabrication physique mais qu'il ne peut pas recevoir certaines opérations à cause d'une ressource critique qui est tombée en panne, il sera marqué comme endommagé et il ne sera plus introduit dans le système physique.

5. Localisation des n'_{wip} holons HF en cours d'exécution qui peuvent être réordonnancés ; après la localisation, une simulation sera débutée avec la configuration courante du système de fabrication. Démarrage de la procédure de négociation inspirée du *Contract Net Protocol* pour le réordonnancement des n'_{wip} holons HF valides ;

6. Démarrage de la procédure de négociation inspirée du *Contract Net Protocol* pour la planification et l'ordonnancement des n''_{wip} holons HF valides qui n'ont pas été introduits dans le système de fabrication physique ;
7. Mise à jour des holons HF dans le niveau exécution des ordres de production ;
8. Reprise de la production.

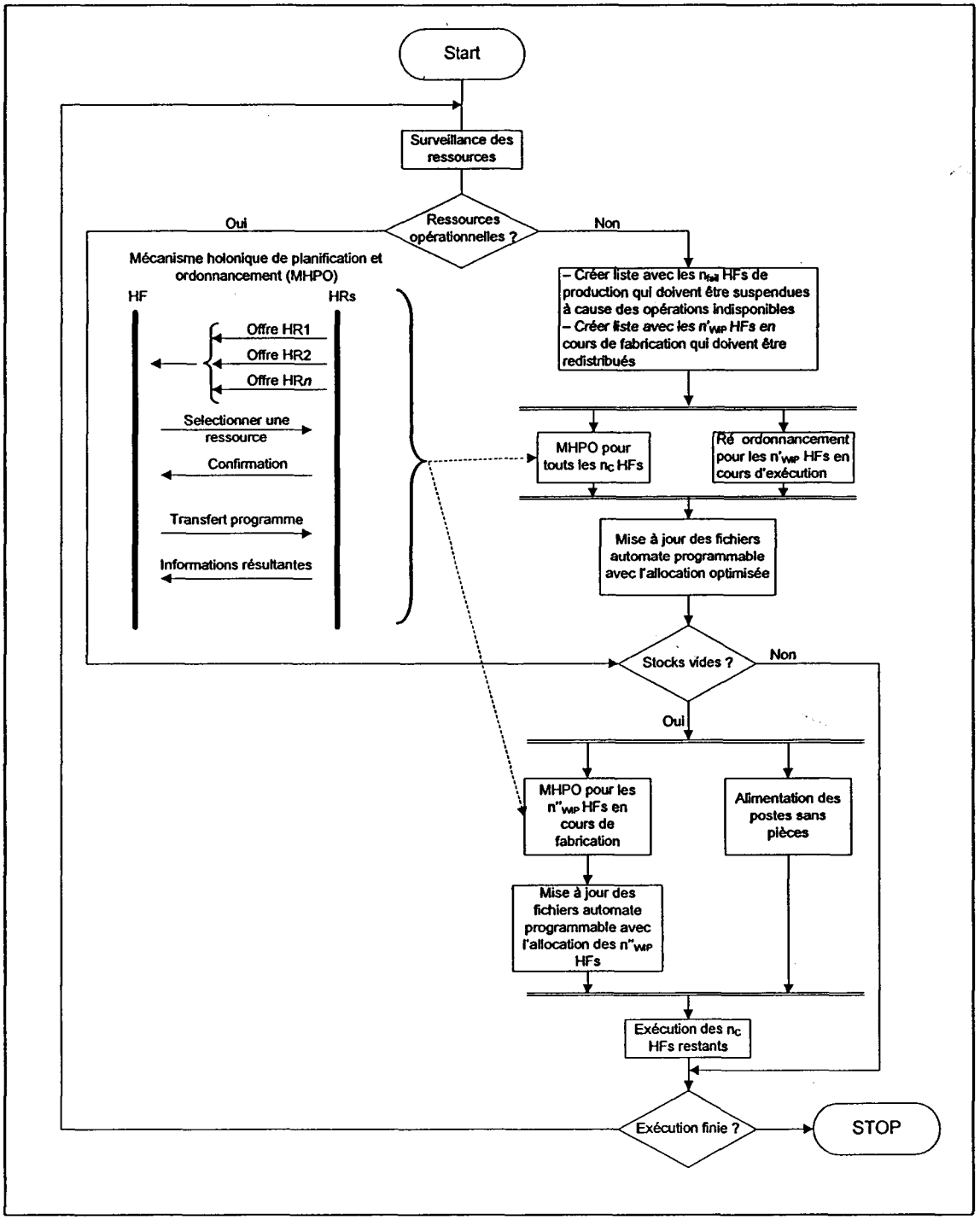


Figure 3.14 : replanification des opérations en cas de panne / alimentation en ligne des ressources

Dans la Figure 3.14 les notations suivantes ont été utilisées :

- n'_{wip} : les holons HF en cours d'exécution, qui peuvent être fabriqué après la défaillance d'une ressource ;
- n''_{wip} : les holons HF qui peuvent être fabriqués après la défaillance d'une ressource ;

- Les processus d'alimentation et réalimentation en ligne des ressources de traitement

La gestion d'un poste qui n'a plus de matières premières (épuisement d'un stock local)

- Avant qu'une palette entre dans un poste de travail, son représentant informationnel (l'image du holon HF résidant dans l'automate programmable AP) débute un dernier dialogue avec le contrôleur de la ressource de traitement du poste concerné pour valider l'opération qui devra être exécutée (figure 3.15). Ce dialogue peut avoir trois résultats :
- L'opération est validée et la production continue selon l'ordonnancement ;
- L'opération n'est pas validée à cause d'une panne très récente de la ressource ;
- L'opération n'est pas validée à cause d'une rupture très récente d'un stock local de matières premières dans le poste concerné – poste temporairement sans pièces.

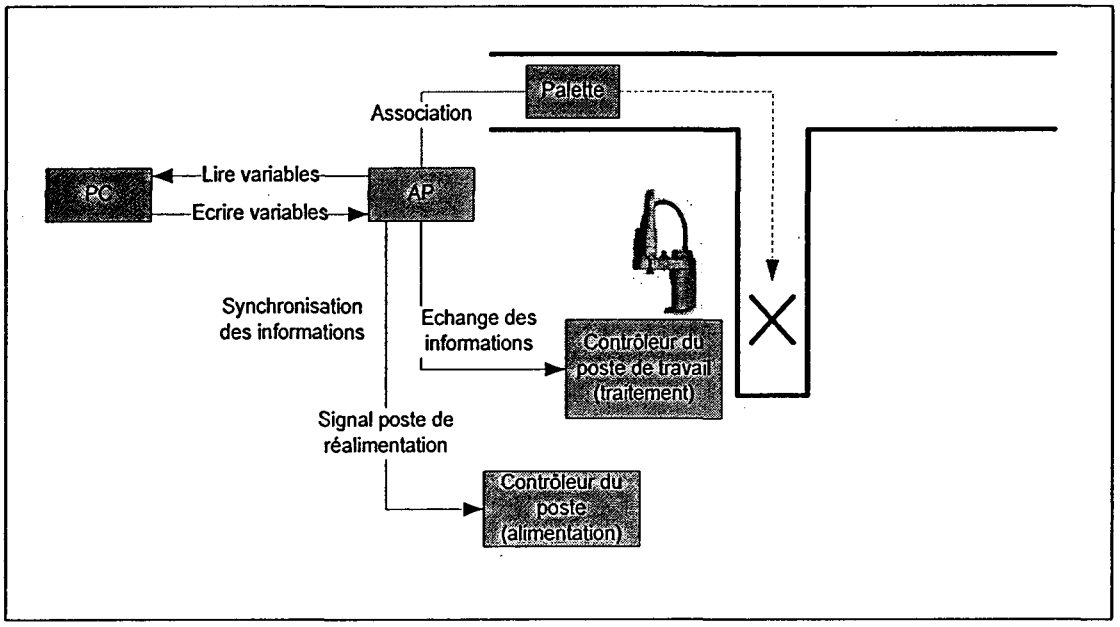


Figure 3.15 : une palette débute un dialogue avec la ressource avant d'entrer dans un poste de travail

L'état « ressource sans pièces » a été introduit parce que cette situation est traitée différemment par rapport à une panne : une replanification est faite pour les ordres qui ont été déjà introduits dans le système, aucun nouvel ordre n'est introduit dans le système, et l'alimentation en ligne du « poste sans pièces » est immédiatement débutée à l'aide d'un ou de plusieurs holons HF d'Alimentation. Une fois cette alimentation réalisée et le cycle de vie des holons HF d'alimentation fini, de nouveaux ordres de production peuvent être introduits. Cette procédure suit les étapes suivantes :

1. Le logiciel qui s'exécute sur l'automate programmable et qui surveille les ressources signale qu'il n'y a plus de pièces d'un certain type dans un poste de travail. Cette situation est détectée quand une palette est en train d'entrer dans le poste de travail ; la procédure d'alimentation en ligne est alors démarrée ;

La procédure de réalimentation comprend les pas :

1.1. Le système de transport est arrêté et les ordres de production en exécution sont évalués ;

1.2. Les holons HF lus en 2.1 sont réordonnancés sur les ressources disponibles, si possible. Si le réordonnancement n'est pas possible (il n'y a aucune autre ressource qui peut exécuter une certaine opération sauf la ressource qui ne dispose plus de matériaux) alors l'ordre est maintenu en attente sur la boucle principale de transport jusqu'au moment où son exécution devient possible dans un poste de travail ;

1.3. Les nouvelles parties informationnelles des holons HF réordonnancés sont transférées à l'automate programmable en charge de l'exécution ;

1.4. Un holon HF d'alimentation en ligne est introduit dans le système pour apporter les pièces au poste de travail identifié en pas1 ;

1.5. L'introduction de nouvelles palettes dans le système est bloquée jusqu'à la fin de la production du lot courant ;

1.6. Les informations des capteurs et dispositifs RFID du convoyeur pour le suivi de l'exécution des holons ordre de production du lot courant et du holon d'alimentation en ligne sont envoyées vers l'automate programmable ;

1.7. La production du lot courant est reprise et les holons HF de production (qui ont été calculés en ligne par le holon HC distribué sur les PC des postes de travail) sont exécutés en parallèle avec le holon HF d'alimentation en ligne ;

1.8. Quand l'alimentation et l'exécution des produits affectés par l'événement « stock vide » sont terminées, le système est déverrouillé et de nouvelles palettes peuvent entrer dans le système ;

2. Reprise de la production planifiée et ordonnancée au début par le holon HC.

- Le réordonnancement pour insérer de nouveaux ordres client

- Le traitement des ordres rapides

La *planification en ligne des ordres de production* consiste en la répartition des ordres sur les ressources d'une cellule de fabrication flexible au fur et à mesure que les commandes du client arrivent et sont introduites dans le système (Albers, 1999). Ainsi, l'application responsable de la planification en ligne prend des décisions (planifie des ordres) sans prévoir des éventuelles autres requêtes qui pourraient arriver. L'avantage de cette méthode par rapport à la planification hors ligne est la *flexibilité* : elle permet un fonctionnement cohérent en présence d'événements et de tâches partiellement connues, mais sans doute n'assure pas un comportement optimal au niveau global des lots de production.

Dans ce cas, c'est le client qui introduit les ordres (un ordre de production représente la demande acceptée de fabrication d'un type de produit). Le client spécifie aussi la date de livraison. Les hypothèses de travail sont :

- le client groupe ses demandes en *lots de production* caractérisés par une date de livraison ;
- la production peut être arrêtée au niveau d'un lot de production ;
- la production ne peut pas être arrêtée au niveau d'un ordre de production : une fois commencée l'exécution d'un ordre, elle ne peut pas être arrêtée ;
- de nouveaux ordres (lots de production avec dates de livraison associées) peuvent apparaître à tout moment et seront pris en compte.

On propose l'approche suivante qui optimise la planification en ligne des ordres de production : disposant d'un système flexible de production de type *job shop*, les ordres de production seront planifiés ainsi que le temps de production (*makespan*) minimisé ou proche du minimal, en autorisant aussi l'introduction, si possible, de nouveaux ordres (au niveau de lots de production) sans perturber les dates de livraison limite des lots déjà planifiés et en cours d'exécution. Si le temps calculé pour traiter une nouvelle demande en présence d'autres demandes déjà planifiés est plus grand que l'intervalle de temps jusqu'à la date de livraison requise, alors cette nouvelle demande est refusée.

Autrement dit, si le système n'est pas utilisé à 100% de ses possibilités sur un certain intervalle de temps qui se termine à la date limite de livraison la plus éloignée, on autorise le calcul à évaluer la possibilité d'insérer de nouveaux ordres. Si l'insertion est possible, tous les lots de production (anciens et nouveaux) doivent être replanifiés et insérés de façon que chaque lot ne dépasse pas la date de livraison désirée. Si l'insertion n'est pas possible, une nouvelle date de livraison sera éventuellement proposée au client.

- Définition de l'allocation en ligne

La planification en ligne des tâches est un problème souvent rencontré dans les systèmes d'exploitation (Albers, 1999), en particulier les systèmes d'exploitation en temps réel où certaines tâches, critiques en général, doivent être exécutées dans une période fixe de temps (ex : dans une vidéo conférence les données qui représentent le son et les images

sont cryptées, envoyées au client et ensuite décodées. Si le signal n'est pas décodé en temps réel les traitements deviennent inutiles puisque que l'information est restituée trop tard). Par analogie les choses se passent de la même manière dans les systèmes de fabrication : si un ordre de production n'est pas exécuté dans un intervalle de temps donné le client peut renoncer à cet ordre ou, éventuellement, accepter sous conditions restrictives. Pour cette raison, la notion d'engagement apparaît : l'opérateur suit une planification et informe le client qu'il s'engage à livrer les produits dans l'intervalle fixé par accord mutuel.

Il y a quatre directions principales pour résoudre la planification en ligne des ordres (http://www.edc.ncl.ac.uk/highlight/rhfebruary2007.php) : (1) la planification en utilisant des priorités fixes; (2) la planification en utilisant des priorités dynamiques; (3) la planification en utilisant des priorités *soft* pour applications en temps réel; (4) la planification en utilisant la réaction.

Prenant en compte les algorithmes précédents on a décide d'utiliser l'algorithme EDF (Sha L., 2004) pour traiter les demandes urgentes d'introduction de nouveaux lots de production dans un contexte de lots de produits déjà planifiés. Ce choix est justifié par la similitude entre les deux problèmes de planification – des tâches et des lots de production (la planification d'une cellule de fabrication est plus facile parce que les tâches ne sont pas régulières), par la complexité réduite de l'algorithme qui peut donner une réponse en temps réel et enfin par la perspective d'une intégration facile de nouveaux lots dans la production en cours de réalisation.

L'algorithme EDF associe des priorités aux tâches. Ces priorités ont des valeurs inversement proportionnelles avec l'intervalle de temps qui reste jusqu'à la date de livraison. Toutes les tâches sont mises dans une queue d'attente en fonction de leurs priorités, et la plus prioritaire des tâches est exécutée (Figure 3.16). Si on tient compte du parallèlle décrit ci-dessus entre un lot de production et une tâche dans un système d'exploitation, on peut très bien utiliser l'algorithme EDF dans les systèmes de production avec quelques changements.

- L'entité lot de production

Pour séparer entre eux les ordres de production avec différentes dates de livraison une nouvelle entité a été introduite – le *lot de production* – qui contient tous les ordres ayant la même date de livraison. Un lot de production a la structure minimale suivante :

- L'ensemble des demandes clients qui doivent être réalisées ;
- Un identificateur unique du lot courant ;
- L'ensemble des holons ordres déjà planifiés en utilisant une certaine configuration de la cellule. Cette configuration est aussi stockée dans la structure du lot de production ;

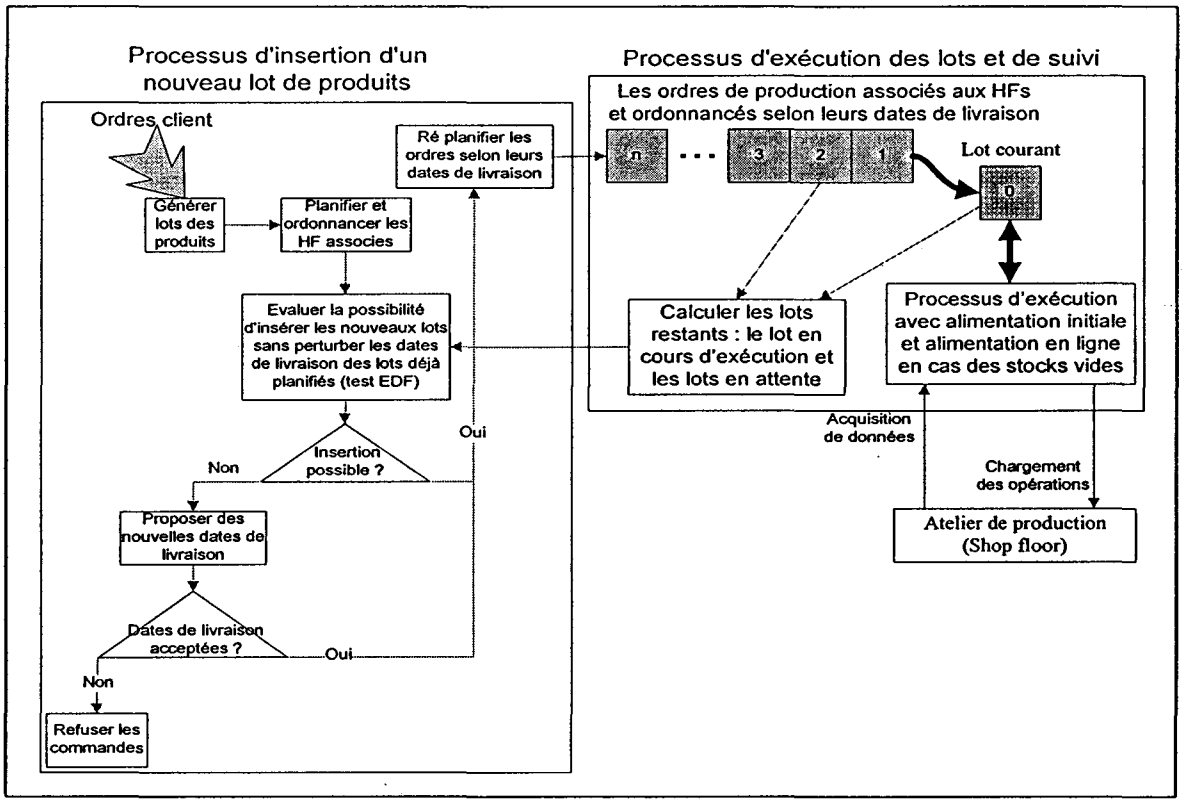


Figure 3.16 : processus d'évaluation permet d'ajouter de nouveaux ordres comme lots de production

- L'ensemble des ressources avec leurs opérations disponibles, ressources qui sont utilisées pour la planification et ordonnancement des holons ordres (le modèle MdAaS mis à jour) ;
- La date limite de livraison imposée, fixée de commun accord entre le client et l'exploitant du système de fabrication ;
- La date de début de la fabrication du lot de produits ;
- L'intervalle de temps nécessaire pour fabriquer le lot complet (*makespan*).

La planification de la production au niveau des lots sera calculée séparément pour chaque lot, et ensuite chaque lot de production sera assigné à la cellule de fabrication en utilisant la procédure EDF.

La procédure d'insertion de nouveaux lots de production

- Pas 0. Calcul du temps restant jusqu'à la fin d'exécution du lot actuel à partir du temps courant (si c'est le cas) ;
- Pas 1. Définition de la nouvelle production : produits, quantités et dates de livraison ;

- Pas 2. Séparation des produits en entités du type "lot de production" selon les clients et les dates limite de livraison ;

- Pas 3. Génération des entités "lot de production" ;

- Pas 4. Génération des ordres bruts (*raw orders*) dans les lots de production ;

- Pas 5. Planification individuelle des lots pour trouver le temps de production (*makespan*) ;

- Pas 6. Analyse de la possibilité d'introduire les nouveaux lots dans la production déjà planifiée utilisant la procédure EDF (équation 3.1 : dans l'intervalle de temps nécessaire pour toute la production le système ne doit pas être utilisé à plus de 100% ; équation 3.2: dans le cas où le s [illegible] 100% les dates de terminaison de chaque lot ne doive $\sum_{i=1}^{n} \frac{C_i}{T_i} \leq 1$, n = *nombre des taches* livraison) :

$$C_i = Temps\ d'execution \tag{3.1}$$

$$\sum_{j=1}^{i} makespan_j \leq date_besoin_i, i = \overline{1,n} \tag{3.2}$$

- Pas 7. Si les conditions (3.1) et (3.2) sont satisfaites alors les lots de production sont acceptés et planifiés comme établi au pas 6 ;

- Pas 8. Si une des conditions 3.1 ou 3.2 n'est pas satisfaite, de nouvelles dates limites de livraison qui peuvent être satisfaites sont proposées ;

- Pas 9. Charger les nouvelles données et continuer la production.

Parce que le processus d'exécution d'un lot est interruptible (système préemptif) les nouveaux lots peuvent être introduits exactement au moment de leur arrivée (Figure 3.16). La seule contrainte est la nécessité que les ordres en cours d'exécution soient maintenus dans le système de fabrication.

4 Conclusion

Ce chapitre a présenté l'architecture générale du système de pilotage envisagé pour un FMS en se basant sur le modèle holonique SEAH du chapitre II. Après avoir présenté la structure du système, sa décomposition en holons et les interactions entre holons, le comportement dynamique du système a été détaillé.

Le chapitre suivant présente des expérimentations et des résultats, réalisées et obtenus sur la cellule flexible CIMR de l'Université Politehnica de Bucarest, qui permettent d'appréhender la validité et le bien fondé de l'architecture de pilotage proposé dans ce chapitre.

Chapitre 4 : expérimentation du modèle de pilotage holonique Cas d'un système de type FMS

Introduction

Dans ce chapitre, l'implémentation et l'évaluation du modèle holonique de pilotage décrit dans le chapitre précédent est présentée.

Les différents équipements utilisés pour la mise en œuvre du système de pilotage sont décrits en détail. On décrit ensuite le protocole expérimental qui a été élaboré pour évaluer le modèle de pilotage holonique.

Les résultats obtenus, en particulier en testant le modèle de pilotage lorsque des perturbations sont introduites dans le système, sont donnés et analysés.

1 La plateforme de fabrication CIMR

1.1 Ressources, types d'opérations et types de pièces

Ce chapitre présente les expérimentations faites sur la cellule de fabrication flexible CIMR de l'Université Politehnica de Bucarest dans le cadre du projet national de recherche RVHOLON (www.rvholon.cimr.pub.ro). La cellule dispose de deux types génériques de ressources de traitement (fraiseuses à commande numérique, robots industriels), d'un type générique de ressource d'inspection de la qualité des produits (système de vision), d'un type générique de ressource de transport (convoyeur en boucle fermée) et d'un type générique de stock d'approvisionnement avec des composants matériels (système flexible d'alimentation avec vision). Ces ressources de traitement, transport, stockage et inspection, placées dans les postes P0, ..., P5 de la cellule (figure 4.1) sont présentées ci-dessous :

- **Robots industriels** de types : robots six axes (figure 4.1- a) pour l'assemblage, le traitement des produits et le transfert des pièces aux fraiseuses CNC en P3 et P4 ; *SCARA* (quatre axes, figure. 4.1 - b) pour l'assemblage des produits en P1, P2 et l'approvisionnement avec des pièces en P5 ; cartésien (quatre axes, Figure4.1- c) pour le chargement / déchargement des palettes en P0.

Les robots six axes et SCARA font eux-mêmes l'alimentation de leurs postes de travail (P1 – P4) et la manutention des produits pour le montage et l'inspection qualité,

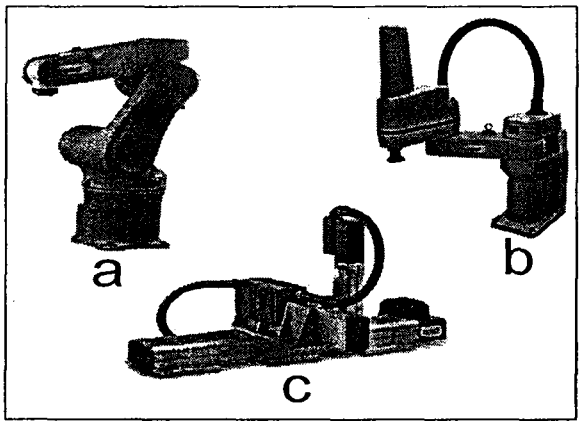


Fig. 4.1 : robots industriels utilisés dans la cellule de fabrication CIMR

La structure informatique associée à une ressource de type robot industriel est composée des données suivantes :

- *type du robot* (Adept Viper s650 en P3, P4 ; Adept Cobra s600 en P1, P2 ; Adept Cobra s800 en P5 ; Adept Python pour les modules linéaires en P0) (Adept, 2010) ;
- *caractéristiques du robot* (nombre de degrés de liberté, répétitivité, précision, type de chemins ; rayon d'action, vitesse maximale, poids transportable maximal) ;
- *caractéristiques de la pince* (mécanique ou à vide ; nombre, forme et dimensions des mors ("empreintes" des mors) ; axiale / parallèle ; transformation "tool");
- *caractérisation de l'extension vision artificielle du système robot* (nombre et types des caméras vidéo ; emplacement des caméras vidéo (fixes ou mobiles) ; transformations caméra-robot [base] et caméra-robot [segment auquel est attachée la caméra mobile);
- *structures des programmes robot* pour la manutention des pièces (assemblage, traitement des produits), pour l'accès aux stocks locaux et pour l'interconnexion avec les autres ressources (robot - convoyeur et robot – caméra). Les programmes des robots, qui composent la partie « automatisation » de l'entité HR (figure 3.8, chapitre précédent) sont localisés sur le contrôleur robot et sont structurés ainsi : un serveur qui interprète les commandes reçues via TCP (partie communication) et un interpréteur des commandes qui lance l'exécution les différents programmes stockés dans la mémoire du contrôleur robot en fonction de la commande reçue.

- **Système de transfert et transport des palettes** de type convoyeur en boucle fermée avec déviations (branches) pour apporter les produits dans les postes (figure 4.2);

La structure informatique associée à la ressource de type convoyeur est composée du modèle du système de transport (figure 3.4), qui est utilisé par l'automate programmable en charge du pilotage du convoyeur : routage des palettes et lancement des ordres de production sous la forme d'un holon HF et d'alimentation des stocks vides (HF d'alimentation). La partie automatisation de l'entité HR qui représente le convoyeur est composée de plusieurs programmes qui s'exécutent « en parallèle » sur l'automate programmable.

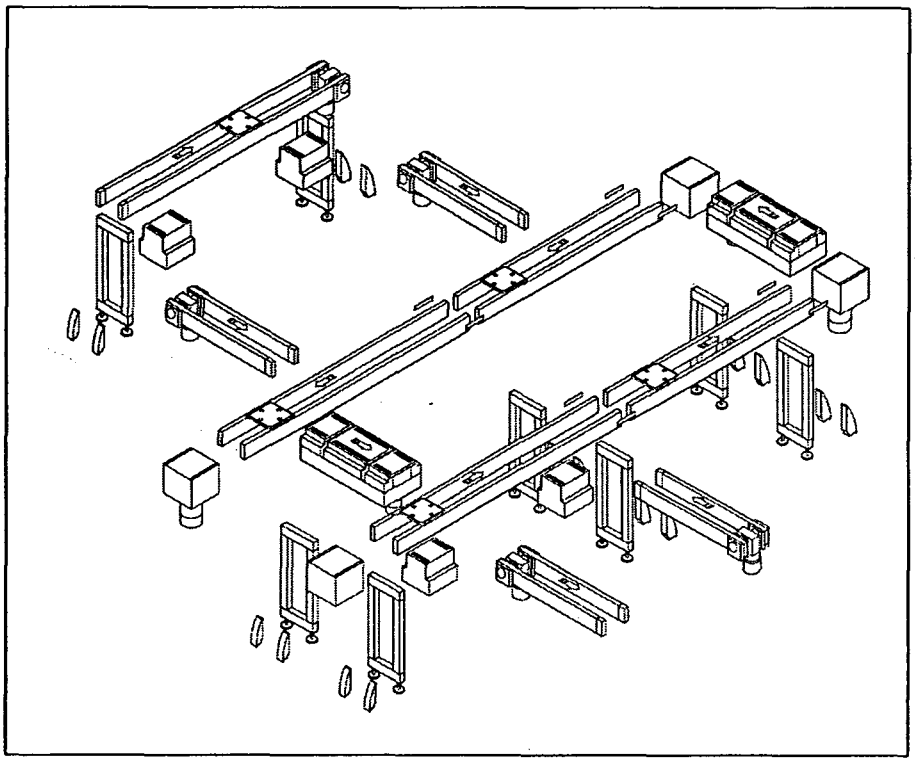


Figure 4.2 : éléments qui composent le système de transport

- **Machines à commande numérique** (figure 4.3) pour le traitement mécanique des produits ou des sous-ensembles des produits (fraiseuse CNC à trois axes en P3 et à quatre axes en P4);

- **Cellule CIMR**

La structure informatique associée a une ressource de type machine à commande numérique est composée des éléments suivants :

  - *type de la machine* à commande numérique : avec trois / quatre degrés de liberté (3D linéaires / 3D linéaires + 1 rotation) ;

  - *caractéristiques de la machine* : valeurs min. / max. de la vitesse d'avance et de la vitesse de rotation principale ; précision de la trajectoire générée ; types des matériaux qui peuvent être usinés ; type d'interpolation des chemins ; correction d'outil (rayon de la fraise) ; puissance électrique max. consommée;

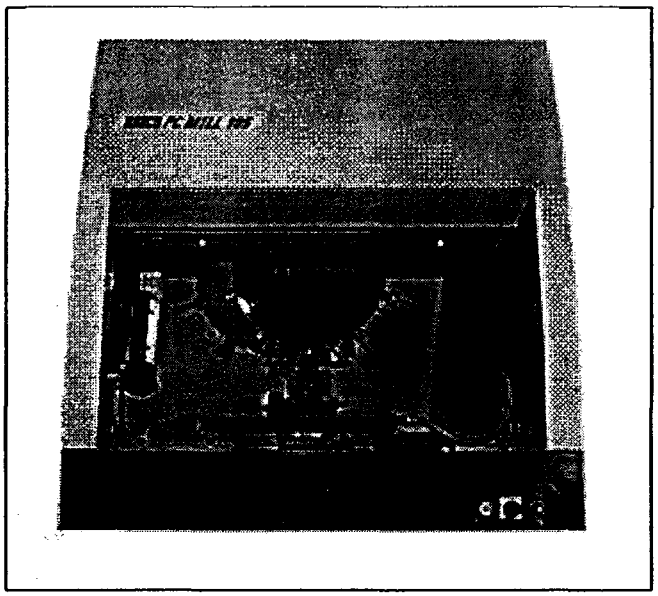


Fig. 4.3 : fraiseuse à commande numérique (EMCO Mill 105) en poste P4 de la cellule CIMR

- *caractéristiques du magasin d'outils* (fraises) : type et capacité du magasin ; type d'outils existants dans le magasin ; temps de changement d'outil ;
- *caractéristiques du dispositif de fixation* des matériaux / pièces ;
- *types et caractéristiques des opérations de traitement* qui peuvent être exécutées par la ressource ;
- *type de communication* avec autres ressources ;
- *spécification du robot qui charge / décharge la ressource* : la ressource de type « machine CNC » est accessible dans la cellule de fabrication via une ressource intermédiaire – le « robot industriel Viper à 6 d.d.l. », maître du poste de travail – qui élargit la gamme des services de traitement des pièces avec des opérations d'usinage (fraisage, perçage) qui nécessitent aussi la manutention des matériaux pour le chargement et déchargement de la machine. Après un chargement de pièce l'opération de traitement est lancée par l'unité de contrôle de la machine. Quand l'opération de traitement se termine le déchargement du poste est débuté.

- **Caméras vidéo** fixes et mobiles pour l'inspection visuelle des opérations effectuées dans les postes de travail (en P1 – P4), du produit fini (en P0) et des zones d'alimentation avec des pièces (en P5), ainsi que pour le guidage en ligne des

robots. Les cameras vidéo sont intégrées dans deux types de structures informationnelles :

- caméra vidéo avec traitement d'image (logiciel *AdeptSight*) localisée sur un serveur indépendant (le PC du poste robot) qui peut être interrogé via réseau ETHERNET ;
- caméra vidéo avec traitement d'image (logiciel *Adept Vision*) localisé directement sur une carte spécialisée du contrôleur robot à laquelle la caméra est branchée. Dans ce cas les informations peuvent être appelées seulement par le robot attaché.

- **Stocks intelligents** pour l'approvisionnement rapide avec des pièces (figure 4.4). Par rapport au stockage classique, les stocks intelligents peuvent être commandés d'une façon flexible suite aux informations fournies par la vision artificielle, afin de permettre l'accès à plusieurs types de pièces simultanément, peu importe leur position et orientation, sans danger de collision de la pince avec les objets voisins. Il y a aussi la possibilité de séparer ou trier les pièces selon des scénarios définis a priori en faisant vibrer l'ensemble. En ce qui concerne la partie informatique des deux stocks intelligents (type *Adept FlexFeeder*) utilisés dans la cellule flexible, l'accès aux fonctions de contrôle se fait par des interfaces standard (série ou ETHERNET).

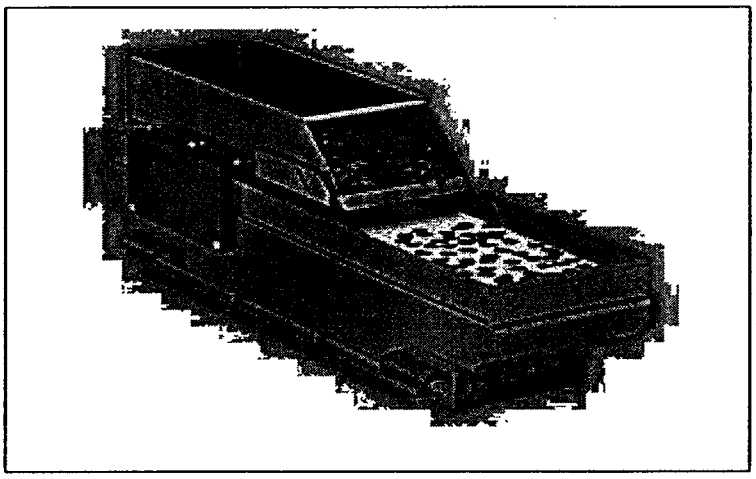


Figure 4.4 : stock intelligent de composants pour l'assemblage au poste P5 de la cellule CIMR

Les **opérations génériques** conçues pour valider le contrôle holonique des applications de fabrication ont été associées aux ressources distribuées dans les postes de travail comme suit :

- **P0**. Ce poste est composé d'un robot industriel cartésien *Adept Python*, 4 d.d.l., deux stocks (un pour les palettes vides, l'autre pour les palettes avec des produits finis) et une caméra vidéo fixe. Le robot du poste P0, en charge avec le processus de manutention des palettes dans la zone d'entrée et sortie du système de

fabrication, débute le processus par lequel une commande de fabrication d'un produit passe du niveau informationnel (HSO avec un client et une date de fin attachée) au niveau physique (produit réalisé dans un ordre planifié des opérations par des ressources ordonnancées selon un critère de coût choisi). En coopération avec l'automate programmable qui réalise le routage de la palette sur laquelle le produit sera successivement traité (usiné, assemblé, vérifié), le robot en P0 est responsable de la mise en place des processus d'*agrégation* et *désagrégation* d'un HF (figure 2.12).

