



**HAL**  
open science

# Approches intelligentes pour le pilotage adaptatif des systèmes en flux tirés dans le contexte de l'industrie 4.0

Nesrine Azouz

► **To cite this version:**

Nesrine Azouz. Approches intelligentes pour le pilotage adaptatif des systèmes en flux tirés dans le contexte de l'industrie 4.0. Autre [cs.OH]. Université Clermont Auvergne [2017-2020], 2019. Français. NNT : 2019CLFAC028 . tel-02401701

**HAL Id: tel-02401701**

**<https://theses.hal.science/tel-02401701>**

Submitted on 10 Dec 2019

**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.

Université Clermont Auvergne  
École doctorale des Sciences Pour l'Ingénieur

THÈSE

En vue de l'obtention du grade de

**Docteur d'Université**

**Spécialité: INFORMATIQUE-PRODUCTIQUE**

*Présentée et soutenue publiquement par*

**Nesrine AZOUZ**

---

**Approches intelligentes pour le pilotage adaptatif  
des systèmes en flux tirés dans le contexte de  
l'industrie 4.0**

---

Thèse dirigée par **Henri PIERREVAL**

Préparée au LIMOS - UMR CNRS 6158

Soutenue publiquement le 28 Juin 2019

**Jury**

Mme. Hind EL HAOUAZI	Maitre de conférences, HDR à l'université de Lorraine
M. David GOLDSMAN	Professeur à H. Milton Stewart School of Industrial and Systems Engineering at the Georgia Institute of Technology. USA
Mme. Maria DI MASCOLO	Directrice de recherche CNRS – Laboratoire G-SCOP Grenoble
M. Jean-Philippe GAYON	Professeur à l'ISIMA Clermont-ferrand
M. Henri PIERREVAL	Professeur à SIGMA Clermont



# Résumé

De nos jours, de nombreux systèmes de production sont gérés en flux « tirés » et utilisent des méthodes basées sur des « cartes », comme : Kanban, ConWIP, COBACABANA, etc. Malgré leur simplicité et leur efficacité, ces méthodes ne sont pas adaptées lorsque la production n'est pas stable et que la demande du client varie. Dans de tels cas, les systèmes de production doivent donc adapter la tension de leur flux tout au long du processus de fabrication. Pour ce faire, il faut déterminer comment ajuster dynamiquement le nombre de cartes (ou de 'e-card') en fonction du contexte. Malheureusement, ces décisions sont complexes et difficiles à prendre en temps réel. De plus, dans certains cas, changer trop souvent le nombre de cartes kanban peut perturber la production et engendrer un problème de nervosité. Les opportunités offertes par l'industrie 4.0 peuvent être exploitées pour définir des stratégies intelligentes de pilotage de flux permettant d'adapter dynamiquement ce nombre de cartes kanban.

Dans cette thèse, nous proposons, dans un premier temps, une approche adaptative basée sur la simulation et l'optimisation multi-objectif, capable de prendre en considération le problème de la nervosité et de décider de manière autonome (ou d'aider les gestionnaires) quand et où ajouter ou retirer des cartes Kanban. Dans un deuxième temps, nous proposons une nouvelle approche adaptative et intelligente basée sur un réseau de neurones dont l'apprentissage est d'abord réalisé hors ligne à l'aide d'un modèle numérique jumeau (simulation), exploité par une optimisation multi-objectif. Après l'apprentissage, le réseau de neurones permet de décider en temps réel, quand et à quelle étape de fabrication il est pertinent de changer le nombre de cartes kanban. Des comparaisons faites avec les meilleures méthodes publiées dans la littérature montrent de meilleurs résultats avec des changements moins fréquents.

Mots-clés :

Industrie 4.0, Pilotage adaptatif des systèmes en flux tirés, Réseau neurone artificiel, Optimisation multi-objectif, Simulation à événements discret, Nervosité, aide à la décision, extraction de connaissances, apprentissage automatique, Optimisation via simulation, Kanban, ConWIP.

# Remerciements

To initiate a venture is relatively easy,  
it is enough to invigorate the fire of enthusiasm,  
to persevere on the venture until success is a different thing,  
that requires continuity and effort  
(Inspired by Thomas Lombardo 2011)

Je voudrais remercier en premier lieu mon directeur de thèse, Monsieur Henri PIERREVAL, professeur des universités à SIGMA Clermont, pour m'avoir accueillie au sein de son équipe. Je lui suis également reconnaissante pour le temps conséquent qu'il m'a accordé, pour toute son aide, ses conseils avisés et son écoute qui ont été prépondérants pour la bonne réussite de cette thèse. Je suis ravie d'avoir travaillé en sa compagnie car outre son appui scientifique, il a toujours été là pour me soutenir et me conseiller au cours de l'élaboration de cette thèse.

Je tiens à remercier chaleureusement tous les membres du jury d'avoir consacré du temps à ce travail et d'avoir accepté de le juger. J'étais, particulièrement et énormément, touchée par votre réponse rapide et enthousiaste.

J'adresse mes plus vifs remerciements à Monsieur David GOLDSMAN, professeur à l'institut des technologies de Georgia Tech, pour m'avoir fait l'honneur d'étudier ce travail et d'être le rapporteur de cette thèse.

De même, je suis particulièrement reconnaissante à Madame Hind EL HAOUZI, maître de conférences, HDR à l'université de Lorraine, pour l'intérêt qu'elle a porté à mes travaux de thèse depuis notre première rencontre aux journées scientifiques du GDR-MACS. Je la remercie pour m'avoir fait l'honneur d'étudier mes travaux de thèse en qualité de rapporteur et pour le temps qu'elle y a consacré.

J'associe à ces remerciements Monsieur Jean-Philippe GAYON, professeur et directeur des études à l'ISIMA de l'Université Clermont Auvergne, de m'avoir fait l'honneur d'examiner ce travail et pour le temps qu'il y a consacré.

Je remercie également Madame Maria DI MASCOLO, directrice de recherche CNRS au Laboratoire G-SCOP Grenoble, de m'avoir fait l'honneur d'examiner ce travail et pour le temps qu'elle y a consacré.

Je souhaite remercier aussi Monsieur Nasser MEBARKI qui a participé aux deux comités de suivi de cette thèse et dont les remarques constructives m'ont permis de mieux avancer.

Je remercie l'ensemble des membres du projet européen Erasmus Mundus EGOV-TN pour leur aide et soutien financier.

Je tiens à remercier en particulier Monsieur Alexandre CABAGNOL qui m'a accueillie le premier jour à Clermont Ferrand et veillé toujours à ce que je sois bien installée.

Mes remerciements vont aussi à mes anciens enseignants et encadrants de mastère Monsieur Lamjed BAN SAID et Slim BECHIKH qui m'ont non seulement initiée à la recherche mais m'ont aussi prodigué de nombreux conseils pour bien débiter cette thèse.

Je tiens également à remercier le laboratoire LIMOS et SIGMA-Clermont pour l'accueil et les conditions de travail privilégiées qui m'ont été offertes. Merci à tout le personnel de SIGMA Clermont, de l'ISIMA et du LIMOS que j'ai côtoyé pendant ces années.

Une thèse est un moment particulier dans la vie qui n'est pas toujours facile. Je ne pourrais donc jamais oublier toutes ces personnes qui ont cru en moi et qui, chacune à sa façon, m'ont permis d'arriver au bout de cette thèse. Un grand merci à tous ceux qui me demandaient sans cesse... quand est-ce que tu la soutiens cette thèse ? Parce que, sans le savoir, vous m'avez encouragée à persévérer.

Merci à tous mes amis que j'ai eu le plaisir de côtoyer durant ces quelques années à Clermont Ferrand et qui ont toujours été là pour me soutenir, à savoir, Aicha, Anissa, Bassem, Essia, Olfa, Seif, Lazher, Zinou, Asma, Lorena, Jean-Baptiste, Hani et Béatrice BOURDIEU.

Je remercie également Sameh pour sa relecture méticuleuse bien que souvent pénible de ce manuscrit. Merci pour tes corrections et pour m'avoir aidée à clarifier mes idées et préciser mon propos.

Mes plus grands remerciements vont bien évidemment à mon mari bien aimé, Mohamed ABDELADHIM, sans lequel cette thèse n'aurait jamais vu le jour, pour m'avoir supportée et m'avoir permis de me lever motivée, le cœur léger et l'esprit tranquille depuis le début de ma thèse. Très humblement, je voudrais te dire merci pour ton soutien pendant mes périodes de doutes et pour tes multiples encouragements perpétuels. Tu as su trouver les mots justes dans les moments les plus difficiles. Est-ce un bon endroit pour dire ce genre de choses ? Je n'en connais en tous cas pas de mauvais. Je t'aime.

Enfin, les mots les plus simples étant les plus forts, j'adresse toute mon affection à mes parents Ridha et Nabiha, ma sœur Nouha et mes frères Wael et Mohamed Amine qui ont toujours été là pour moi. Leur soutien inconditionnel et leurs encouragements sont pour moi les piliers fondateurs de ce que je suis et de ce que je fais. Merci pour avoir fait de moi ce que je suis aujourd'hui.

Merci à mes beaux-parents, Abdelfateh et Mabrouka, ma belle-sœur Amal, mon beau-frère Djo et Mima qui m'ont reçue les bras ouverts. Ils ont toujours été là pour m'encourager et ont manifesté de l'intérêt pour ce que je fais et ce fut important pour moi de sentir que j'avais leur soutien et leurs encouragements.

Une pensée pour terminer ces remerciements pour mes grands-parents qui n'ont pas vu l'aboutissement de mon travail mais je sais que vous en auriez été très fiers!!!

# Table des matières

Résumé.....	i
Remerciements.....	ii
Introduction Générale.....	1
Chapitre 1. Pilotage de systèmes de production en flux tiré.....	7
1. Introduction.....	8
2. La philosophie du juste à temps.....	9
3. La méthode Kanban.....	10
3.1. Le principe de la méthode Kanban.....	10
3.2. Types de carte kanban.....	12
3.3. Fonctionnement du système kanban à simple carte.....	12
3.4. Extensions de la méthode Kanban.....	14
3.4.1. La méthode Kanban spécifique.....	14
3.4.2. La méthode ConWIP.....	15
3.4.3. COBACABANA.....	16
3.4.4. Autres.....	16
3.5. Dimensionnement d'un système Kanban.....	17
4. Pilotage adaptatif des systèmes en flux tirés.....	18
4.1. Méthodes de pilotage adaptatif basé sur des prévisions.....	19
4.2. Méthodes de pilotage adaptatif en temps réel.....	20
4.2.1. Cartes de contrôle ou seuils.....	20
4.2.2. L'intelligence artificielle.....	21
5. Conclusion.....	22
Chapitre 2. Evolution des E-Kanban dans l'industrie 4.0.....	25
1. Introduction.....	26