L'agrégation d'un HF est réalisée par l'opération générique du robot en P0 qui amène sur le convoyeur une *palette vide* sur laquelle le produit sera réalisé, et celle de l'automate programmable qui lui associe un fichier qui décrit, en format standard, les destinations successives du produit pour recevoir les traitements nécessaires pendant son cycle de fabrication.

La désagrégation d'un HF signifie *l'évacuation de la palette avec le produit achevé* et le *stockage* dans un stock spécial en P0. Ainsi, les opérations génériques exécutées par P0 sont :

- *Alimentation avec palette*

  APAL(*pal,stor,rob*)[pal(typ.pal),car.stor(typ.stor,cap.stor,

  avail.stor,org.stor)...; loc.portepal; robot(typ.rob,

  typ.grip, TOOL); message()], avec les arguments :

- pal( ) = {work_pal,supply_pal}: le type de la palette vide (*de travail* – sur laquelle un produit nouveau sera construit (HF) ou *d'alimentation* – sur laquelle des pièces seront ensuite placées par une opération APFF ou GSTO) qui sera enlevée du dépôt de palettes (la zone d'entrée ASRS) et placée sur une porte palette qui attend d'entrer dans le système de production ;
- car.stor[ ] = [...;[#IDstor.area;typ.pal;cap_stor.area; avail_stor.atea; org_stor.area()];..]: caractérisation de la zone d'entrée ASRS d'où la palette sera prise ;
- loc.portepal: emplacement sur le convoyeur d'entrée de la porte palette sur laquelle la palette vide sera déposée ;
- robot(tip.rob,typ.grip,TOOL) : caractérisation du robot identifié par #IDrob qui effectuera l'opération APAL ;
- message(STATUS(APAL,temps.r)): message qui indique l'état (terminée : ok / n_ok ; en exécution ; suspendue) de l'opération et le temps dépensé.

APAL() assure l'alimentation robotisée d'une porte palette avec une palette vide à l'entrée de la cellule; l'opération est exécutée par le robot cartésien Python et précède le lancement en exécution d'un HF (pour production ou pour l'alimentation des quatre postes de travail – initialement, respectivement du poste de travail où une rupture de stock a été signalé – lors de la production) ; le

robot utilise une opération « *pick and place* » qui est exécutée une seule fois (Borangiu et al. 2002) ;

- *Retrait d'une palette du flux de fabrication* et *stockage*

  DPRO(*pal,stor,rob*)[pal(typ.pal),car.stor(typ.stor,cap.stor,

  avail.stor, org.stor)...; loc.portepal;robot(typ.rob,

  typ.grip,TOOL); message()], avec les mêmes arguments de APAL.

  Cette opération réalise l'évacuation robotisée d'une palette avec un produit fini et le stockage dans la zone de sortie ASRS (*automatic storage and retrieval system*) du convoyeur ; elle est exécutée par le robot cartésien Python quand un HF a terminé l'exécution du produit attaché ; le robot exécute alors une seule fois l'opération « *pick-and-place* ».

- **P1** et **P2**. Traités simultanément en raison du fait que les deux postes sont équipés avec le même type de robot industriel (SCARA, avec 4 degrés de liberté - *Adept Cobra*), ils peuvent exécuter les opérations génériques suivantes :

  - *Montage d'un ou plusieurs composant(s) sur un produit*

    MTC(*pr,*[*comp*],*rob*))[pr(#IDpr,loc.pr);[m(t,nm,loc.mat,

    tr.mt)];...; [stor(typ.stor,loc.stor)];robot(typ.rob,

    typ.grip,TOOL); message()] avec les paramètres :

  - pr(#IDpr,loc.pr): le code du produit qui se trouve sur la palette amenée par un OH dans le poste de travail, où un robot du type typ.rob assigné par (#IDrob) devra monter le (les) composant(s) décrit(s) dans la liste [m( )] ;

  - [m(t,nm,loc.mat,tr.mt)] = [...;(typ_*k*.mat,nm_*k*, [loc.mat_*ki*,[tr.mt_*ki*]);...| $1 \le k \le f_k, 1 \le i \le \text{nm_}k$]: caractérisation des composants qui seront montés sur le produit, en termes de : type, nombre, emplacement de montage, points significatifs et type de segments de déplacement sur les trajectoires robot *appro / depart* ;

  - [stor(tip.stor,loc.stor)]: vecteur de données qui décrit les f_k dépôts de matériaux de type typ_*k*.mat : [...;(typ_*k*.stor, loc_*k*.stor);...], $1 \le k \le f_k$ en P1 ou P2 ;

  L'opération générique MTC() consiste en le montage d'un ou plusieurs composant(s) matériel(s) sur le produit qui a été amené dans le poste de travail ; elle nécessite un ou plusieurs programme(s) déclarés dans [IDprg(MTC)], chacun avec une routine "*pick-and-place*" pour l'assemblage des composants.

  - *Changement d'outil pour un robot*

    SCT(*old.tool,new.tool,rob*)[old.tool(typ.otl,loc.otl);

    new.tool(typ.ntl,loc.ntl);...; robot(typ.rob,typ.grip,

old.TOOL,new.TOOL); message()], avec les paramètres :

- old.tool(typ.otl,loc.otl): caractérisation de l'outil existant, qui sera libéré par le robot ;
- new.tool(typ.ntl,loc.ntl): caractérisation du nouvel outil qui sera pris par la pince du robot après l'éventuel abandon de l'ancien outil ;

L'opération, réalisée par le robot lui-même, cause le changement d'outil dans la pince du robot ; elle nécessite un programme spécifié dans [IDprg(STC)], avec 1 routine "*pick-and-place*" (quand un instrument est introduit dans / libéré de la pince de type typ.grip du robot de type typ.rob) ou 2 routines "*pick-and-place*" (quand un instrument existant est changé avec un nouveau outil).

o *Fixation composant(s) sur un produit*

```
FIX(pr,[sub],[el.fix],[tool],rob))[pr(#IDpr,loc.pr);
 [sub(t,loc.sub,[mod.fix,el.fix,tool])];...
   ...
[el.fix(t,stor.fix,loc.fix)];[tool(t,tool.hld,
   loc.tool)];   ...   robot(typ.rob,typ.grip,TOOL);
message()]
```

avec les arguments :

- `pr(#IDpr,loc.pr)`: code du produit sur la palette amenée par un HF dans le poste de travail, où un robot type `typ.rob` assigné par `(#IDrob)` fixera avec l'élément (les éléments) de fixation décrit(s) par `[el.fix()]` en utilisant l'outil (les outils) spécifié(s) par `[tool()]` la (les) composante(s) déjà montée(s) sur le produit, décrite(s) en liste `[sub()]`;
- `[sub(t,loc.sub,[mod.fix,el.fix,tool])]` = `[...;(typ_i.sub,loc_i.sub,[mod_ij.fix,loc_ij.fix,[tr.fix_ij], el_ij.fix, tool_ij]);...` | $1 \le i \le ns$, $1 \le j \le n.fix$]: caractérisation des composants à fixer en termes de : type et emplacement de montage du sous ensemble no. i sur le produit identifié par `pr(#IDpr)`, mode de fixation (presser, visser, appliquer couvercle et visser, etc.), emplacement de l'outil (de la pince) pour la fixation no. j du sous ensemble no. i, points initiaux, finaux et intermédiaires et type des segments *appro / depart* lors des mouvements de fixation ;
- `[el.fix(t,stor.fix,loc.fix)]` =`[...;(tip_k_el.fix, stor_k.fix, loc_k.stor); ...]`, $1 \le k \le f$: vecteur de données qui décrit f types d'éléments de fixation des ns sous ensembles du produit pr ;

- [tool(t, tool.hld, loc.tool)]= [...;(tip_*m*.tool, tool_*m*.hld, loc_*m*.tool, *m*_TOOL); ...], $1 \leq m \leq nt$: vecteur de données qui spécifie les *nt* types d'outils de fixation (outil implicite – la pince du robot) nécessaires pour fixer les *ns* sous ensembles du produit *pr* ;

L'opération permet de fixer une ou plusieurs composants [sub] qui ont été déjà assemblés sur le produit *pr*, éventuellement avec un ou plusieurs éléments de fixation [*el.fix*] (vis, couvercle, tige), éventuellement avec un ou plusieurs instruments [*tool*] (tourne vis, axe) ; elle utilise un ou plusieurs programmes spécifiés par #IDprg(), contenant une ou plusieurs routines "*pick-and-place*".

o *Traitement d'un produit*

PRP(*pr*,*op*,*tool*,*rob*) [pr(#IDpr,loc.pr);op(typ.op,typ.tool);

tool(typ.tool,tool.hld,loc.tool);robot(typ.rob, typ.grip,TOOL);message()] avec les paramètres :

- pr(#IDpr,loc.pr) : code du produit qui se trouve sur la palette amenée par un HF dans le poste du robot (emplacement du produit - loc.pr) et sur lequel le robot du type *typ.rob* doit effectuer une opération de traitement direct du type *typ.op*, utilisant éventuellement un instrument de type *typ.tool* ;
- op(typ.op,typ.tool): décrit l'opération qui sera exécutée sur le produit pr(#IDpr), par exemple : polissage, marquage, peinture, soudure ;
- tool(typ.tool,tool.hld,loc.tool,TOOL): données qui caractérisent l'outil dans la pince du robot avec lequel l'opération de traitement de type *typ.op* sera effectuée ;
- message(STATUS(PRP),temps.r,temps.p) où les deux arguments temps.r, temps.p expriment respectivement le temps effectif d'exécution de l'opération PRP et le temps effectif d'utilisation de l'outil de type *typ.tool* lors de cette opération.

L'opération commande le traitement d'un produit *pr* à l'aide d'un outil robot monté sur sa pince [tool] (tourne vis, axe) ; elle utilise un programme spécifié par #IDprg(), contenant une ou plusieurs routines "*pick-and-place*".

o *Alimentation d'un stock local*

ASTO(*mat*,*stor*,*rob*) [mat(typ.mat,*k*,supply_pal,loc.mat);

stor(typ.stor,loc.stor)... ;
robot(typ.rob,typ.grip,
TOOL); message()] avec les arguments :

- mat(typ.mat,supply.pal,loc.mat): décrit les composants matériels en nombre de k, $1 \leq k \leq$ cap(#IDstor) avec lesquels le poste de travail sera alimenté ; les pièces se trouvent sur une palette spéciale, de type *supply pallet* qui a été activée par un *Holon Flux d'alimentation ;*
- stor(typ.stor,loc.stor) : type du stock local où les k matériaux de type *typ.mat* seront deposés par le robot ; emplacement des matériaux ;
- robot(typ.rob,typ.grip,TOOL): caractérisation du type de robot qui fait l'alimentation de *stor* (COBRA, VIPER) et de sa pince (type, nombre de mors / à vide, forme et dimension des mors) ;
- message (STATUS(ASTO,temps.r) : ce message indique l'état d'exécution de l'opération - terminée / en exécution / suspendue en attente de l'arrivée d'une nouvelle palette de type *supply pallet.*

Cette opération réalise l'alimentation robotisée du stock de pièces dans le poste de travail du robot ; l'opération est exécutée par le robot lorsque son stock est vide ; le robot a besoin d'une opération « *pick-and-place* » qui est exécutée k fois, une fois pour chaque pièce apportée par une palette qui arrive sur le convoyeur du poste robot.

o *Déchargement d'un stock local*

GSTO(*mat,stor,rob*)[mat(typ.mat,*k*,supply_pal,loc.mat);

stor(typ.stor,loc.stor)... ; robot(typ.rob,typ.grip,

TOOL); message()] avec les arguments :

L'opération décrit un déchargement du stock typ.stor placé dans le poste de travail du robot typ.rob par ce même robot, si le stock n'est pas vide et qu'une requête d'alimentation avec k unités de matériel de type typ.mat est formulée; les matériaux demandés seront placés sur une ou successivement sur plusieurs palettes de type *supply pallet* qui arrivent sur le convoyeur du poste robot ; le robot exécute k fois un sous programme « *pick-and-place* ».

P3 et **P4**. Les deux postes sont traités ensemble parce qu'ils sont équipés avec le même type de ressources : un robot industriel à 6 degrés de liberté (*Adept Viper*), une fraiseuse à commande numérique (3 axes en P3, 4 axes en P4) et deux cameras vidéo (fixe et mobile montées sur le robot). En plus des opérations des postes **P1** et **P2**, les opérations suivantes peuvent être exécutées :

o *Chargement d'une machine à commande numérique*

AMOU(*pr,mou,rob*)[pr(#IDpr,loc.pr); mat(typ.mat,loc.mat);

robot(typ.rob,typ.grip,TOOL);mou(#IDmou,loc.wp);

message()] avec les paramètres :

- pr(#IDpr,loc.pr): code du produit (pièce) qui se trouve sur une palette arrivée dans le poste robot sur le convoyeur en loc.pr et qui sera placé dans le poste de travail wp de la machine outil mou ;
- mat(typ.mat,loc.mat): décrit respectivement le type et l'emplacement d'un composant matériel de type typ.mat qui sera pris par le robot *rob* et déposé sur le poste d'usinage *wp* de la fraiseuse CNC *mou* ;
- robot(typ.rob,typ.grip,TOOL): caractérisation du type de robot (VIPER) qui fait l'alimentation du poste d'usinage *wp* de la fraiseuse *mou* ;
- mou(#IDmou,loc.wp): code d'identification de la fraiseuse CNC (EMCO 3/4 axes) et position loc.wp où le robot *rob* déposera le produit *pr* ou le composant *mat* dans le poste d'usinage *wp* de la machine ;
- message(STATUS(AMOU,temps.r) : ce message indique l'état d'exécution de l'opération - terminée [ok/not_ok(code erreur)] ; en exécution ; suspendue en attente d'une alimentation avec des matériaux de type *mat* quand le stock local du poste qui n'a plus ce type de pièces (une opération ASTO est automatiquement insérée, ainsi qu'un HF d'alimentation dans ce cas).

L'opération réalise l'alimentation robotisée du poste d'usinage d'une machine outil en P3 ou P4 avec 1 produit (qui se trouve sur une palette dans le poste robot du convoyeur) ou avec 1 matériel (qui se trouve dans le stock local du poste *mou*).

- *Déchargement d'une machine à commande numérique*

DMOU(*pr,mou,rob*)[pr(#IDpr,loc.pr); mat(typ.mat,loc.mat);

robot(typ.rob,typ.grip,TOOL);mou(#IDmou,loc.wp);

message() avec les mêmes arguments robot et mou qu'en AMOU et :

- pr(#IDpr,loc.pr): code du produit qui a été usiné et se trouve en loc.wp dans le poste de la machine mou et qui sera pris par le robot et placé en loc.pr sur la palette arrivée dans le poste robot sur le convoyeur ;
- mat(typ.mat,loc.mat): décrit respectivement le type et l'emplacement d'une pièce de type typ.mat qui vient d'être usinée en position loc.wp du poste de travail de la fraiseuse mou et qui sera prise par le robot *rob* et déposée en loc.mat - une des deux destinations possibles :
  - loc.mat=loc.pr : emplacement de montage de la pièce usinée, sur un produit identifié par #IDpr qui se trouve sur une palette en attente sur le convoyeur ;
  - loc.mat=loc.stor : emplacement de stockage de la pièce usinée, dans le stock local du poste de travail de la *mou* ;

- message(STATUS(DMOU,temps.r) : ce message indique l'état d'exécution de l'opération - terminée [ok/not_ok(code erreur)] ; en exécution ; suspendue en attente :
  - de l'arrivée d'une palette avec le produit #IDpr (sur laquelle la pièce usinée doit être assemblée) dans le poste robot sur le convoyeur ;
  - du déchargement du stock local dans le poste de travail de *mou* (une opération GSTO a été déclenchée et sera exécutée par le robot rob.

o *Usinage sur une machine à commande numérique*

USIN(*mou,pr*)[typ.pr;typ.mat;[sequ_usin]; message()], où :

- typ.pr : type du produit, identifié par #IDpr, sur lequel s'effectue l'opération USIN() ;
- typ.mat : type du matériel, identifié par #IDmat, sur lequel s'effectue l'opération USIN()et qui est destiné à un produit. Si #IDpr≠ 0 et #IDmat=0, l'op opération USIN() s'exécute directement sur le produit ;

[sequ_us]=[op.us_1,temps_1,[out_11,...,out_1f_1],#IDprg_1;

op.us_u,temps_u,[out_u1,...,out_uf_u],#IDprg_u;

message()];

les opérations d'usinage op.us_*i*, $1 \leq i \leq u, 1 \leq u \leq 256$ peuvent être : fraisage ou perçage, et les temps temps_i,$1 \leq i \leq u$ des opérations d'usinage incluent les temps de fixation du produit / matériel et les temps de changement des outils out_ij, $1 \leq i \leq u$, $1 \leq j \leq f_i$;

- message (): STATUS(USIN());[temps.r_1,...,temps.r_u];...

[temps.uo_#IDou_1, ... , temps.uo_#IDou_10]$_{\#mou}$

l'état de l'opération peut être: en exécution; terminée (ok/not_ok code_erreur);suspendue. Les temps temps.r_i,$1 \leq i \leq u$ indiquent les durées des opérations d'usinage, et les temps temps.uo_#IDou_i,$1 \leq i \leq 10$ indiquent les durées réelles d'utilisation des 10 outils des magasins CNC.

Dans les quatre postes de travail **P1-P4** il est possible de gérer des stocks locaux de pièces non structurés par rapport à leur contenu, de conditionner la tâche du robot en fonction du résultat d'une mesure de géométrie effectué, ou de réaliser un contrôle qualité du produit qui vient d'être traité dans le poste de travail. Deux classes d'opérations de vision génériques ont été développées dans ce but :

o *Détection, reconnaissance, localisation et accès aux objets*

DRLA(fonction,[objets[]],strategie,AOI,cam.video[]), où les arguments signifient :

- fonction: D / D+R / D+R+L / D+R+L+A
  - D : Detection[blob_AOI]
  - R : Reconnaissance[blob AOI,instances des modeles AOI]
  - L : Localisation[any/particular, ordre]
  - A : Acces[strategie(O_1,..,O_u);[MS_j(O_i)];[MP_j(O_i)]
    $1 \leq i \leq u$(u = no.objets),$1 \leq j \leq i$,MS = modele de saisie ;
    MP = modele de la pince(ampreintes de ses mors)
- objets[]: O_i[typ_O;nombre;ordre;MS;MP;dest], $1 \leq i \leq u$;
- strategie(O_1,...,O_dernier) : any/particular ;
- AOI : Aire d'intérêt pour la recherche des objets ;
- cam.video : #ID caméra physique; #ID caméras virtuelles.

o *Mesure d'un produit*

MESP(*O*)[typ_prod,mes_1(),...,mes_k(),validation(),
message(),action()] avec les arguments :

- typ.prod : blob ou instance d'un modèle a prori appris ;
- mes(typ.mes,AOI(mes_i),crit[typ.crit,repere,val_std]
  (mes_i),...,res[typ.crit,repere,val_std](mes_i)

Plusieurs types de mesures (globales - descripteurs de frontières / de région scalaires / vectoriels) ont été conçus, ainsi que l'élaboration de critères typ.crit qui utilisent des repères repere et des valeurs standards val_std pour l'obtention des résultats res.

Les opérations MESP()sont toujours précédées par une opération DRLA configurée soit comme DL_AOI soit comme DRL_AOI pour une pièce qui devra être mesurée. Les figures 4.5 a et b présentent deux types de mesures développées pour les pièces usinées, qui utilisent des références linéaires et circulaires.

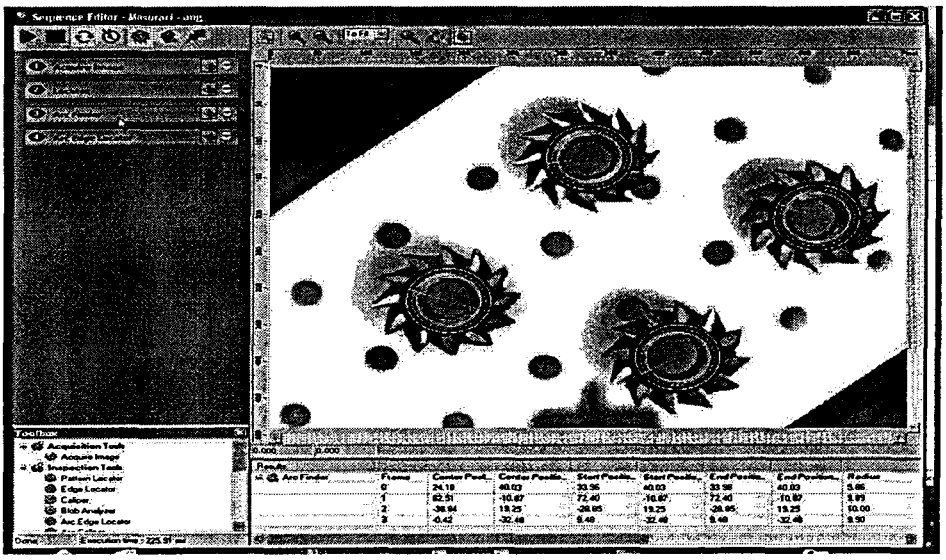


Figure 4.5 – a : contrôle de la géométrie des pièces avec règles de mesure linéaires

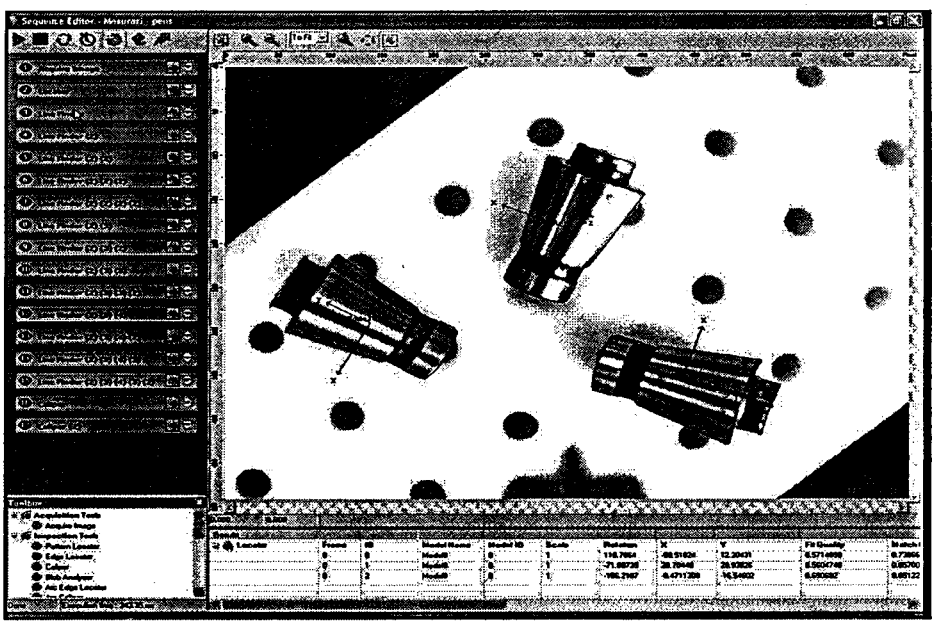


Figure 4.5 - b : contrôle de la géométrie des pièces avec règles de mesure circulaires

- **P5.** En charge avec le remplissage des stocks locaux des postes de travail **P1-P4** à l'aide d'un nombre de palettes dirigées sur le convoyeur de la cellule par un nombre égal de *Supply Holons*, le poste **P5** a la structure de contrôle présentée figure 4.6.

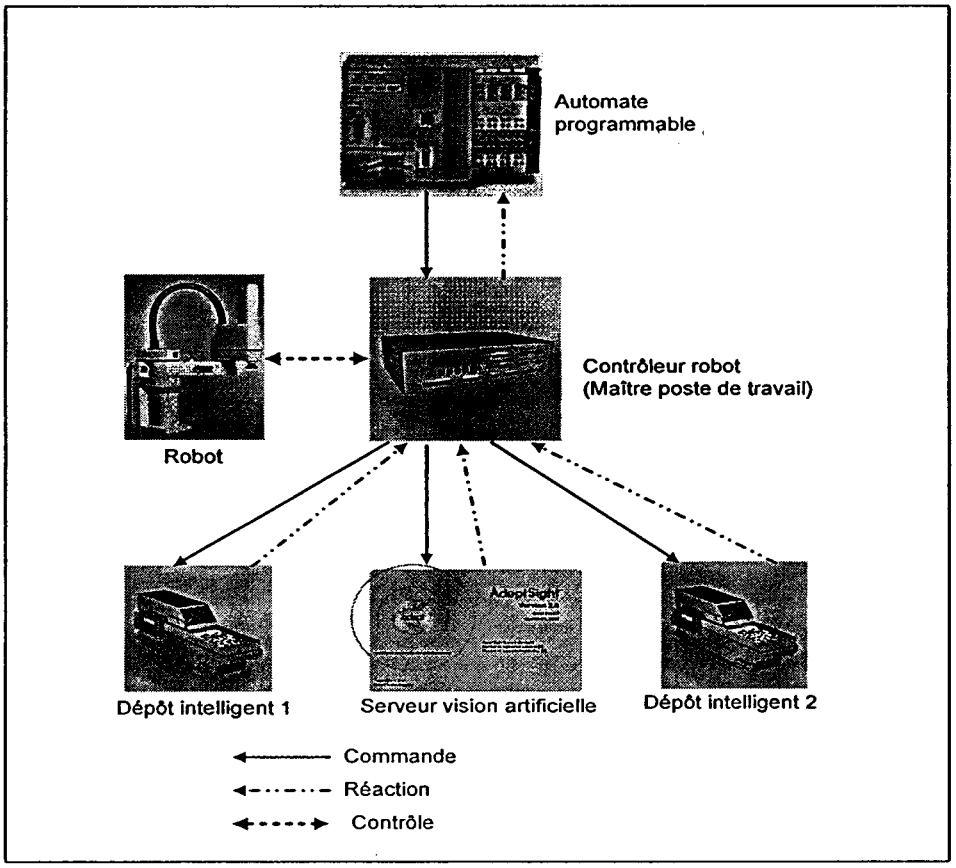


Figure 4.6 : architecture de pilotage du poste intelligent d'approvisionnement

Ce poste de travail a été développé autour du robot SCARA *Adept Cobra* s800 en utilisant une architecture type maître esclave. Ainsi, le contrôleur robot est le maître du poste et des autres ressources (dépôts/stocks intelligents du type *Adept AnyFeeder* et un système de vision artificielle *AdeptSight*) lui sont subordonnées. L'opération générique exécutée dans ce poste est :

- *Approvisionnement d'un alimentateur intelligent robotisé*

  APFF(*ifeed,mat,rob*)[ifeed(typ.ifeed[];ifeed.strg[];

  [ifeed.mat()]);rep(typ.rep[];no.rep[];

  [supply_pal()]);robot(typ.rob,typ.grip,TOOL);

  message()], où :

- ifeed(typ.ifeed[];ifeed.strg[];[ifeed.mat()])décrivent le dispositif intelligent d'approvisionnement (*Adept AnyFeeder*) duquel un nombre de $\sum_{1 \le j \le m} k_j$ matériaux de m types $1 \le j \le m$ sera extrait d'après une stratégie spécifiée: source des pièces (unité droite / gauche / alternance) ;

  mode des vibrations (amplitude, fréquence, moments de déclenchement, durée) ; ordre d'extraction. Les repères extraits rep seront déposés sur une ou plusieurs palettes supply_pal dirigées par un nombre associé de Supply Holons, par le robot du poste P5 (typ.rob=SCARA).

On peut constater que la cellule de fabrication CIMR dispose d'une grande flexibilité au niveau des postes de travails, comme suggéré sur la figure 4.7 : presque toutes les opérations génériques peuvent être exécutées par au moins deux postes de travail ce qui assure la disponibilité de l'ensemble des ressources par rapport aux demandes des clients, formulées par lots de fabrication.

Les ressources de traitement (robots, machines) avec des stocks locaux de palettes et pièces sont disposées sur cinq postes de travail comme indiqué en Fig. 4.7:

- 4 postes de travail (P1-P4) reliés par un système de transfert et transport - convoyeur en boucle fermée avec branches de déviation vers les ressources de traitement ;
- 1 poste P5 d'approvisionnement avec double dépôts/stocks intelligents ;
- 1 poste P0 qui gère l'insertion et l'évacuation des palettes du système de transport.

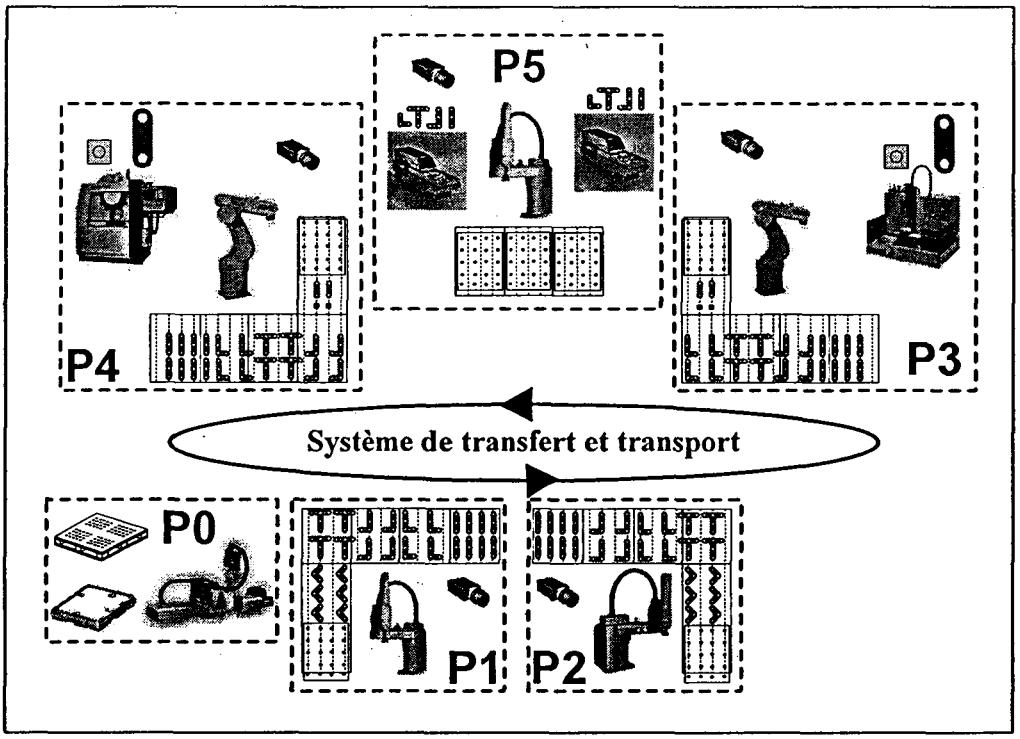


Figure 4.7 : les postes de travail de la cellule CIMR, les composants à assembler et les opérations qu'ils sont capables d'effectuer

Pour valider la solution développée de planification, d'ordonnancement et de routage, les opérations génériques conçues ont été particularisées ; des scénarios ont été proposés pour réaliser les produits présentés figure 4.8, en combinant les opérations génériques conçues :

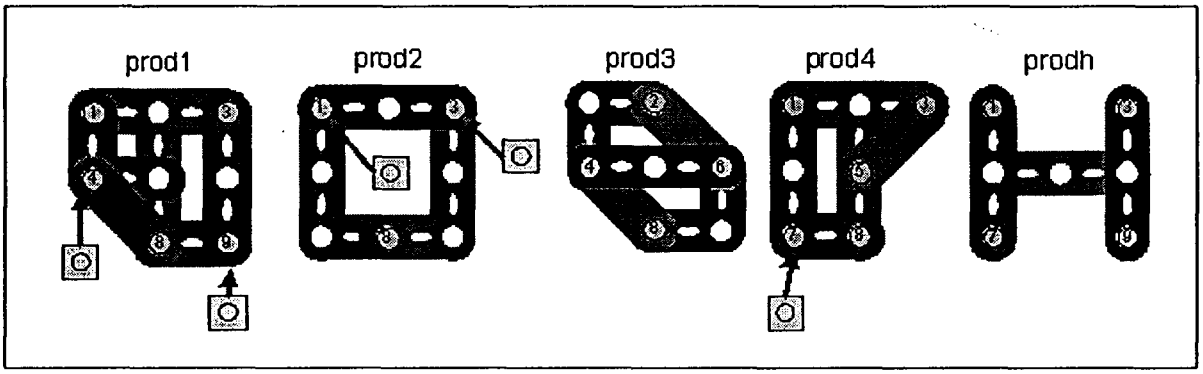


Figure 4.8 : types des produits choisis pour la validation du système de contrôle de la production

- **Produit de type 1** (prod1) : montage axe1, axe3, axe8, axe9, T, axe4, inspection visuelle intermédiaire, montage rondelle4, rondelle9, nonL, r, I modifié, inspection visuelle finale ;
- **Produit de type 2** (prod2) : montage axe1, axe3, axe8, L, rondelle3, inspection visuelle intermédiaire, montage nonL, rondelle1, I, inspection visuelle finale ;

- **Produit de type 3** (prod3) : montage axe2, axe4, axe6, axe8, r, r, inspection visuelle intermédiaire, montage I modifié, I modifié, I, inspection visuelle finale ;
- **Produit de type 4** (prod4): montage axe1, axe3, axe8, axe7, T, rondelle7, inspection visuelle intermédiaire, montage nonL, axe5, I modifié, inspection visuelle finale ;
- **Produit de type h** (prodh) : montage axe1, axe3, axe7, axe9, T, I, inspection visuelle finale.

Les successions d'opérations décrites représentent des séquences standard, mais, dans certains cas l'ordre des opérations peut être changé si ceci est physiquement possible (la pince du robot ne doit pas entrer en collision avec le produit). Par exemple, l'ordre d'insertion des premiers axes pour l'assemblage de chaque produit peut changer.

En plus, des scénarios définis pour valider le contrôle des lots de fabrication des produits montrés figure 4.8 dans la cellule physique avec la distribution des opérations indiquée en Fig. 4.7, un lot de produits virtuels a été défini pour expérimenter différents scénarios de production sur le simulateur de production basé sur un tableau de transport (*Transportation Time Matrix* - TTM).

Ce lot est composé de trois types de produits de complexités différentes : réduite (Fig. 4.9.a), moyenne (figure 4.9 - b) et élevée (figure 4.9 - c), définis en termes de services génériques (S1, S2, S3, S4) et précédences entre ces services. Le modèle pour les expérimentations reste le même (figure 3.4), mais la distribution des opérations dans les postes de travail change de façon à évaluer la relation entre le temps total de production et la distribution des opérations dans les postes de travail.

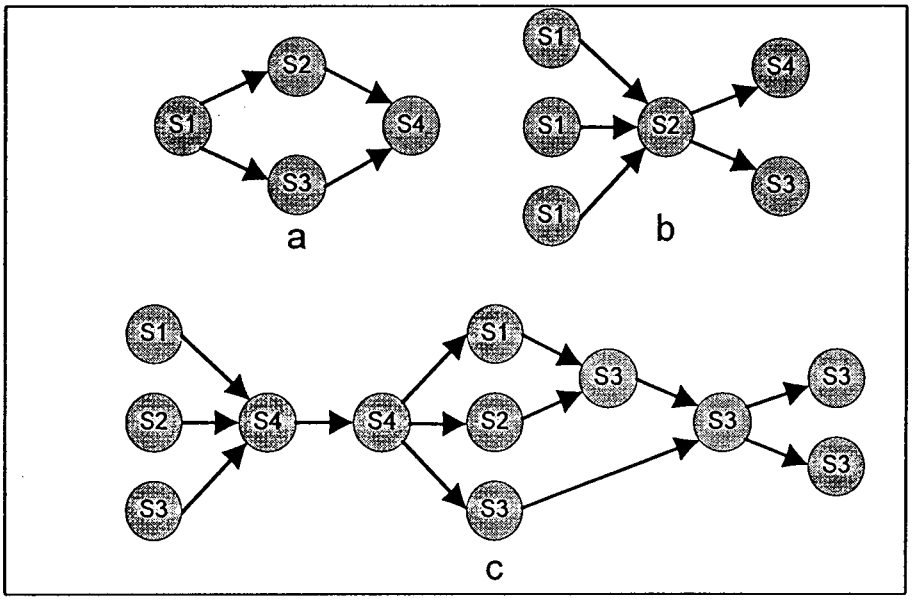


Figure 4.9 : complexité des produits pour la simulation d'un processus de production : a) réduite; b) moyenne, c) élevée

1.1. Modalité de pilotage

Au niveau logique, le système de fabrication est composé de trois types des ressources :

- de *calcul* - le serveur de la cellule et les ordinateurs type IBM PC (*Station Computers*) sur lesquels tourne l'application de gestion de la production, planification et ordonnancement des ordres de production ;
- de *contrôle* - l'automate programmable (AP) qui est en charge avec le contrôle du routage des palettes avec des produits sur le convoyeur ;
- de *calcule et contrôle* - les contrôleurs des robots industrielles (*Station Controllers*). Toutes ces ressources sont interconnectées dans une structure représentée figure 4.10.

En utilisant ces trois types de ressources, le système de pilotage de la cellule de fabrication CIMR de type *shop floor* peut être analysé selon plusieurs points de vue :

1. **Point de vue planification.** Considérant la solution adoptée pour la planification des lots de production (au début de la production ou pendant son exécution), il y a deux modalités qui coexistent : planification hors ligne / en ligne :
   - *planification hors ligne* (sans considérer les ordres rapides de production) : après leur réception, les ordres clients sont planifiés et ordonnancés au niveau supérieur (Niveau planification et ordonnancement figure 4.10) et envoyés ensuite aux niveaux inférieurs pour être exécutés. La planification hors ligne marque le début du processus de production ;
   - *planification en ligne* avec la procédure EDF (figure 3.14), pour permettre d'insérer des ordres rapides dans une commande déjà planifiée et ordonnancée. Dans ce cas il y a au moins un lot de produits déjà planifié et ordonnancé et un nouvel ordre arrive sous la forme d'un lot avec une date de livraison. Le nouveau lot est planifié et ordonnancé en ligne, et en suivant la *procédure d'insertion de nouveaux lots de production* décrite en 3.3.3 une décision est prise pour accepter ou non le lot.
2. **Point de vue prise de décision.** L'infrastructure de commande de la cellule de fabrication CIMR permet le pilotage centralisé (une seule entité est en charge du pilotage) ainsi que le pilotage décentralisé (un ensemble d'entités entre lesquelles il n'y a pas de relations maître esclave sont en charge du pilotage). Ces deux modalités de pilotage influencent les processus de planification et ordonnancement comme suit :
   - *planification et ordonnancement centralisés* : les deux processus sont réalisés par une seule entité informationnelle qui possède une vue complète du système et qui peut optimiser globalement la production au niveau des lots de produits,

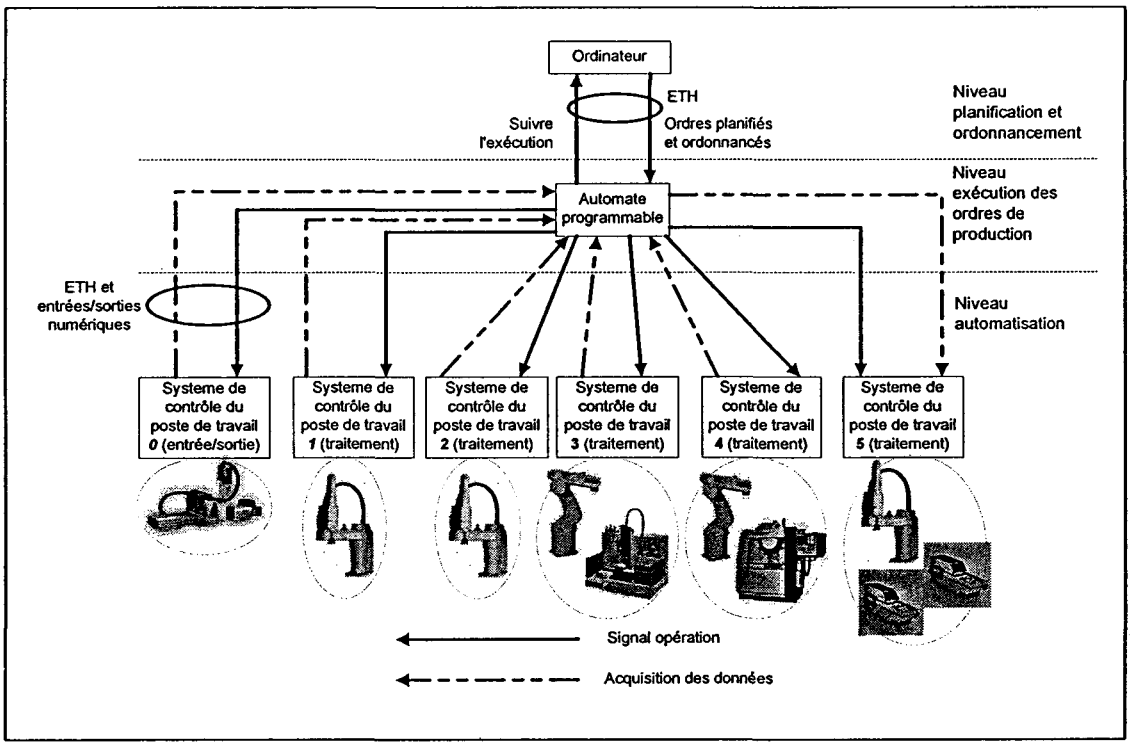


Figure 4.10 : interconnexions entre les ressources de calcul et contrôle

- *planification centralisée* (optimale au niveau des lots), faite par une entité qui a une vue complète du système, et l'*ordonnancement décentralisé*, fait par un ensemble d'entités (les HFs) qui dialoguent entre eux pour établir un ordonnancement valide en essayant aussi de l'optimiser au niveau du lot de produits en exécution simultanée. Cette modalité de pilotage représente la solution courante, qui est implémentée sur la cellule de fabrication ;
- *sans planification* et avec *ordonnancement décentralisé*. Dans ce cas les ordres client sont transférés directement, dans l'ordre de leur réception, aux entités responsables de leur production (HF). Comme dans le cas précédent, deux stratégies d'allocation ont été développées et utilisées avec de bons résultats :
  - allocation successive des opérations sur la ressource la plus proche pour minimiser le temps de transport ;
  - allocation uniforme des opérations sur les ressources existantes pour équilibrer leur chargement.

3. **Point de vue commutation entre les régimes de pilotage.** La solution de pilotage proposée permet deux types de régimes :

- *Sans commutation* : le régime de pilotage établi initialement (d'habitude hiérarchique ou centralisé) reste le même pendant tout le processus de production s'il n'y a pas de perturbations. Ainsi, pour une structure matérielle robuste de la cellule et un environnement stable, l'optimalité est assurée (minimisation du temps de fabrication des lots, chargement uniforme des ressources) ;

- Avec *commutation entre le fonctionnement centralisé et le fonctionnement décentralisé.* Ce passage est déclenché par des perturbations qui invalident la planification et l'ordonnancement faits au début de la production. Les perturbations considérées pour déclencher la commutation sont : panne de ressource, changement des temps de réception des services de traitement ou des temps de transport et réinsertion d'une ressource dans le flux de production (Borangiu et al., 2010). Ce régime de pilotage permet l'utilisation en parallèle des deux modes de prise de décision pour l'allocation des ressources (figure 4.11) :
  - *centralisé pour les ordres qui attendent l'exécution* (d'habitude ce type d'ordres représente la majorité et nécessite un temps significatif pour être suffisamment optimisés) et
  - *décentralisé pour le lot d'ordres qui sont en cours d'exécution* (étant moins nombreux, ces ordres ont besoin d'un intervalle de temps plus court pour être ordonnancés). L'idée de base est que, suite à une panne, l'ordonnancement du lot en cours d'exécution (la nouvelle allocation des opérations sur les ressources qui restent disponibles) est fait en ligne dans un temps court, et l'exécution physique des ordres peut commencer en parallèle avec la reprise centralisée de la planification des produits en attente, qui dure plus longtemps. Ainsi, la différence de temps entre l'ordonnancement décentralisé, qui est exécuté en ligne, et la planification et l'ordonnancement centralisés, exécutés hors ligne (Δt en figure 4.11), est utilisée pour l'exécution physique des ordres en cours ce qui conduit à la minimisation du temps total de production.

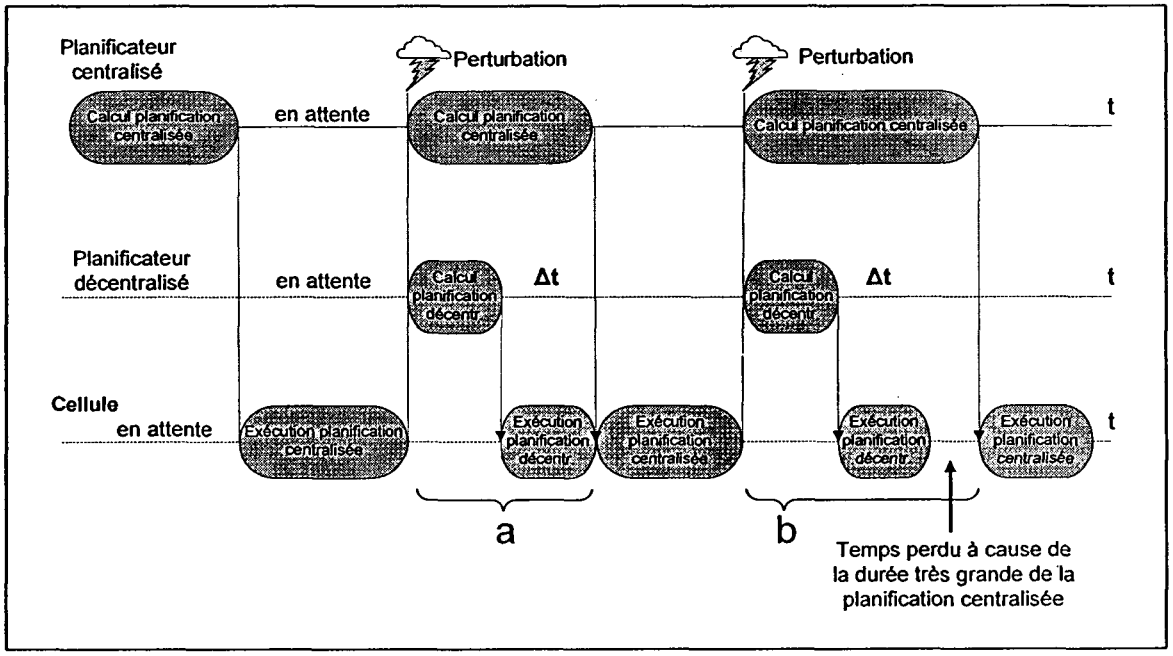


Figure 4.11 : fonctionnement parallèle du planificateur centralisé (a) et décentralisé (b)

La partie pilotage et contrôle du système conçu est composée de plusieurs programmes structurées sur les ressources de calcul & contrôle comme suit :

- **Serveur de la cellule :**
  - application de haut niveau pour la *collecte des ordres client, base de données production* (types de produits), *planification, ordonnancement et simulation* ;
  - serveur OPC pour la *communication avec l'automate programmable* AP en charge avec le routage des lots produits en exécution) ;
- **Ordinateurs des postes de travail de la cellule :**
  - application distribuée, type middleware, en charge avec l'*uniformisation de la modalité d'accès aux ressources* (le lancement en exécution d'une commande de traitement passe par l'ordinateur du poste de travail, associé au contrôleur de la ressource ; sur cet ordinateur s'exécute un programme ayant une interface d'accès standard) ;
  - application qui réalise la *mise à jour du Modèle d'Accès aux Services* (MdAaS) ;
  - application qui assure la *communication entre les HRs et les HFs* basée sur le protocole CNP ;
- **Automate programmable :**
  - un programme en charge du *contrôle de la partie transport* (convoyeur) du système de fabrication. Des capteurs et des actionneurs sont connectés à l'automate programmable par des lignes d'entrée / sortie numériques pour saisir les événements importants et agir ensuite en conséquence ;
  - un programme responsable du *lancement en exécution des opérations* des composants des HFs. Pour réaliser cette tâche, il communique avec les contrôleurs robots par une combinaison de réseaux ETHERNET (TCP AP – le AP spécifie quelle est l'opération demandée (D1), TCP Robot – le robot informe quel est son état (D2) et l'état de l'opération demandée (D3)) et environnement E / S numériques (READY – robot prêt a recevoir une opération, RQST-JOB – le AP demande une opération, Pal En Pos – le AP signale le début de l'opération demandée, Job Done – le robot signale la fin de l'opération demandée) ((figure 4.12).

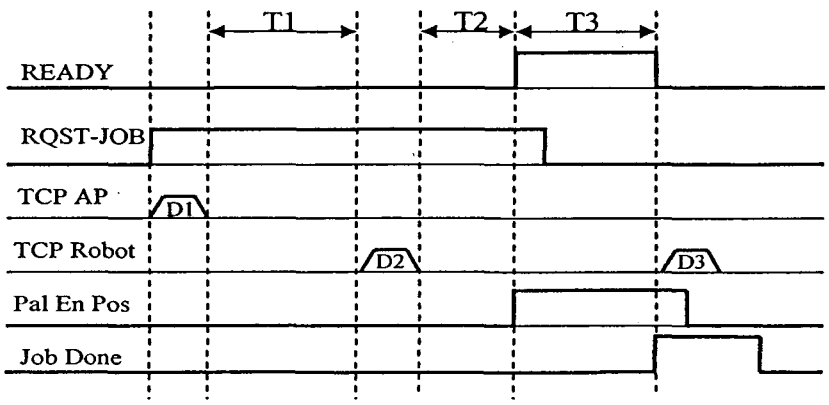


Figure 4.12 : protocole mixte de communication AP – contrôleur robot

- un programme responsable de *la gestion des holons HF* (*product tracking* and *history*). Ce programme communique avec le niveau supérieur (planification et allocation) à travers un réseau ETHERNET (protocole OPC, décrit en détail dans le chapitre précédent § 3.2), pour recevoir les informations nécessaires à l'exécution des HFs et pour envoyer les informations obtenues après leur exécution, le programme communique aussi avec les contrôleurs robots pour la mise à jour des états des HFs en cours d'exécution. Ces informations sont disponibles sur l'écran d'une interface graphique qui offre une vue globale du système de production (figure 4.13) :
- **Contrôleurs robots :**
  - un programme type serveur ETHERNET pour la *réception et interprétation des commandes* ; ce programme réalise aussi la partie *contrôle du robot* (ex. : opérations *pick and place*, mesure des produits, inspections qualité par vision, etc.). Le contrôleur robot communique en permanence avec le AP (entité en charge avec l'exécution des holons HF) en confirmant la possibilité de recevoir des holons HF. Cette confirmation assure la tolérance aux défauts du type défaillance de ressources; quand la communication est interrompue la ressource est déclarée hors ligne et les holons HF sont routés vers les ressources disponibles.

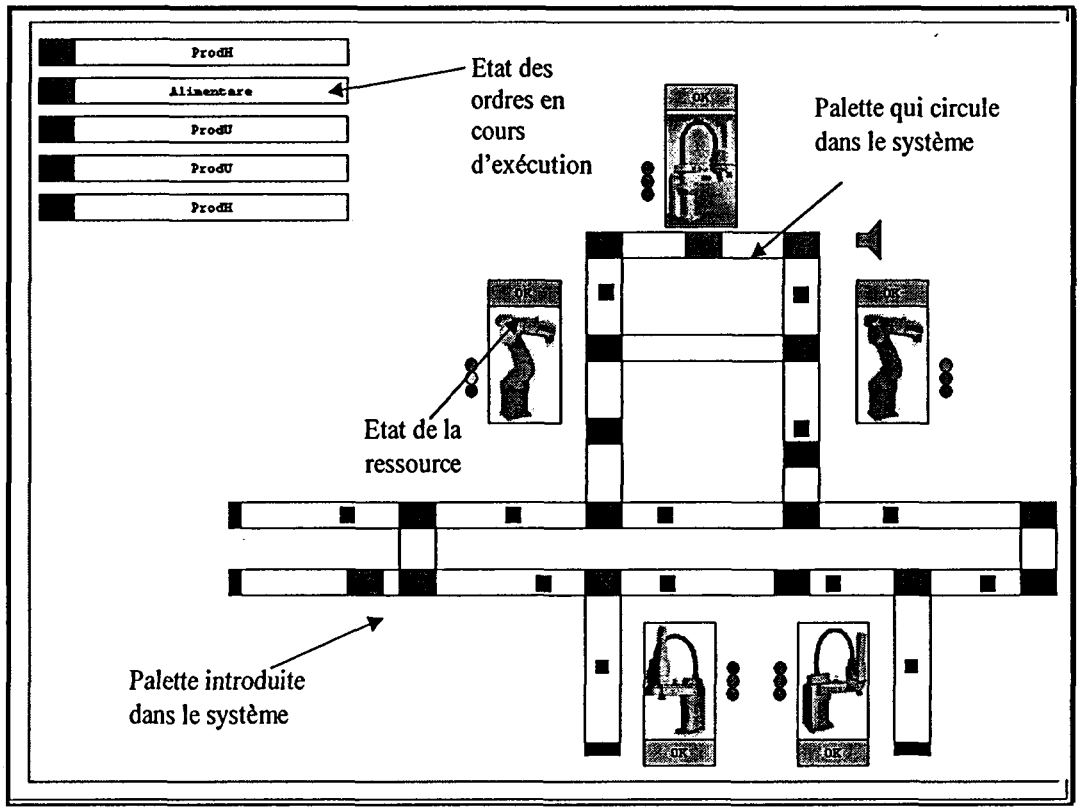


Figure 4.13 : interface graphique pour la gestion et le suivi de la fabrication dans la cellule CIMR

En parallèle avec le projet RVHOLON (www.rvholon.cimr.pub.ro) dans le cadre duquel l'architecture holonique proposée a été expérimentée, un deuxième projet SOFHICOR (www.sofhicor.cimr.pub.ro) dans le domaine manufacturier a permis le développement du concept de système de contrôle holonique semi-hétérarchique de la fabrication.

Dans le cadre de ce projet on a développé la solution d'implémentation HMES (*Holonic Manufacturing Execution System*) basé sur un groupement de ressources en "équipes spécialisées" ou *consortiums* qui offrent une gamme de services complémentaires pour réaliser un lot de produits. En reconfigurant ces consortiums, un comportement agile est obtenu par rapport à la dynamique des changements dans les commandes client.

2 Le protocole

Dans le cadre du projet RVHOLON, les expérimentations suivantes ont été effectuées pour vérifier le fonctionnement du système de fabrication holonique avec contrôle semi-hétérarchique :

1. *Expérimentations* pour la validation *de la planification, de l'ordonnancement, et du simulateur de routage des produits*, en partant des séquences produites par les deux processus précédents ;
2. *Expérimentations sur la communication PC – AP – contrôleur robot.* Le système de communication est un point clé du système de fabrication ; il assure l'échange des données entre les entités autonomes de pilotage et contrôle. Pour cette raison, deux expérimentations ont été réalisées : la première pour tester la communication PC – AP en utilisant le service OPC et la deuxième pour tester la communication AP – contrôleur robot. Le but de ces expérimentations est d'optimiser la communication (minimiser le nombre de variables échangées et le temps total de communication) et de valider le fonctionnement de chaque entité ;
3. *Expérimentations pour vérifier le fonctionnement du système de fabrication* : définition d'un lot de produits, planification et ordonnancement, transmission des ordres de production vers l'AP, exécution des ordres de production, mise à jour en ligne des états des ressources (défaillance et remise en état des ressources, dépôts avec et sans matières premières) et historique des produits après leur exécution et celle du lot. Le but de ces expérimentations est de valider le fonctionnement globale du système de fabrication avec tous ses sous processus et d'ajuster les temps des opérations de traitement et de transport dans le modèle du système de fabrication implémenté dans le simulateur ;
4. *Expérimentations avec différents types de lots de production* (diversité : nombre de produits, nombre d'opérations pour un produit, types de produits dans un lot, précédences entre opérations) pour analyser la dépendance entre le degré d'utilisation du convoyeur et des ressources, et le temps total de fabrication ;

5. *Expérimentations avec des ressources qui tombent en panne et rentrent dans le flux de production* pour valider la *tolérance aux défauts* / la *robustesse* du système de fabrication ainsi que le mécanisme de replanification ;
6. *Expérimentations avec les holons HF d'alimentation* pour valider les deux processus dans lesquels est impliqué ce type de holon : l'alimentation initiale des postes de travail pour commencer la production et la réalimentation des dépôts avec stocks épuisés pendant le processus de fabrication pour utiliser le plus possible les ressources ;
7. *Expérimentations avec l'insertion des ordres rapides* pour valider la capacité du système à s'adapter à la dynamique du marché (*agilité en face des changements fréquents dans les ordres clients*).

Pour toutes ces expérimentations les types des données enregistrées et comparées sont les temps d'exécution :

- *Le temps nécessaire pour la planification et l'ordonnancement.* L'application en charge avec ces calculs a besoin d'un intervalle de temps qui est une fonction du nombre total de produits et d'opérations, avec leurs précédences, nécessaires pour réaliser chaque type de produit ;
- *Le temps nécessaire pour l'exécution d'un lot de produits* ;
- Le *taux de chargement des postes de travail* qui représente le rapport entre le temps total de production des toutes les lots de produits et le temps d'utilisation de chaque poste de travail ;
- *Le temps de réaction à une perturbation de type défaillance / maintenance d'une ressource* : l'intervalle de temps entre la détection de la défaillance / maintenance de la ressource et le moment auquel la fabrication recommence. Cet intervalle est directement proportionnel au nombre des produits restants pour l'exécution physique et leur complexité (nombre d'opérations nécessaires pour chaque produit et précédences entre eux) ;
- *Le temps de réalimentation d'un poste de travail* : l'intervalle de temps écoulé à partir du moment où la rupture d'un certain type de pièce a été détectée dans un stock local d'un poste de travail et le moment où ce poste a pu être réutilisé. Le temps de réalimentation est directement proportionnel avec : (a) la distance entre le poste de réalimentation (**P5** avec les deux équipements *Adept AnyFeeder*) et le poste où il n'y a plus de pièces et (b) la dimension du stock de pièces dans le dépôt. En parallèle avec le processus de réalimentation, les HF déjà créés et en exécution dans le lot sont réordonnancés.

3 Résultats

Ce paragraphe présente les résultats obtenus lors des expérimentations effectuées pour valider les solutions de planification, ordonnancement, routage, suivi de la production, tolérance aux perturbations des ressources (ex : défaillance des ressources) et aux

changements de l'environnement de production (ex : apparition d'ordres urgents, modifications fréquentes dans les ordres de fabrication).

Les tests effectués sur l'application de haut niveau pour la planification et l'ordonnancement des holons HFs et le simulateur de la production associé TTM ont eu comme objectifs de :

- valider le fonctionnement de l'application et ajuster la durée du cycle d'opération du simulateur TTM (figure 4.14) ;
- tester différentes stratégies de contrôle du convoyeur et établir le nombre maximal des palettes qui coexistent dans le paquet de holons HF en exécution simultanée dans le système.

Suite aux nombreux tests effectués, l'application de haut niveau pour la planification et l'ordonnancement des holons HF a été validée, parce qu'elle a produit des temps totaux de production toujours inférieurs aux séquences non planifiées.

D'autre part, la dimension maximale du lot de produits qui peuvent être introduits pour exécution simultanée sans induire des blocages temporaires ou permanents dans le système de transport est de quatre; ainsi, cinq holons HF peuvent coexister pour un temps court dans le système ; une cinquième palette peut être introduite dans le système dés qu'un holon HF a subi toutes les opérations ordonnancées pour son produit associé et débute son chemin vers le poste de sortie **P0** (Raileanu et al., 2009), voir figure 4.15.

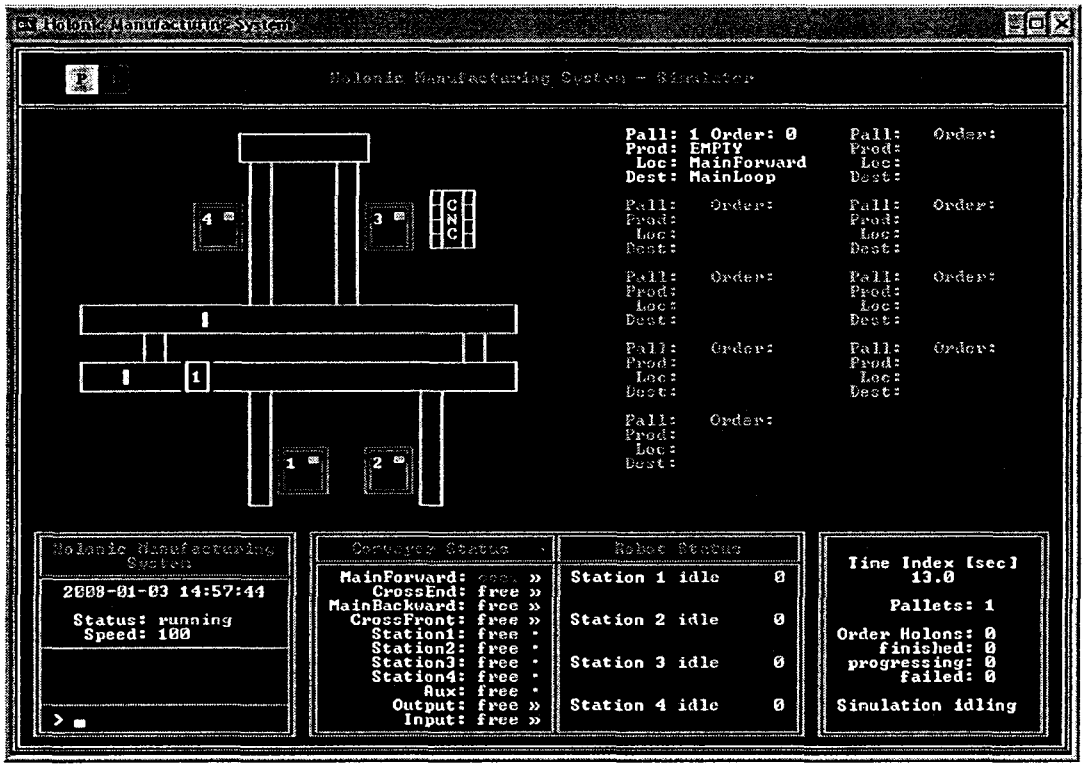


Figure 4.14 : outil de Simulation Dynamique (OSD)

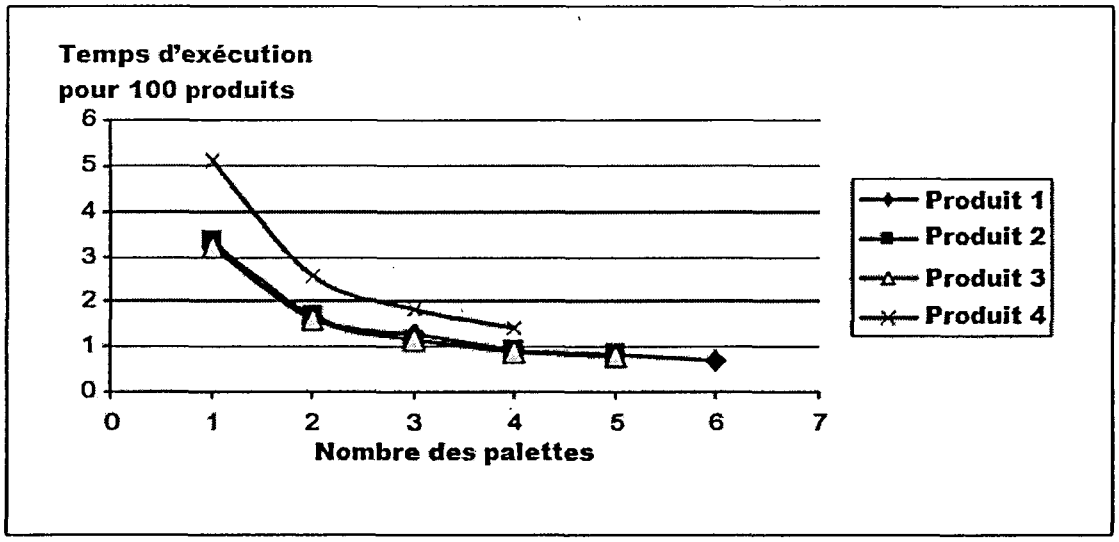


Figure 4.15 : temps d'exécution pour différents nombres de palettes insérées dans le système physique

Le deuxième groupe de tests a été fait pour valider les protocoles de communication conçus, ainsi que les séquences de transfert de données et d'informations entre les ressources de calcul du système de pilotage de la cellule de fabrication CIMR de type *shop floor*, reliées par un système de communication multi réseau ayant la topologie présentée en Fig. 4.16 :

- serveur de la cellule, du type IBM xSeries 3500 ;
- ordinateurs des six stations de travail **P0-P5**, du type IBM PC ;
- automate programmable distribué en réseau de deux clusters : (a) un cluster principal qui crée les holons flux de fabrication et d'approvisionnement; (b) un cluster secondaire qui gère les holons HF du type approvisionnement ; l'AP est du type *Bosch Rexroth IndraControl* L40 ;
- contrôleurs des robots des six stations de travail **P0-P5**, du type *Adept Smart Controller*.

Le système de communication multi réseau implémenté est basé sur quatre types de connexions, représentées sur la figure. 4.16:

1. *Réseau Ethernet* : directe, hiérarchique (Serveur SC [Serveur Cellule] - Client OS [Ordinateurs des Stations]) et commuté, hétérarchique (Serveur OS - Client CR [Contrôleur Robot]) ;
2. *Connexion Directe* : Série RS 232 OS *i* - CR *i* [Ordinateur - Contrôleur Station *i*, $0 \leq i \leq 5$;
3. *Anneau (Ring)* : CR *i*-1 - CR *i* - CR *i*+1 ;
4. *Point -à-Point* (PTP) : par environnement d'E / S digitales

La tolérance aux pannes diverses est assurée par ce réseau multiple dans les situations suivantes qui ont été testées et validées lors des expérimentations :

- La défaillance d'un contrôleur robot CR (qui est aussi contrôleur de station de travail, car il assure la liaison entre les flux de transport et le flux de traitement des matériaux) est détectée par surveillance continue des lignes série directes OS *i* - RC *i* et cause la mise à jour du modèle d'accès aux services des robots MdAaS et par conséquent le réordonnancement des tâches spécifiques vers d'autres robots ;
- Si un OS tombe en panne, sa place este prise par un autre OS valide, car les données OS sont répliquées en temps réel entre les OS via Ethernet directe (par rapport au serveur) et Ethernet commuté (par rapport aux CR) ;
- Si le commutateur (*switch*) tombe en panne, la communication entre les clusters OS est toujours possible due aux connexions directes [OS *i* – RC *i*] et aux connexions en anneau [RC *i* – RC *j*] ;
- Le conditionnement inter opérations entre les tâches RC (exclusion mutuelle, synchronisation, etc.) est assuré par le réseau PTP formé par lignes d'E/S digitales.

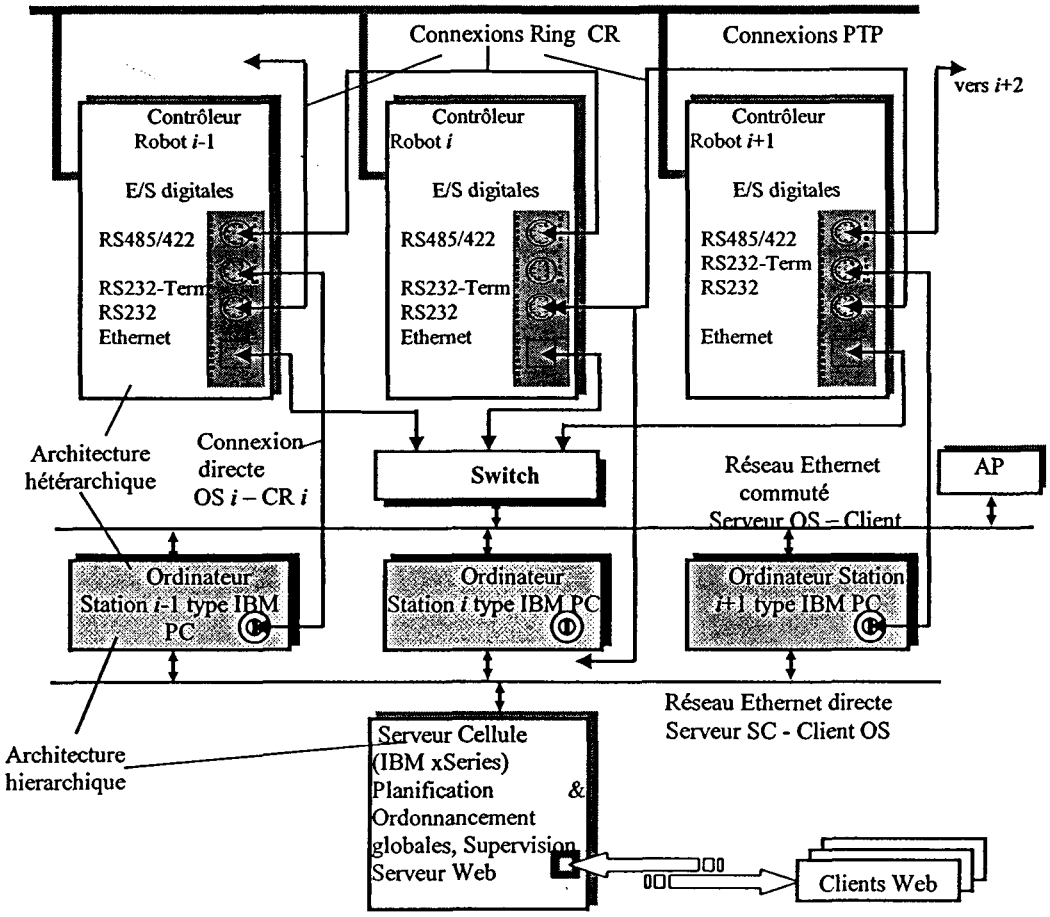


Figure 4.16 : système de communication multi réseau tolérant aux pannes pour le pilotage de la cellule

Des tests ont été effectués pour vérifier la correspondance entre les temps de production planifiés de manière optimale et les temps réellement obtenus lors de l'exécution des séquences des holons HF planifiés. Ainsi, un lot composé de 6 produits (4 produits type

prodH, 1 produit type prod2 et un produit type prod3 - voir figure 4.8) avec la distribution des opérations associées indiquée figure. 4.7 a été planifié et ordonnancé avec l'algorithme *SchedulerV2* présenté en chapitre 3 et ensuite exécuté sur le système réel de fabrication de la cellule CIMR.