2.	Industrie 4.0.....	26
2.1.	Historique.....	26
2.2.	Les technologies clés de l'industrie 4.0.....	28
3.	Evolution du pilotage en flux tirés des systèmes de production.....	31
3.1.	Kanban électroniques « E-Kanban ».....	31
3.2.	Implémentations de l'E-Kanban.....	31
3.2.1.	Les technologies utilisées.....	31
3.2.2.	Implémentation industrielle de l'E-Kanban.....	32
3.2.3.	Approches de pilotage en flux tirés via l'E-Kanban.....	34
3.3.	Les avantages de l'implémentation de l'E-Kanban.....	34
4.	Conclusion.....	35
Chapitre 3. Optimisation multi-objectif via simulation pour le pilotage adaptatif.....		37
1.	Introduction.....	38
2.	Principes généraux de l'approche.....	41
3.	Le problème de la nervosité.....	44
3.1.	La nervosité dans les systèmes de production.....	44
3.2.	La nervosité dans les approches de pilotages des systèmes en flux tirés.....	45
4.	Résolution par hybridation de l'optimisation Multi-objectif et de la simulation.....	48
4.1.	Prise en compte de la nervosité : période gelée adaptative.....	48
4.2.	Stratégie de pilotage.....	49
4.3.	Formulation mathématique du problème : Optimisation Multi-objectif.....	52
4.4.	Usage de la simulation à évènement discret.....	55
5.	Implémentation.....	56
5.1.	NSGAI : vocabulaire et principe de fonctionnement.....	56
5.2.	Application de NSGA-II au problème d'adaptation dynamique.....	58
5.2.1.	Représentation vectorielle des solutions.....	58

5.2.2.	Initialisation.....	59
5.2.3.	Croisement.....	60
5.2.4.	Mutation .....	61
5.2.5.	Procédure de réparation.....	61
6.	Application de l’approche à un système Kanban adaptatif .....	62
6.1.	Description du système .....	62
6.2.	Module décisionnel : fonctionnement de la stratégie du pilotage du système étudié.....	64
6.3.	Conditions des expérimentations.....	65
6.4.	Résultats .....	68
6.4.1.	Scénario 1 .....	69
6.4.2.	Scénario 2.....	70
7.	Conclusion.....	73
Chapitre 4. Apprentissage à partir du jumeau numérique et l’optimisation multi-objectif pour le pilotage adaptatif dans l’industrie 4.0 .....		
		74
1.	Introduction .....	75
2.	Apprentissage à partir d’un jumeau numérique.....	76
3.	Approche proposée : Apprentissage à partir du jumeau numérique et l’optimisation Multi-objectif	79
3.1.	Principe général de l’approche .....	79
3.2.	Apprentissage du réseau de neurones par simulation et optimisation multi-objectif .....	81
4.	Implémentation.....	84
4.1.	Optimisation Multi-objectif.....	84
4.1.1.	Représentation vectorielle des solutions .....	85
4.1.2.	Initialisation.....	85
4.1.3.	Opérateurs génétiques de reproduction .....	86
4.2.	Implémentation logicielle de l’approche .....	87

5. Exemples d'application .....	88
5.1. Cas du système ConWIP .....	89
5.1.1. Description du système .....	89
5.1.2. Configuration du MLP-NN .....	90
5.1.3. Conditions des expérimentations.....	91
5.1.4. Résultats .....	93
5.2. Cas du système Kanban.....	96
5.2.1. Description du système .....	96
5.2.2. Configuration du MLP-NN .....	96
5.2.3. Conditions des expérimentations.....	97
5.2.4. Résultats .....	99
6. Conclusion.....	101
Conclusions et perspectives.....	103
Bibliographie.....	109

## Liste des Figures

Figure 1: Organisation du manuscrit.....	5
Figure 2: Systèmes de contrôle en flux tirés et flux poussés .....	11
Figure 3: Le fonctionnement du système kanban à simple carte. ....	13
Figure 4: Système de contrôle en flux tiré de type Kanban .....	14
Figure 5: Système de contrôle en flux tiré de type ConWIP.....	15
Figure 6: Les quatre révolutions industrielles .....	27
Figure 7: Les technologies clés de l'industrie 4.0.....	28
Figure 8: Le système E-kanban de Toyota (Kotani 2007) .....	33
Figure 9: Interaction entre la composante physique et le module décisionnel.....	42
Figure 10: Exemples de signaux des variations de la demande clients.....	43
Figure 11: Couplage de la simulation et de l'optimisation multi-objectif .....	44
Figure 12: La conséquence du délai entre une décision et son effet .....	46
Figure 13: Effet de la période gelée .....	49
Figure 14: Un système en flux tiré à plusieurs stations.....	50
Figure 15: Optimisation multi-objectif des paramètres de pilotage et évaluation via simulation..	56
Figure 16: Schéma de l'évolution de l'algorithme NSGA-II .....	58
Figure 17: Représentation génétique d'un chromosome .....	59
Figure 18: Un exemple de croisement uniforme .....	61
Figure 19: Exemple d'application de la procédure de réparation .....	62
Figure 20: Patterns utilisés pour la variation de la demande dans les deux systèmes étudiés .....	63
Figure 21: La convergence de la mesure de performance des coûts de production durant l'optimisation pour notre approche et l'approche proposée dans (Sivakumar et Shahabudeen, 2008).....	69
Figure 22: Une vue d'ensemble durant l'évolution de NSGA-II .....	70
Figure 23: Le front de Pareto obtenu par notre approche .....	71
Figure 24: Apprentissage du réseau de neurones à partir d'ensemble généré par simulation .....	77
Figure 25: L'interaction entre le MLP-NN, la simulation et l'optimisation multi-objectif .....	80
Figure 26: Diagramme de séquence décrivant les interactions dans notre approche.....	81
Figure 27: La transmission des informations dans le MLP-NN pour la prise de décision .....	82

Figure 28: Un exemple d'une représentation vectorielle d'un MLP-NN.....	85
Figure 29: Un exemple de croisement un-point .....	86
Figure 30 : La connexion entre Arena, NSGA-II et MLP-NN.....	88
Figure 31: Le modèle ConWIP étudié (Tardif et Maaseidvaag, 2001).....	90
Figure 32: Patterns triangulaires utilisés pour la variation de la demande clients .....	90
Figure 33: La configuration du MLP-NN dans le cas d'un système ConWIP .....	91
Figure 34: Les courbes de la convergence du coût de production durant l'apprentissage selon les deux stratégies d'initialisation dans le cas d'un système ConWIP .....	94
Figure 35: Le front de Pareto obtenu par notre approche dans le cas du système ConWIP .....	95
Figure 36: Le modèle Kanban étudié (Sivakumar et Shahabudeen, 2008).....	96
Figure 37: La configuration du MLP-NN dans le cas d'un système Kanban.....	97
Figure 38: Les courbes de la convergence du coût de production durant l'apprentissage selon les deux stratégies d'initialisation dans le cas d'un système Kanban.....	100
Figure 39: Le front de pareto dans le cas d'un système Kanban .....	100

## Liste des Tableaux

Tableau 1: Les différentes causes de la nervosité. ....	45
Tableau 2: Les valeurs des paramètres de NSGA-II. ....	66
Tableau 3: Les paramètres utilisés dans les expérimentations .....	67
Tableau 4: Les valeurs des paramètres de NSGA-II pour le problème d'apprentissage .....	92
Tableau 5: Les paramètres des approches de comparaisons .....	93
Tableau 6: Résultats des meilleures performances obtenues pour le cas d'un système ConWIP ..	94
Tableau 7: Les paramètres des systèmes Kanban à optimiser .....	99
Tableau 8: Résultats des meilleures performances obtenues pour le cas d'un système Kanban ....	99

## Liste des abréviations

MRP	Materials Requirements Planning
JAT	Juste à Temps
JIT	Just In Time
APICS	American Production and Inventory Control Society
OD	Ordre de Déplacement
OP	Ordre de Production
ConWIP	Constant Work In Process
COBACABANA	Control Of BALance by CARd BAsed NAVigation
WIP	Work In Process
TIC	Technologie de l'Information et de la Communication
IaaS	Infrastructure as a Service
PaaS	Platform as a Service
SaaS	Software as a Service
IoT	Intenet of Things
IIoT	Industrial Intenet of Things
CPS	Cyber Physical System
MAC	Multi-Agent Control
HMS	Holonic Manufacturing System
PID	Proportionnel, Intégral et Dérivé
NSGA-II	Non Dominated Sorting Genetic Algorithm
NN	Neural Network
MLP-NN	Multi-Layer Perceptron Neural Network
GA	Genetic Algorithm

---

# Introduction Générale

Beaucoup d'entreprises se tournent vers des stratégies de pilotage de type Juste À Temps, reposant sur une production tirée par la demande du client. Elle consiste à fournir au client le produit demandé, au moment demandé et dans la quantité demandée, tout en maîtrisant le niveau d'en-cours tout au long du processus de fabrication. Un des moyens les plus connus de mise en œuvre le JAT est la méthode Kanban, qui est un outil de pilotage reposant sur l'association à tout produit (ou conteneur de produits) une carte kanban dont le nombre fixé permet de réguler le flux de produit et par conséquent le niveau d'en-cours de production. Malgré sa simplicité et son efficacité, cette méthode est adaptée à un environnement de production déterministe avec une demande client stable et des temps de processus constants, ce qui souvent peut ne pas être le cas.

Aujourd'hui, de nombreux systèmes de production sont sujets à la complexité et l'imprévisibilité de l'évolution dynamique et changeante de l'environnement de production et des demandes client. Ces systèmes de production doivent souvent faire face à des problèmes opérationnels tels qu'une annulation ou modification de commande, la prise en compte d'une commande urgente ou encore des aléas de production (pannes, etc.). Au niveau stratégique, ils sont aussi appelés à s'adapter aux mutations technologiques et à suivre les évolutions du marché. Afin de faire face à ces difficultés, les systèmes de production doivent réguler la tension de leur flux de produit tout au long du processus de fabrication. Pour ce faire, ils doivent ajuster dynamiquement le nombre de cartes kanban, en ajoutant ou retirant des cartes de la ligne de fabrication afin d'améliorer les performances (encours, satisfaction des clients, etc.). Malheureusement, ces décisions sont complexes et demandent une capacité à prendre en compte un environnement de production où les incertitudes et les variabilités sont importantes et rapides. De plus, la difficulté de prendre de telles décisions ne peut que s'amplifier si l'on cherche à les prendre en temps réel. Dans ce cas, les gestionnaires doivent prendre des décisions en fonction des informations venant de l'environnement de production et de la ligne de fabrication en temps réel. Mais comment peuvent-ils déterminer quand et dans quelle étape de la ligne de fabrication il faut changer le



---

nombre de cartes kanban ? De plus, une fois le besoin détecté et la décision prise, comment savoir si cette décision est cohérente ou pas avec les décisions passées et celles qui devront être prises par la suite?

En effet, la prise de décision est un processus cognitif qui vise à sélectionner parmi différentes alternatives pour n'en garder qu'une. Prendre les meilleures décisions peut s'avérer difficile en fonction de la situation, des risques et des conséquences potentielles. Par exemple, dans certaines industries, la carte kanban peut être attachée à un conteneur de produit. Par conséquent, changer le nombre de cartes kanban revient à changer le nombre de conteneurs. Cela peut engendrer des coûts supplémentaires et avoir des effets indésirables sur l'environnement de production, ce qui n'est pas souvent le cas si la carte kanban est attachée à un produit. De plus, la complexité et la multitude des informations alliées à la rapidité et l'imprévisibilité des changements de l'environnement rendent la prise de décision de plus en plus délicate, d'où la nécessité de se faire assister par des moyens adéquats capables d'aider à la prise de décision.

Cette aide à la décision peut se faire, par exemple, avec des règles de contrôle basées sur des seuils qui déterminent selon l'état actuel du système de production, la décision à prendre afin d'améliorer les performances (encours, satisfaction des clients, etc.). Malgré la simplicité de ces règles à mettre en œuvre dans un environnement de production, elles restent limitées et dépendantes des paramètres qui sont difficiles à déterminer, surtout si l'on prend en considération les caractéristiques de l'environnement de production qui est évolutif et incertain ainsi que les spécificités de chaque système : une règle performante pour un système ne l'est pas forcément pour un autre. Comment peut-on alors déterminer ces paramètres tout en prenant en considération l'évolution dynamique et changeante de l'environnement de production lors de la prise de décision ? et comment peut-on savoir si une règle de contrôle (ou une stratégie de pilotage adaptatif) est performante ou non selon le type du système ?