Pendant le processus de fabrication, les informations concernant l'exécution des produits (date de début et date de fin de chaque opération et produit) ont été mémorisées pour être ensuite comparées avec les données théoriques, calculées par le serveur de la cellule. Les résultats sont décrits dans le diagramme figure 4.17.

Cette comparaison met en évidence une différence entre les temps planifiés [prodx(th)] et les temps réels de fabrication [prodx(rx)] de chaque produit, qui varie entre 10% et 20%, ce qui ralentit la production entière. L'explication est due au fait que le modèle interne de l'Outil de Simulation Dynamique OSD utilise des temps de transport fixes, qui sont la moyenne des temps de transport mesurés pendant plusieurs expérimentations, et qui ne sont pas forcément réactualisés pendant l'exécution de la production.

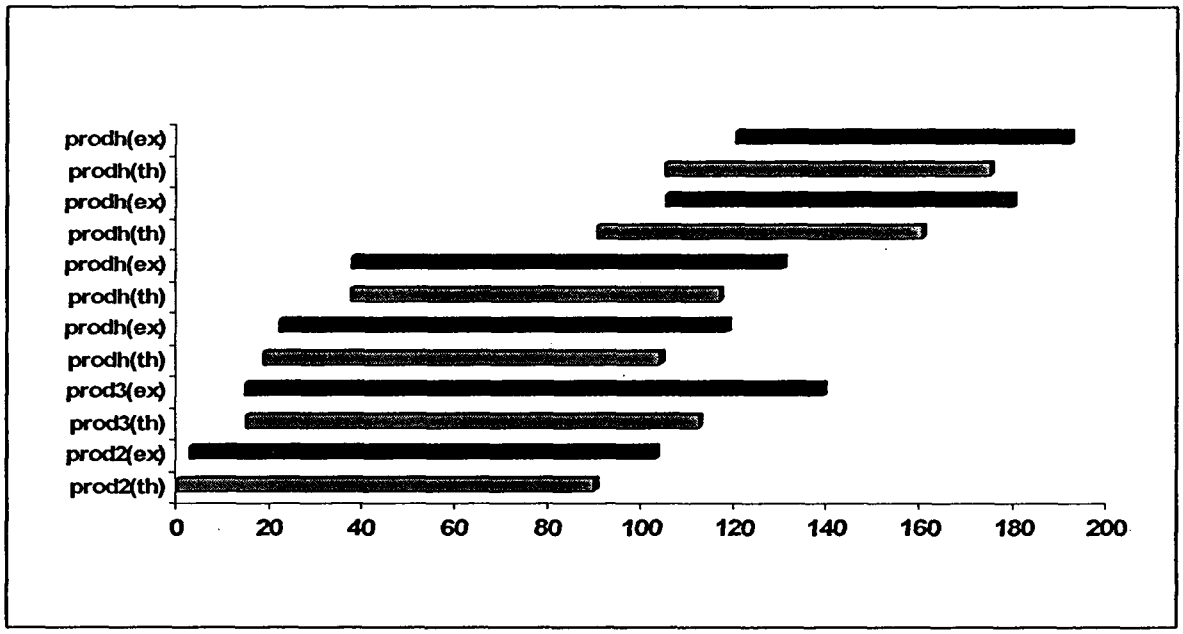


Figure 4.17 : comparaison entre les temps planifiés (vert) et les temps de production mesurés (bleu)

De cette façon, parce que les temps réels de transport peuvent augmenter sans que le niveau de planification et d'ordonnancement soit informé, les ordres sont ordonnancés théoriquement plus vite. Dans le cas ci-dessus (figure 4.17), ces ralentissements n'affectent que la date de fin de chaque produit ce qui prolonge la date de fin de toute la production.

Pour offrir aux clients des dates de fin plus précises dans les conditions présentées ci-dessus il y a deux possibilités : la première est d'ajouter un mécanisme de suivi en temps réel de la production qui modifie le modèle du système ; la deuxième possibilité est d'estimer comment les différences entre le modèle et le système réel varient et ajouter

ensuite des valeurs de compensation aux dates initialement proposées. Dans le premier cas chaque fois que les temps de transport/traitement physique changent, l'algorithme *SchedulerV2* sera relancé et si une meilleure solution (planification et ordonnancement) est trouvée elle devient la nouvelle planification de base. Parce que le temps de changement de la production n'est pas négligeable pour le type de produits considérés (environ 2 minutes) le changement est admis seulement si la nouvelle planification et le nouvel ordonnancement sont nettement supérieurs aux anciennes valeurs.

Pour un quatrième scénario la même configuration de ressources et le même lot de produits (1 prod2, 1 prod3, et 4 prodH) ont été considérés. Dans ce cas, l'algorithme *StepScheduler* a été exécuté plusieurs fois et les résultats (la correspondance entre le degré d'utilisation de chacune des quatre ressources disponibles et le temps total de production) sont présentés figure 4.18. Chaque *pie chart* représente le pourcentage des opérations qui ont été exécutées sur chaque ressource et la valeur en bas de chaque représentation graphique indique le temps total de fabrication mesuré en unités de temps de 0,5 secondes.

Même si le lot des produits n'a pas une grande taille (6 produits), la figure 4.18 montre qu'une utilisation égale des ressources (le premier cas à gauche en haut - 319 unités temps) et la réduction du nombre de boucles qu'un produit doit faire sur le convoyeur principal contribue à la réduction du temps total de production.

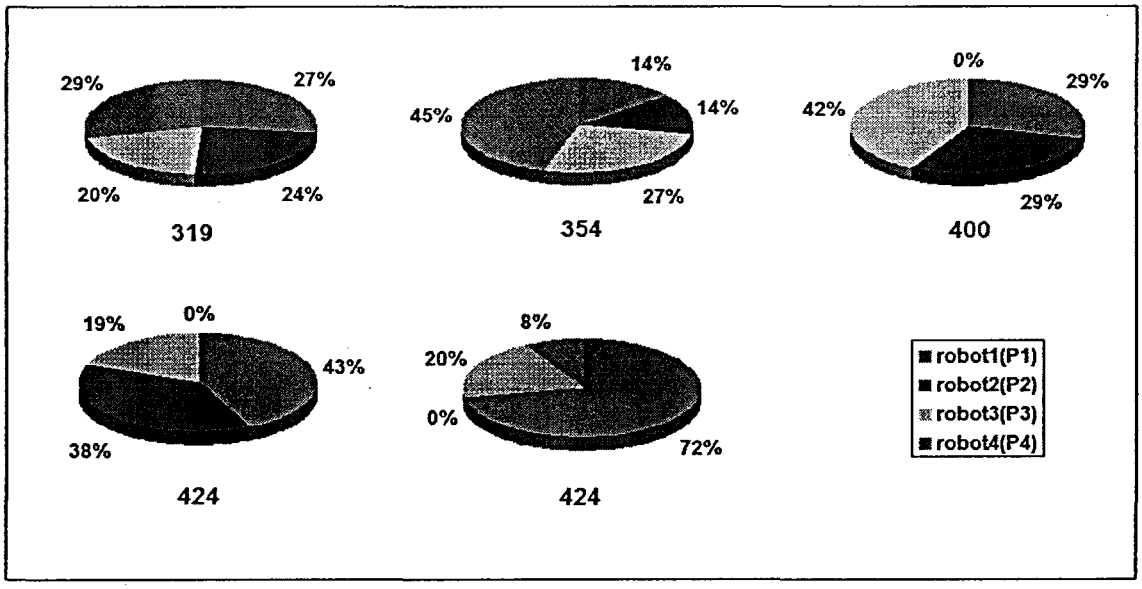


Figure 4.18 : relation entre le temps total de production et l'utilisation des ressources pour un lot de 6 ordres de production

Utilisant la même configuration mais avec un lot plus grand de produits (touts les produits précédents 10 fois chacun) l'application de planification et d'ordonnancement a été exécutée plusieurs fois et les résultats obtenus sont représentés figure 4.19.

Comme dans le cas précédent, le temps total de fabrication diminue si les ressources de traitement sont utilisées le plus uniformément possible.

Pour analyser l'influence de la diversité de complexité des produits inclus dans le même lot de production, un lot de 30 produits (10 simples, 10 de complexité moyenne et 10 complexes) a été planifié et ordonnancé en utilisant plusieurs distributions des opérations dans les postes de travail :

1. les ressources R1-R4 offrent les mêmes services (S1-S4) avec les mêmes temps de réalisation ;
2. R1, R2 offrent {S1, S3} et R3, R4 offrent {S2, S4} ;
3. R1, R3 offrent {S1, S2} et R2, R4 offrent {S3, S4} ;
4. R1 offre S1 (ressource critique), R2 offre S2, S3; R3 offre S3, S4; R4 offre S2, S4 ;
5. R3 offre S1 (ressource critique), R1 offre S2, S3; R2 offre S3, S4; R4 offre S2, S4.

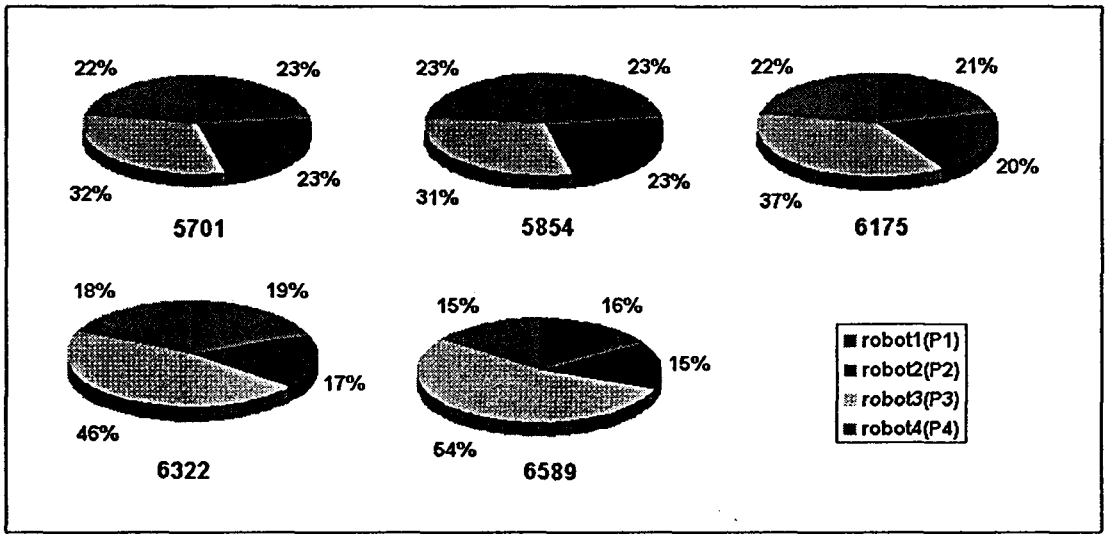


Figure 4.19 : relation entre le temps total de production et l'utilisation des ressources pour un lot de 100 ordres de production

L'interprétation des résultats obtenus pour les scénarios testés, présenté ci-dessus, est faite dans la figure 4.20; elle conduit aux conclusions suivantes :

Le temps total de fabrication est inversement proportionnel avec la *flexibilité de la cellule* par rapport aux opérations nécessaires pour l'exécution des produits : si la flexibilité diminue (scénario 1 le plus flexible, scénarios 4 et 5 les moins flexibles), le temps total de fabrication augmente. Le deuxième scénario est une exception (temps de production plus grand) à cause de la structure particulière des produits (figure 4.8) par rapport à la distribution des opérations, ce qui fait que les holons HF doivent boucler plusieurs fois. Cette circulation multiple augmente le temps total de production ;

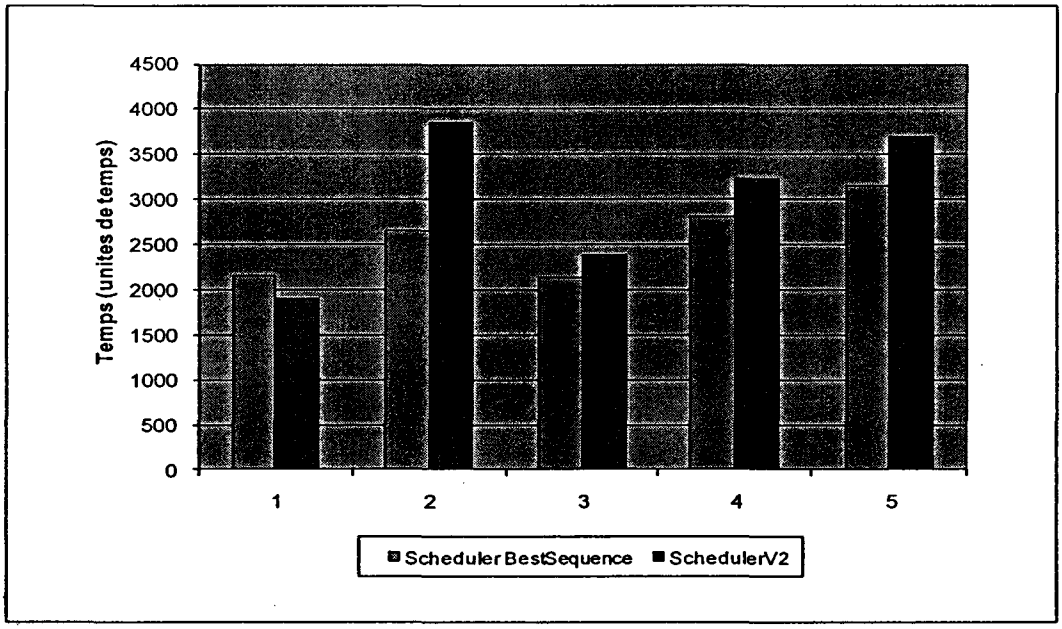


Figure 4.20 : comparaison entre les temps de production (*makespan)* pour les cinq modalités de distribution des opérations dans les postes de travail

Il est envisageable d'éviter le cas où il y a une *ressource critique* (elle est la seule qui peut exécuter un certain type d'opération) dans le système : les deux derniers scénarios ont une ressource critique ce qui augmente beaucoup le temps de production à cause des blocages qui se forment devant cette ressource. Cependant, s'il y a une ressource critique, elle doit être bien positionnée par rapport aux types des produits qui doivent être exécutés (ex : l'exécution de touts les produits représentés figure 4.8 commence avec une opération S1 ; dans le scénario 4 cette opération est attribuée au poste **P1** qui se trouve près de l'entrée dans le système de fabrication et le produit ne doit pas être « recyclé » ; dans le scénario 5 l'opération est attribuée à un poste vers la sortie du système ce qui fait que le produit soit recyclé) ;

Une conclusion finale est que l'algorithme *SchedulerV2* est plus efficace pour des configurations flexibles avec produits identiques tandis que l'algorithme *StepScheduler Best Sequence* est plus efficace pour des configurations non flexibles avec produits homogènes.

Un cinquième scénario est l'expérimentation du pilotage de la production quand une ressource tombe en panne. Ce scénario a été testé sur le simulateur pour valider la tolérance aux défauts du système de pilotage et sa capacité de replanification en fonction de l'état réel des ressources.

Avec la configuration du système de fabrication décrite figure 4.7, la procédure de replanification et de réordonnancement a été testée sur un lot de 8 produits (2 prod1, 2 prod2, 2 prod3 et 2 prod4, voir figure 4.8) en utilisant la procédure *StepSchedulerV2.*

Le test montre les temps d'exécution du lot de production précédent, avant et après la défaillance de la ressource du poste **P1** (figure 4.21). Cet événement est repéré sur la figure

4.21 par l'éclair jaune qui apparaît immédiatement après le départ du troisième produit du poste **P1**.

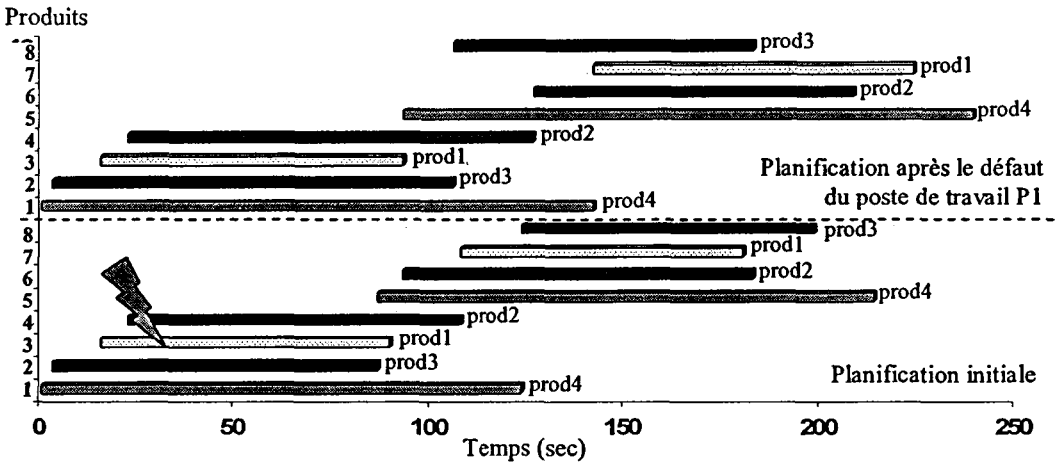


Figure 4.21 : temps d'exécution des ordres de production avant et après le défaut du poste de travail P1

Le résultat présenté dans figure 4.21 indique que le système de pilotage fonctionne bien en présence de perturbations, mais à cause de la structure particulière du système de transport (convoyeur principal en boucle fermée avec déviations qui apportent les palettes dans les postes de travail) et aussi à cause des files d'attentes avec des dimensions limitées, que les palettes doivent obligatoirement parcourir, des blocages se forment et ils ralentissent les autres palettes. Dans ce cas, le dernier produit prod3, porté par la palette numéro 8, est exécuté plus vite après la replanification en comparaison avec la planification initiale, mais au niveau du lot de produits la planification après le défaut du poste de travail P1 est moins bonne (temps total d'exécution plus grand) que la planification initiale.

Dans le cadre du même type de scénario, des expérimentations ont été faites sur le simulateur DST pour connaître les avantages de la capacité du système à tolérer les défauts et continuer la production sans intervention extérieure. Ainsi, un lot de 30 produits (10 simples, 10 de complexité moyenne et 10 complexes) avec la première distribution des opérations du scénario précédent a été utilisé comme entrée pour le simulateur. Les expérimentations se sont déroulées comme suit :

(1) Le lot a été planifié et ordonnancé en utilisant toutes les ressources disponibles ; les informations résultantes : temps total de fabrication, chargement des ressource (figure 4.22 - a), allocation des opérations des holons HF on été retenues ;

(2) Avec les informations obtenues comme ci-dessus, le simulateur de la production DST a été lancé et après l'exécution des premiers quatre produits l'état du poste de travail **P4** a été changé en hors ligne (ressource associée en panne). A cet instant l'application de gestion de la production a détecté la perturbation et a refait la

planification et l'ordonnancement en éliminant la ressource en panne. Les résultats obtenus sont présentés sur la figure 4.22 - b ;

(3) Pour avoir un étalon de comparaison, le lot a été planifié et ordonnancé en utilisant seulement les postes de travail P1, P2 et P3 qui restent disponibles après la défaillance du poste P4. Les résultats obtenus sont présentés sur la figure 4.22 - c.

L'analyse comparative de ces trois expérimentations (figure 4.22) montre que le système est capable de continuer la production sans s'arrêter (des ressources peuvent être ajoutées ou retirées du flux de production) et grâce à la flexibilité des postes de travail les holons HF sont re-routées vers des ressources alternatives, sans que le temps total de production dépasse la limite supérieure imposée par le scénario **c**. La différence de temps entre les scénarios **b** et **c** de 94 unités de temps est comparable avec le temps en début de la production quand toutes les ressources sont disponibles (233 unités de temps). Cette différence contient aussi le temps de replanification et de réordonnancement qui interviennent dans le scénario **b** augmentant le temps total.

Les temps de replanification et réordonnancement sont négligeables (moins d'une seconde pour un lot de 30 produits), mais pour un lot plus grand de produits la différence sera plus faible. Pour cette raison nous avons proposée de séparer l'allocation des holons HF en cours d'exécution (qui peut s'exécuter plus rapidement) de la planification et de l'ordonnancement des holons HF en attente (qui réclament un temps beaucoup plus important), et d'exécuter les deux processus en parallèle. Ainsi, la production peut recommencer directement après le réordonnancement des holons HF du lot en cours d'exécution, ce qui a comme effet la réduction du temps total de production (figure 4.11).

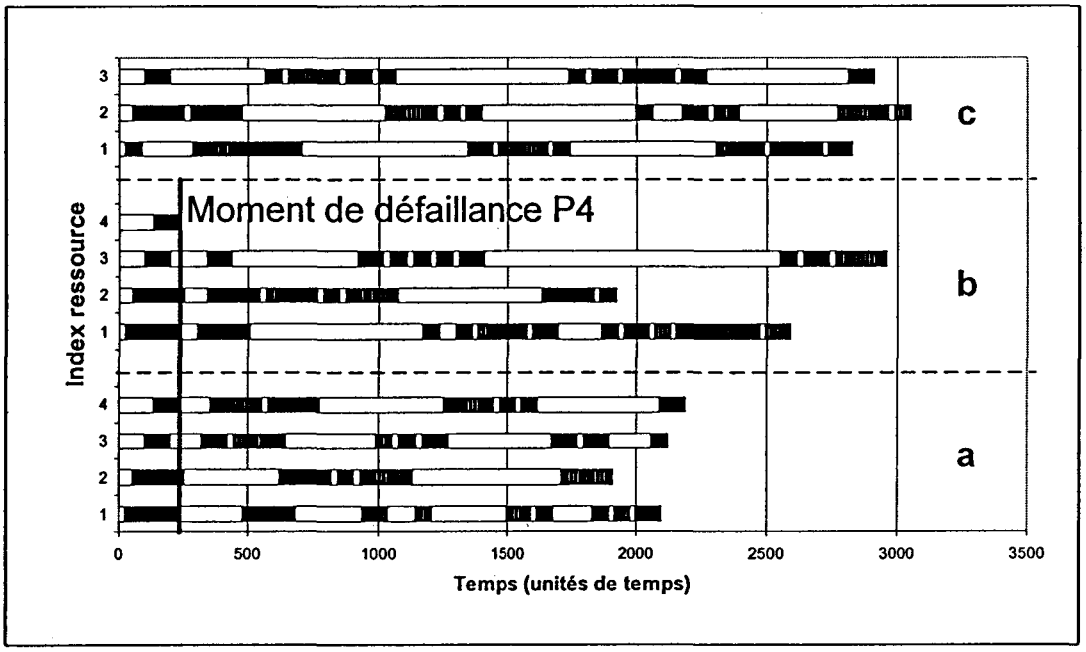


Figure 4.22 : comparaison des chargements des postes de travail a) avant la panne de P4, b) après la panne de P4 et c) en utilisant seulement les postes P1-P3 du début à la fin

Pour le sixième scénario un lot de dix produits de type prodH a été choisi pour mesurer les temps d'alimentation initiale et de réalimentation pendant le processus de fabrication ;

les deux temps représentent les indicateurs de qualité des processus. Dans la composition de ces deux indicateurs entrent plusieurs facteurs qui seront mesurées indépendamment :

- *Le temps de chargement d'une palette* avec les pièces nécessaires, qui est un processus statistique. Ainsi, une valeur moyenne (Tableau 4.1) pour un nombre fixe de pièces (Tableau 4.2), sera utilisée dans le calcul des indicateurs d'intérêt ;

I	T	r	L	Non L	Moyenne
68	66	59	108	204	75

Tableau 4.1 – Le temps moyen de chargement d'une palette (sec)

I	T	R	L	Non L
24	25	29	7	7

Tableau 4.2 – Le nombres de pièces utilisées dans les stocks intelligents

- *Le temps de transport de la palette* entre le poste d'approvisionnement **(P5)** et le poste qui nécessite l'alimentation de son stock local (Tableau 4.3) ;

	Poste 1	Poste 2	Poste 3	Poste 4
Aller	38	48	29	12
Retour	40	37	12	24

Tableau 4.3 – Les temps de routage entre le poste P5 et le poste qui nécessite l'alimentation

- *Le temps d'alimentation* (les *n* opérations d'alimentation du stock) des postes avec stock épuisé (Tableau 4.4) ;

I	T	R	L	Non L
15sec	15sec	15sec	15sec	15sec

Tableau 4.4 – Temps de réalimentation des postes de travail

- *Les temps de communication* entre le contrôleur robot et l'automate programmable interviennent aussi dans les indicateurs de qualité, mais sont négligeables (moins d'un seconde) en comparaison avec les autres temps.

La conclusion de ce scénario est que l'alimentation en ligne des postes de travail permet de continuer la fabrication des produits sans arrêt, ce qui optimise le temps total de production : dans le cas d'un ou plusieurs lots de production ayant beaucoup d'ordres qui nécessitent un nombre de pièces plus grand que les stocks des postes de travail, il est nécessaire de faire la réalimentation pendant le processus de production. Si la cellule est arrêtée complètement et la réalimentation se fait en mode manuel, cela prend plus de temps

et la production est arrêtée (Tableau 4.5) ; par contre, si la réalimentation est faite automatiquement la production continue et les postes sont réalimentées en parallèle avec la fabrication des produits du lot courant qui se poursuit.

Type de production	Temps (sec)
Production avec alimentation manuelle: 10 produits type H + alimentation manuelle + réinitialisation manuelle des ressources + 1 produit type H	420
Production avec alimentation automatisée	1200 (alimentation et réalimentation inclues)
Réalimentation poste 1	78
Réalimentation poste 2	100

Tableau 4.5 – Comparaison des temps de production

Même si la production avec alimentation et réalimentation automatisées prend plus de temps que celle avec alimentation manuelle, la différence est moindre que le temps nécessaire au remplissage manuel de tous les postes de travail avec la réinitialisation des ressources associées ; ces deux derniers processus étant répétitifs et laborieux et nécessitent beaucoup de temps. En conclusion, l'automatisation des processus d'alimentation et de réalimentation, et leur intégration dans le processus de production apporte des avantages importantes quand la demande de produits est très grande, comme dans le cas de la production en série : le personnel en charge avec le processus de production remplit les deux dispositifs d'approvisionnement du type *AnyFeeder* avec des pièces ; ensuite le système trie les composants par type, les distribue dans les postes de travail et débute la fabrication.

Le dernier scénario, **traitement des ordres rapides**, a été implémenté en utilisant la méthode EDF (Sha et al., 2004). La figure 4.23 illustre les résultats obtenus quand deux lots de produits $T_24 = (4, 17)$ si $T_25 = (1, 3)$ sont reçus au moment $T = 2$ après le début de l'exécution des trois premiers lots déjà planifiés $T_11 = (2, 18)$, $T_12 = (3, 20)$, $T_13 = (7, 11)$. Le terme $T_{ij} = (m, dd)$ a la signification suivante : (j) est le numéro du lot, (i) est le numéro de la session d'insertion des lots (1 – les premiers lots insérés, 2 – les lots insérés quand les premiers lots sont en cours d'exécution). Ce lot a un temps de production m et une date de livraison dd, les deux étant exprimés en unités de temps relativement au début de la production. Sur la figure 4.23 la date de fin de chaque lot est marquée avec **lot***i*, i=1…3 pour les premiers lots et i=1…5 pour les lots suivants.

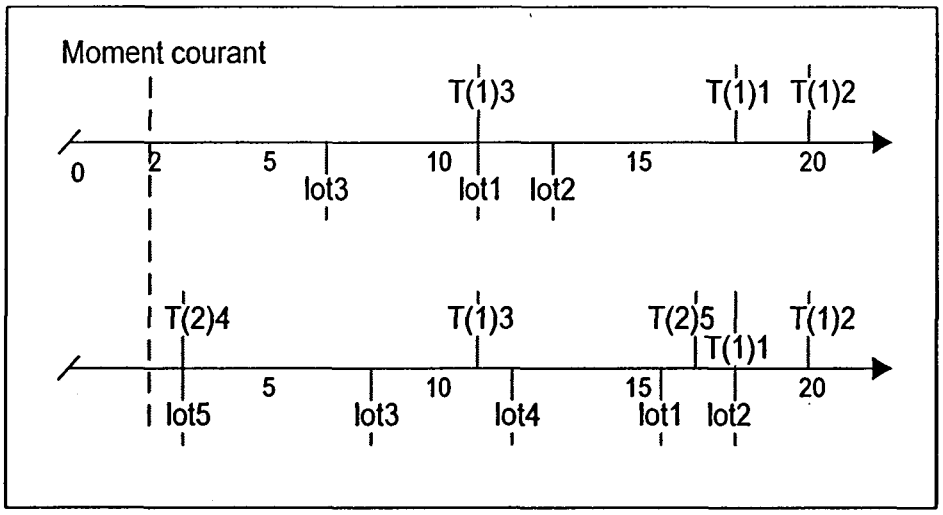


Figure 4.23 : replanification pour l'insertion des ordres rapides de production

4 Analyse et synthèse

Le but de cette recherche est de développer une architecture de contrôle, inspirée par des concepts holoniques, pour un système de fabrication décentralisé et de valider cette architecture par une implémentation réelle sur une plateforme. Un point fort de l'implémentation est le développement de l'application de gestion intégrée de la fabrication qui prend en charge :

- la description des produits en termes d'opérations nécessaires et des précédences entre elles ;
- la réception des ordres clients ;
- la planification, l'ordonnancement et le lancement en exécution des ordres de production ;
- l'alimentation et la réalimentation des postes de travail ;
- l'insertion des ordres de production rapides ;
- le suivi de l'exécution des ordres de production dans le système de fabrication réel.

L'application a été développée a partir des concepts généraux qui se trouvent dans la littérature spécialisée (van Brussel, 1998 ; Barata, 2006 ; Sauer, 2008 ; Nylund, 2008) en s'appuyant sur les systèmes de fabrication holoniques et essayant de maintenir un degré de généricité élevé pour que l'architecture soit réutilisable dans le cadre d'autres projets dans des domaines qui impliquent le transport automatisé : productique, services ou transport de passagers.

Pour le moment l'application a été testée sur la plateforme réelle CIMR pour la planification, l'ordonnancement et l'exécution de lots de production de maximum dix types de produits nécessitant jusqu'à 16 opérations d'usinage, de montage / assemblage, et de contrôle visuel de la géométrie des produits (dimensions, positions relatives entre composantes assemblées). Les tests effectués confirment la qualité des solutions proposées

en termes de planification, d'ordonnancement et de routage des lots de produits pour une exécution optimisée (temps de fabrication des lots, tolérance aux défauts, agilité).

Les expérimentations faites sur la cellule flexible de fabrication CIMR de l'Université Politehnica de Bucarest (www.cimr.pub.ro) ont confirmé les avantages suivants :

- *production optimale* (temps d'exécution d'un lot de production minimisé) quand le système de pilotage fonctionne avec planification et ordonnancement centralisés ; le système de pilotage de la cellule de fabrication a une vue générale de la fabrication ce qui permet l'optimisation avec contraintes des processus impliqués ;
- *agilité* : il est possible de traiter et insérer des commandes clients rapides même si le processus de fabrication est en cours d'exécution (la procédure EDF proposée en 3.3.3 a été testée dans le sous chapitre précédent) ;
- *robustesse en face de perturbation et tolérance aux défauts* : en cas de défaillance des ressources ou de rupture dans les stocks locaux, la fabrication continue en mode dégradé suite à une replanification faite en ligne ;
- *auto-apprentissage* – le MdAaS est mis à jour grâce au système de suivi des ressources en temps réel et en utilisant l'historique des produits exécutés. Le MdAaS, en entrée dans le processus de décision pour la planification et l'ordonnancement des holons HF, permet la reconfigurabilité du système cible en opposition avec la reprogrammation - plus rigide et qui nécessite un temps beaucoup plus important ;
- *l'extension du contrôle explicite* au *contrôle implicite de type sociétal* qui permet l'implémentation d'un système de pilotage décentralisé ;
- *l'utilisation du contrôle implicite de type environnemental* pour tenir compte des caractéristiques physiques des holons HF (corrections des actions des robots, qualification des composants matériels, interactions avec le tapis du convoyeur en mouvement).

Les avantages évoqués ci-dessus sont parties intégrantes de la solution de pilotage semi-hétérarchique développée, qui permet d'adapter le mode de fabrication aux états de fonctionnement des ressources utilisées, aux changements de l'environnement de travail et aux demandes variables et imprévisibles des clients.

5 Conclusion

En conclusion, nous avons constaté que les résultats des expérimentations réalisées sur l'architecture de pilotage holonique semi-hétérarchique sont satisfaisants en termes de modularité (de nouvelles ressources peuvent être ajoutées ou retirées pour le traitement physique des produits sans apporter de modifications au niveau des modèles de gestion), mais il reste cependant beaucoup d'améliorations à apporter au niveau de l'allocation des ressources.

Parce que les paramètres du système de fabrication sont évolutifs, en particulier les temps de transport et d'occupation des tronçons de transport, cette architecture sera adaptée par la suite dans une approche donnant davantage d'autonomie aux produits, en se basant sur le concept de "Système Contrôlé par le Produit" et de "Produit intelligent".

Chapitre 5 - Proposition d'un modèle de pilotage holonique pour un système de transport de type PRT

Introduction

Ce chapitre présente un modèle d'un système de pilotage d'un système de transport de type PRT (l'annexe 1 fournit une description détaillée d'un système PRT) s'appuyant sur le modèle holonique SEAH proposé au chapitre 2. Nous proposons un système de type "Open Control" à deux niveaux, constitué d'une part, d'un système centralisé chargé de l'allocation des véhicules et, d'autre part, d'un système de routage dynamique exploitant l'autonomie décisionnelle des véhicules du réseau.

La démarche de modélisation holonique s'appuie sur le modèle structurel holonique (figure 2.10) présenté au chapitre 2. Elle comporte les étapes suivantes:

- modélisation structurelle d'un système de transport de type PRT (structure statique, structure dynamique). Elle est décrite dans la première partie du chapitre, et illustrée par un exemple,
- modélisation comportementale : nous proposons dans la deuxième partie du chapitre deux approches illustratives du comportement dynamique des holons du système PRT. La première approche est fondée sur une communication indirecte, via l'environnement, entre les holons. La seconde approche est basée sur une communication directe entre holons, par échange de messages. Cette dernière approche a fait l'objet d'une mise en œuvre et d'une validation expérimentale présentée dans le chapitre suivant,
- identification des flux : les flux sont définis conformément au tableau 2.2 établi au chapitre 2.

1 Modèle structurel holonique d'un système de transport PRT

1.1 Structure statique d'un système de transport PRT

Elle est constituée des holons coordinateur HC, des Holons Ressources et du Holon Connaissance Système.

1.1.1 Holon coordinateur HC

Le holon coordinateur HC est constitué par le système de pilotage centralisé. Il gère l'ensemble des informations lui permettant d'assurer l'allocation des véhicules, de maintenir à jour l'état du réseau en fonction des informations reçues des véhicules et des stations, de gérer l'exploitation du réseau (répartition et allocation des véhicules libres présents dans les stations, gestion des demandes...).

1.1.2 Holon Ressource HR

Les Holons Ressources HR correspondent aux composants du système PRT permettant d'assurer l'ensemble des services nécessaires au déplacement des passagers : les véhicules, les stations et les voies de transport.