Une autre solution consiste à faire appel aux techniques de l'intelligence artificielle et plus particulièrement aux méthodes d'apprentissage automatiques afin de piloter dynamiquement les systèmes en flux tirés adaptatifs. Dans la plupart de ces approches, l'objectif est d'extraire des connaissances à partir d'exemples fournis par des experts du domaine, des observations disponibles ou des bases de données. Cependant, dans le contexte des systèmes de production, il est

---

malheureusement difficile de trouver de bons exemples indiquant comment ajuster dynamiquement le nombre de cartes kanban. En plus, Les experts n'ont pas toutes les connaissances requises et ne sont donc pas en mesure de fournir des exemples ou des cas (suffisamment représentatifs) à partir desquels il serait possible d'apprendre. Ceci revient à l'état changeant dynamiquement au cours du temps de ces types de systèmes, qui rend difficile la détermination des « bonnes pratiques ». De plus, puisque plusieurs décisions sont prises et donc s'enchaînent au cours du temps, c'est cet enchaînement de décisions qui s'avère être performant ou non, et non pas chaque décision individuellement. C'est cet aspect dynamique du problème et des systèmes de production eux-mêmes qui rend difficile la tâche d'évaluation intrinsèque à la génération d'exemples pour l'apprentissage. Comment alors peut-on apprendre au système la façon dont il doit adapter dynamiquement le nombre de cartes Kanban dans les systèmes en flux tirés adaptatifs, sans la disponibilité d'une base d'apprentissage afin d'améliorer les performances du système de production ?

Aujourd'hui, à l'ère de l'industrie 4.0, une large communauté de chercheurs et d'industriels témoignent de l'intérêt de l'exploitation des nouvelles opportunités offertes par cette nouvelle révolution. Plus particulièrement, les technologies des objets connectés et intelligents offrent des perspectives d'innovation prometteuses. Nous citons, parmi celles-ci, la possibilité de connaître à tout moment et à chaque étape de la ligne de fabrication le contexte de production : l'emplacement, le statut, et ce qui est arrivé aux produits et aux ressources. La collecte, le stockage et la communication d'informations précises et en temps réel pour chacun des produits. Le recueil de toutes ces données permet de construire une réplique virtuelle de la ligne de fabrication ou de l'usine, ce qu'on appelle l'usine virtuelle ou le jumeau numérique, qui permettra non seulement de suivre et piloter le système de production, mais également de le simuler pour en optimiser le fonctionnement. Comment peut-on alors exploiter ces nouvelles opportunités offertes par l'industrie 4.0 pour mieux ajuster les flux dans un environnement de production en flux tiré dynamique et incertain ?

La modernisation de l'outil productif et l'exploitation des nouvelles opportunités offertes par l'industrie 4.0 sont clairement devenues l'objectif de nombreuses entreprises afin d'améliorer leur productivité, leur flexibilité et leur réactivité aux variations de l'environnement. Cependant, cela nécessite l'acquisition de nouvelles compétences, de nouvelles technologies et beaucoup

---

d'investissements, ce qui n'est pas évident et possible pour la plupart des entreprises dans la réalité. Par conséquent, l'émergence de ce type d'entreprises vers l'industrie 4.0 nécessite beaucoup de temps. Dans ce cas, comment peut-on améliorer le pilotage adaptatif des flux de production en temps réel, sans recours aux nouvelles technologies, dans un contexte plus réaliste et éviter les hypothèses restrictives ?

Pour répondre à ces questions, nous proposerons dans cette thèse deux approches de pilotage adaptatif et intelligent des systèmes en flux tirés capables de prendre en considération les aléas de fabrication et les variations dynamiques et changeantes de la demande clients afin de décider en temps réel quand et dans quelle étape de fabrication il faut changer le nombre de cartes kanban.

Dans une première partie, nous proposerons une approche d'optimisation multi-objectif via simulation.

Puis, dans une deuxième partie, nous proposerons une approche appliquée à un environnement de production 4.0 doté d'un jumeau numérique. Cette approche est basée sur l'apprentissage automatique d'un réseau de neurone à partir de la simulation et de l'optimisation multi-objectif.

Afin d'atteindre ces objectifs, ces travaux de thèse ont été organisés en quatre chapitres, qui développent chacun des points particuliers s'inscrivant dans l'une ou l'autre des deux approches proposées (comme décrit sur la Figure 1).

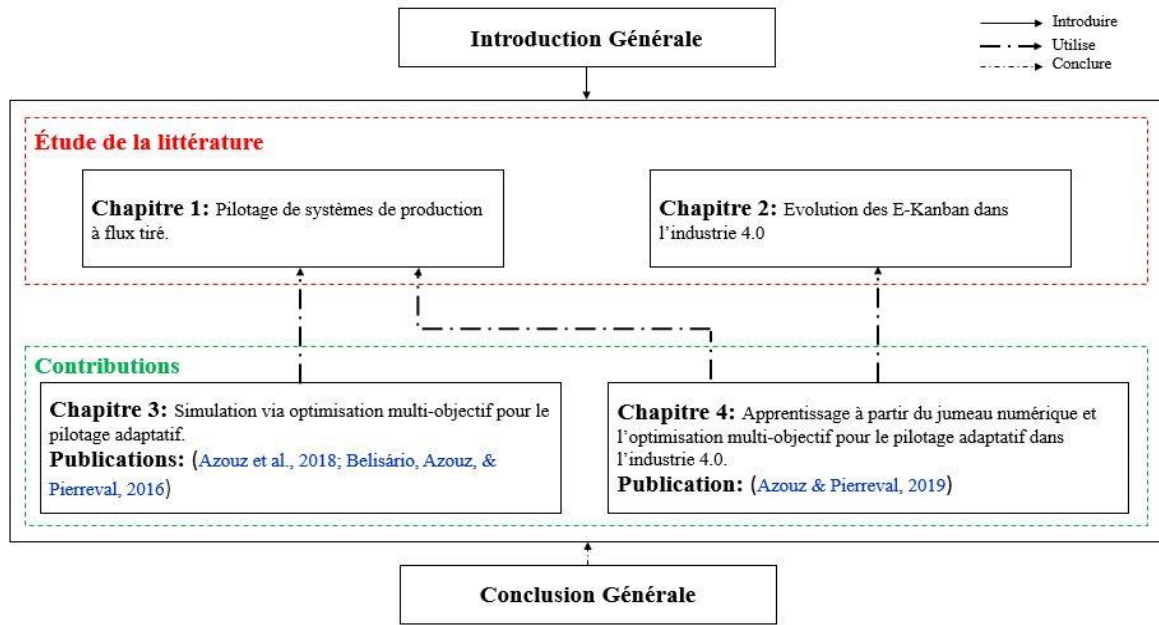


Figure 1: Organisation du manuscrit

Dans le premier chapitre, nous présenterons les problématiques liées au pilotage adaptatif des systèmes en flux tirés. Pour ce faire, nous introduirons, dans une première partie, le cadre général dans lequel s'inscrit notre travail et nous nous intéresserons aux systèmes basés sur une production en flux tiré qui fonctionnent à base de cartes. Dans une deuxième partie, nous présenterons différentes stratégies de pilotages adaptatifs qui ont été proposées dans la littérature afin d'ajuster dynamiquement le nombre de cartes kanban. Pour conclure ce chapitre, nous analyserons les différentes méthodes utilisées dans ces approches ainsi que leurs différents objectifs. Cette démarche globale permettra de mieux nous positionner par rapport aux différentes contributions existantes.

Dans le deuxième chapitre, nous présenterons la nouvelle révolution industrielle, l'industrie 4.0, et nous nous intéresserons à l'évolution des systèmes Kanban dans le contexte de l'industrie 4.0. Par la suite, nous décrirons quelques travaux de référence qui ont été proposés dans la littérature pour améliorer le pilotage adaptatif des systèmes de production en flux tirés dans le contexte de l'industrie 4.0. L'objectif de ce chapitre est de présenter les différentes opportunités offertes par l'industrie 4.0 dans le domaine du pilotage des systèmes en flux tirés.

---

Dans le troisième chapitre nous proposerons notre première approche d'optimisation multi-objectif via simulation. Pour ce faire, nous formaliserons, dans un premier temps, le problème du pilotage adaptatif dans un environnement dynamique et incertain. Par la suite, nous présenterons la solution adéquate pour résoudre ce problème. Enfin, afin de montrer la faisabilité de l'approche proposée, nous l'illustrerons sur un système Kanban et nous montrerons son intérêt en la comparant avec une approche de pilotage adaptatif qui a été largement utilisée dans la littérature.

Le dernier chapitre sera consacré à notre approche d'apprentissage automatique du réseau de neurone à partir d'un jumeau numérique et de l'optimisation multi-objectif. Nous montrerons dans ce chapitre comment on peut exploiter les données collectées à partir des capteurs pour mieux ajuster les flux dans un environnement de production dynamique et incertain. Afin de montrer la faisabilité de notre approche et illustrer son potentiel, nous l'appliquerons sur différents types de système en flux tirés. Puis, nous les comparerons à différentes approches de pilotages adaptatifs des systèmes en flux tirés bien reportées dans la littérature.

# Chapitre 1.

## Pilotage de systèmes de production en flux tiré

## 1. Introduction

Pour rester performants et compétitifs, les systèmes de production doivent être capables de s'adapter pour faire face à de nombreuses perturbations dues à ses caractères stochastiques ou encore aux changements externes tels que l'évolution de la demande des clients. Il leur est alors essentiel de pouvoir déterminer quand et comment s'adapter (capacités, etc.). Malheureusement, de tels problèmes sont connus pour être difficiles.

La prise en compte et la gestion des perturbations représentent des enjeux majeurs et des défis à relever lors du pilotage des systèmes de production de biens et de services. En effet, le pilotage de la production peut être défini comme l'ensemble des consignes pertinentes à donner à un système soumis à des perturbations pour atteindre un objectif donné. Ces consignes répondent principalement aux questions « quand, où, et que produire ? ».

Contrairement à la méthode MRP qui gère les perturbations et l'incertitude par une amélioration des prévisions, la philosophie du Juste A Temps (JAT) aborde le même problème par la recherche de réactivité en temps réel. Pour ce faire, elle repose sur une logique dite de production en flux tiré dans laquelle la production d'un produit est déclenchée par la demande effective des clients.

Dans cette thèse, nous nous intéresserons plus particulièrement aux systèmes utilisant une production en flux tirés qui fonctionnent à base de cartes. Dans ce type de système, les cartes autorisent et contrôlent ainsi la production. Dans un système dit traditionnel, l'optimisation consiste à trouver le nombre de cartes qui minimise les coûts de production. Une telle optimisation part de l'hypothèse que la demande est stable, ce qui est rarement le cas avec les marchés et les besoins des clients qui changent. Ceci est dû à la complexité même des systèmes de production qui se traduit généralement par leurs aspects combinatoire, stochastique, dynamique ou encore multi-objectif. Ces difficultés ont conduit de nombreux chercheurs à proposer des approches de pilotage adaptatives pour l'adaptation dynamique du nombre de cartes via des cartes dites supplémentaires afin de rendre ces systèmes plus réactifs aux changements et donc performants sur le long terme. Malgré les recherches, le développement de tels systèmes reste encore un défi scientifique.

Dans ce chapitre, nous présenterons, dans une première partie, le cadre général dans lequel s'inscrit notre travail. Nous introduirons les principaux concepts liés à la philosophie du JAT. Puis, nous nous intéresserons aux systèmes basés sur une production en flux tiré qui fonctionnent à base de cartes. Pour cela, nous expliquerons, donc, leur principe de fonctionnement ainsi que leurs différents types. Ces parties font référence à de nombreuses notions de base et introduisent un ensemble de concepts et de définitions. Dans une deuxième partie, nous nous intéresserons aux différentes stratégies de pilotages adaptatifs qui ont été proposées dans la littérature afin d'ajuster dynamiquement le nombre de cartes kanban. Pour cela, nous analyserons, alors, les différentes méthodes utilisées dans ces approches ainsi que leurs différents objectifs. Cette démarche globale permettra de mieux nous positionner par rapport aux différentes contributions existantes.

## **2. La philosophie du juste à temps**

Le JAT est une traduction littérale du terme anglais « Just In Time » (JIT). Etymologiquement, ce terme signifie "juste, à temps". On y retrouve deux concepts : un concept de quantité qui consiste à avoir seulement la quantité nécessaire dont on a besoin et un concept de temps qui nécessite d'avoir la quantité demandée au moment voulu.

Le principe du JAT consiste à fournir au client la quantité de produits demandée au moment demandé en respectant le standard de qualité et des coûts fixés. Cela implique, en premier lieu, que l'ensemble de la production soit assujetti à la demande du client (interne ou externe).

L'APICS (American Production and Inventory Control Society) définit le JAT comme « une philosophie basée sur l'élimination de tous types de gaspillage et sur la mise en œuvre d'une stratégie de progrès permanent en terme de productivité ».

Cette philosophie a pour objectif l'amélioration continue de la performance industrielle et la réduction, voire l'élimination des opérations sans valeur ajoutée. Ces opérations sont considérées comme des gaspillages.

Il existe trois formes de gaspillage dans la philosophie du JAT regroupées régulièrement sous l'appellation « 3M » : les Muda, les Mura et les Muri. Muda (signifie gaspillage en



japonais) inclut les activités non créatrices de valeur. Sept catégories de Muda ont été décrites par Womack J et Jones Daniel T (1990) :

- Surproduction : produire alors qu'il n'y a pas eu de demande client.
- Temps d'attente : attendre des pièces ou une machine à cause d'un goulot d'étranglement ou pendant une interruption.
- Transports et déplacements inutiles : tous transports ou déplacements qui n'apportent pas de valeur pour le client.
- Opérations inutiles : toutes opérations ou tâches non nécessaires.
- Stocks excessifs : stocks inutiles qui occupent de l'espace et génèrent une gestion coûteuse et consommatrice de temps.
- Gestes inutiles : tous les déplacements et les gestes inutiles du personnel et qui n'apportent pas de valeur aux clients.
- Défauts/rebut : tous produits ou services non conformes aux exigences des clients entraînent la mise en rebut ou le retraitement de ces produits/ services.

Les 2 autres formes de gaspillage sont, quant à elles, souvent oubliées et sont également plus difficiles à identifier :

- Muri (ou L'excessif) : correspond au surcharge de travail (pour les hommes ou les équipements).
- Mura (ou la variabilité) : correspond aux déséquilibres dans le travail à accomplir ou dans une activité.

Afin d'atteindre l'efficacité et éliminer les différents types de gaspillage, le JAT fait appel à différentes techniques et méthodes. La méthode Kanban est l'une des méthodes les plus connues du JAT. Elle est une méthode de gestion d'atelier qu'il ne faut pas confondre avec le JAT lui-même.

### **3. La méthode Kanban**

#### **3.1. Le principe de la méthode Kanban**

La méthode Kanban est une méthode d'organisation et de gestion de production qui permet d'assurer une production JAT à chaque étape du processus. La production est réalisée au bon moment en réponse à une demande de client et non pas en avance. Elle a été fortement associée au JAT puisqu'elle a été expérimentée sur les lignes de production de Toyota Motor Company par M. Ohno à la fin des années 50.

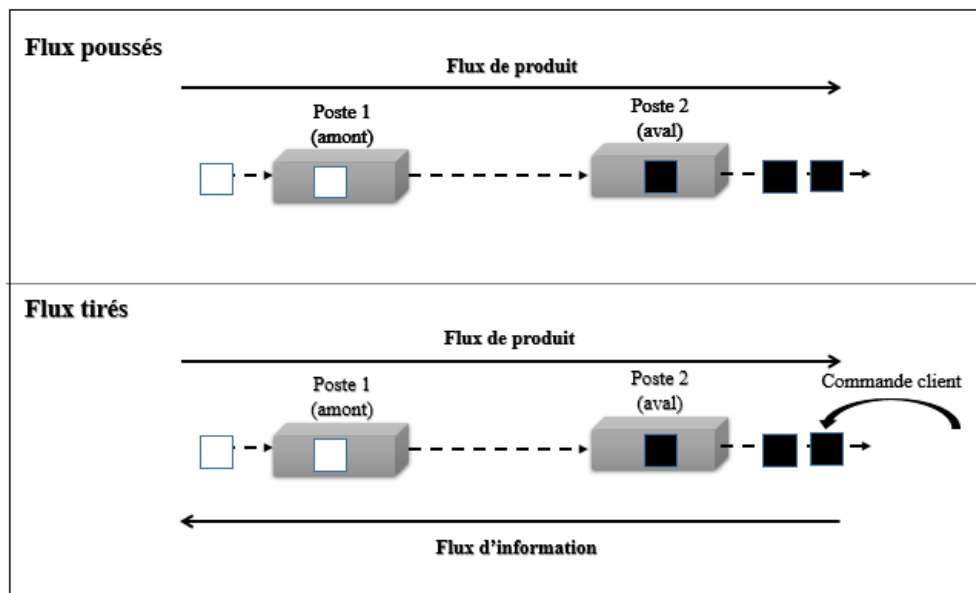


Figure 2: Systèmes de contrôle en flux tirés et flux poussés

Cette méthode repose sur un mode de pilotage de la production en flux tiré, par opposition au flux poussé qui anticipe la demande client. La Figure 2 illustre leur principe de fonctionnement. En effet, dans le système à flux poussé, le déclenchement de la production est basé sur des planifications et des prévisions pour déterminer un programme de production. Dans ce type de système, c'est la disponibilité de la machine et du produit venant de l'amont qui lancent la production. On parle alors d'une stratégie de pilotage à flux poussés. Le pilotage en flux tirés repose, quant à lui, sur une gestion par l'aval, chaque machine ne lance la production que sur demande de la machine située en aval. Le système est alors tiré par l'aval. D'une manière générale, on peut distinguer deux types de fonctionnement du pilotage en flux tiré.

- Une production sur stock (make to stock) : Elle consiste à maintenir un stock minimum spécifié de produits finis. Autrement dit, si un produit quitte le stock, un ordre de production est lancé au système afin de pouvoir le reconstituer.

- Une production sur demande (make to order) : La production est déclenchée dès la réception d'une commande client. De ce fait, ce type de fonctionnement vise à maintenir un stock de produits finis nul.

Si d'un point de vue théorique cette classification reste pertinente, son déploiement dans la réalité industrielle est moins viable. En effet, dans la pratique, une production en flux tiré est utilisée à la place de celle à flux poussé dans certain environnement et en complément à celle-ci dans beaucoup d'autres environnements. On parle, dans ce cas, d'une stratégie de pilotage dite : hybride où une production, en flux poussé et tiré, est utilisée. Par exemple, dans un système de production, les produits simples sont fabriqués en flux poussés et les produits complexes sont assemblés en flux tirés.

### **3.2. Types de carte kanban**

La méthode kanban repose sur un système d'information simple et performant pour faire monter l'information de l'aval vers l'amont. L'information est transmise par l'intermédiaire du « Kanban » (carte ou étiquette en japonais), qui est attaché à une pièce (ou plus généralement à un conteneur). Il existe deux types de cartes Kanban (Kumar et Panneerselvam 2007): une qui représente un ordre de déplacement/transport (OD) et une autre qui représente un ordre de production/fabrication (OP). La carte OD permet de transférer un produit (ou un conteneur) d'un poste de travail au poste suivant (de l'amont vers l'aval) ou d'un poste de travail à une aire de stockage. Ainsi, la carte OP permet de faire signe ou d'envoyer un ordre de production au poste en amont (de l'aval vers l'amont).

Le système kanban peut être à simple carte ou à double cartes (Huang et Kusiak 1996). Par exemple, lorsque les postes de travail se situent dans des ateliers de production distants où le recours à un magasin de stockage est nécessaire, un système kanban à double cartes (deux boucles) est alors mis en place. Dans le cas contraire, si les postes de travail se situent dans le même atelier de production, un système kanban à simple carte (simple boucle) est mis en place. Dans cette thèse, nous nous concentrerons sur les systèmes kanban à simple carte.

### **3.3. Fonctionnement du système kanban à simple carte**

La méthode Kanban vise à créer un échange simple d'informations entre les postes de travail via une ou plusieurs boucles de cartes Kanban. Dans un système à simple carte (simple boucle), un nombre de cartes kanban (OP) fixe est associé à chaque boucle de cartes.

Le parcours d'une carte kanban est illustré dans la Figure 3 :

- Lorsqu'une demande client arrive, s'il y'a des produits dans le stock final, la demande est satisfaite.
- La carte kanban est détachée du produit (ou du conteneur) qui a été consommé par le client et revient au poste amont, propageant à son tour la demande.
- A son arrivée, la carte kanban est déposée sur un tableau, le planning à kanban, en attente d'usinage.
- Lorsque le poste amont termine la fabrication du produit (ou du conteneur de produits), il lui attache la carte kanban et l'achemine vers le poste aval.

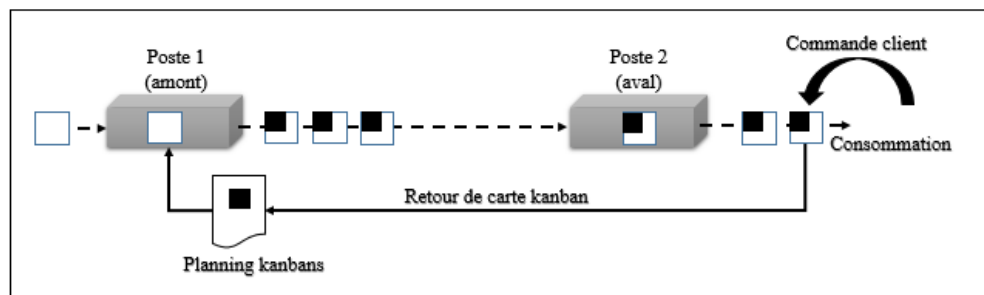


Figure 3: Le fonctionnement du système kanban à simple carte.

Cependant, si le planning des cartes kanban est vide ou si aucune pièce n'est disponible dans le stock du poste amont, la production de ce poste s'arrête. Notons qu'aucune carte kanban ne peut être libre, soit elle est attachée à un produit (ou conteneur), soit elle est placée sur le planning des carte kanban.

Le déploiement de la méthode Kanban dans le milieu industriel a donné lieu à différentes extensions incluant les systèmes Kanban, Kanban généralisé (Buzacott 1989; Duri et al. 1997; Frein, Di Mascolo, and Dallery 1995), Kanban étendu (Dallery et Liberopoulos, 2000), ConWIP, POLCA, COBACABANA, etc. Plusieurs travaux dans la littérature ont présenté des analyses avancées de ces différents systèmes (Duri et al. 2000; Berkley 1992; Framinan et al.