Les informations essentielles relatives au pilotage associées aux Holons Ressources sont :

- HR véhicule : identificateur, état du véhicule (libre, en service, nombre de passagers transportés...), les services disponibles et les informations associées à chacun de ces services offerts par le HR, destination du véhicule, position du véhicule, historique des déplacements.

- HR station : identificateur, localisation, état de la station (accessibilité, nombre de véhicules présents, nombre de véhicules libres disponibles), historique du service (nombre de véhicules en entrée et sortie, nombre et caractéristiques des demandes de transport effectuées à la station...).

- HR voies de transport : disponibilité, taux de charge (nombre de véhicules sur le tronçon).

1.1.3 Holon Connaissance Système HCS

Le système est dans le cas d'un réseau de transport, l'ensemble des constituants et des services que le réseau est capable de fournir aux passagers. Par conséquent, le Holon HCS a en charge la gestion de l'ensemble des connaissances caractéristiques des constituants (véhicules, stations, infrastructure...) du réseau et des services de transport offerts par le réseau :

- informations géographiques : cartographie du réseau, localisation et caractéristiques des stations et des voies, stations intermodales et réseaux associés...),

- les informations temporelles : horaires prévisionnels si gestion en cadence du réseau ou en heure creuse, plans de déplacements types optimisés (en temps et/ou en distance) : stations à emprunter, horaires prévisionnels d'arrivée à destination, horaires de passage aux stations intermédiaires...),

- les informations techniques : les caractéristiques des véhicules (nombre de places, vitesse maximum, autonomie dans le cas d'un fonctionnement sur batterie...), les stations (nombre maximum de véhicules en service et de véhicules vides pouvant être accueillis...), les voies (type de voie (au niveau du sol, aérienne), géométrie (linéaire, embranchement, aiguillage...), capacité en nombre de véhicules, vitesse maximum à respecter...).

1.2 Structure dynamique d'un système de transport PRT

1.2.1 Le Holon Flux HF

La structure dynamique correspond au Holon Flux HF. Il comporte le Holon Objet Receveur des Services HORS, le Holon Services à Obtenir, le Holon Mobilité et le Holon conduite HCO.

1.2.2 Holon Objet Receveur des Services HORS

Le holon Objet Receveur des Services est constitué par le passager qui reçoit un service de déplacement sans apporter de service lui-même. C'est, avec le Holon Mobilité, un des éléments du flux physique du réseau.

1.2.3 Holon Services à Obtenir

Le service à obtenir est, dans le cas d'un système PRT, le déplacement d'un passager. Le holon Service à Obtenir HORS comporte ainsi l'ensemble des éléments définissant le déplacement à effectuer pour répondre à la demande spécifique d'un passager :

- informations géographiques (stations de départ et de destination, liste des stations intermédiaires potentielles pouvant constituer le trajet du véhicule...),

- les informations techniques (identification et caractéristiques du (des) véhicule(s), identification et caractéristiques des voies à emprunter...).

1.2.4 Holon Mobilité

Le véhicule constitue le Holon Mobilité. C'est un des composants mobiles du flux physique.

1.2.5 Holon conduite HCO

Le HCO est constitué, dans le cas d'un système hybride, par le PRT. Il détermine, en temps réel, la trajectoire du PRT dans le réseau de transport.

Il détermine, en fonction de l'état du réseau (demandes de transport des voyageurs, stations et tronçons de voies accessibles, position du véhicule...) le cheminement du véhicule.

Le HCO gère les informations concernant :

- la liste des stations et tronçons réellement visités, les horaires de passage à ces stations,

- la liste des évènements survenus : pannes, réactions à la présence d'obstacles sur la voie, liste des tronçons indisponibles, arrêt à une station pour embarquement non prévu de passagers...).

Mécanisme d'agrégation – désagrégation

Le mécanisme d'agrégation/désagrégation a été décrit au chapitre 2 (§ 4.5). Il est rappelé, figure 5-1.

Le Holon Flux HF est initialement composé de tout ou partie des holons HORS, HSO, HM et HCO. L'agrégation et la désagrégation des holons composant le holon HF se manifeste physiquement par l'agrégation et la désagrégation partielles ou totales des supports des holons composant un holon HF.

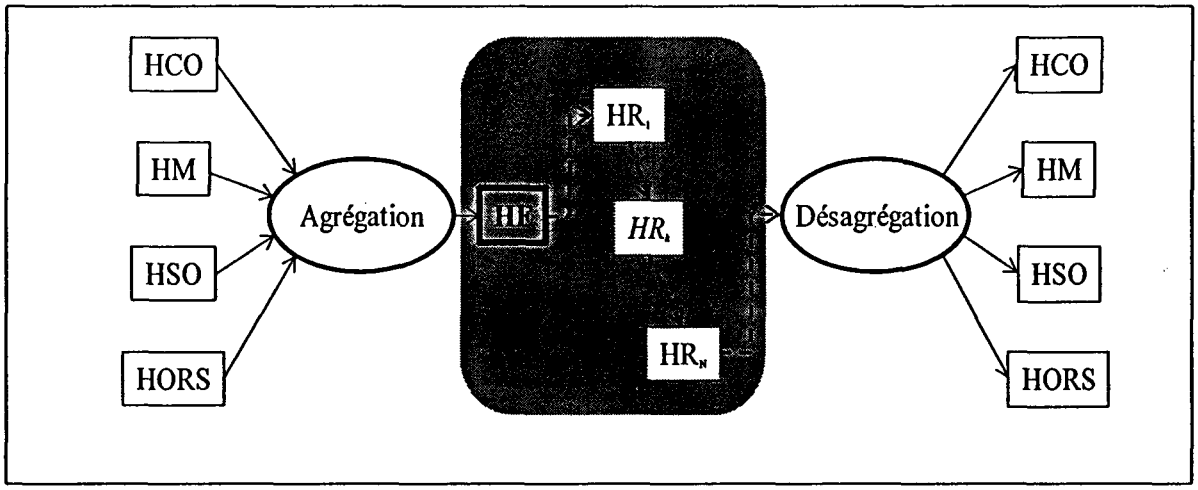


Figure 5.1 : mécanisme d'agrégation – désagrégation

1.3 Exemple de modélisation holonique d'un système PRT

Pour illustrer la modélisation holonique dans le cas d'un système PRT, nous présentons un modèle du système PRT Urbanaut dont on trouvera une description détaillée sur le site (http://www.urbanaut.com/index.htm).

1.3.1 Le système PRT Urbanaut

- **Le réseau**

Le système Urbanaut est exploité en Corée du Sud. Le réseau, représenté figure 5-2, comporte 10 stations et 2 boucles couvrant :

- boucle 1 : le Parc de loisirs Wolmido,

- boucle 2 : un quartier de la ville d'Incheon.

Ces deux boucles sont reliées par un tronçon central bidirectionnel (tronçon T, situé entre les stations 101 et 102).

Le système de contrôle est centralisé. Il comporte trois systèmes principaux (propulsion, opération, supervision).

L'équipement de contrôle est réparti le long des rails de guidage et à la station de contrôle centrale.

- **Véhicules**

Les véhicules sont de type monorail et peuvent être regroupés pour former des trains de capacités allant de 6 personnes assises à 58 personnes assises et 72 debout (figure 5-3). La vitesse maximum est de 100 km/h.

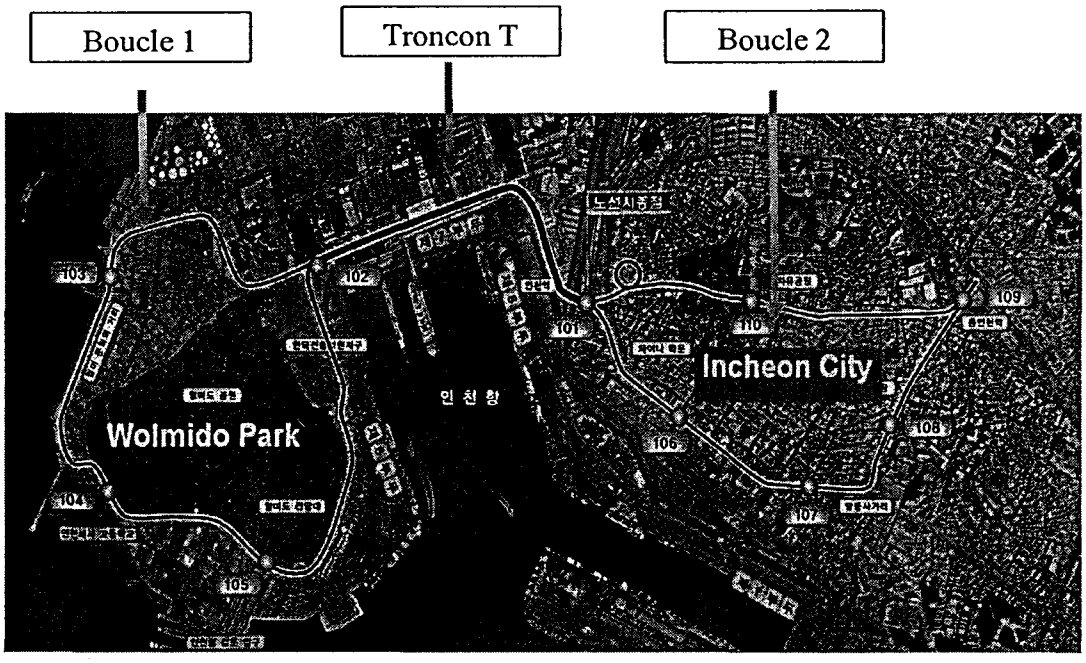


Figure 5.2 : le réseau PERT Urbanaut

Figure 5.3 : les véhicules du système Urbanaut

- **Les stations**

Le système PRT Urbanaut comporte deux types de stations :

- stations d'intermodalité : c'est la station représentée figure 5-4,

- stations intermédiaires : ces stations sont accessibles par des escaliers ou des escalators. Elles peuvent notamment surplomber une voie de circulation (figure 5-5). Certaines stations possèdent des zones de réserve (véhicules vides, maintenance).

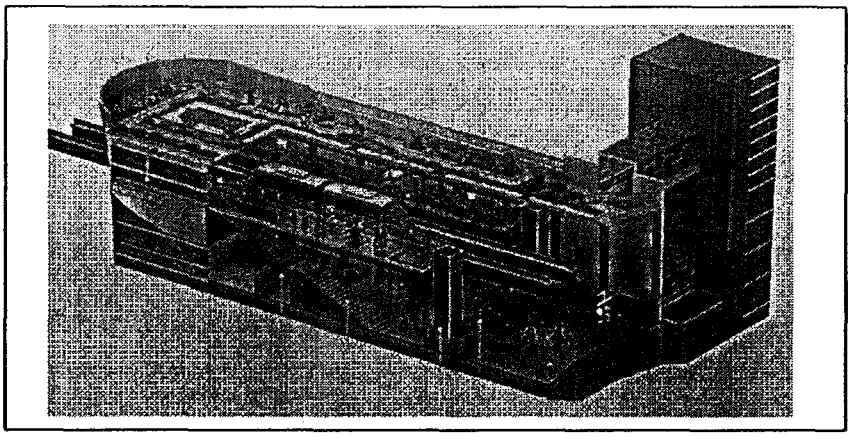


Figure 5.4 : station aérienne (Véhicule porté) : Système PRT – Projet Urbanaut

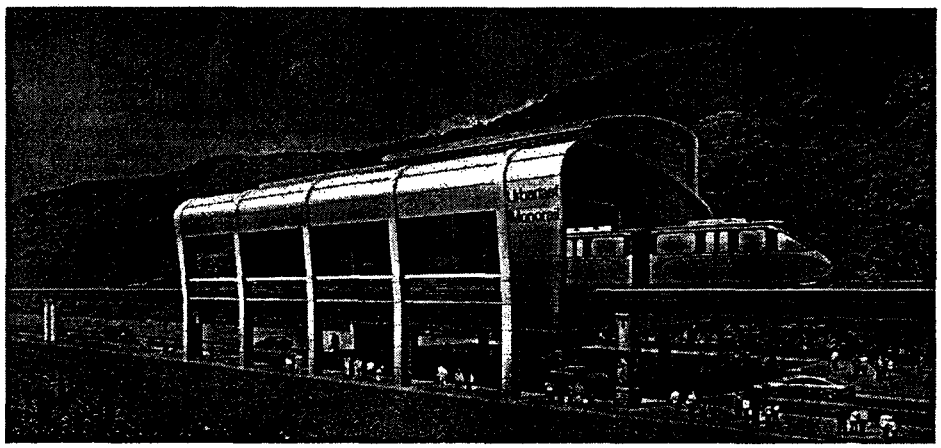


Figure 5.5: station intermédiaire du système Urbanaut

- **L'infrastructure**

L'infrastructure de support des véhicules est essentiellement composée de poteaux (figure 5.6). Les aiguillages présentent des degrés de flexibilité variables : flexible multivoies (figure 5.7 a), fixe (figure 5.7 b).

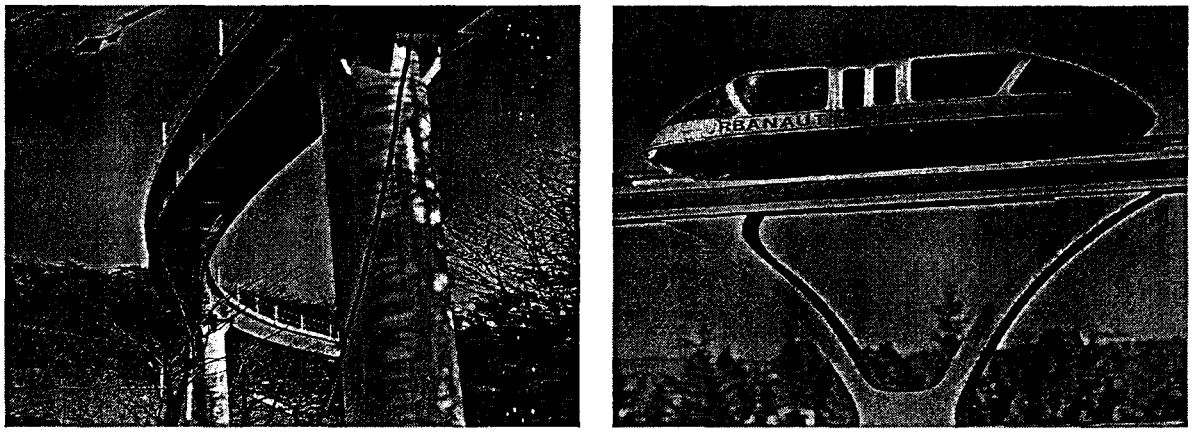


Figure 5.6 : infrastructure du système Urbanaut

(a) (b)

Figure 5.7: aiguillages du système Urbanaut

Le modèle holonique présenté au chapitre 2 permet d'obtenir un modèle holonique du système Urbanaut (figure 5-8).

1.4 Modèle holonique du système Urbanaut.

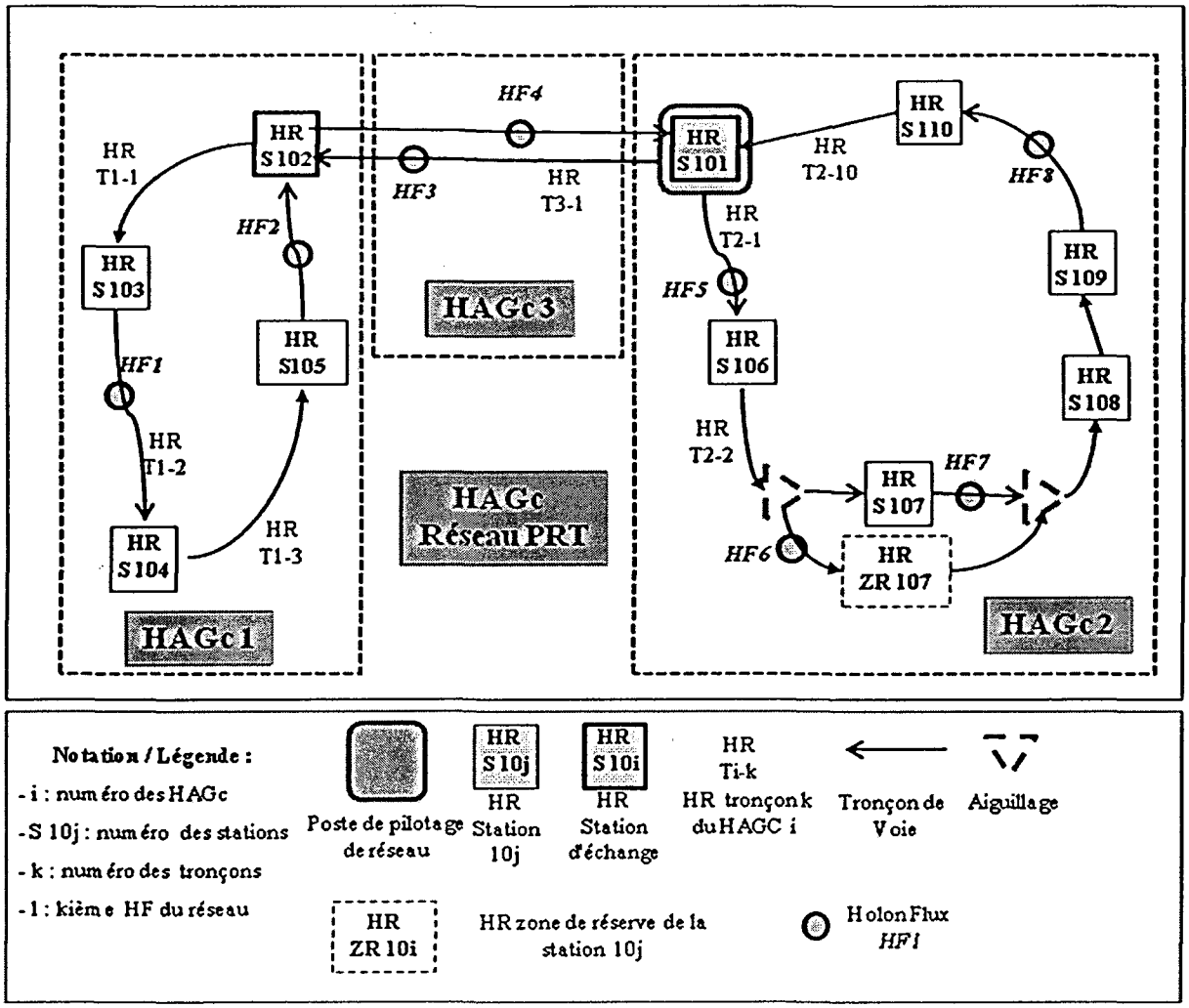


Figure 5.8 : modèle holonique du système Urbanaut

Nous considérerons que le réseau (HAGc réseau PRT) est constitué :

- de HAGc 1 : correspondant au secteur 1

- de HAGc 2 : correspondant au secteur 2

- de HAGc 3 : correspondant au tronçon T

- de chaque véhicule, et ses équipements de traitement non décisionnel (au regard du pilotage) embarqués, qui est modélisé par un holon HF (*HFl)*,

- des stations qui sont représentées par un holon ressource HR non actif. On a supposé le système de pilotage centralisé localisé à la station S 101. Les stations HR S101 et HR S102 sont des pôles d'échange,

- de la station S 107 : elle correspond à une station possédant une zone de réserve. La station 107 est donc modélisé par :

  - un HR S 107 associé à la station S 107,

  - un HR ZR 107, modélisant la zone de réserve de la station (sur le schéma, le *HF 6* est engagé vers cette zone),

- des tronçons de voie sont modélisés par des HR (HR Ti-k). Pour des raisons de lisibilité, tous les tronçons de voie HR Ti-k constituant l'infrastructure du réseau ne sont pas représentés sur la figure 5.8.

2 Modélisation du comportement d'un système PRT

Nous présentons tout d'abord une architecture générique pour le pilotage d'une flotte de PRT. Un mécanisme d'optimisation environnemental et un mécanisme d'optimisation sociétal sont ensuite décrits dans le contexte du pilotage d'une flotte de PRT. Un PRT correspond à un holon HF.

2.1 Architecture de pilotage d'une flotte de PRT

Le pilotage d'une flotte de PRT concerne dans le contexte de la thèse le processus de routage. Il doit être en mesure de répondre aux demandes dynamiques de déplacements des usagers/passagers et faire face aux perturbations sur les voies de circulation. L'architecture de pilotage se compose, conformément au chapitre 2, de deux niveaux (figure 5.9) :

Le niveau inférieur comprend les PRT, les voies de circulation et l'infrastructure associée. Les PRT interagissent directement ou indirectement afin de transporter à destination des passagers. Chaque PRT est vu comme une entité décisionnelle autonome se déplaçant, sur un graphe fortement connexe, d'un nœud source à un nœud destination. Le PRT est plongé dans un espace informationnel où il peut être influencé par un Mécanisme d'Optimisation (MO) sociétal ou environnemental. Sur ce niveau inférieur, seuls les problèmes de routage sont traités, et le système s'accommode des perturbations affectant la fluidité du trafic. Un PRT est capable de trouver son meilleur chemin en temps-réel et

s'adapter aux conditions variables du trafic, quitte à changer son chemin face à des ralentissements, blocage...

Le niveau supérieur gère globalement la flotte de PRT et s'occupe principalement des problèmes d'affectation de PRT aux usagers et de répartition de charge (ex : gestion des demandes, répartition des PRT libres...)

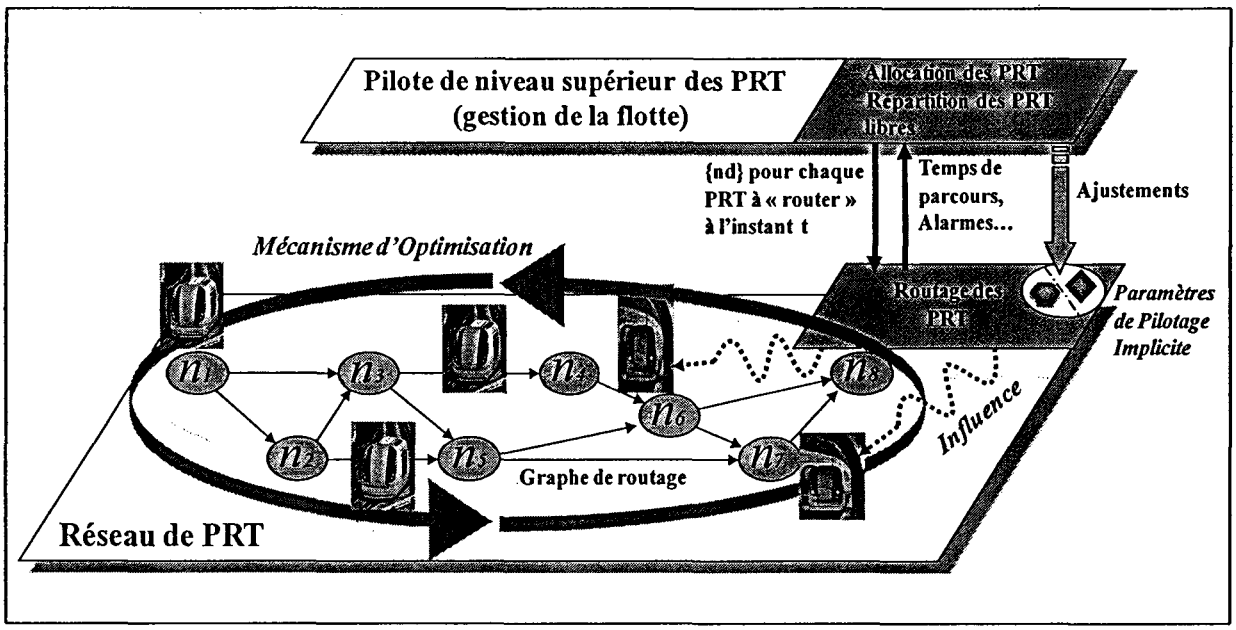


Figure 5.9. Architecture de pilotage d'une flotte de PRT

Ce niveau supérieur envoie (i.e. pilotage explicite) à chaque PRT libre le nœud destination qu'il doit atteindre. En retour ce niveau reçoit les temps de déplacements de nœud en nœud, mesurés par le PRT, qui sont exploités par exemple par un processus d'allocation pour mettre à jour sa connaissance du réseau de PRT.

Ce niveau supérieur n'assigne pas explicitement de chemins aux PRT, ceci est de la responsabilité de chaque PRT sur le niveau inférieur. Cependant ce niveau supérieur peut influencer le comportement de chaque PRT en appliquant un pilotage implicite.

Cette approche favorise une optimisation à long terme au niveau supérieur, éventuellement centralisé, en ayant recours par exemple aux outils usuels de la recherche opérationnelle (Pasquier et al, 2001; Garaix et al, 2010). Ce niveau supérieur est couplé à un niveau inférieur hétérarchique. Ce dernier niveau réagit aux événements imprévisibles grâce à sa forte réactivité, et en traitant les perturbations au plus proche de leurs sources.

Les parties suivantes illustrent le pilotage implicite d'une flotte de PRT. Dans un premier temps un MO environnemental fondé sur la stigmergie est présenté. Dans un deuxième temps, un MO sociétal basé sur une diffusion de messages est introduit.

2.2 Pilotage implicite via un MO environnemental

Le Mécanisme d'Optimisation environnemental présenté est basé sur la stigmergie. Ce concept est alors, dans un premier temps, décrit.

2.2.1 Concept de stigmergie

Le terme stigmergie a été introduit pour décrire le mécanisme par lequel des termites se coordonnent pour édifier une termitière (Grassé, 1959). Dans une telle activité, plusieurs individus participent à la tâche collective d'édification, et les stimuli produits par la structure de la termitière émergente influencent les actions des individus. Un mécanisme semblable coordonne également les fourmis lors de leur quête de nourriture (Theraulaz and Bonabeau, 1999). Les fourmis cherchent de la nourriture et la transporte au retour sur le même chemin que pour l'aller, ceci tout en déposant sur le sol des phéromones. D'autres fourmis, détectant les phéromones, sont alors recrutées et suivent directement un chemin conduisant à de la nourriture. Plus un chemin conduisant à de la nourriture est utilisé et plus les fourmis qui l'empruntent renforcent son intérêt en déposant plus de phéromones. Les phéromones s'évaporent avec le temps ; seuls les chemins les plus attractifs accumulent des phéromones et sont prépondérant sur les autres chemins qui ont leur marquage, via phéromones, qui finit par disparaître. Au cours du temps, dû au renforcement, le chemin le plus court reste (Dorigo et Stützle, 2004).

Dans la nature, trois opérations de base sont associées au processus de la stigmergie : la fusion de l'information (apportée par les phéromones), le retrait d'information et la diffusion locale de l'information. Dans la première opération, les phéromones sont naturellement agrégées par superposition, ce qui correspond à une fusion d'information. Dans la deuxième, l'évaporation des phéromones au cours du temps à pour effet d'éliminer l'information obsolète ou inconsistante. Pour la dernière opération, l'information est accessible que dans le voisinage immédiat des phéromones, ce qui limite l'effet attracteur de ces dernières à la localité.

Dans ces opérations, le champ de phéromones possède trois caractéristiques principales :

- *Indépendance* : l'émetteur d'un message de phéromone ne connaît pas l'identité du receveur potentiel et donc n'attend pas d'accusé de réception. Ceci rend les phéromones particulièrement efficaces lors d'une communication au sein d'une population constituée de nombreux individus.
- *Gestion locale* : puisque la diffusion des phéromones s'atténue rapidement avec la distance, la gestion (naturelle dans le cas des phéromones) est locale, distribuée et ne nécessite pas de gestion centrale.
- *Dynamique :* les cycles de renforcement et d'évaporation agissent respectivement pour intégrer de nouvelles informations et pour supprimer les informations obsolètes.

2.2.2 MO environnemental basé sur la stigmergie

Plusieurs algorithmes performants de routage ont été inspirés par le comportement de fourmis (Dorigo and Stützle, 2004). Dans un contexte dynamique, le plus approprié est AntNet (Di Caro and Dorigo, 1998), qui est basé sur une algorithmique "Ant Colony

Optimization" (ACO) et appliqué à l'apprentissage adaptatif de tables de routage dans des réseaux de communications. L'approche présentée est directement inspirée d'AntNet.

La figure 5.10 présente un réseau de transport où des PRT se déplacent sur un graphe fortement connexe, nœud par nœud, du nœud source n1 au nœud destination n8. Au nœud courant n3, le PRT 1 choisit le nœud suivant en lisant une table de phéromones virtuelles de routage qui contient le meilleur chemin à suivre vers la destination. Par analogie aux systèmes biologiques, la table de routage fournit le taux de phéromones virtuelles sur les différents arcs du graphe. Lors de son déplacement, chaque véhicule mémorise son expérience (i.e. la succession des nœuds traversés et les temps de parcours) dans une mémoire embarquée. Comme les fourmis réelles, qui déposent des phéromones lors du retour au nid, à chaque fois qu'un véhicule atteint son nœud destination, son expérience de parcours est exploitée pour mettre à jour les coefficients des phéromones virtuelles sur chaque nœud traversé. La mise à jour est détaillée ci-après.

Chaque nœud n_k contient :

- une table de routage P^{nk}, où les colonnes contiennent toutes les destinations n_d possibles et les lignes tous les voisins Vn_k possibles à partir du nœud nk ;

- une table μ_{dk}, contenant le temps moyen pour aller de n_k à n_d pour toutes les destinations possibles.

Soit le coefficient p_{dn} d'une table de routage P^{nk}, sur le nœud n_k, qui représente l'intérêt de passer par le nœud immédiatement voisin n_n en permettant le temps de transport le plus court pour aller du nœud n_k vers le nœud destination n_d via ce nœud n_n, qui appartient à l'ensemble des nœuds voisins.

La notation suivante :

$$p_{dn} \in [0;1],\ p_{dj} > p_{dp} \qquad (1)$$

implique que n_j permet au PRT d'atteindre n_d plus rapidement que n_p, où n_j et n_p sont voisins de n_k :

$$Vn_k \Leftrightarrow \{\text{voisins immédiats de } n_k\}; \qquad (2)$$

Après normalisation, il est obtenu que :

$$\sum_i p_{di} = 1; i \in Vn_k \qquad (3)$$

T_{dk} étant le temps mesuré pour aller de n_k à n_d, T_{dk} et la moyenne μ_{dk} des T_{dk} précédents sont comparés. Une valeur r de renforcement est utilisée pour augmenter le coefficient p_{dc} (représentant l'intérêt de choisir le nœud voisin n_c, à partir du nœud n_k, pour se rendre de proche en proche au nœud destination n_d) et pour diminuer les autres coefficients p_{do} de telle sorte que :

$$p_{dc} \Leftarrow p_{dc} + r^*(1 - p_{dc}) \qquad (4)$$

$$p_{do} \Leftarrow p_{do} - r^* p_{do},\ n_o \in Vn_k \text{ avec } n_o \neq n_c \qquad (5)$$

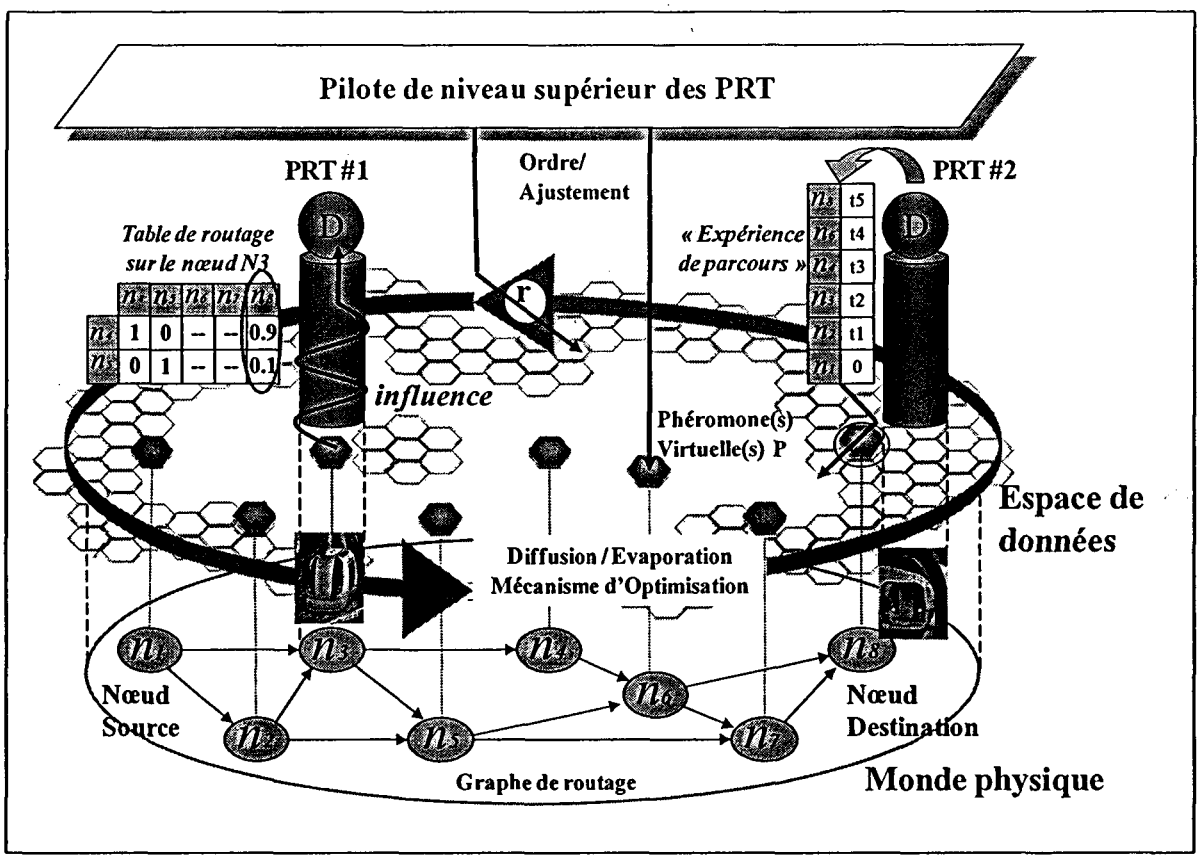


Figure 5.10 : routage stigmergique

Le MO environnemental utilise l'expérience T_{dk} et la matrice μ_{dk} pour mettre à jour toute table de routage P^{nk}.

Par analogie aux systèmes biologiques, la valeur de renforcement r peut être interprétée comme le taux de dépôt/évaporation des phéromones (Dorigo et Stützle, 2004).

2.2.3 Fonctionnement du pilotage implicite

Au travers du pilotage implicite, le niveau haut peut influencer un PRT de deux façons :

En modifiant directement et localement un coefficient p_{dn} dans la table de routage P^{nk} d'un nœud n_k particulier, influençant ainsi, par exemple, les choix de routage d'un PRT suite à une augmentation artificielle et locale du poids d'un nœud voisin n_n particulier ; ou

En modifiant le MO environnemental par ajustement de la variable de renforcement r, ce qui a une incidence globale puisque r est utilisé par chaque nœud pour mettre à jour les coefficients p_{dn} de sa table de routage P^{nk}.

2.2.4 Etude critique de la stigmergie pour le pilotage d'une flotte de PRT

L'approche à base de stigmergie, inspirée de AntNet, est naturellement distribuée, ceci signifie qu'il n'y a pas de mémorisation centrale ni de traitement central. Cette propriété a orienté notre choix pour une modélisation multi-agents et l'environnement de simulation multi-agents Netlogo. Avec Netlogo (Netlogo, 2010), chaque entité modélisée peut être

décrite comme un agent indépendant en interaction avec son environnement. Les règles de comportement définies pour les agents permettent de décrire l'interaction agent-environnement, qui est très importante lors de la simulation du processus "stigmergique". Ce choix est également justifié par le fait que les simulateurs (présentés sur le site Web dédié "PRT Simulator (2009)") sont très spécifiques et ne peuvent pas supporter le pilotage implicite.