2003; Lage Junior et Godinho Filho 2010; Framinan et Pierreval 2012; Thürer et al. 2016; Vercraene et al. 2011). C'est sur les principaux systèmes, cités ci-dessus, que nous nous attarderons.

### 3.4. Extensions de la méthode Kanban

#### 3.4.1. La méthode Kanban spécifique

La méthode Kanban spécifique est l'une des plus connues avec une boucle de cartes pour chaque étape du processus de production. Dans cette méthode, les cartes OP, encore appelées carte kanban, sont utilisées pour limiter les encours dans chaque étape de production. La Figure 4 montre une ligne de production composée de trois machines et produisant un seul type de produit avec une gestion de production de type : Kanban spécifique. Le mouvement des cartes kanban est représenté par une ligne continue.

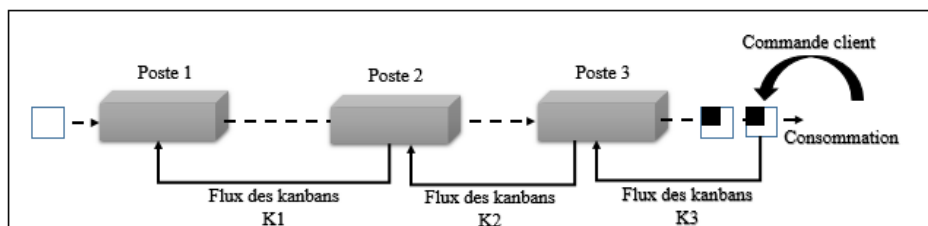


Figure 4: Système de contrôle en flux tiré de type Kanban

Cette méthode fonctionne comme suit. A chaque fois qu'un client consomme un produit fini, le Kanban K3 attaché à ce produit est libéré et transféré en amont (Poste 3). Ce Kanban autorise la production d'une nouvelle pièce au niveau du Poste 3. Si une pièce est disponible dans le stock amont du poste 3, elle est détachée du Kanban K2 et attachée au Kanban K3. A cet instant, le poste 3 lance la production d'une nouvelle pièce et le Kanban K2 est envoyé au poste 2.

La circulation des cartes kanban ne se fait qu'entre deux postes. (Kumar et Panneerselvam, 2007) affirment que le kanban fonctionne comme un système de planification à plusieurs étapes utilisant moins d'encours. Il a la capacité de gérer des volumes de production élevée et de mieux utiliser : la capacité du système de production.

La méthode Kanban spécifique dépend seulement d'un seul paramètre par étape de production, à savoir le nombre de cartes kanban (K1, K2 et K3 dans notre exemple). Ces paramètres influencent la performance du système de production (coûts, encours, satisfaction client, etc.) (Martin et al. 1998; T. M. Chang et Yih 1994).

### 3.4.2. La méthode ConWIP

La méthode ConWIP (Constant Work In Process), proposée par Spearman Mark L et Woodruff David L (1990), est apparue comme une amélioration du Kanban classique dans le cas d'une ligne de production produisant des produits très différents. Cette méthode utilise une seule boucle reliant la fin de la ligne de production avec le début de la ligne. Dans le ConWIP, les cartes nommées « carte ConWIP », sont génériques quand elles sont sur le planning carte kanban, puis associées à un type de produit quand elles entrent en début de ligne. Elles redeviennent génériques en fin de ligne quand elles sont libérées. La Figure 5 illustre la circulation des flux physiques et des flux d'informations dans la méthode CONWIP.

La méthode ConWIP fonctionne comme suit : A chaque fois qu'un client consomme un produit fini, la carte ConWIP attachée à ce produit est libérée et transférée en début de la ligne de production (poste 1). Une fois qu'une matière première est mise en production à l'entrée de la ligne de production, elle est poussée à travers le système aussi vite que possible vers la fin de la ligne. En effet, la méthode ConWIP peut être considérée comme un système de type flux mixte. Elle utilise un flux tiré à la fin de la ligne et un flux poussé dès le début de la ligne vers sa fin. Elle peut également être considérée comme une méthode Kanban avec une seule étape.

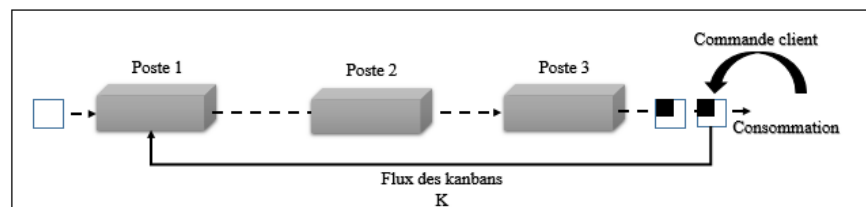


Figure 5: Système de contrôle en flux tiré de type ConWIP

D'une manière générale, l'objectif principal du ConWIP consiste, comme son nom l'indique, à contrôler la quantité d'encours présente sur une ligne de production et la maintenir

constante. Elle dépend seulement d'un seul paramètre pour contrôler toute la ligne de production, à savoir le nombre de cartes ConWIP (K dans notre exemple) (Ajourlou, Shams, and Aryanezhad 2011; Jaegler et al. 2017).

### **3.4.3. COBACABANA**

La méthode COBACABANA (Control of Balance by CArd BAsed NAvigation) a été proposée par Land (2009) puis a été améliorée par Thürer, Land et Stevenson (2014). C'est une méthode de contrôle de la charge de travail basée sur des cartes. Cette méthode a été conçue spécialement pour le contrôle des ateliers à cheminement multiples (Job shops). La charge de travail de l'atelier est contrôlée en établissant des boucles de cartes entre la validation de la commande client et chaque poste de travail dans l'atelier. COBACABANA est une méthode à double cartes (deux boucles) : une carte de production et une carte d'autorisation. La carte de production permet de faire signe ou envoyer un ordre de production aux différents postes de la ligne de production et la carte d'autorisation permet l'acceptation d'une commande client.

La méthode COBACABANA fonctionne comme suit : à l'arrivée d'une demande client, si une carte d'autorisation est disponible, la demande sera déposée sur un tableau (le planning des demandes clients) en attente de production. Une commande client ne peut être transmise à l'atelier que si elle peut obtenir toutes les cartes de production requises pour tous les postes de son acheminement. En effet, chaque poste de travail possède une carte de production différente. Lorsque la commande client est lancée dans l'atelier, chaque fois que le produit termine son traitement sur un poste de travail, la carte associée à ce poste est détachée et retournée au planning des cartes kanban. Cette carte devient alors disponible pour lancer une nouvelle commande.

### **3.4.4. Autres**

Il existe également de nombreuses adaptations du Kanban. Ainsi, 32 identifiées par Lage Junior et Godinho Filho (2010) ont été développées pour améliorer ou étendre la logique du système Kanban à des environnements plus complexes : Stock nominal, Kanban Etendu, Kanban Généralisé, POLCA, etc.

### 3.5. Dimensionnement d'un système Kanban

Le dimensionnement des paramètres kanban (nombre de cartes kanban et nombre de pièces par carte kanban) s'appuie sur les caractéristiques de chaque boucle de cartes (lot économique, capacité des conteneurs, délais de réapprovisionnement, demandes journalières, etc.). Lorsque la demande client varie de façon significativement forte, il est nécessaire de remettre rapidement en cause le paramétrage des boucles kanban dans le temps en ajustant sur le nombre de cartes kanban en circulation, dans le but d'améliorer la performance du système de production.

Deux approches différentes ont été proposées dans la littérature pour ajuster le nombre de cartes kanban en circulation (Framinan, González et Ruiz-Usano, 2006) :

- L'approche « card setting » : Permet de définir, dans des conditions de production données, le nombre de cartes kanban à mettre en circulation afin d'atteindre les performances souhaitées (niveau d'encours, taux de service, etc.).
- L'approche « card controlling » : Permet de définir des règles pour ajuster dynamiquement le nombre de cartes kanban en circulation, en fonction de certains événements (comme par exemple une variation de la demande client, une panne de machine, etc.).



Plusieurs travaux ont traité le problème de « card setting », à commencer par Yasuhiro (1983), qui a établi la formule suivante permettant de calculer le nombre optimal de cartes kanban à mettre en circulation :

$$Nbr_{Kanbans} = \frac{CMJ * (t_w + t_{pc}) * S}{UC}$$

Avec :

- CMJ : consommation moyenne journalière prévisionnelle sur la période considérée
- UC : unité de conditionnement (nombre de produits finis par palette)
- $t_w$  : temps de production d'un conteneur
- $t_{pc}$  : temps d'attente moyen
- S : facteur de sécurité

Cette formule a été améliorée par plusieurs chercheurs dans la littérature. Cependant, cette approche permet de définir un nombre de cartes kanban fixe pour la période considérée.

Le système Kanban a montré son efficacité, pour contrôler le niveau d'en-cours et réapprovisionner les lignes de production efficacement, dans un environnement de production stable et répétitive (Di Mascolo 1990). Selon Yasuhiro (1983), ce type de système est conçu pour une faible variabilité de la demande client, ce qui n'est pas très souvent le cas. En effet, de nombreux systèmes de production sont davantage sujets à la complexité et l'imprévisibilité de l'évolution dynamique et changeante des demandes client.

Afin de remédier à ce problème, plusieurs travaux ont proposé des approches de « card controlling ». Dans ces approches, le nombre de cartes kanban en circulation est adapté aux changements dynamiques et imprévisibles de l'environnement de production. On parle alors de système de contrôle en flux tiré flexible (Gupta et Al-Turki, 1997), réactif (Takahashi, Morikawa et Nakamura, 2004) ou adaptatif (Tardif et Maaseidvaag, 2001). Notons que dans ce qui suit, le terme « adaptatif » sera utilisé de la façon la plus générale possible pour faire référence indifféremment à toutes ces notions.

#### **4. Pilotage adaptatif des systèmes en flux tirés**

---

Plusieurs chercheurs comme (Tardif et Maaseidvaag, 2001; Sivakumar et Shahabudeen, 2008; Renna, Magrino et Zaffina, 2013; Belisário et Pierreval, 2015) ont prouvé que dans un environnement de production instable, le système kanban traditionnel avec un nombre de cartes kanban fixe ne permet plus de s'adapter aux changements dynamiques de l'environnement de production et peut entraîner des risques de rupture de stock ou de sur-stockage.

Plusieurs approches de pilotage adaptatif ont été proposées dans la littérature pour ajuster la tension de flux dans un environnement de production en flux tiré dynamique et changeant. Pour cela, ils cherchent à déterminer quand et où changer le nombre de cartes kanban afin d'améliorer les performances du système de production (coûts, satisfaction client, etc.). Pour résoudre ce problème, certains auteurs se sont basés sur des prévisions de la demande client pour changer dynamiquement le nombre de cartes kanban. D'autres ont proposé des approches de pilotage adaptatif en temps réel où aucune information sur la future demande client n'est disponible. Dans ce qui suit, nous analyserons les différentes approches de pilotage adaptatif proposées dans la littérature. Nous nous focaliserons, alors, sur les méthodes proposées pour l'ajustement dynamique du nombre de cartes kanban ainsi que les mesures de performances utilisées.