Cette approche de pilotage, à base de stigmergie, fonctionne et permet d'obtenir de bons résultats expérimentaux en simulation. Le lecteur intéressé peut consulter Sallez et al. (2009) pour plus de détails.

Les résultats obtenus sont cohérents avec ceux trouvés par Di Caro en utilisant l'algorithme AntNet. Di Caro et Dorigo (1998) ont montré qu'AntNet surpasse, en performance, la plupart des algorithmes de routage (par exemple, Bellman-Ford, OSPF) face à des perturbations fortes. Dans notre contexte de gestion de flotte, les PRT sont capables de faire face aux perturbations en cherchant de nouveaux chemins, qui permettent de les contourner, tout en atteignant la destination souhaitée. Ces capacités indiquent que notre approche offre un bon niveau d'adaptabilité et de robustesse face aux fluctuations de l'environnement.

Certains paramètres (par exemple, le taux d'évaporation des phéromones, la valeur de renforcement, le pourcentage des entités utilisées pour la tâche d'exploration) ont été introduits pour affiner le modèle et pour permettre un pilotage implicite. Comme c'est le cas traditionnellement dans les applications de l'ACO, et plus généralement dans les applications de l'apprentissage par renforcement, le choix des paramètres influe directement sur deux caractéristiques : l'optimalité et l'évolutivité. Plus de détails sur ces deux caractéristiques (par exemple, le respect du critère d'optimalité de Bellman, l'ajustement des paramètres de réglage) peuvent être trouvés dans la bibliographie (Di Caro et Dorigo, 1998; Dorigo et Stützle, 2004).

La principale conclusion de l'étude d'AntNet est que le pilotage implicite est très intéressant pour les applications qui impliquent une grande flotte de véhicules qui ne peuvent pas être pilotés directement ou individuellement. Néanmoins, il existe certaines limites :

Aucune preuve de l'optimalité du routage ne peut être donnée pour ce type d'approche (Dorigo et Stützle, 2004).

La stigmergie nécessite une phase d'exploration, pour découvrir le réseau et ajuster des paramètres, qui nuit à l'efficacité si les PRT réels sont utilisés.

Ces limites nous ont conduites à considérer une autre approche qui peut fournir des chemins optimaux et qui plus est pour éventuellement des flottes de véhicules à l'effectif restreint. Cette approche, qui illustre également la notion de pilotage implicite en utilisant un MO sociétal est décrite ci-après.

2.3 Pilotage implicite via un MO sociétal

Cette partie se focalise sur une possible réalisation d'un pilotage implicite, via un MO sociétal, reposant sur une version embarquée de l'algorithme de Dijkstra.

2.3.1 Algorithme de Dijkstra embarqué pour un MO sociétal

L'algorithme, très connu, de Dijkstra (1959) est usuellement utilisé pour résoudre le problème de recherche du plus court chemin dans un graphe. La littérature sur les problèmes de routage est abondante dans le domaine de la Recherche Opérationnelle. (Nous renvoyons le lecteur intéressé vers les articles suivants : Cormen et al, 2001; Fu et al, 2006; Kumari et Geethanjali, 2010) Dans cette approche, de pilotage implicite, proposée dans la thèse, chaque PRT comprend une carte de routage contenant le temps de parcours, réactualisé en temps-réel, sur les différents arcs du graphe. Cette seconde approche induit également d'inclure une version embarquée de l'algorithme de Dijkstra, qui est déclenché chaque fois qu'un PRT doit choisir le prochain nœud voisin à utiliser, à partir de la carte de routage, pour atteindre finalement le nœud de destination.

Le choix des voisins de proche en proche permet à chaque PRT de progresser sur un chemin optimal (au sens le plus court) et de s'adapter aux perturbations (par exemple, ralentissements) lors de la progression dans le réseau. Sur la base de l'algorithme de Dijkstra, chaque choix est calculé en utilisant les valuations des arcs qui sont mises à jour en temps réel et basées sur les temps mesurés par les PRT lors de leurs déplacements. Ces valuations sont les temps de parcours sur les arcs, d'une extrémité à l'autre, connus suite aux mesures des durées de déplacements réalisées par chaque PRT. La diffusion par chaque PRT de ses temps mesurés est à la base du MO sociétal proposé dans notre approche. En fonction de l'ajustement de la diffusion, par exemple son étendue, les cartes de routage peuvent différer d'un PRT à l'autre.

La figure 5-11 montre un réseau de transport où des PRT se déplacent sur un graphe fortement connexe, de proche en proche, du nœud source n_1 au nœud destination n_8. Chaque PRT k contient une matrice T^k (i.e., carte de routage), où $t_{i,j}$ à l'intersection de la ligne i et de la colonne j est le temps, mis à jour, mesuré par le dernier PRT qui a emprunté, lors de son déplacement, l'arc $[n_i, n_j]$. Sur le nœud divergent n_3, PRT 1 choisit le prochain nœud voisin à utiliser en calculant (i.e. l'algorithme de Dijkstra embarqué sur le PRT) le chemin optimal vers le nœud destination, en tenant compte des valuations dynamiques dans sa carte de routage. Le PRT 1 sur le nœud n_3 exécute l'algorithme de Dijkstra pour choisir entre n_4 ou n_5, en utilisant sa table de routage. Seul le nœud "meilleur voisin" sur le chemin optimal est retenu. Ainsi, en réitérant le processus de nœud en nœud, le PRT s'adapte aux perturbations et atteint son nœud destination. A chaque fois qu'un PRT se déplace sur un arc $[n_i, n_j]$, il mesure le temps qui a été nécessaire pour progresser du nœud n_i au nœud n_j. Une fois arrivé sur le nœud n_j, le PRT diffuse ce temps mesuré vers les autres PRT.

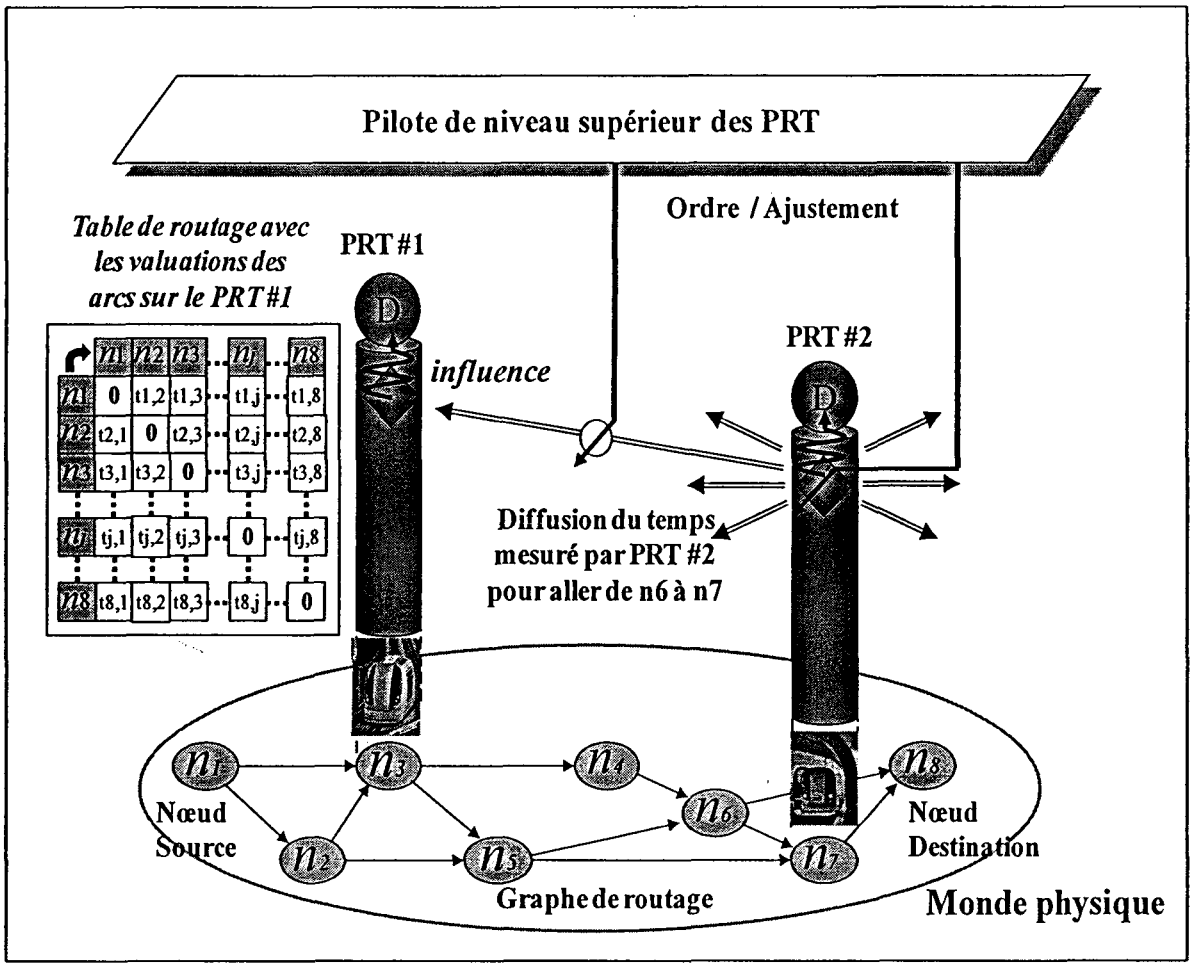


Figure 5.11 : routage dynamique basé sur notre approche " Dijkstra embarqué".

2.3.2 Fonctionnement du pilotage implicite

Via le pilotage implicite, le niveau supérieur peut influencer un PRT de deux façons :

- La première consiste à ajuster la vue partielle du PRT sur la fluidité du trafic en modifiant la valuation $t_{i,j}$ de chaque arc dans sa table de routage T^k. Cette modification peut être vue comme une influence externe qui va modifier indirectement le comportement du PRT suite à l'enclenchement de son algorithme de Dijkstra embarqué.
- La seconde façon consiste à altérer le MO sociétal en modifiant les lois de diffusion. Par exemple, les temps mesurés peuvent être diffusés à tous les PRT du réseau ou seulement aux PRT dans une localité assez proche du PRT "émetteur" de la diffusion. De cette manière, la performance globale du collectif, pour une flotte importante de véhicule, peut être indirectement affectée.

Contrairement à l'approche basée sur la stigmergie, cette dernière approche, basée sur l'algorithme de Dijkstra embarqué, ne nécessite pas que les PRT explorent l'environnement, i.e. le graphe. Pour cette raison, puisque nous souhaitons, conjointement

au modèle holonique, également valider sa partie comportementale, i.e. l' "open-control", sur un système réel, nous avons retenu la dernière approche.

3 Les flux au sein du système PRT.

La figure 5-12 donne quelques exemples de flux au sein d'un système de transport de type PRT.

		Référence flux	**Type de flux**	**Type de flux dans un réseau PRT**	Exemple de flux
Flux internes	Flux informations	1	Flux entre HA d'un même HAG (ou HAGc)	Communication entre PRT d'un même secteur	coordination entre deux PRT pour la gestion de la priorité à un aiguillage
		2	Flux entre un HA et HNA d'un même HAG (ou HAGc)	Communication entre véhicule PRT et une station	Transmission de l'information de position du PRT à la station
		3	Flux entre HNA d'un même HAG (ou HAGc)	synchronisation entre aiguillages d'une même voie unidirectionnelle	Vérification sécuritaire du positionnement des aiguillages
	Flux physique	4	Flux entre HNA d'un même HAG (ou HAGc)	Flux entre deux stations d'un même secteur d'un réseau	PRT circulant entre deux stations d'un même secteur
Flux externes	Flux informations	5	Flux entre HA appartenant à des HAG différents	Coopération entre 2 PPC de deux secteurs d'un réseau	Répartition de véhicules vides pour équilibrer leur allocation entre secteurs
		6	Flux entre un HA d'un HAG (ou HAGc) et un HNA d'un HAG (ou HAGc) différent	Communication entre PRT et une station ou un aiguillage	Demande à un aiguillage d'établir un chemin particulier
		7	Flux entre HNA d'HAG (ou HAGc) différents	Communication entre deux PRT	Synchronisation entre deux PRT (partie véhicule HNA) pour éviter de rentrer en collision
	Flux physique	8	Flux entre HNA d'HAG (ou HAGc) différents	Flux entre réseaux différents	PRT (partie HNA véhicule) circulant entre des stations appartenant à deux réseaux différents

Figure 5.12 : flux au sein d'un système de transport PRT

4 Conclusion

Le chapitre 5 a permis de montrer la capacité offerte par le modèle proposé au chapitre 2 d'élaborer un modèle holonique d'un système à flux guidé. L'approche retenue a consisté à considérer tout d'abord un système PRT générique, puis à modéliser un système PRT opérationnel.

Par rapport aux objectifs fixés dans le chapitre 2, relatifs aux caractéristiques du modèle, nous disposons d'un modèle semi-formel prenant explicitement en compte les flux au sein d'un SEA, permettant d'en décrire aisément les différents éléments et facilitant la tâche du modélisateur.

Le chapitre suivant traite des aspects de la mise en œuvre d'un tel modèle et présente l'expérimentation réalisée et les résultats obtenus.

Chapitre 6 – Expérimentation d'un système de pilotage holonique d'un système de transport de type PRT

Introduction

Ce chapitre décrit l'expérimentation d'un SEAH réalisée dans le but d'évaluer le modèle de système de pilotage proposé avec un type de holon flux représentatif d'un PRT.

Nous présentons tout d'abord la plateforme qui a servi de site expérimental et en donnons un modèle holonique. Nous précisons ensuite le modèle de pilotage utilisé. Ce dernier concerne essentiellement la fonction de routage, partie comportementale "open-control" du modèle SEAH, des holons flux. Le modèle de pilotage s'appuie sur l'approche présentée dans le paragraphe 3.3 (Pilotage implicite via un MO sociétal) du chapitre 5 et utilise un algorithme de routage dynamique basé sur l'approche "Dijkstra embarqué" au sein de chaque PRT. Les résultats obtenus sont ensuite exposés et analysés.

1 Le site expérimental

La plateforme expérimentale est constituée de la cellule flexible du pôle AIP-Priméca Nord Pas-de-Calais de Valenciennes (figure 6.1). La raison de ce choix est qu'une analogie très forte existe entre une cellule flexible de production et un système PRT.

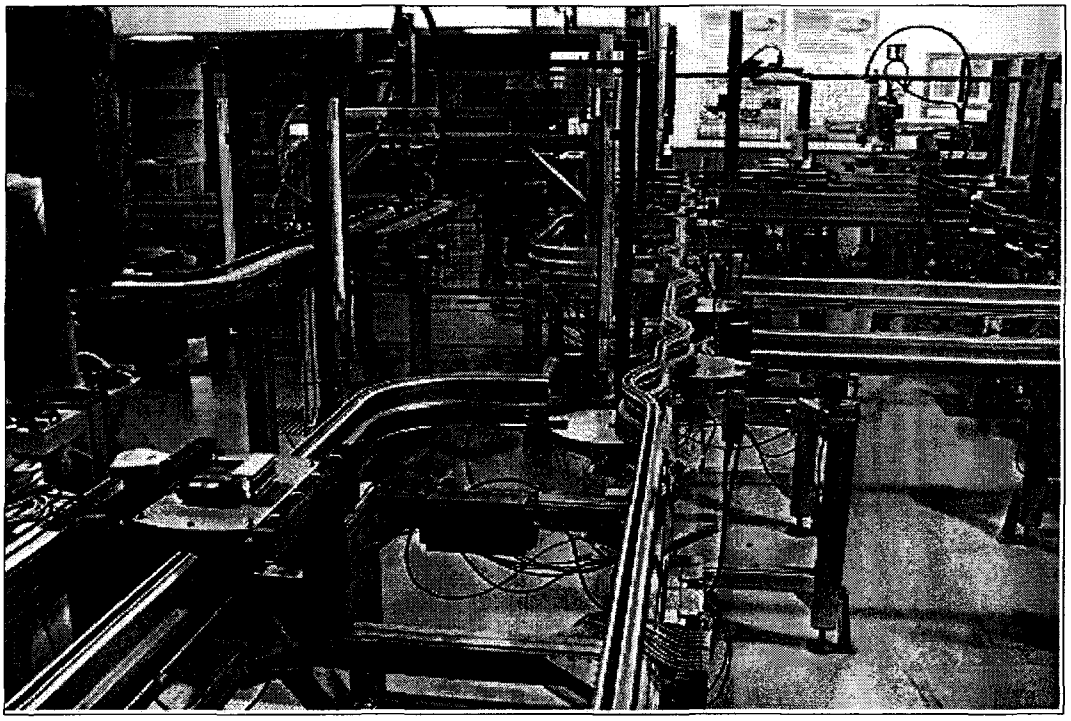


Figure 6.1 : la plateforme expérimentale

A échelle réduite, un système de convoyage peut être vu comme le réseau d'un système PRT. Les équipements de la cellule de l'AIP-Priméca Nord Pas-de-Calais de Valenciennes permettent ainsi la mise en œuvre et l'évaluation du système de pilotage :

- les shuttles (palettes instrumentées de transport) peuvent être considérés comme des PRT,

- les postes de travail sont assimilés aux stations (i.e. point de montée/descente des usagers du système PRT),

- le système de transport monorail de la cellule est l'équivalent des tronçons de transport d'un système PRT.

Nous allons décrire brièvement les différents composants de la cellule.

2 Le système de transport

2.1 Les shuttles

Les shuttles, appelés par la suite "shuttle-PRT", sont autopropulsés. Ils évoluent sur un système de transport monorail Montrac de Montech (http://www.montech.com/).

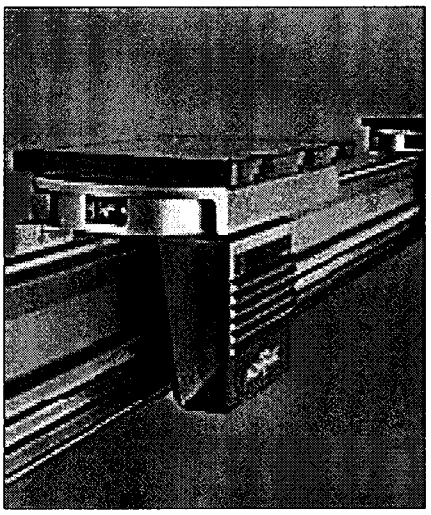


Figure 6.2 : shuttle sur monorail Montrac

2.2 Le réseau de transport

La Figure 6.3 représente le système de transport PRT expérimental, il comporte18 nœuds :

- n_1, n_6, n_9, n_{12}, n_{17}, n_{22} et n_{25} (en blanc) sont des nœuds-stations ; ils constituent des nœuds possibles de départ/source et d'arrivée/destination.

- n_3, n_5, n_8, n_{11}, n_{14}, n_{16}, n_{19}, n_{21}, n_{24}, n_{27} et n_{29} (en gris) sont des nœuds divergents de transfert vers, à un moment donné, le tronçon choisi.

Les autres nœuds (exemple n_{28}) ne sont pas considérés. Ils sont de type convergent et aucune décision de routage n'y est requise.

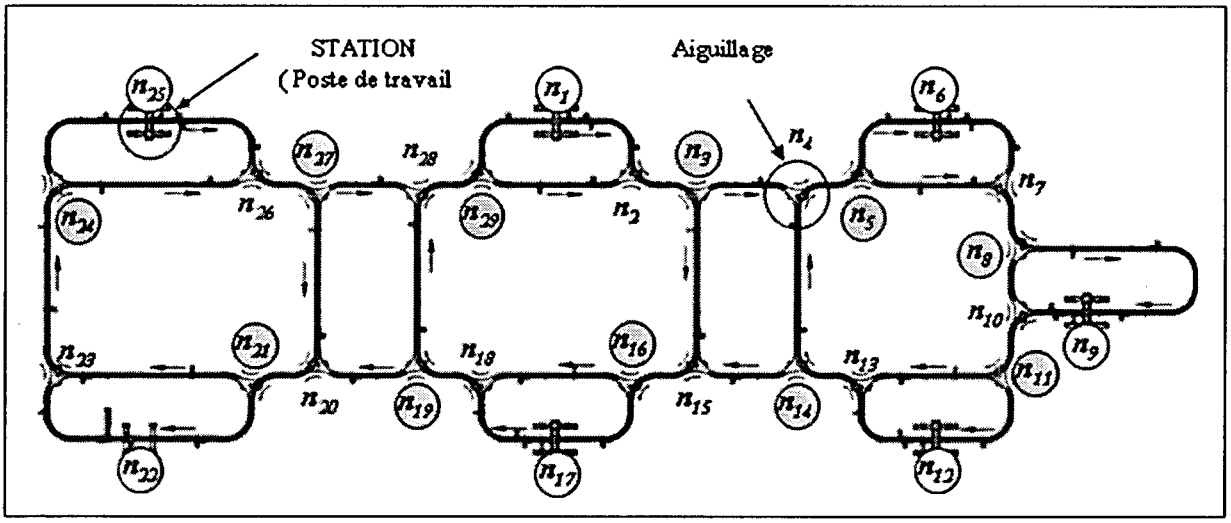


Figure 6.3 : représentation du système de transport PRT expérimental.

Le réseau comporte 22 aiguillages bidirectionnels (figure 6-4-a) et des postes de travail (figure 6-4-b) considérés comme les stations du réseau PRT.

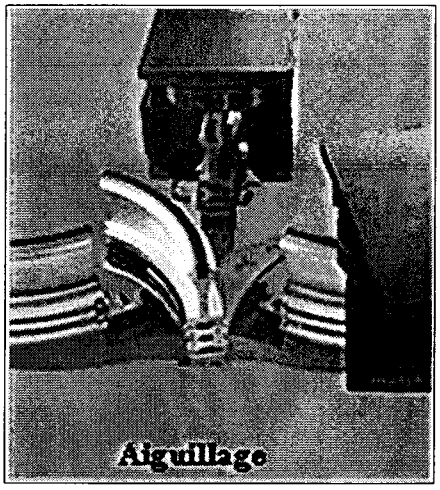


Figure 6.4-a : aiguillage

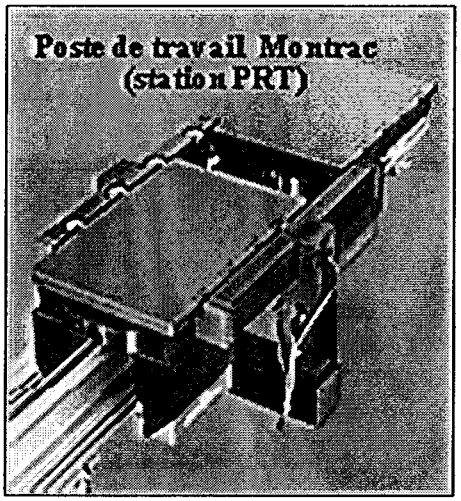


Figure 6.4-b : poste de travail

2.3 Système de communication et de traitement des données

Dans la mise en œuvre qui a été réalisée, pour des raisons technologiques, les shuttles-PRT peuvent communiquer avec leur environnement uniquement quand ils sont arrivés à un nœud. Chaque nœud est utilisé comme un intermédiaire pour diffuser une table de routage à jour composée des différents temps valuant chaque arc/tronçon du réseau. Notons que cette limitation sur les communications pourrait être levée en ayant recours à

un lien hertzien, permettant alors une communication en continue entre les shuttles-PRT et l'environnement.

Le contrôleur de nœud a un rôle clé. Il contrôle le transfert physique, en évitant les collisions, et il met à jour sa table de routage avec les temps de progression sur chaque arc traversé, mesurés par les shuttles-PRT transmis au contrôleur de nœud au moment où il est positionné sur le nœud.

La figure 6.5 illustre les interactions entre un Shuttle-PRT et un contrôleur de nœud. Lorsque le shuttle-PRT arrive sur un nœud divergent de routage, il dialogue avec le contrôleur de nœud via une liaison infrarouge IrDA.

Une succession d'opérations débute alors :

1. le shuttle-PRT transmet sa mesure de temps, concernant le dernier arc/tronçon emprunté, au contrôleur de nœud.
2. le contrôleur de nœud diffuse (via Ethernet) ce temps mesuré aux autres contrôleurs de nœuds.
3. chaque contrôleur de nœud met à jour sa propre table de routage.
4. le shuttle-PRT accède aux valeurs souhaitées dans la table de routage du contrôleur de nœud courant.
5. le shuttle-PRT met à jour sa table de routage T^k, choisit le meilleur nœud voisin et demande au contrôleur de nœud de réaliser le transfert sur le tronçon menant au nœud voisin sélectionné.
6. le contrôleur de nœud positionne l'aiguillage et le shuttle-PRT poursuit sa route.

La figure 6.6 donne des détails sur les équipements utilisés sur la cellule flexible AIP-Priméca Valenciennes. Les contrôleurs de nœuds sont des calculateurs embarqués Wago (Wago, 2011).

Notre approche de pilotage implicite sociétal est hétérarchique. Ce qui laisse toute latitude pour la mise en œuvre entre deux solutions extrêmes :

Solution 1: un système de traitement pour tous les nœuds, ou

Solution 2: un système de traitement pour chaque nœud.

Nous avons choisi une solution intermédiaire, proche de la solution 2, dans laquelle plusieurs nœuds sont parfois associés à un seul système de traitement.

Le système de traitement retenu est un Wago 750-841, qui embarque les programmes, tâches et données nécessaires.

Par exemple, le Wago 11 supporte seulement le nœud 12 (nœud station), alors que le Wago 12 supporte à la fois le nœud convergent 13 et le nœud divergent 14, sur lequel une décision de routage doit être prise (Figure 6.7).

La figure 6.7 montre une vue de la disposition réelle des nœuds 12, 13 et 14 contrôlés par le Wagos 11 et 12.

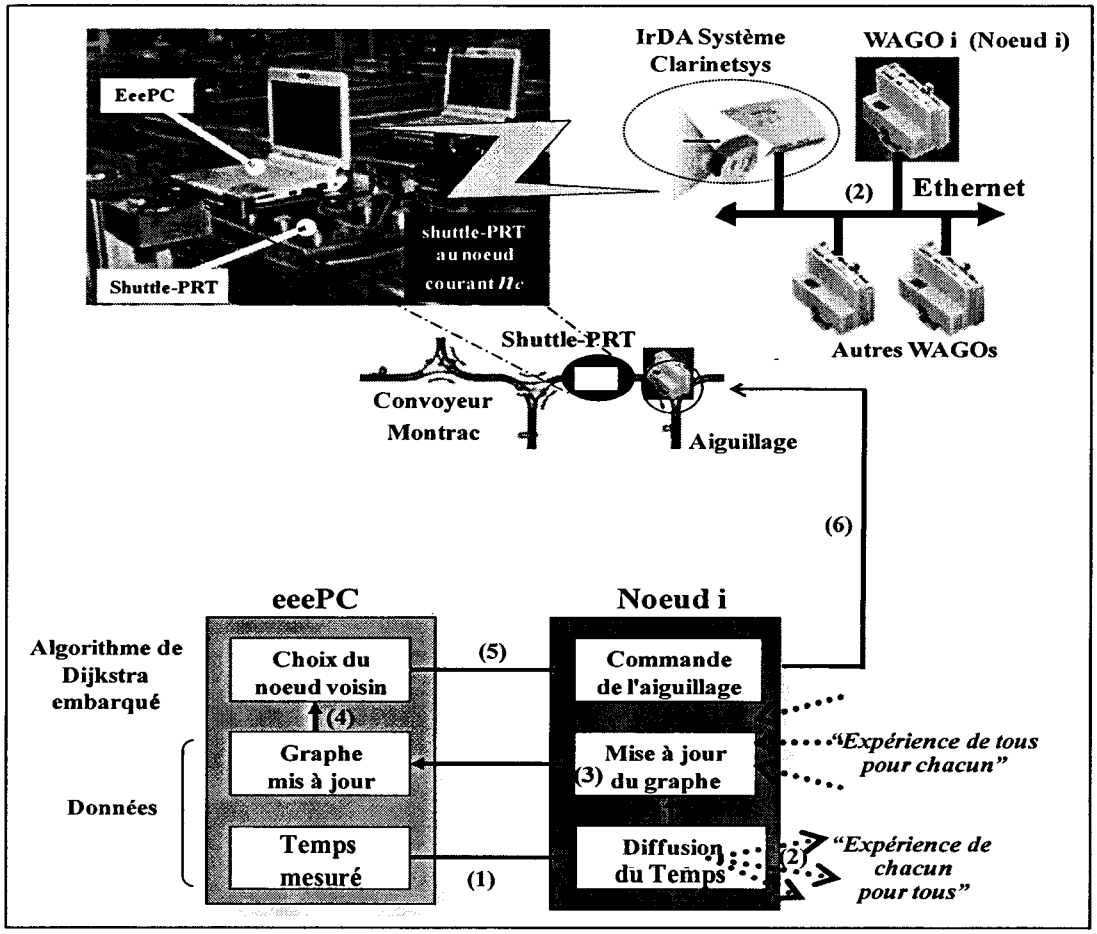


Figure 6.5 : interactions entre un shuttle-PRT- et un contrôleur de nœud

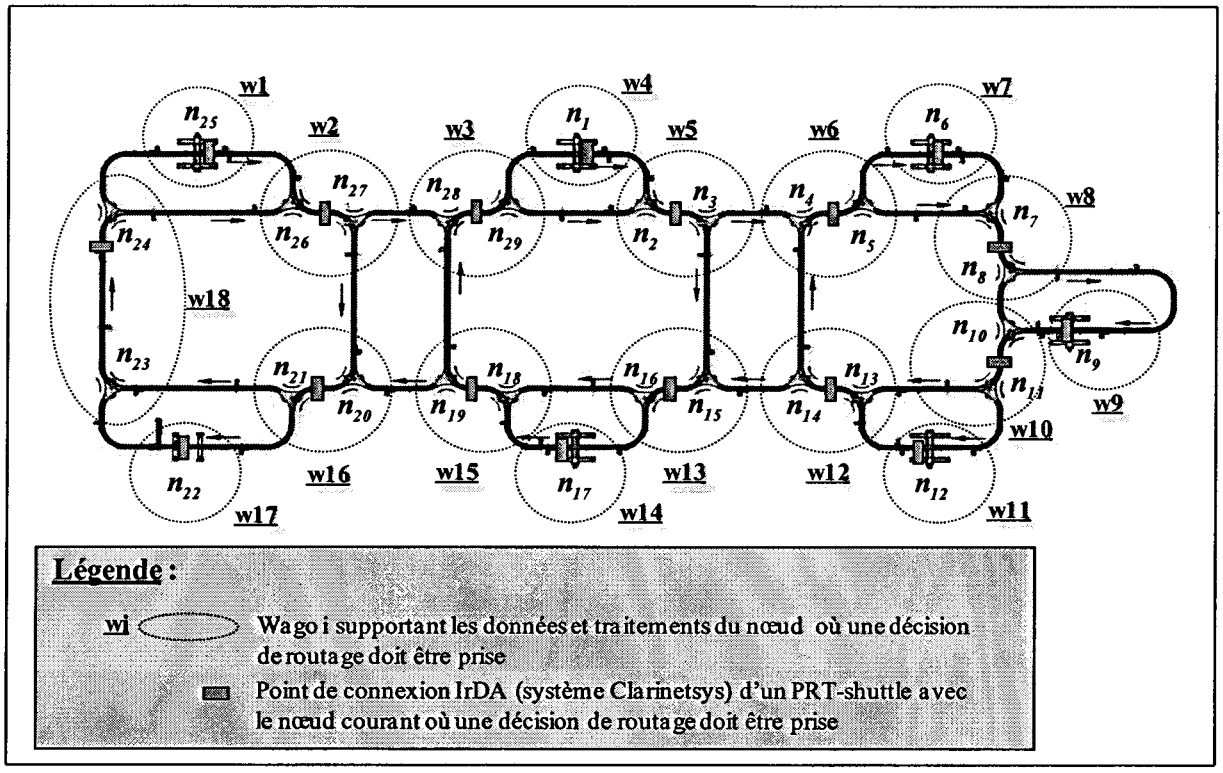


Figure 6.6 : équipements utilisés sur la cellule flexible AIP-Priméca / Valenciennes

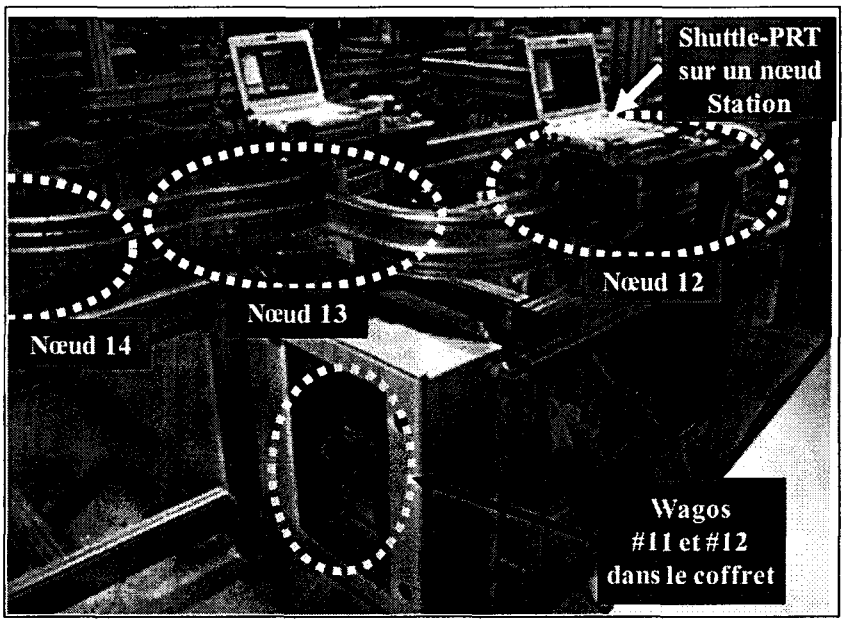


Figure 6.7: Vue de la disposition réelle des nœuds 12, 13 et 14.

3 Modélisation holonique du système PRT expérimental

La modélisation holonique du système PRT expérimental est réalisée avec le modèle SEAH présenté dans le chapitre 2 de cette thèse. Le modèle holonique favorise une démarche "bottom-up" et sous-tend comme préalable une idée des composants de base qui seront utilisés. Ces derniers ont été décrits précédemment.

Cette modélisation a été un préalable à la mise en œuvre effective du système PRT expérimental (constitué de shuttles-PRT). Le modèle comporte une partie structurelle et une partie comportementale.

3.1 Modèle structurel du système PRT expérimental

3.1.1 Partie statique

Les différents holons de la partie statique du système PRT expérimental sont :

- *le Holon Coordinateur* (HC), qui correspond à la couche de pilotage supérieur du système PRT expérimental. Cette couche a vocation à piloter directement un PRT (pilotage explicite) ou à piloter indirectement un PRT (pilotage implicite). Sur le système PRT expérimental elle n'est pas explicitement supportée.
- *le Holon Connaissance Système* (HCS), qui correspond à la connaissance du système utilisée pour élaborer le graphe du système (exploité lors du routage).
- *les Holons Ressources* (HR), qui correspondent :

o aux tronçons/arcs de convoyage sur le système expérimental et aiguillages associés,

o aux nœuds divergents où une décision de routage doit être prise par les shuttle-PRT,

- o aux nœuds convergents,
- o aux stations où des usagers peuvent monter/descendre.