#### **4.1. Méthodes de pilotage adaptatif basé sur des prévisions**

Parmi les travaux relatifs au pilotage adaptatif des systèmes en flux tiré, certains supposent la disponibilité de plans ou de prévisions de production, liés à des périodes de temps. On trouve par exemple l'approche proposée par Rees et al (1987), qui consiste à déterminer le nombre de cartes kanban en circulation au début de chaque période sur la base des prévisions de la demande client et des observations du temps de passage des pièces dans l'atelier (lead time) : afin de minimiser la quantité d'encours dans le système. Pour atteindre les mêmes objectifs, Moattar Husseini et al. (2006) utilisent une technique de programmation linéaire en nombres entiers pour déterminer le nombre de cartes kanban dans chaque station du système de production, et ce pour chaque période de temps. Dans (Gupta et Al-Turki 1997), des prévisions sont plutôt utilisées avec la notion de périodes glissantes. Les auteurs se basent sur la moyenne et l'écart-type des temps de traitement pour estimer le temps nécessaire afin de satisfaire la demande client et par conséquent, si besoin, le nombre de cartes kanban supplémentaires à rajouter et quand. Ces cartes sont retirées à la fin de la période où la décision a été prise. Guion et al. (2011) et Talibi

et al. (2013) proposent une heuristique basée sur le stock final et les délais de réapprovisionnement des boucles kanban afin de détecter des éventuelles ruptures de stock. Des cartes kanban supplémentaires peuvent ainsi être rajoutées. Ils utilisent des plans d'expérience et la simulation pour déterminer des facteurs de contrôle et étudier l'impact des cartes supplémentaires sur différents critères de performance.

Dans ces différentes approches, nous constatons, dans un premier temps, que le nombre de cartes kanban est changé une seule fois durant chaque période, dans la plupart de temps, au début de chaque période. Par conséquent, si des changements et/ou des perturbations surviennent durant la période, ils ne peuvent pas être pris en considération. De plus, la durée de la période peut avoir un effet significatif sur les performances globales du système de production. Dans un deuxième temps, ils utilisent des prévisions ou des plans de production. Dans ce cas, si des prévisions précises et fiables sont disponibles, les approches suggérées par ces auteurs sont très intéressantes et il est parfois possible d'utiliser des méthodes d'optimisation pour déterminer quand changer le nombre de cartes kanban et dans quelle station. Or, ce n'est pas très souvent le cas. Pour remédier à ce problème, des chercheurs proposent des approches de pilotage adaptatif en temps réel, visant à déterminer quand changer le nombre de cartes kanban et dans quelle station de façon dynamique dans l'atelier lorsque c'est nécessaire.

## **4.2. Méthodes de pilotage adaptatif en temps réel**

### **4.2.1. Cartes de contrôle ou seuils**

Des cartes de contrôle sont utilisées par exemple dans (Takahashi et Nakamura, 1999b, 2002). Les auteurs se servent plus précisément des cartes de contrôle utilisant la moyenne mobile pondérée exponentiellement (EWMA) pour contrôler la moyenne des durées de temps qui séparent les arrivées des demandes clients et changer le nombre de cartes kanban dans un système en flux tiré. Pour construire leur carte, une analyse par simulation sur l'attente des clients est faite selon le nombre de cartes et le taux d'arrivée. L'objectif est de maintenir le temps d'attente moyen au plus proche d'un temps d'attente cible. Dans un autre travail (Takahashi, Morikawa et Nakamura, 2004), les mêmes auteurs utilisent une carte de même type pour la variance et analysent le compromis entre l'attente et les encours. Hopp et Roof (1998) proposent également un contrôle statistique pour le taux de sortie vérifié par rapport à une valeur

de référence à chaque demande livrée. Selon les mêmes principes, (Framinan, González et Ruiz-Usano, 2006; Liu et Huang, 2009; González-R, Framinan et Usano, 2011) contrôlent le taux de sortie par rapport à un taux de production de référence. Des cartes kanban supplémentaires sont ajoutées ou retirées de la ligne de production, une par une, pour atteindre un taux de sortie de référence et minimiser les encours.

Nous trouvons aussi plusieurs approches basées sur des règles simples avec des seuils déclenchant des décisions comme celle de (Tardif et Maaseidvaag, 2001). Dans cette approche, une carte kanban supplémentaire est ajoutée au système lorsqu'une demande arrive et que le niveau de stock est en dessous d'un certain seuil bas  $R$ . Un nombre maximum de cartes kanban supplémentaires en circulation  $E$ , permet de maîtriser le niveau d'en-cours. Ces cartes sont retirées de la circulation lorsque le stock de produits finis atteint un seuil haut  $C$ . Leur but est de minimiser les coûts de stockage (quantité d'encours et produits finis) et de demandes en attente. Ils utilisent une méthode de recherche locale gloutonne pour essayer d'optimiser leurs seuils et les performances sont évaluées par simulation. De nombreux travaux s'en sont inspirés. Par exemple, Shahabudeen et Sivakumar (2008) utilisent une combinaison d'un algorithme génétique et d'une heuristique de recuit simulé pour déterminer les seuils auxquels des cartes kanban supplémentaires seront ajoutées ou retirées de la ligne de production. Les résultats numériques obtenus indiquent que l'algorithme génétique (GA) est très efficace. Pour cette raison, les mêmes auteurs ont étendu leur approche pour gérer un système kanban adaptatif à plusieurs postes. Dans des travaux ultérieurs, les mêmes chercheurs (Sivakumar et Shahabudeen, 2009) ont utilisé le recuit simulé pour définir les valeurs de seuil d'un système kanban adaptatif. Leurs résultats indiquent que l'algorithme de recuit simulé est également efficace pour optimiser un tel problème.

#### **4.2.2. L'intelligence artificielle**

Récemment, des approches utilisant des méthodes issues de l'intelligence artificielle ont été proposées pour ajuster le nombre de cartes kanban dans un système en flux tirés adaptatifs. Dans (Papadopoulou et Mousavi, 2007), une approche basée sur les systèmes multi-agent a été proposée pour ajuster le nombre de cartes kanban dans un système ConWIP. Leur objectif est de minimiser le temps d'exécution et les encours.

La logique floue a également été utilisée pour ajuster le nombre de kanban. Par exemple, Chang et Yih (1994) ont utilisé une base de règles floue pour contrôler dynamiquement le nombre de cartes kanban en réponse à des variations de la demande clients, des travaux en attente, de l'utilisation de la machine et des délais de traitement. Renna (2015) propose une méthodologie basée sur le développement d'un contrôleur flou qui ajuste le nombre de cartes kanban. L'approche proposée est testée à l'aide d'un modèle de simulation.

Takahashi et Nakamura (1999a) ont utilisé des réseaux de neurones pour détecter des changements dans les séries chronologiques caractérisant les arrivées de la demande client. Leur objectif est d'ajuster la taille des conteneurs tout en maintenant le temps d'attente moyen de la demande au-dessous d'un certain niveau. À cet égard, un réseau de neurone est formé hors ligne à l'aide d'expériences préliminaires issues d'un jeu de simulation dans un environnement avec une variation stable de la demande client. Par conséquent, leur approche se base sur un échantillon de séries chronologiques généré dans des conditions stables, ce qui n'est pas le cas dans le comportement réel de l'environnement de production

Des arbres de décisions ont été utilisés par Belisário et Pierreval (2015) pour ajuster le nombre de cartes kanban. Les auteurs proposent une approche basée sur la simulation et la programmation génétique pour apprendre à décider. Un arbre de décision utilise uniquement l'état actuel du système en entrée et renvoie les modifications suggérées du nombre de cartes en sortie. La fonction objectif utilisée est la minimisation des coûts exprimée en encours, stock final et demandes en attente.

## **5. Conclusion**

Dans ce chapitre, nous avons commencé par une présentation des systèmes de production en flux tirés qui s'inscrivent dans la philosophie du JAT ainsi que les différents concepts et paradigmes qui l'entourent. Nous avons ensuite présenté les différents travaux de référence qui ont étudié le pilotage adaptatif des systèmes en flux tirés et nous nous sommes plus particulièrement intéressés aux différentes méthodes et mesures de performance utilisées.

Malgré l'intérêt grandissant des chercheurs pour les problèmes liés au pilotage adaptatif des systèmes en flux tirés, nous pouvons dresser diverses constatations à partir des approches existantes dans la littérature :

- Plusieurs travaux considèrent que des prévisions et/ou des plans de production sont disponibles, ce qui leur permet de déterminer quand et dans quelle étape de la ligne de fabrication adapter le nombre de cartes kanban pour la prochaine période afin d'améliorer les performances du système de production (coûts, satisfaction client, etc.). Cependant, des prévisions adéquates ne sont pas toujours disponibles dans le contexte actuel des systèmes de production qui est caractérisé par l'incertitude et l'imprévisibilité des variations de la demande clients.
- D'autres approches utilisent des ajustements périodiques des cartes plutôt que des ajustements dynamiques pouvant être effectués à tout moment. Cette décomposition du problème en périodes facilite l'usage des modèles mathématiques et des méthodes d'optimisation classiques. Cependant, de tels types d'approches limitent intrinsèquement la réactivité du système.
- Dans le cas où un pilotage adaptatif en temps réel est utilisé, les informations utilisées pour déterminer quand changer le nombre de cartes kanban sont limitées à certains paramètres liés à l'état actuel du système de production. Cependant, comme le notent Takahashi et Nakamura (1999b), d'autres données pourraient être utiles et pourraient être prises en compte, en particulier la série chronologique de la demande client.
- Toutes les approches de pilotage adaptatif mentionnées visent à optimiser des coûts de production. Ces coûts combinent généralement les coûts de stockages (WIP et stock final) et un coût lié à la satisfaction du client (commandes en retard). Malheureusement, on n'accorde pas suffisamment d'attention à l'impact des changements excessifs du nombre de cartes kanban effectués sur l'environnement de production.
- La simulation a aussi largement été utilisée, ce qui n'est pas sans lien avec sa capacité à représenter la dynamique des systèmes de production ainsi qu'à prendre en compte leur complexité. Cependant, tout le potentiel de la simulation n'est pas exploité.
- Nous pouvons remarquer que ces dernières années, les méthodes d'apprentissage issues de l'intelligence artificielle regagnent de l'intérêt. Néanmoins, elles restent limitées.

Aujourd'hui, l'ensemble du secteur industriel est entré dans une phase de profonde mutation qui voit les technologies numériques s'intégrer au cœur des processus industriels donnant naissance à une nouvelle génération d'usine. Qu'on l'appelle « Cyber-usine », « Usine digitale », « Integrated Industry », « Innovative Factory » ou « Industrie 4.0 », cette nouvelle révolution industrielle offre des opportunités prometteuses et amène les industriels à revoir l'organisation du mode de production avec les outils existants afin de faciliter le travail des opérateurs et pour améliorer sa performance industrielle. Dans le chapitre 2, nous présenterons cette nouvelle révolution et nous nous intéresserons à l'évolution des systèmes Kanban dans le contexte de l'industrie 4.0.

## Chapitre 2.

Evolution des E-Kanban dans l'industrie

4.0



## **1. Introduction**

L'évolution des technologies de l'information et de la communication (TIC) ainsi que leur diffusion massive sont des sources permanentes d'innovation aussi bien au niveau des solutions et des usages que des techniques et des organisations de production. Aujourd'hui, cette évolution a conduit à la naissance d'une nouvelle révolution industrielle. Plusieurs initiatives stratégiques sont nées à travers le monde : le « smart manufacturing » aux États-Unis, « l'internet+ » en Chine, « l'industrie du futur » en France et « l'industrie 4.0 » en Allemagne. Le terme industrie 4.0 semble s'imposer au niveau international. Nous avons décidé d'exploiter ce terme tout au long de nos travaux.