3.1.2 Partie dynamique

Les différents holons de la partie dynamique du système PRT expérimental composant un Holon Flux (HF) sont :

- *les Holons Objet Receveur des Services* (HORS), qui correspondent aux usagers bénéficiant d'un service de transport par un PRT. Sur le système PRT expérimental les usagers sont virtuels mais les déplacements sont réels.
- *le Holon Services à Obtenir* (HSO), qui correspond à la consigne, ou aux consignes, de déplacement du/des usager(s) que le PRT va satisfaire.
- le *Holon Mobilité* (HM), de chaque PRT (shuttle-PRT) qui correspond à un shuttle qui permet les déplacements effectifs des usagers (virtuels).
- le *Holon Conduite* (HC), qui prend en charge le pilotage du shuttle-PRT et réalise, dans notre étude, son routage au sein du système.

Les holons HSO et HC sont supportés par un eeePC embarqué sur un shuttle, holon HM, pour déplacer réellement des usagers (virtuels), holon HORS. La figure 6-8 illustre ces différents "éléments".

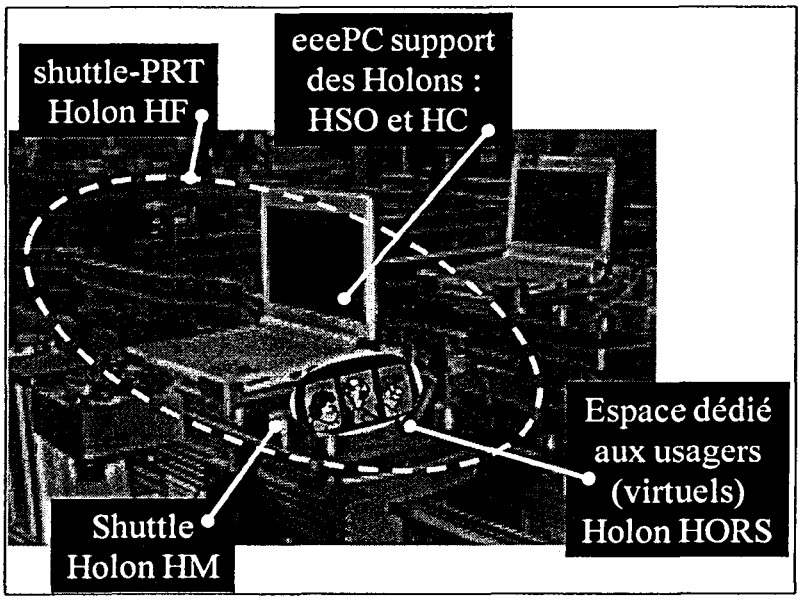


Figure 6.8 : réalisation du holon HF pour un PRT (shuttle-PRT) du système expérimental

Le pilotage du système PRT expérimental, au niveau inférieur, est un pilotage implicite basé sur un Mécanisme d'Optimisation (MO) sociétal conformément au paragraphe 3.3 du chapitre 5 (Pilotage implicite via un MO sociétal).

3.2 Modèle comportemental du système PRT expérimental

Chaque holon HF-PRT choisit lui-même dynamiquement son chemin et s'adapte aux perturbations (ex : ralentissements) pouvant affecter les temps de parcours sur les chemins empruntés lors des déplacements. Ce processus de routage est hétérarchique. Chaque holon HF choisit son chemin à partir d'une carte de routage embarquée qui est un modèle de la topologie du graphe orienté du système de transport. Dans ce modèle, chaque arc du graphe est valué par son temps de parcours. Ce temps est remis à jour sur la base d'informations disponibles au niveau des nœuds à chaque fois que ceci est possible. Il s'ensuit, par exemple, que chaque nœud divergent du graphe, où une décision de routage doit être prise, contient une valuation mise à jour en temps-réel des temps de parcours de chacun des arcs du graphe et donc du système de transport.

Lorsqu'un holon HF arrive au niveau d'un nœud divergent, un échange d'information s'instaure, (figure 6.9) :

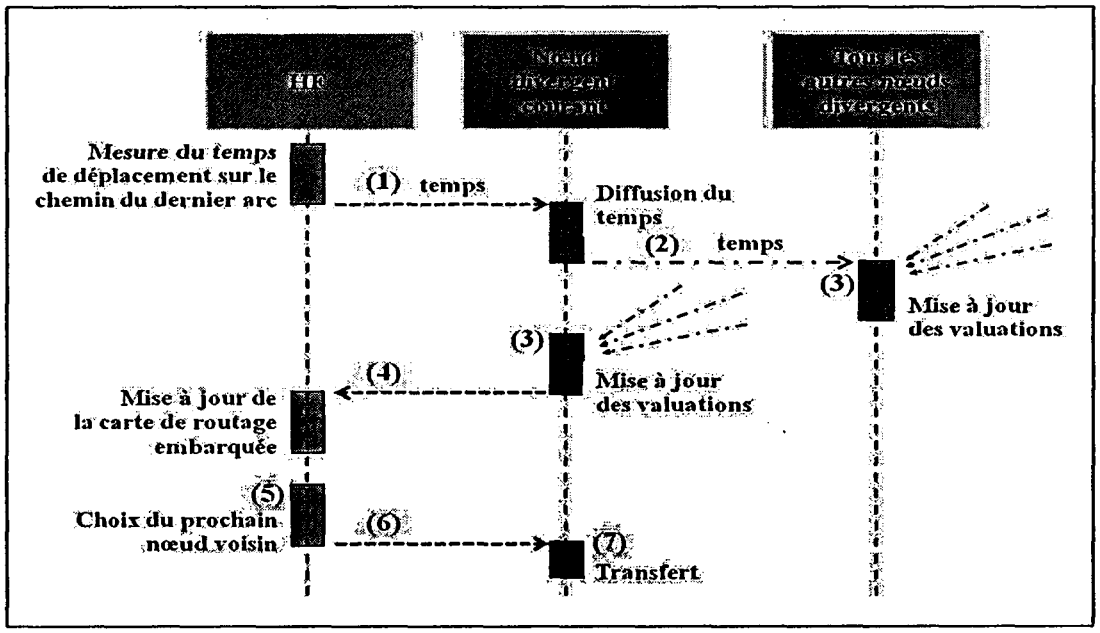


Figure 6.9 : détails du processus de routage.

1. le holon HF transmet le temps mesuré correspondant à son déplacement du nœud précédent au nœud divergent courant,
2. le nœud divergent courant diffuse ce temps à tous les autres nœuds,
3. chaque nœud met à jour la valuation de l'arc concerné par la diffusion du temps,
4. le nœud courant transfert les valuations des arcs du système de transport au holon HF pour qu'il puisse mettre à jour sa carte de routage embarquée,
5. le holon HF choisit le meilleur nœud voisin, immédiatement successeur du nœud courant, en appliquant un algorithme d'optimisation embarqué. L'algorithme utilisé peut être basé sur l'algorithme de Dijkstra (Dijkstra, 1959) dans un contexte embarqué,

6. une fois le choix effectué, le holon HF demande au nœud de réaliser les opérations nécessaires pour être mis sur le chemin qui mène au nœud suivant choisi,
7. le nœud en accord avec la demande de transfert du holon HF réalise donc les opérations nécessaires (par exemple, positionnement d'aiguillage).

4 Scénarios et résultats expérimentaux

Trois scénarios sont présentés, mettant en évidence la capacité d'auto-adaptation aux perturbations du holon HF Shuttle-PRT dans notre approche. Dans ces scénarios, nous supposons que la destination de chaque shuttle-PRT est déterminée et fournie par le niveau de pilotage supérieur.

Les deux premiers scénarios mettent en évidence la capacité de réaction en temps-réel du niveau bas, sur lequel évoluent les shuttles-PRT (holons HF), aux perturbations, tandis que le troisième scénario illustre la possibilité que possède le niveau supérieur à influencer le niveau bas en ayant recours à un pilotage implicite.

4.1 Scénario 1 : évolutivité/"scalabilité"

Le premier scénario montre la capacité de notre approche de pilotage à faciliter l'introduction ou le retrait en temps-réel d'un(de) shuttle(s)-PRT du système. Cette caractéristique montre qu'il est possible de faire évoluer le système facilement.

Dans ce scénario, un shuttle-PRT est introduit sur un tronçon à un endroit quelconque sur l'arc [n_{19}-n_{20}] avec pour nœud destination nd=n_1 (voir la Figure 6-10).

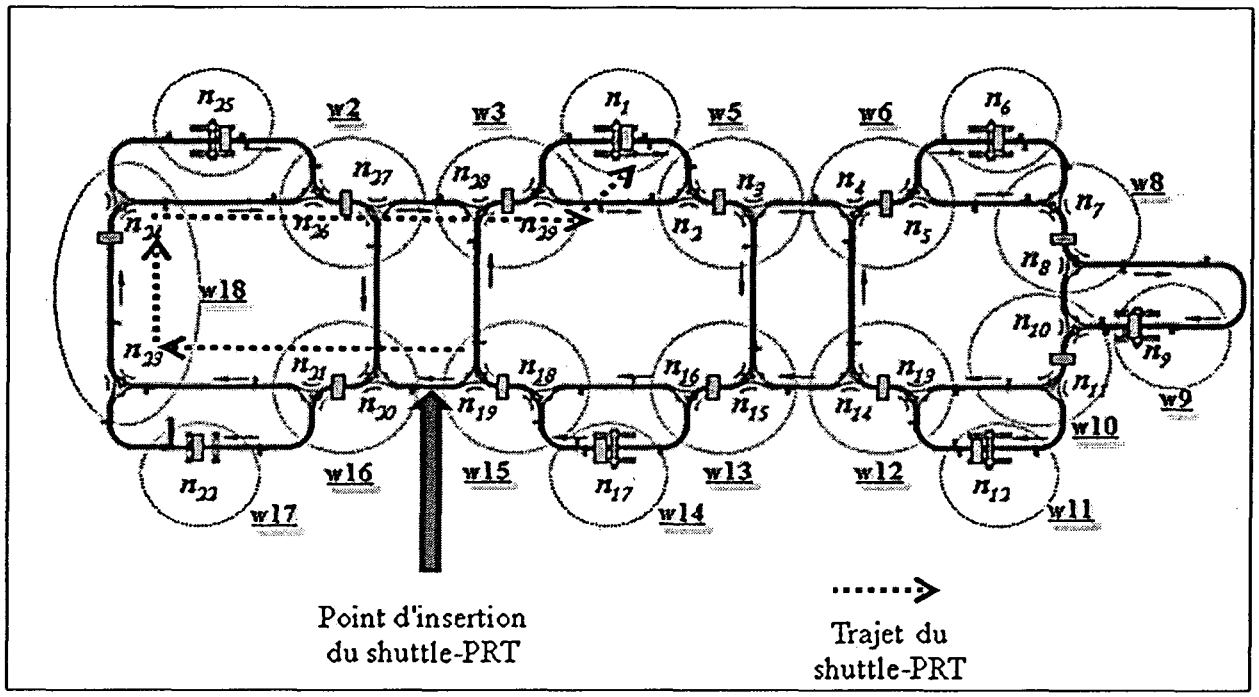


Figure 6.10 : insertion d'un shuttle dans le réseau - Scenario 1

Le Shuttle-PRT progresse sur le rail en suivant les sens de progression jusqu'à atteindre un nœud décisionnel divergent. Dans notre cas, il est allé vers le nœud convergent n_{20} et est arrivé au nœud divergent n_{21}. Le Shuttle-PRT a alors accès aux valuations des arcs via la table de routage localisé sur le nœud n_{21} et met à jour sa propre table de routage. L'algorithme de Dijkstra embarqué sur le PRT est déclenché et le Shuttle-PRT calcule successivement, de nœud en nœud, le chemin [n_{21}-n_{23}-n_{24}-n_{26}-n_{27}-n_{28}-n_{29}-n_1] et atteint finalement sa destination.

A aucun moment le niveau de pilotage supérieur n'est intervenu. Le Shuttle-PRT a géré son routage en toute autonomie.

Il est nécessaire de noter que le Shuttle-PRT a été introduit dans la cellule sans aucune reconfiguration matérielle ou logicielle. Il est également possible de retirer un shuttle-PRT durant sa progression et de le réintroduire plus tard.

4.2 Scénario 2 : gestion des perturbations de routage par le pilotage de niveau inférieur

Ce scénario illustre comment un Shuttle-PRT peut dynamiquement adapter son parcours face à des perturbations de trafic, sans intervention du pilotage de niveau supérieur.

- Scénario de routage :

Nous étudions l'impact d'une perturbation sur un arc qui peut être contourné (voir la Figure 6-11). Les nœuds n25 et n22 sont, respectivement, les nœuds source et destination. Nous supposons que trois usagers, qui ne souhaitent pas voyager ensemble, se déplacent du nœud n25 au nœud n22, chacun utilisant alors un PRT différent.

Le PRT 1 est " lancé" à t = 0 s, le PRT 2 à t = 150 s et le PRT3 à t = 350 s.

- Déplacements des shuttles-PRT :

- le PRT 1 se déplace via le chemin [n_{25}-n_{26}-n_{27}-n_{20}-n_{21}-n_{22}] en 23s.
- à l'instant τ = 200s, une perturbation (i. e. un ralentissement dû à un dysfonctionnement simulé d'un tronçon) a été introduite et a affecté la fluidité du flux sur l'arc [n_{27}-n_{20}]. En conséquence PRT 2 a été ralenti et un temps total de parcours de 125 s en a résulté.
- lorsque PRT 2 est arrivé au nœud divergent n_{21}, le temps mesuré par PRT 2 pour l'arc [n_{27}-n_{20}] a été transmis au contrôleur de nœud (n_{21}) qui l'a diffusé aux autres contrôleurs de nœuds, ainsi chaque contrôleur de nœud a pu mettre à jour sa valuation de l'arc [n_{27}-n_{20}] dans sa table de routage. Cette valeur pourra être utilisée par de futurs PRT.
- lorsque PRT 3 est "lancé" à partir du nœud n_{25}, il met à jour sa propre table de routage avec les valeurs fournies par les contrôleurs des nœuds successivement traversés lors de son déplacement. A partir des nouvelles valeurs de sa table de routage, le troisième PRT utilise l'algorithme de Dijkstra embarqué pour

calculer, nœud après nœud, un chemin alternatif [n_{25}-n_{26}-n_{27}-n_{28}-n_{29}-n_2-n_3-n_{15}-n_{16}-n_{18}-n_{19}-n_{20}-n_{21}-n_{22}]. PRT 3 a parcouru ce chemin alternatif en 67s.

Il en résulte, qu'après la perturbation, seul PRT 2 a été retardé ; PRT 3 a pu prendre en compte la perturbation affectant l'arc [n_{27}-n_{20}] et choisir dynamiquement un chemin alternatif.

La figure 6-11 montre le chemin que chacun des PRT a parcouru, et la figure 6-12 l'évolution des temps de parcours.

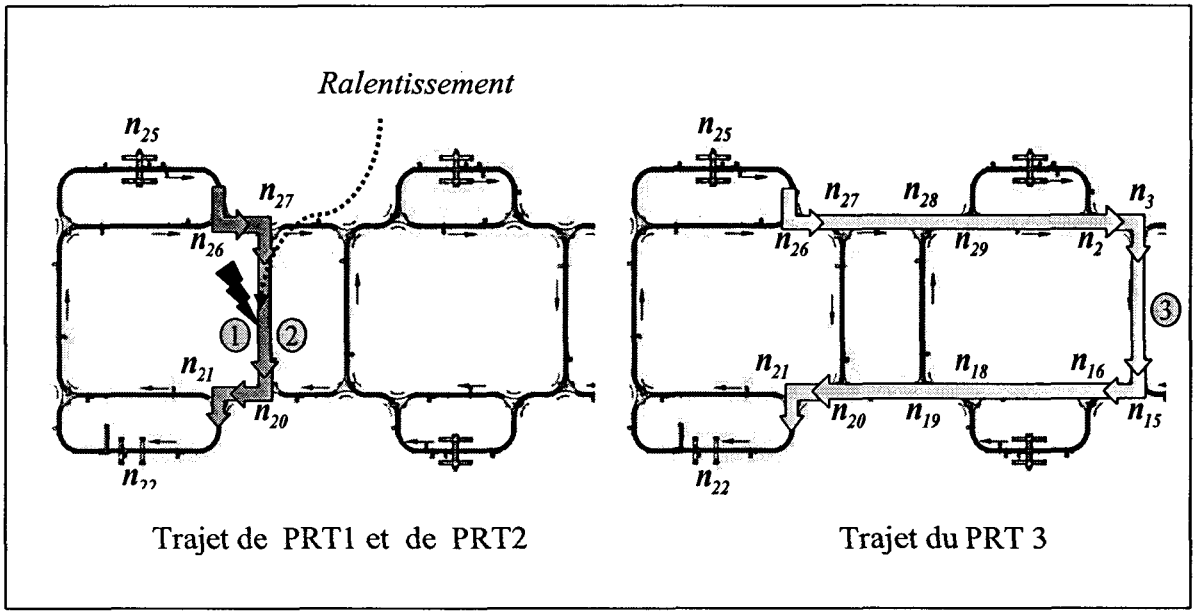


Figure 6.11 : chemin des PRT pour le scénario 2.

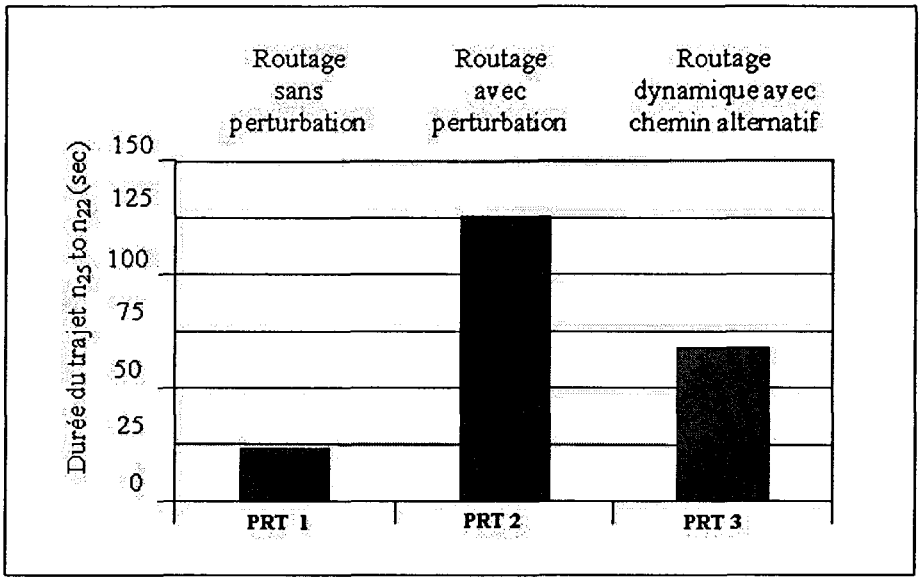


Figure 6.12 : evolution du temps de parcours des PRT - Scénario 2.

4.3 Scénario 3: pilotage implicite

Dans ce scénario, une décision d'arrêt temporaire du trafic sur l'arc [n8-n10] a été effectuée par le pilote de niveau supérieur, et ceci de façon à permettre, par exemple, une opération de maintenance sur le tronçon supportant physiquement l'arc concerné (Figure 6-13). Pour gérer ceci, le niveau de pilotage supérieur n'a pas besoin d'envoyer un ordre explicite aux PRT, il doit seulement agir au niveau du contrôleur de nœud associé au nœud n8. Le pilote supérieur écrit directement dans la table de routage du contrôleur de nœud n8 une valeur significative de l'infini pour la valuation du temps t8,10 pour l'arc [n8-n10]. Lorsqu'un PRT arrive sur le nœud n8, il met à jour sa propre table de routage avec notamment la valeur t8,10 qui a été changée par le pilote de niveau supérieur. Le PRT déclenche alors son « Dijkstra embarqué » qui calcule un chemin alternatif [n8-n9-n10].

Le comportement du PRT a été influencé par le pilote de niveau supérieur via un pilotage implicite.

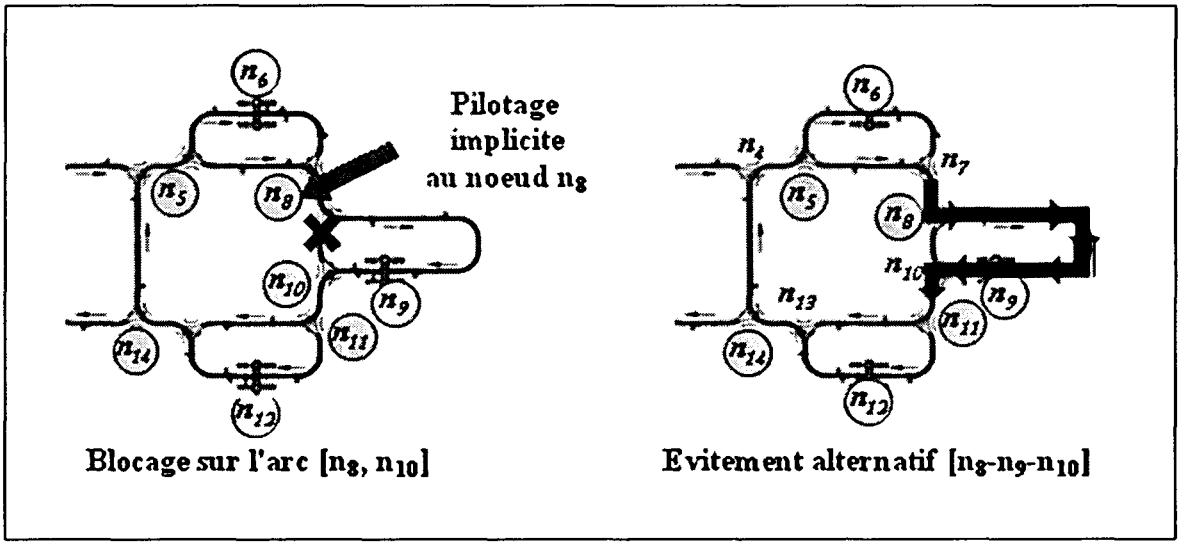


Figure 6.13 : chemins de routage du PRT - Scenario 3

Les résultats montrent que la prise de décision locale permet une bonne robustesse du système aux perturbations et une bonne réactivité. Cette prise de décision locale est particulièrement adaptée à la prise en compte de l'état réel du système au plus proche des éléments de base le constituant.

5 CONCLUSION

L'expérimentation du modèle de pilotage (routage) proposé, réalisée sur le site de la cellule flexible du Pôle AIP-PRIMECA de Valenciennes, a permis de montrer sa faisabilité et son intérêt.

Trois scénarios (insertion d'un shuttle dans le réseau, perturbations de trafic, blocage d'un tronçon du réseau) ont été expérimentés. Dans les trois cas, la capacité d'autonomie et d'auto-adaptation aux perturbations du système de pilotage (pas de reconfiguration du réseau, bon comportement en temps en cas de perturbation avec le pilotage autonome) a montré la flexibilité apportée par l'architecture holonique SEAH du système grâce au holon HF.

Conclusion générale

L'objectif de cette thèse était la proposition d'un modèle générique de pilotage de systèmes à flux guidés (SFG) fondé sur des concepts holoniques.

La thèse a été réalisée en cotutelle entre deux laboratoires (CIMR et TEMPO/PSI).

- la plateforme du CIMR est orientée vers le domaine manufacturier (flexibilité au niveau des postes de travail, système de transport de type convoyeur en boucle fermée avec des routes fixes),

- la plateforme du Pôle PRIMECA – AIP de Valenciennes, est également orientée production mais elle dispose d'un système de convoyage fortement connexe, permettant l'expérimentation de scénarios de routage dynamique et la mise en œuvre d'un pilotage hétérarchique. Elle permet ainsi d'évaluer des systèmes de pilotage de systèmes de transport à flux guidés comme les systèmes de type PRT.

Les fortes similitudes existant entre un système FMS et un système PRT ont conduit à envisager l'utilisation des plateformes des deux laboratoires pour les expérimentations nécessaires à l'application et la validation du modèle générique de pilotage de systèmes à flux guidés (SFG) proposé.

Le premier chapitre a présenté le contexte de la thèse qui comprend la définition d'un système à flux guidé, l'analyse critique de l'état de l'art des systèmes à flux guidé, ainsi que leur problématique, et une étude des architectures de pilotage classiques, récentes et émergentes. Le chapitre se termine avec des études de cas sur l'application des concepts holoniques pour le pilotage des systèmes à flux guidé. La première et plus importante conclusion est la forte similitude entre les systèmes de transport de voyageurs, type PRT, et les systèmes de fabrication - ce qui permet de les traiter de façon générale comme un seul cas de systèmes à flux guidé. La deuxième conclusion est que chaque type d'architecture de pilotage (hiérarchique ou hétérarchique) a ses avantages et inconvénients :

- *les architectures hiérarchiques* offrent un comportement optimal global mais manquent d'agilité face à des perturbations,

- dans *les architectures hétérarchiques* les entités ont une autonomie importante et le système peut alors réagir rapidement aux changements non prévus (d'origines techniques ou/et liés au marché). Toutefois, parce que le comportement d'une entité dépend des comportements des autres entités, il est, dans certains cas impossible d'optimiser globalement le fonctionnement du système et ses performances sont difficiles, voir impossibles, à prévoir.

On propose alors de combiner les deux types d'architectures en essayant de profiter des avantages de l'une et de l'autre. Ce type de pilotage est repris des concepts de l'« open-control ».

Le deuxième chapitre propose un modèle holonique générique d'un système de pilotage d'un SFG. On introduit en premier lieu la notion d'entité comme élément constitutif d'un SFG. Certaines entités - entités actives - peuvent participer au processus décisionnel de pilotage, allocation et routage des différents flux ; les autres entités constituent des entités non actives. La structure générale d'un système à entités actives (SEA) est alors présentée et le concept d'entité active généralisée (EAG) est défini. Pour élaborer le modèle holonique du SEA, l'holonification du SEA est réalisée en introduisant le holon actif générique (HAG) qui dans sa forme normale est un holon composé. Le modèle holonique de pilotage d'un SEA (SEAH) est ensuite proposé en présentant d'abord le modèle statique des holons du SEAH, ensuite les modèles dynamique et de comportement.

L'apport essentiel du modèle holonique générique SEAH proposé réside dans l'adoption d'un principe systémique de faible couplage et de forte cohésion qui conduit au concept de **holon flux**. Le holon flux est capable, en toute autonomie, de prendre en charge la réalisation des services offerts par un SFG (ex : fabrication d'un produit au sein d'un FMS, ou, dans le domaine des PRT, assurer le transport de bout en bout de la course d'un usager). Le flux, habituellement passif, est ici explicite et actif au regard du pilotage.

Le chapitre trois a présente un modèle de système de pilotage semi-hétérarchique qui, en conditions de stabilité opérationnelle de l'environnement de production, optimise d'une manière globale les lots de production même si des commandes rapides apparaissent. En cas de perturbations l'allocation des ressources et le routage sont faits en temps réel, ce qui rend le système de pilotage agile et assurent une optimalité au niveau des lots de fabrication en cours de réalisation.

L'avantage de cette approche est que la planification et l'allocation des ressources bénéficient au départ d'une durée suffisante pour calculer l'ordre d'insertion des produits et l'allocation des ressources aux opérations nécessaires pour leur réalisation en permettant d'obtenir le plus court temps total de production, le chargement le plus uniforme possible des ressources ou d'autres indicateurs tel que le coût.

Le chapitre quatre décrit les expérimentations réalisées pour évaluer les propositions du modèle proposé au chapitre trois. Les résultats, présentés dans plusieurs conférences scientifiques, et dans des revues montrent le bien fondé de l'architecture et la capacité du système de pilotage à optimiser la production, à réagir vite (agilité) aux perturbations internes et externes, en préservant un fonctionnement robuste du système cible, ainsi que d'intégrer le pilotage implicite de FMS selon les deux modalités : pilotage implicite sociétal et pilotage implicite environnemental.

Le chapitre cinq concerne les systèmes de transport de type PRT. L'architecture générique du chapitre deux est reprise ici avec des adaptations concernant les entités actives, celles-ci étant totalement autonomes du point de point de vue décisionnel.

Dans le chapitre six, l'expérimentation du modèle de pilotage (routage), proposé dans le chapitre 5, réalisée sur le site de la cellule flexible du Pôle AIP-PRIMECA de Valenciennes, a permis de montrer sa faisabilité et son intérêt. Trois scénarios (insertion

d'un shuttle dans le réseau, perturbations de trafic, blocage d'un tronçon du réseau) ont été expérimentés. Dans les trois cas, la capacité d'autonomie et d'auto-adaptation aux perturbations du système de pilotage (pas de reconfiguration du réseau, bon comportement en temps en cas de perturbation avec le pilotage autonome) a montré la flexibilité apportée par l'architecture holonique SEAH du système grâce au holon HF.

En conclusion, les principales contributions originales de cette thèse sont :

- le développement d'un modèle générique de pilotage holonique pour des systèmes à flux guidé basés sur des holons flux,

- la conception d'une architecture avec pilotage variable (modes hiérarchique, semi-hétérarchique, hétérarchique) configurable selon les objectifs optimalité/agilité imposés au système cible (FMS ou PRT) qui commute automatiquement entre les différents modes de pilotage en réaction aux perturbations ;

- l'orientation services, des fonctionnalités d'allocation de ressources et de routage, basée sur le Modèle d'Accès aux Services des Ressources (RSAM - *Resource Service Access Model*) mis à jour en permanence avec l'état de fonctionnement des ressources, leur disponibilité et la qualité de leurs services ;

- l'implémentation du concept de holon flux: avec intelligence embarquée (plateforme PRT de l'AIP) ou avec intelligence distante dans la cellule CIMR) ;

- la mise en œuvre d'une implémentation tolérante aux pannes du système de pilotage hétérarchique dans le cas des FMS,

- l'extension des modèles classiques de pilotage FMS avec l'ajout de la fonctionnalité d'alimentation et de réalimentation des stocks en ligne,

- le développement d'une procédure heuristique de planification et d'ordonnancement basée sur des règles de production. Cette procédure peut être utilisée pour refaire la planification et l'ordonnancement en cas de perturbations, de défaillance des ressources ou de ruptures de stocks,

La conception d'un outil de simulation de la fabrication basé sur une matrice temporelle de transport (TTM - *Time Transport Matrix*) permettant le calcul et la validation simultanée de la planification et de l'ordonnancement des lots de fabrication, ainsi que la traçabilité des opérations sur les produits ;

Le développement d'une procédure (EDF) pour insérer des ordres de production rapides avec des ordres déjà acceptés et en cours d'exécution ;

Le développement d'une procédure de routage en ligne des entités actives dans un système de transport perturbé, en utilisant des informations en temps réel.

Le modèle SEAH de pilotage est générique. Sa mise en œuvre nécessite des raffinements et des choix conduisant à des solutions différentes dans le cas des FMS, et des PRT.

En accompagnement des résultats expérimentaux, les activités de conception et de mise en œuvre effectuées dans le cadre de la thèse ont fourni des résultats pratiques sous forme de logiciels utilisés à présent dans les cellules CIMR et AIP :

- le logiciel SOFHICOR pour la gestion de la cellule avec la prise en charge des ordres des clients, la planification des ordres et l'ordonnancement des opérations sur les machines disponibles et le routage développés au CIMR,

- le logiciel de routage pour un système expérimental PRT, basé sur un système à l'origine de production flexible utilisé à l'AIP,

Le travail présenté dans cette thèse ouvre plusieurs perspectives de recherche, particulièrement en ce qui concerne :

- les concepts de pilotage dirigé par le produit et de "produit intelligent" permettront d'étendre la notion de holon flux pour des structures plus complexes,

- l'orientation "services" permettant d'intégrer la gestion des processus "business" et le pilotage des processus techniques dans une architecture orientée services (SOA) en tenant compte du cycle de vie du produit,

- le pilotage de "systèmes tolérants" prenant en compte les caractéristiques et variations de l'environnement, du flux des matériaux, des énergies consommées, du comportement des ressources... en modifiant en conséquence les modes de planification, d'ordonnancement, d'allocation de ressources, de routage et de contrôle pour réaliser les processus à des coûts acceptables, préserver les performances des ressources, intégrer le contrôle qualité des produits et la maintenance des ressources.