Les révolutions industrielles successives ainsi que l'évolution de la demande ont conduit à la multiplication des contraintes de pilotage donnant à cette activité une importance grandissante au sein des industries. Au fur et à mesure du temps, les entreprises dotées de systèmes de pilotage de la production doivent gérer des systèmes de production de plus en plus complexes et des informations toujours plus nombreuses, dispersées et hétérogènes. Dans ce contexte, les systèmes kanban électroniques (E-kanban) émergent comme une solution fluide/naturelle qui combinent les principes lean avec les évolutions technologiques.

Dans ce chapitre, nous présenterons, dans une première partie, l'historique des révolutions industrielles et donnerons la définition de l'industrie 4.0. Dans une seconde partie, nous réaliserons une synthèse sur les recherches actuellement menées dans le domaine de l'E-kanban et nous nous intéresserons plus particulièrement aux différents travaux menés dans la littérature pour améliorer le pilotage adaptatif en flux tirés des systèmes de production dans le contexte de l'industrie 4.0.

## **2. Industrie 4.0**

### **2.1. Historique**

"Industrie 4.0, usine du futur, usine 4.0, smart factory..." autant de termes pour désigner ce nouveau concept né de la 4ème révolution industrielle qui succède aux 3 autres révolutions

déclenchées par l'introduction de la machine à vapeur (la mécanisation), de la chaîne de montage (l'industrialisation) et du développement de l'électronique (l'automatisation) comme représenté sur la Figure 6.

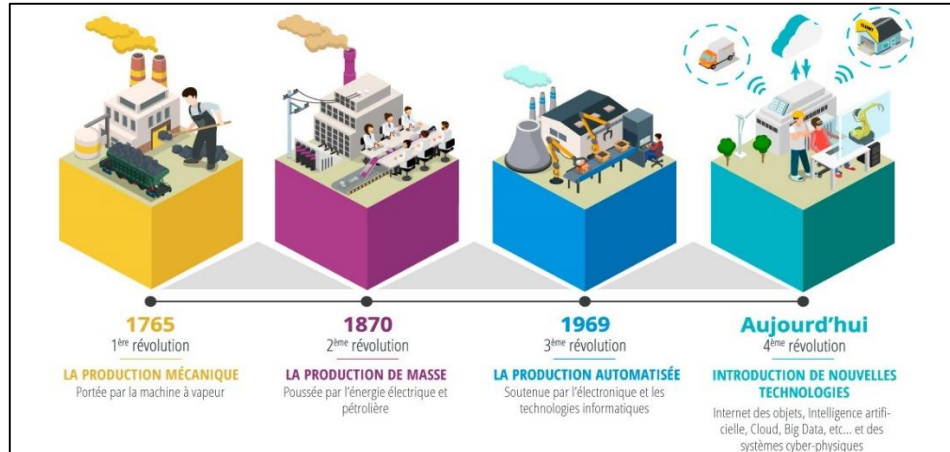


Figure 6: Les quatre révolutions industrielles

- La Première Révolution Industrielle (industrie 1.0) : Elle s'étale entre le milieu du 18<sup>ème</sup> siècle et la fin du 19<sup>ème</sup> siècle. D'un point de vue technologique, elle est caractérisée par l'exploitation du charbon et l'invention de la machine à vapeur par l'écossais James Watt en 1769. Cette révolution a transformé radicalement le mode de fabrication : d'une fabrication artisanale à une production mécanique.
- La Deuxième Révolution Industrielle (industrie 2.0) : Elle arrive à la fin du 19<sup>ème</sup> siècle. Elle est caractérisée par l'utilisation du pétrole et de l'électricité. Dans l'industrie, les machines de production ne sont plus "à vapeur" mais "électriques". Cette époque correspond à la mise en place du taylorisme, du travail à la chaîne et de la production de masse dont le but est d'augmenter le volume de production en minimisant les coûts.
- La Troisième Révolution Industrielle (industrie 3.0) : Elle a eu lieu au milieu du 20<sup>ème</sup> siècle. Elle est caractérisée par l'avènement des innovations technologiques reposant sur l'informatique et l'automatisation. Cette révolution a donné naissance à la robotique, la flexibilité des outils de production et de la production en grandes séries.

- La Quatrième Révolution Industrielle (industrie 4.0) : Ce concept est apparu en 2011 au Forum mondial de l'Industrie de Hanovre. En 2013, un rapport décrivant le concept a été présenté par un groupe de travail transdisciplinaire à la foire de Hanovre (Kagermann, Helbig et Wahlster, 2013). L'objectif premier de cette révolution n'est pas d'augmenter l'automatisation de la production, mais de rendre plus intelligents les modes de production à travers la meilleure exploitation des technologies de l'information et de la communication. L'industrie 4.0 vise à connecter l'ensemble des objets et intervenants tout au long de la chaîne de valeur au sein de l'usine de demain afin d'adapter le système de production aux différentes variations de l'environnement.

L'industrie 4.0 repose sur des technologies numériques existantes ; tels que l'internet des-objets, les systèmes cyber-physique, le big data, le cloud computing, les capteurs intelligents, etc. C'est la conjonction de toutes les possibilités offertes et leur combinaison qui ouvre cet énorme potentiel d'évolution.

## 2.2. Les technologies clés de l'industrie 4.0

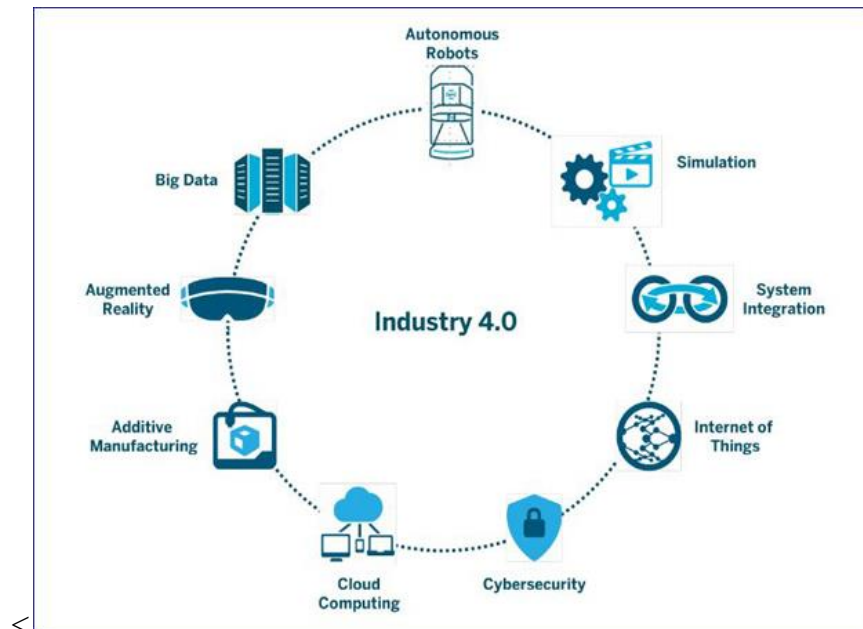


Figure 7: Les technologies clés de l'industrie 4.0.

Selon plusieurs chercheurs (Bril El-Haouzi 2017; Lu 2017; Saucedo-Martínez et al. 2018), l'industrie 4.0 s'appuie sur 9 technologies qui sont décrites dans la Figure 7:

- Internet des objets (IoT)

Ce terme a été introduit par Kevin Ashton (Ng et Wakenshaw, 2016) puis a évolué vers l'internet industriel des objets (IIoT). L'IIoT permet de disposer de services évolués en interconnectant des objets (physiques ou virtuels) grâce aux Technologies de l'Information et de la Communication interopérables.

- Système cyber-physique (CPS)

Ce système peut être défini comme : un système embarqué complexe conçu pour interagir avec son environnement de manière continue via l'association d'éléments physiques, informatiques et de communication. Une analyse de la littérature montrant la prolifération des travaux autour des CPS a été faite par (Cardin 2016).

- Fabrication additive

Egalement appelée impression 3D, prototypage rapide ou outillage rapide (Ponche 2013). Elle désigne l'ensemble de procédés permettant de fabriquer, couche par couche, par ajout de matière, un objet physique directement à partir de son modèle numérique. Cette technologie permet la réduction des coûts et des délais de réalisation de prototypes et de pièces de petite série.

- Robotique avancée

Aujourd'hui, les robots classiques, dangereux pour l'homme et difficiles à programmer, cèdent la place au robot collaboratif ou cobot (Peshkin et al. 2001). En effet, le cobot permet d'assister l'opérateur dans la réalisation de tâches pénibles ou répétitives. Ils ont été conçus pour améliorer les conditions de travail des opérateurs et non pas pour les éliminer.

- Cloud Computing (nuage informatique)

L'Institut national des normes et de la technologie définit le « cloud computing » comme : un modèle permettant un accès pratique et à la demande à un réseau partagé de ressources informatiques configurables (par exemple : réseaux, serveurs, stockage, applications et services)

qui peuvent être rapidement provisionnées et diffusées avec un effort de gestion ou une interaction minimale avec les fournisseurs de services (Qi, Lu, and Raouf 2010). La plupart des services de cloud computing peuvent être classés en trois grandes catégories : IaaS (Infrastructure as a Service), PaaS (Platform as a Service) et SaaS (Software as a Service).

- Jumeau numérique (twin models)

Le jumeau numérique ou double numérique est un modèle logiciel dynamique d'un objet ou d'un système physique qui repose sur des données issues de capteurs en temps réel. C'est une technologie en fort développement dans le domaine de l'industrie car ces applications sont multiples (Boschert et Rosen, 2016).

- Réalité augmentée

Elle a été introduite par Sutherland (1965) afin de visualiser des éléments et des objets virtuels dans un environnement réel. Elle peut être considérée comme une nouvelle forme d'interaction homme-machine fondée sur l'association sémantique et spatiale d'objets réels et virtuels, c'est-à-dire générée par un ordinateur. Les applications industrielles de la réalité augmentée sont réduites au commerce électronique, à l'administration et au marketing (Wolter et al. 2015). La réalité augmentée est un élément important de l'industrie 4.0, notamment pour l'amélioration de la production (Kolberg et Zuhlke, 2015) et la réduction des coûts de maintenance (Simonis, Gloy et Gries, 2016).

- Cyber-sécurité

L'émergence du cloud computing et de l'IoT, ainsi que l'augmentation des communications qui l'accompagnent font de la cyber-sécurité un enjeu majeur pour les entreprises industrielles (Von Solms et Van Niekerk, 2013). La mise en place des moyens de prévention du piratage des données, et du risque de prise de contrôle de l'outil de production à distance, est devenue primordiale.

La maturité technologique offerte par l'industrie 4.0 met en évidence de nouveaux défis liés notamment au pilotage des systèmes de production. Par exemple, en mettant en œuvre des systèmes CPS flexibles et robustes. Parmi les nouveaux domaines d'application, citons l'amélioration des systèmes Kanban existants. En effet, les cartes kanban classiques, qui sont

représentées par des étiquettes attachées aux produits ou aux conteneurs, sont remplacées par des Kanban virtuels ou électroniques (Lage Junior et Godinho Filho, 2010).