Références bibliographiques

1. Albers S., Leonardi S. Online algorithms: Trends and perspectives. ACM Computing Surveys, 31:3 (electronic edition), 1999
2. Anderson, J.E., Control of Personal Rapid Transit Systems, JAT, 32:1(1998)
3. Anton, F.D., Borangiu, Th., Anton, S., An Implementation of High Availability in Networked Robotic Systems, Proceedings Vol. of the 4th Int. Conf. on Informatics in Control, Automation and Robotics – ICINCO 2007, May, 9-12, Angers, 2007
4. Bagchi, T.P., Multiobjective Scheduling by Genetic Algorithms, Kluwer Academic Publishers, 1999
5. Baker, K., Introduction to Sequencing and Scheduling, John Wiley & Sons, 1974.
6. Barata, J., Camarinha-Matos, L.M., Onori, M., A Multiagent Based Control Approach for Evolvable Assembly Systems. INDIN'05 - 3rd IEEE International Conference on Industrial Informatics, pp. 478-483 (IEEE Xplore, Perth - Australia, 2005)
7. Barata, J., The Cobasa architecture as an answer to shop floor agility, manufacturing the future - concepts, technology, Visions. Pro Literatur Verlag, 31-76, 2006
8. Barták, R., Mixing planning and scheduling to model process environments, PACLP, Manchester, UK, 2000
9. Berger, T., Sallez, Y., Raileanu, S., Tahon, C., Trentesaux, D., Borangiu, Th., Personal Rapid Transit in an Open Control framework, Computers & Industrial Engineering, doi:10.1016/j.cie.2010.12.010, 2011
10. Berger, T., Zbib, N., Sallez, Y., Trentesaux, D., Active product driven control of dynamic routing in FMS. Journal Européen des Systèmes Automatisés, 43(4-5), 407-434, 2009
11. Bond, A. H., Gasser, L., Readings in Distributed Artificial Intelligence. Morgan Kaufmann Publishers: San Mateo, CA, 1988
12. Borangiu, Th., Raileanu, S., Ivanescu, N., Anton, S., Anton, F.D., Dogar, A., RVHOLON – A Holonic Control Platform, Proceedings of the 3rd National Conference Cercetarea de Excelenta, October 24-26, Brasov, Romania, 2007
13. Borangiu, Th., Raileanu, S., Ivanescu, N., Modelling and Design of a Holonic Manufacturing Control System with Optimal Job Shop Scheduling, Proceedings Volume 7e Conférence Internationale de MOdélisation et SIMulation MOSIM 2008, Mars 31-Avril 2, Paris, 2007
14. Borangiu, Th., Tomas Balibrea, L.M., Anton, F.D., Ivanescu, N., Silvia Tunaru, Dogar, A., Raileanu, S., Holonic Fault-Tolerant Control of Networked Robot-Vision Assembly

Stations, Intelligent Assembly and Disassembly IAD 2007, May 23-25, Alicante, Spain, 2007

15. Borangiu, Th., Anton, F., Anton, S., Dogar, A., Ivanescu, N., Raileanu, S., Bruja, A., Embedded Control of a Bricklaying Robot from CAD Production Data, 16th International Workshop Robotics in Alpe-Adria-Danube Region RAAD 2007, June 7-9, Ljubljana, Slovenia, 2007

16. Borangiu, Th., Ivanescu, N., Raileanu, S., Rosu, A., Anton, F.D., Anton, S., Dogar, A., Dumitrache, A., Holonic Manufacturing Control with Robot-Vision Part Supply, Proc. of the National Conf. "Excellence Research – A Way to Innovation", July 27-29, 2008, Brasov, Romania, ISSN 1844-7090, pp. 146-1 – 146-6

17. Borangiu, Th., Ivanescu, N., Raileanu, S., Rosu, A., Vision-Guided Part Feeding in a Holonic Manufacturing System with Networked Robots, Proc. of the 17th International Workshop on Robotics in Alpe-Adria-Danube Region RAAD 2008, September 15-17, 2008, Ancona, Italy, ISBN 978-88-903709-0-8, pp. 97

18. Borangiu, Th., Anton, F.D., Anton, S., Dogar, A., Raileanu, S., Knowledge-Based Holonic Robot Control of Job Shop Assembling, The Journal of Control Engineering and Applied Informatics CEAI, Vol 8, No 4, Bucharest, pp. 22-31, ISSN 1454-8658, 2009

19. Borangiu, Th., Raileanu, S., Rosu, A., Parlea, M., Anton, S., Anton, F. D., Semi-heterarchical distributed control of a holonic manufacturing cell, The Journal of Control Engineering and Applied Informatics- CEAI, Vol. 11, Nr. 2, pp. 14-21, 2009

20. Borangiu, Th., Raileanu, S., Rosu, A., Parlea, M., Anton, F. D., Management of changes in a holonic manufacturing system with dual-horizon dynamic rescheduling of production orders, 13th IFAC Symposium on Information Control Problems in Manufacturing- INCOM, Moscow, June 3-5, 2009

21. Borangiu, Th., Gilbert, P., Ivanescu, N., Rosu, A.: An implementing framework for holonic manufacturing control with multiple robot-vision stations, EAAI, 22, 4-5, Elsevier, June 2009, p. 505-521

22. Borangiu, Th, Raileanu, S., Rosu, A., Parlea, M., Holonic Robot Control for Job Shop Assembly with Dynamic Simulation, Programmable Logic Controller, In-Tech Publications, Vienna, Austria, ISBN 978-953-7619-X-X, 2009

23. Borangiu, Th, Raileanu, S., Trentesaux, D., Berger, T.,Open Manufacturing Control with Agile Reconfiguring of Robot Services, 19th International Workshop on Robotics in Alpe-Adria-Danube Region, RAAD 2010, Budapest, Hungary June 23-25, 2010, ISBN: 978-1-4244-6884-3

24. Borangiu Th, Raileanu, S., Berger, T., Trentesaux, D., Semi-heterarchical agile control architecture with intelligent product-driven scheduling, 10th International Workshop on Intelligent Manufacturing Systems, IMS2010, Lisbon, Portugal, 1-2 July 2010

25. Brussel, H.V., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P., Reference Architecture for Holonic Manufacturing Systems: PROSA, Computers in Industry, Volume 37, Issue 3 (November 1998), Special issue on manufacturing systems, Pages: 255 – 274, ISSN:0166-3615, 1998

26. Buchanan, M., Anderson, J.E, Tegnér, G., Fabian, L., Schweizer, J., Emerging Personal Rapid Transit Technologies Introduction, State of the Art, Applications, In Proceedings of the AATS conference, Bologna, Italy, 7-8 Nov.2005

27. Buckminster Fuller R., Alternative Metaheuristics, chapter Alternative Metaheuristics, pag 547-621, ISBN 978-0-387-33254-3, 2007

28. Bussmann, S., Schild, K., Self-organizing manufacturing control: an industrial application of agent technology, in: Proceedings of the MAS'00, 2000, pp. 10–12

29. Carnegie, J., Hoffman, P., "Viability of Personal Rapid Transit in New Jersey," February 2007, pp. A3-9

30. Castagna P., Maza S., Une bibliotheque dediee au routage de chariots bidirectionnels. 4 ème Conférence Francophone de MOdélisation et SIMulation «Organisation et Conduite d'Activités dans l'Industrie et les Services», MOSIM'03 – Avril 2003 - Toulouse (France).

31. Cea Ramirez, A.A., Contribution à la Modélisation et à la Gestion des Interactions Produit-Processus dans la Chaîne Logistique par l'Approche Produits Communicants, Thèse, 2006

32. Cheng, R.M., Gen, M., Tsujimura, Y., A Tutorial Survey of Jobshop Scheduling Problems using Genetic Algorithm: Pt I Representation, International Journal of Computers and Industrial Engineering, 1996

33. Chirn, J.L., McFarlane, D. C., Application of a holonic component-based architecture for the control of a robot assembly cell. Proceedings of IEEE Robotics and Automation, San Francisco, 2000.

34. Christensen, J., Holonic manufacturing systems: initial architecture and standards directions. First European Conference on Holonic Manufacturing Systems, European HMS Consortium, Hanover, Germany, 1994.

35. Clarinet System, Network Connectivity for Mobile, Devices, http://www.clarinetsys.com, 2010.

36. Concepts for Holonic Manufacturing, http://www.mech.kuleuven.be/goa/concepts.htm, 2010.

37. Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C., Introduction to algorithms. Section 24.1: The Bellman-Ford algorithm (2nd ed.). MIT Press and McGraw-Hill (pp. 588–592.), 2001

38. Dijkstra, E.W, A note on two problems in connexion with graphs. Nu-merische Mathematik, 1:269, 271., 1959.

39. Dilts, R.B., Epstein, T., Dilts R.W., Tools for Dreamers, (1st ed.). Cupertine: Meta Publications, 1991.

40. Di Caro, G., Dorigo, M., AntNet: Distributed stigmergic control for communications networks. Journal of Artificial Intelligence Research, 9, 317–365, 1998

41. Dorigo, M., Stützle, T., Ant colony optimization. The MIT Press, 2004

42. Ferber, J., Les systèmes multi agents – vers une intelligence collective, InterEdition, Paris (ISBN 2-7296-0572-X), 1995

43. Fischer K., Schillo, M., Siekmann, J., Holonic Multi Agent Systems: The Foundation for the Organization of Multi Agent Systems. Proceedings of the First International Conference on Applications of Holonic and Multi Agent Systems (HoloMAS'03), LNAI 2744:71-80, Springer-Verlag 2003

44. Fischer K., Self-Organization in Holonic Multi Agent Systems, Book Mechanizing Mathematical Reasoning, Copyright 2005, ISBN 978-3-540-25051-7

45. Franklin, S., Graesser, A., Is it an Agent, or just a Program? : A Taxonomy for Autonomous Agents, Proceedings of the Third InternationalWorkshop on Agent Theories, Architectures, and Languages, Springer-Verlag, 1996

46. Fu, L., Sun, D., Rilett, L. R., Heuristic shortest path algorithms for transportation applications: State of the art. Computers & Operations Research, 33, 3324–3343, 2006

47. Garaix, T., Artigues, C., Feilletand, D., Josselin, D., Vehicle routing problems with alternative paths: An application to on-demand transportation. European Journal of Operational Research, 204(1), 62–75, 2010

48. Garrido A., Salido M.A., Barber F., López M.A.., Heuristic Methods for Solving Job-Shop Scheduling Problems, Proc. 'ECAI-2000 Workshop on New Results in Planning, Scheduling and Design (PuK2000)', pp. 44-49. 2000.

49. Giret A., Botti V., Valero S. MAS Methodology for HMS. HoloMAS 2005.

50. Gopalan K., Chiueh T., Integrated Real-time Resource Scheduling, Technical Report TR-56, Experimental Computer Systems Labs, Dept. of Computer Science, SUNY at Stony Brook, Stony Brook, NY, USA, Jan 2001

51. Grassé, P., La reconstruction du nid et les coordinations inter-individuelles chez Bellicositermes natalensis et Cubitermes sp. La théorie de la stigmergie: Essai d'interprétation des termites constructeurs. Insectes Sociaux, Vol. 6, pp. 41-84, 1959

52. Greenberg, H., A Branch-and-Bound Solution to the General Scheduling Problem, Operations Research, 1968

53. Groover, M., Automation, Production Systems and CIM. Prentice-Hall, 1987

54. HMS I and II Holonic Manufacturing Systems – Phase I and II IMS Project No. 95003

55. Höpf M., Holonic Manufacturing Systems (HMS): The Basic Concept and a Report of IMS Test Case 5, Sharing CIM Solutions, J.K.H. Knudsen et al., 1994

56. Innovative Transportation Technologies, http://faculty.washington.edu/jbs/itrans/ , consulted in November 2010

57. Jennings, N.R., Sycara, K., Wooldridge, M., A Roadmap of Agent Research and Development, Autonomous Agents and Multi-Agent Systems, 1, 7–38, 1998

58. JESA : Journal Européen des Systèmes Automatisés, numéro special "Systèmes contrôlés par le produit", Vol 43/4-5, 2009

59. Koestler A., The Ghost in the Machine, Arkana Books, London, 1967

60. Kuehn W., Draschba C., Simulation based job-shop production analyzer, Proceedings 18th European Simulation Multiconference Graham Horton, SCS Europe, 2004

61. Kumari, S.M., Geethanjali, N., A survey on shortest path routing algorithms for public transport travel. Global Journal of Computer Science and Technology, 9(5), 73–76, 2010

62. Kusiak, A., Intelligent Manufacturing Systems, Prentice Hall, Englewood Cliffs, N.J., 1990

63. Leitão, P., An Agile and Adaptive Holonic Architecture for Manufacturing, Control. PhD Thesis, 2004

64. Leitão, P., Restivo, F., ADACOR: A holonic architecture for agile and adaptive manufacturing control, Computers in Industry, Volume 57, Issue 2, Pages 121-130, February 2006

65. Leitão, P., Casais, F., Restivo, F., Holonic Manufacturing Control: A Practical Implementation. Proceedings of the 6th IFIP International Conference on Information Technology for Balanced Automation Systems, 2004

66. Le Moigne, J. L., La théorie du système général, Théorie de la modélisation, Second edition, Paris : PUF, 338p, (1994)

67. Lowson, M., A New Approach to Effective and Sustainable Urban Transport. Transportation Research Board 2003: Paper 03-2140, 2003

68. McCulloch, W. S., A « Heterarchy of values Determined by the Topology of Nervous Nets », Bull. math. biophys., vol. 7, 1945, p. 89-93

69. McFarlane D., Bussmann S., Holonic Manufacturing Control: Rationales, Developments and Open Issues, Agent-Based Manufacturing, chapter 13, pages 303-326, Springer-Verlag, Berlin, 2003

70. McFarlane, D., Zaharudin, A., Wong, C.Y., Agarwal, V., The Intelligent Product Driven Supply Chain, in Proceedings of IEEE SMC 2002, Tunisia, 2002.

71. Maza S., Castagna P., A performance-based structural policy for conflict-free routing of bi-directional automated guided vehicles. Computers in Industry 56 (2005) 719–733.

72. Meyer, G.G. , Främling, K., Holmström, J., Intelligent Products: A survey, Computer Industry, doi:10.1016/ j.compind.2008.12.005, Computers in Industry, Volume 60, Issue 3, April 2009, Pages 137-148

73. Montana, D., Hussain, T., Vidaver, G., A Genetic-Algorithm-Based Reconfigurable Scheduler, Evolutionary Scheduling, K. Dahal, K.C. Tan and P. Cowling (eds.), Springer, 2007

74. Netlogo (2010). <http://ccl.northwestern.edu/netlogo/docs>.

75. Neubert, G., Contribution à la spécification d'un pilotage proactif et réactif pour la gestion des aléas, thèse, 1997

76. Noubissié Tchako, J.F., Contribution à la conception d'un système de pilotage distribué pour les systèmes automatisés de production, thèse, 1994

77. Nylund H., Kai Salminen, Paul H Andersson, A multidimensional approach to digital manufacturing systems, Proceedings of DET2008 5th International Conference on Digital Enterprise Technology Nantes, France 22-24 October 2008

78. PABADIS'PROMISE project, 2008, Industrial Application of the PABADIS'PROMISE System

79. Parunak, H. V. D., Brueckner, S., Sauter, J., Digital pheromone mechanisms for coordination of unmanned vehicles. In Proceedings of the 2002 AAMAS. Bologna, Italy, 2002

80. Pasquier, M., Quek, C., Tan, B. T., Chee, S., Opportunistic planning for a fleet of transportation robots. Engineering Applications of Artificial Intelligence, 14(3), 329–340, 2001

81. Pechoucek, M., Marik, V., Industrial deployment of multi-agent technologies: review and selected case studies, Autonomous Agents and Multi-Agent Systems, 2008, 17:397–431

82. Pétin, J.F., Gouyon, D., Morel, G., Supervisory synthesis for product-driven automation and its application to a flexible assembly cell Control Engineering Practice, Volume 15, Issue 5, May 2007, Pages 595-614.

83. PRT Simulators (2009). Innovative transport design, planning and operational simulation software packages. <http://faculty.washington.edu/jbs/itrans/simu.htm>.

84. Pujo, P., Broissin, N., Fouzia, O. (2009) PROSIS: An isoarchic structure for HMS control Engineering Applications of Artificial Intelligence 22: 7. 1034-1045 October.

85. Pujo, P., Kieffer, J., 'Fondements du pilotage des systèmes de production'. Lavoisier, 2002

86. Qianjun Xu, Introduction to Job Shop Scheduling Problem, Oct. 30, 2001.

87. Qiu L., Hsu W.J., Huang S.Y., International of production research, 2002, vol. 40, No 3, 745-760.

88. Rahimifard, S., 2004. Semi-heterarchical production planning structures in the support of team-based manufacturing, International Journal of Production Research, 42, 17, September 1, 3369-3382(14), Taylor and Francis Ltd.

89. Raileanu S., Production scheduling in a holonic manufacturing system using the open-control concept, U.P.B. Science Bulletin, 2010

90. Raileanu, S., Berger, Th., Sallez, Y., Borangiu, Th., Trentesaux, D., The open-control concept for holonic manufacturing, International Journal of Mechanics and Control Vol.11, No.01, 2010, ISSN: 1590-8844

91. Raileanu, S., Berger, Th., Sallez, Y., Borangiu, Th., Trentesaux, D., The Open-Control Concept in a Holonic Manufacturing System, Proc. of the 18th International Workshop on Robotics in Alpe-Adria-Danube Region - RAAD, Brasov, Romania, May 25-27, ISBN 978-606-521-315-9, pp. 86, ISSN 2066-4745, 2009

92. Raileanu, S., Sallez, Y., Berger, Th., Borangiu, Th., Trentesaux, D., Holonic implementation of the open-control paradigm, International Conference on Industrial Engineering and Systems Management, ISBN 978-2-9600532-2-7,IESM'09, May 13 - 15, 2009, Montreal, Canada

93. Raileanu, S., Sallez, Y., Berger, T., Trentesaux, D., Personal Rapid Transit in Open Control framework, 2009 International Conference on Computers & Industrial Engineering, 6-8 July, 2009, Troyes, France

94. Raileanu, S., Sallez, Y., Berger, Th., Trentesaux, D., Du produit passif au produit actif: le concept d'augmentation, 11ème Colloque National AIP PRIMECA, 22-24 avril 2009, La Plagne, France

95. Reid G. Smith, 1980, IEEE Transactions on Computers, Vol. C-29, No. 12, December 1980

96. Sallez, Y., Berger, T., Raileanu, S., Chaabane, S., Trentesaux, D., Semi-heterarchical control of FMS : From theory to application, Engineering Applications of Artificial Intelligence, vol. 23, pp. 1314–1326, 2010

97. Sallez, Y., Berger, T., & Trentesaux, D., A stigmergic approach for dynamic routing of active products in FMS. Computer in Industry, 60(3), 204–216, 2009

98. Sallez, Y., Berger, T., Trentesaux, D., Open-control: a new concept for integrated product-driven manufacturing control, 13th IFAC Symposium on Information Control Problems in Manufacturing, ISBN: 978-3-902661-43-2, DOI: 10.3182/20090603-3-RU-2001.00347, pp. 2061-2066, 2009

99. Sauer, O., Automated engineering of manufacturing execution systems – a contribution to "adaptivity" in manufacturing companies, Proceedings of DET2008 5th International Conference on Digital Enterprise Technology Nantes, France 22-24 October 2008

100. Schillo, M., Fischer, K.: Holonic Multi Agent Systems. KI Zeitschrift, 4:54, arendtap 2003

101. Schillo, M., Multiagent Robustness: Autonomy vs. Organisation. PhD Thesis, Department of Computer Science, Universität des Saarlandes. 2004

102. Sha, L., Abdelzaher, T., Arzen, K., Cervin, A., Baker, T., Burns A., Buttazzo G., Caccamo, M., Lehoczky, J., Mok, A.K., Real Time Scheduling Theory: A Historical Perspective, Real-Time Systems, Vol. 28, No. 2-3, 101-155, 2004

103. Shen, W. M., Will, P., Galstyan, A., Chuong, C. M., Hormone-inspired self-organization and distributed control of robotic swarms. Autonomous Robots, 17, 93–105, 2004

104. Szillat, M.T., A Low-level PRT Microsimulation, PhD thesis, 2001

105. Theraulaz, G., Bonabeau, E., A brief history of stigmergy. Journal of Artificial Life, 5(2), 97–116, 1999

106. Todd, D., Multiple Criteria Genetic Algorithms in Engineering Design and Operation, PhD Thesis, Engineering Design Centre, University of Newcastle Upon Tyne, October 1997.

107. Trentesaux D., Berger T., Sallez Y., Product & resource driven control of allocation and routing processes in FMS, MOSIM, Paris, France, 2007

108. Trentesaux D., Les systèmes de pilotage hétérarchiques : innovations réelles ou modèles stériles ? Journal Européen des Systèmes Automatisés, 9-10, pp. 1165-1202, ISSN 1269-6935, 2007

109. Upton D., Flexible Structure for Computer Controlled Manufacturing System. Manufacturing Review, 5(1):58–74l, 1992.

110. Vis I., Survey of research in the design and control of automated guided vehicle systems. European Journal of Operational Research, 170 (2006) ,677–709.

111. Wago (2011). Wago system. <http://www.wago.com>.

112. www.2getthere.eu/ , About Group Rapid Transit and Dual-Mode Applications, Robert H.C. Lohmann, consulted in January 2009

113. www.adept.com, 2010

114. www.edc.ncl.ac.uk/highlight/rhfebruary2007.php/, basic job-shop scheduling, consulted in November 2010

115. www.fipa.org, 2010

116. www.speedylook.com/Scheduling_of_workshop.html , Scheduling of workshop consulted in November 2010

117. www.speedylook.com/Scheduling_of_workshop.html, 2009

118. www.taxi2000.com, 2010

119. www.ultraprt.com, 2010

120. www.vectusprt.com, 2010

121. Zambrano G., Pach C., Adam E., Berger T., Trentesaux D., Myopic Behaviour in Heterarchical Control of FMS. International Conference on Industrial Engineering and Systems Management, Metz (France), 25-27th May, 2011

122. Zbib, N., Raileanu, S., Sallez, Y., Berger, T., Trentesaux, D., From Passive Products to Intelligent Products: the Augmentation Module Concept. 5th International Conference on Digital Enterprise Technology, Nantes, France, 2008

Annexe 1 : Composants d'un système de transport PRT

Nous considérons qu'un système de transport de type PRT comporte 4 composants essentiels : les véhicules, les stations, l'infrastructure et les passagers. Dans cette annexe, nous présentons pour chacun d'eux une description physique et une description fonctionnelle. Toutefois, cette description ne se veut pas exhaustive car, comme nous l'avons signalé au chapitre 1, il existe actuellement un nombre important de systèmes PRT, à des stades différents de développement et correspondant à une grande variété dans les solutions adoptées. Par ailleurs les informations concernant ces différents systèmes ne sont pas toujours disponibles.

On trouvera sur les sites (http://faculty.washington.edu/jbs/itrans/prtquick.htm), (http://faculty.washington.edu/jbs/itrans/techtable.htm) et dans (Berger et al. 2011) un état de l'art et une présentation des systèmes de transport PRT actuels et de leur niveau de développement. La description des différents composants s'appuie sur ces différents documents.

1- Le véhicule

1 - 1 Description physique

Typiquement, les véhicules utilisés dans les systèmes PRT peuvent accueillir de 1 à 6 personnes avec leurs bagages, mais certains systèmes utilisent des véhicules pouvant accueillir 18 passagers. Ils sont généralement équipés d'une motorisation électrique (moteurs linéaires ou tournants) et de batteries. Certains projets prévoient d'utiliser une suspension magnétique.

Selon les modes de circulation et de guidage ils sont équipés de roues et d'un ou deux rails de guidage. Pour limiter la consommation énergétique et diminuer les dimensions des infrastructures, on cherche à limiter le poids du véhicule par des choix de structures et de matériaux adaptés (Exemple figure 1 - véhicule PRT ULTra : cadres en aluminium, coque en thermoplastique ABS)

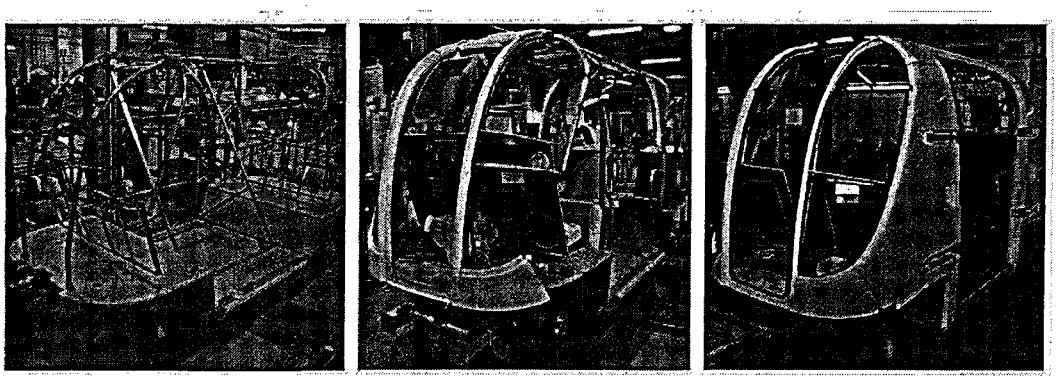


Figure 1 : véhicule PRT ULTra

Les figures 2 (Système PRT Vectus) et 3 (système PRT Serpentine) montrent deux autres exemples de véhicules.

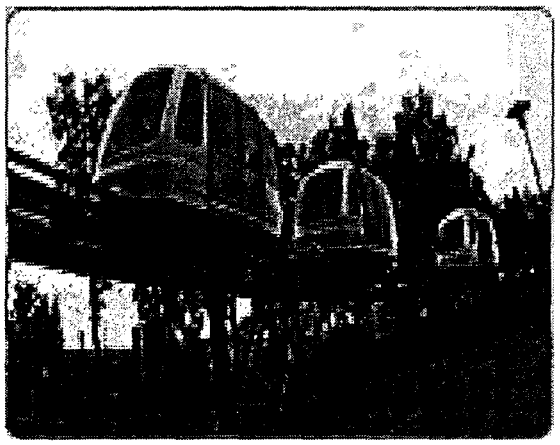


Figure 2 : véhicules PRT Vectus

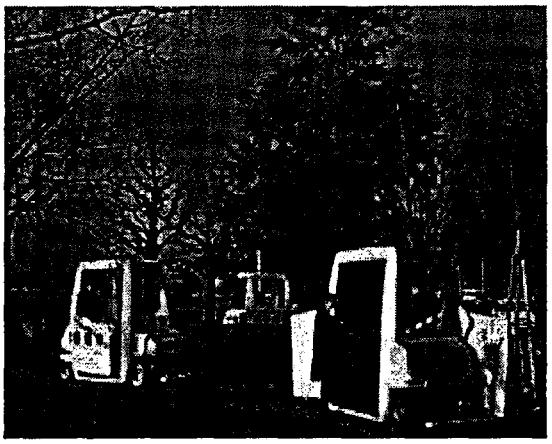


Figure 3 : véhicules PRT Serpentine

1 - 2 Fonctions

- Accueil et transport des passagers

Le véhicule accueille les passagers et permet leur déplacement entre un point du réseau de transport (station origine) et un autre point du réseau (station de destination). La conception du véhicule facilite l'accès et la sortie du véhicule (dimensionnement des portes, alignement des niveaux quai-véhicules...).

- Guidage

Le guidage est assuré, au niveau du véhicule, par un ensemble de capteurs qui définissent la position axiale et latérale du véhicule. Ces informations sont transmises au système de positionnement du véhicule (PC de calcul, automatismes et actionneurs de position...).

- Communication : les besoins en communication d'un véhicule concernent :

- dans le cas d'un système de pilotage de la flotte de véhicules centralisé : l'échange bidirectionnel d'informations avec le système de pilotage en charge de l'allocation et du routage prévisionnel des véhicules,

- dans le cas d'un système de pilotage hybride : les échanges bidirectionnels d'informations avec, d'une part, un système de pilotage centralisé qui fournit les points de destination de chacun des véhicules PRT, qui assure l'allocation des véhicules et met à jour l'état du réseau, et, d'autre part, avec les autres véhicules pour permettre le routage dynamique.

Pour assurer les échanges d'informations nécessaires, le véhicule est doté d'un ou plusieurs ordinateurs de bord et systèmes de communication associés, spécialisés ou utilisés également pour les autres fonctions (informations aux voyageurs, communication, sécurité...).

- Information aux voyageurs

Les interfaces utilisateurs peuvent être très variées : écrans de visualisation, messages vocaux, plans du réseau, informations sur le parcours.

- Sécurité

Le véhicule est muni de dispositifs qui offrent une sécurité intrinsèque (ouverture/fermeture des portes, positionnement par rapport aux points d'embarquement/débarquement des passagers aux stations...) ou provoquant l'arrêt d'urgence (pare-chocs, bouton d'arrêt d'urgence...). Par ailleurs, le véhicule peut être également muni d'un ensemble de capteurs permettant la détection d'obstacles (ex. capteurs laser). Des systèmes d'évacuation d'urgence sont prévus (toboggans dans le cas d'un système aérien, voie piétonne et/ou routière...).

2 - Les stations

2 – 1 Description physique

La structure des stations est fonction du mode de circulation des véhicules et une grande variété d'architectures ont été conçues. Deux types de systèmes sont les plus utilisés : station au niveau du sol, station aérienne (véhicule suspendu ou porté). La station peut également être enterrée dans le cas d'un système PRT en circulation souterraine.

D'une manière générale, quatre types de zones peuvent exister au sein d'une station :

- la zone de circulation (ZC) : c'est la ligne de passage à la station,

- la zone de dégagement (ZD) : c'est la zone d'accès aux véhicules et de sortie des passagers. Elle libère la zone de circulation pour les véhicules circulant sur le réseau mais ne s'arrêtant pas à la station,

- la zone de réserve (ZR) : elle permet de stocker des véhicules vides.

- zone(s) d'aiguillage (ZA) : nécessaires pour diriger les véhicules au sein de la station et les replacer dans la zone de circulation.

La figure 4 donne un exemple de station du projet Cybertran (http://cybertran.com/) comportant une ZC à deux voies, une ZD, une ZA et une ZR.

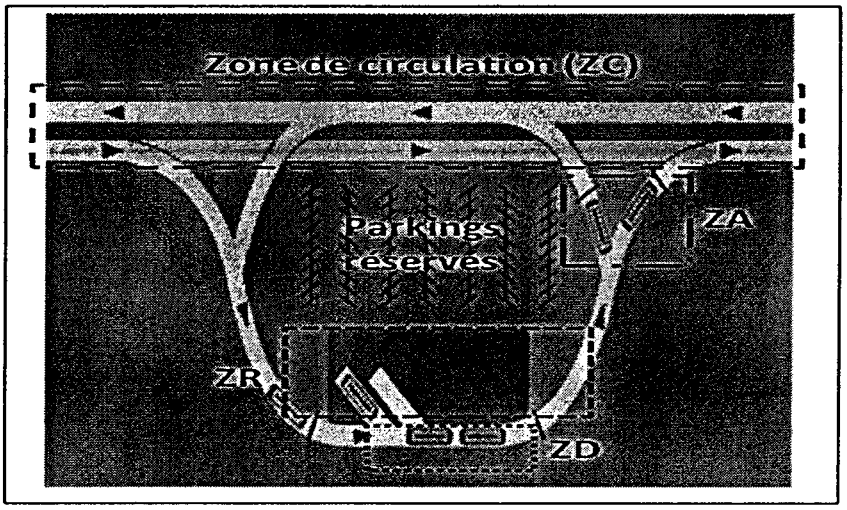


Figure 4 : station au niveau du sol - Système PRT Projet Cybertran.

La figure 5 donne un exemple de station aérienne du projet PRT Urbanaut (http://www.urbanaut.com/).

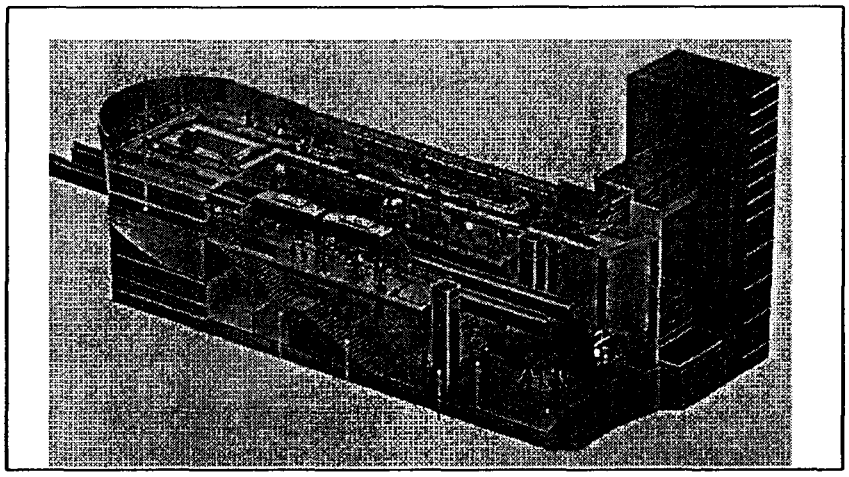


Figure 5 : station aérienne (Véhicule porté) : Système PRT – Projet Urbanaut

2 – 2 Fonctions

- Embarquement/débarquement des passagers

La station permet l'entrée des passagers à bord des véhicules et leur sortie au niveau d'un emplacement pour chaque véhicule. Ces emplacements sont généralement situés dans une zone spécifique (zone ZD figure 4).

- Aiguillage intra-station : l'aiguillage concerne les véhicules en arrêt à la station qui sont remis en circulation. Le système d'aiguillage permet d'orienter les véhicules dans un des sens de circulation.

- Stockage de véhicules vides

Une station peut disposer d'une zone de réserve (ex : zone ZR figure 4) permettant d'accueillir des véhicules vides durant les périodes de demande faible. En période de demande normale ou en cas d'une pointe de la demande, ces véhicules sont à disposition pour absorber les besoins en véhicules supplémentaires de transport à la station considérée ou une autre.

- Intermodalité

La station dispose de zones de stationnement spécifiques pour permettre l'intermodalité entre différents modes de transports : bus, vélos, automobiles.

- Paiement

Les stations proposent aux passagers divers modes de réservation de transport et de paiement (jetons, monnaie, carte d'abonnement, carte bancaire).

3 - Les infrastructures

3 – 1 Description physique

Les caractéristiques les plus largement répandues sont : voie de circulation au niveau du sol ou en aérien (véhicule suspendu ou porté), voie protégée ou en site propre pour les véhicules à roues, voie monorail ou à deux rails, voie protégée avec système de guidage enterré.

La figure 6 illustre le projet de système PRT Cabintaxi où les véhicules sont portés et suspendus. La figure 7 montre l'infrastructure rails utilisée dans le projet Autrans.

3 - 2Fonctions

Les infrastructures remplissent trois rôles essentiels :

- Guidage

Il est effectué par contact direct (ex : rail infrastructure – rail véhicule), par système électromagnétique (ex : plots magnétiques enterrés), par suspension magnétique pour certains projets.

- Alimentation en énergie

Les systèmes d'alimentation font appel à des technologies diverses (rails, plots électromagnétiques enterrés...)

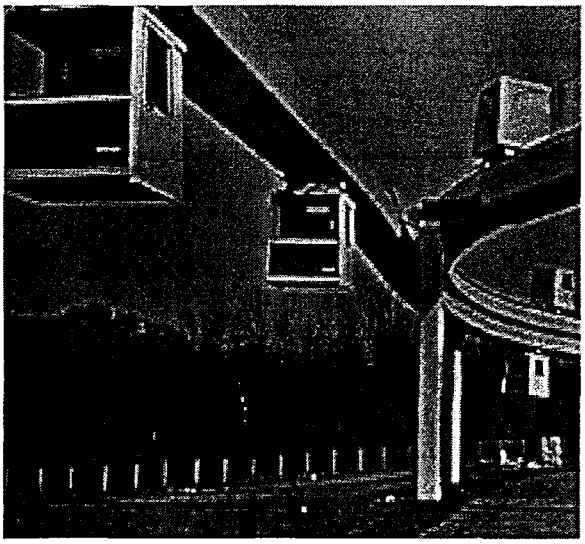


Figure 6 : système PRT Cabintaxi

(Véhicule porté et suspendu)

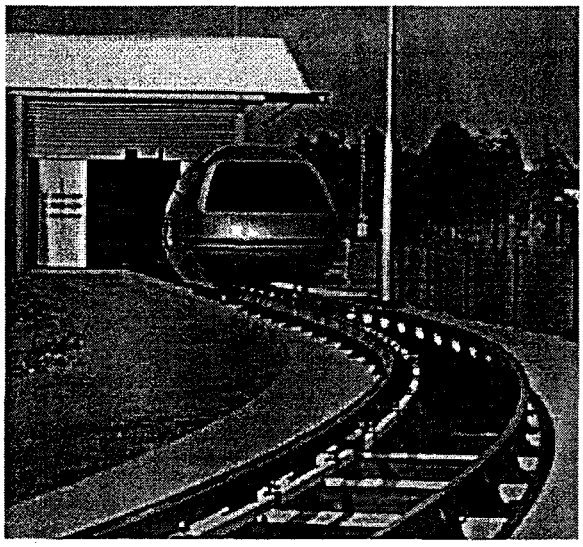


Figure 7 : système PRT Austrans

(Infrastructure rails)

- Communication

Les infrastructures supportent les divers systèmes de communication utilisés notamment les systèmes de communication sans fil.

4 - Les passagers

4 – 1 Description physique

Le système PRT est prévu pour offrir un service de déplacement aux passagers et à leurs bagages. Par ailleurs, les stations et les véhicules sont conçus pour faciliter l'accès et la descente de passagers handicapés. Notons que plusieurs projets PRT ont prévu une adaptation des véhicules afin de pouvoir assurer le transport de fret ou un transport mixte.

4 – 2 Fonctions

Les passagers définissent une destination en fonction de leur besoin de transport. Le déplacement est réalisé "à la demande". Dans certains cas, le passager peut indiquer les stations par lesquelles il veut passer, ses contraintes horaires et le nombre de véhicules nécessaires.

Remarque : il peut actionner le système d'arrêt d'urgence et signaler tout incident survenant lors de son trajet. Cependant cet aspect n'est pas abordé dans ce travail.

Bibliotheque..Universitaire..de..Valenciennes

00900788