### **3. Evolution du pilotage en flux tirés des systèmes de production**

#### **3.1. Kanban électroniques « E-Kanban »**

Nous trouvons dans la littérature plusieurs définitions et nominations du kanban électronique, parmi lesquelles :

- E-Kanban (Electronic Kanban): Une variation de la méthode Kanban classique avec des modifications, où le signal physique de la carte Kanban est remplacé par des signaux électroniques (Lage Junior et Godinho Filho 2010).
- Kanban informatisé (computer-based Kanban): Un micro-ordinateur qui affiche le calendrier et le nombre de pièces disponibles et requises sur la chaîne de montage et autorise la production ou le retrait des cartes Kanban selon les besoins, il s'agirait en fait d'un poste Kanban électronique (Lee-Mortimer 2008).
- Kanban intelligent (Smart Kanban) : La combinaison des avantages des technologies RFID et du système Kanban traditionnel (Zhang, Jiang et Huang, 2008).
- Kanban sans fil (Wireless Kanban) : Une nouvelle forme émergente de la méthode Kanban classique basée sur l'intégration de la communication sans fil dans la production JAT et la gestion des stocks kanban (Ansari et Modarress, 1995).

D'une manière générale, selon Wan et Chen (2008), les systèmes kanban électroniques émergent comme une solution fluide/naturelle qui combinent les principes du système kanban classique avec les évolutions technologiques et plus particulièrement les technologies d'identification automatiques (code à barres, RFID, etc.).

#### **3.2. Implémentations de l'E-Kanban**

##### **3.2.1. Les technologies utilisées**

Technologiquement, la carte kanban classique peut être remplacée par différentes technologies :

- La technologie code à barres 1D et 2D : La technologie la plus utilisée pour l'identification rapide de produits, notamment au travers du standard EAN 13 (Paolo, Antje, and Fabio 2014; Preiss 2016).
- La technologie RFID (Radio Frequency Identification) : Cette technologie repose sur des puces électroniques, capables de stocker de l'information et de la transmettre à distance par ondes radio (Brintrup, Ranasinghe, and McFarlane 2010; G. Q. Huang, Zhang, and Jiang 2008; Luo, Fang, and Huang 2015; Y. Zhang, Jiang, and Huang 2008).

L'intégration de telles technologies dans le système de production amène une amélioration de la visibilité des flux de produits et de l'environnement de production.

### **3.2.2. Implémentation industrielle de l'E-Kanban**

Ceci a motivé les fournisseurs de progiciels de gestion intégrés (ERP) à étendre leurs produits avec des modules Kanban. Ils fournissent des modules additionnels basés sur l'alimentation constamment en données synchrones et en temps réel des évolutions des produits ainsi que des informations concernant l'état du système de production (Bell 2005). Les fournisseurs d'ERP tels que SAP ont inclus la fonctionnalité kanban dans leurs produits (Chai 2008; Powell 2013). La solution SAP utilise des étiquettes RFID attachées aux conteneurs pour automatiser l'identification des lots et le suivi de localisation. Les décisions de fabrication et de pilotage peuvent être prises sur la base d'informations précises sur l'état du lot de production.

Dans les dernières décennies, un certain nombre de grandes entreprises, telles que Toyota (Cullen 2002), BMW (Kochan 2006), Ericsson (Eriksson 2010) et Ford (Zarbo et al. 2009), ont mis en œuvre un système E-kanban dans leurs systèmes de production. Prenons l'exemple de Toyota qui a mis en place un système E-kanban pour la commande de pièces auprès de ses fournisseurs. L'un des objectifs de ce système était d'introduire un moyen efficace d'ajustement du nombre de cartes kanban. En effet, comme décrit dans la Figure 8, Les fournisseurs apposent un code à barres sur chaque conteneur livré, utilisé pour l'identification du lot. Une fois le

conteneur utilisé dans l'usine, le code à barres est retiré. Les informations contenues dans le code à barres sont enregistrées dans des bases de données et partagées via l'IIoT. Ceci a permis une réduction des fluctuations dans la quantité de commande et des changements appropriés dans le nombre de cartes kanban. Il a aussi amélioré la réactivité du système aux variations de la demande client.

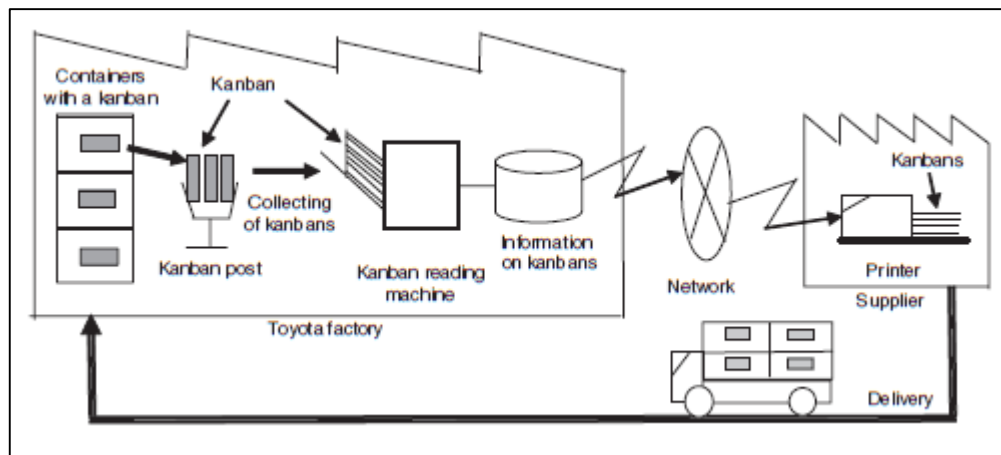


Figure 8: Le système E-kanban de Toyota (Kotani 2007)

BMW a mis en place un système E-Kanban en collaboration avec l'un de ses fournisseurs. En fonction de la demande quotidienne de BMW et grâce à une interface de planification des ressources de l'entreprise (ERP), les programmes de livraison prévisionnels sont envoyés à son fournisseur en temps réel, ce qui donne au fournisseur 300 minutes pour produire et livrer les produits directement à la chaîne de montage. Au cours de sa première année de fonctionnement, le système E-Kanban a permis de minimiser le temps de transfert d'informations entre BMW et son fournisseur et réaliser des économies de plus de 63 millions d'euros.

De nombreux fabricants constatent de fortes améliorations de la rentabilité des systèmes E-Kanban. Pour cette raison, des systèmes Web Kanban commerciaux ont récemment été développés par des fournisseurs de logiciels, tels que Datacraft Solutions, eBots, Manufactus, SupplyWorks et un Web libre Heijunka-Kanban de Bo Ahlberg. Ces systèmes ont été développés sur la base d'une infrastructure Web offrant un accès multiplateforme via des navigateurs Web. Ces systèmes ont différents objectifs, tels que le pilotage des systèmes en flux tirés, la gestion des stocks ou l'analyse de la performance en temps réel.



### **3.2.3. Approches de pilotage en flux tirés via l'E-Kanban**

Dans la littérature du JAT, certains travaux proposent des approches de pilotages en flux tirés utilisant des cartes kanban électroniques au lieu des cartes kanban classiques. Leur but est d'étudier l'opportunité d'utiliser les technologies d'identification automatique pour le pilotage des flux de produits.

(El Haouzi 2008) a développé un système E-Kanban pour le pilotage d'une ligne de production au sein de l'entreprise TRANE. Il a été déployé pour la gestion des mouvements de stock entre le magasin et l'atelier et pour l'optimisation des déplacements des « RUNNERS ». Avec l'application du E-Kanban, ils ont pu réduire le temps de suivi des boucles kanban et le traitement des cartes kanban. De plus, la qualité des produits ainsi que la visibilité du processus de production se sont nettement améliorées.

Plusieurs chercheurs ont proposé des méthodes conceptuelles pour l'implémentation d'un système E-kanban basée sur l'identification des produits par des étiquettes RFID (Jiang et al. 2010; Zhang et al. 2008; Baudin et Rao 2005). Leur objectif est de suivre et visualiser les flux de produits dans le système de production.

Chow et al. (2007) proposent un système multi-agents de pilotage d'entrepôt basé sur l'identification par RFID. Leur approche montre une amélioration considérable des performances du système de production étudié.

### **3.3. Les avantages de l'implémentation de l'E-Kanban**

De nombreuses études montrent les avantages de l'implémentation du E-Kanban par rapport au Kanban classique (A. Ansari et Modarress 1995; Svirčević et al. 2013; Henderson 1986). Il permet, entre autres :

- La traçabilité en temps réel des flux de produits ;
- L'échange d'information en temps réel ;
- L'amélioration de la visibilité des flux de produits et d'informations (kanban);
- La réduction des erreurs humaines (par exemple : la perte des cartes kanban) ;

- La formalisation du processus de communication entre les fournisseurs ;
- La synchronisation du flux d'objets physiques du procédé avec le flux des objets logiciels du système d'information.

## 4. Conclusion

Dans ce chapitre, nous avons présenté le concept d'industrie 4.0 ainsi que les différentes technologies qui l'entourent. Par la suite, nous nous sommes plus particulièrement intéressées à l'intégration des technologies d'identification automatique dans les systèmes en flux tirés. L'intégration de telles technologies a substitué la carte kanban classique par une carte électronique (E-Kanban) assurant ainsi une meilleure visibilité des flux de produits et d'informations.

En effet, les technologies d'identification automatique offrent la possibilité de connaître à tout moment et à chaque étape de la ligne de fabrication le contexte de production : l'emplacement, le statut et ce qui est arrivé aux produits et aux ressources. Ainsi, la collecte, le stockage et la communication d'informations précises et en temps réel pour chacun des produits. Ceci permet une amélioration des décisions prises sur le pilotage des systèmes en flux tirés.

Malgré l'intérêt et les opportunités offertes par ces technologies, leurs utilisations à l'échelle industrielle restent limitées. Cela est dû en partie aux coûts des investissements nécessaires à l'installation de ces technologies, la validation de la pertinence des infrastructures (nombre de lecteurs, leurs emplacements, l'objet porteur de l'étiquette (produit, palette, lot...)), l'intégration aux systèmes d'informations, etc. En effet, cette évolution industrielle se fait au détriment des entreprises classiques qui doivent supporter un coût du travail élevé.

Si certaines entreprises choisissent d'adopter et d'émerger vers l'industrie 4.0, en intégrant les nouvelles technologies dans leur industrie, d'autres choisissent au contraire de rechercher de nouvelles solutions leur permettant des gains substantiels de productivité sans recours aux nouvelles technologies. Une solution pour ces entreprises consiste à leur proposer un outil performant d'aide à la décision, capable d'aider les gestionnaires sur le terrain à ajuster

dynamiquement la tension des flux de production dans les systèmes en flux tirés. Cela sera l'objectif du prochain chapitre, qui a été déjà publiée dans (Azouz et al. 2018).

## Chapitre 3.

# Optimisation multi-objectif via Simulation pour le pilotage adaptatif

## 1. Introduction

Aujourd'hui, l'évolution des marchés et des besoins des clients ainsi que les aléas de fabrication exigent que les systèmes de production soient capables de s'y adapter au mieux, de mettre en place des modes de pilotages intelligents et d'être plus flexible et réactif. Face à ces enjeux industriels, plusieurs travaux de recherche ont été menés dans la littérature.

Dans le contexte des systèmes de production en flux tirés, nous avons souligné, dans le premier chapitre, que les systèmes de production en flux tirés traditionnels, avec un nombre fixe de cartes kanban, sont adaptés à un environnement de production stable et répétitif, ce qui n'est pas toujours le cas. Pour éviter des problèmes de rupture de stock (ou de sur-stockage) dans un environnement instable et changeant, l'adaptation dynamique du nombre de cartes kanban via des cartes dites supplémentaires a été proposée par plusieurs chercheurs. Elle consiste à ajouter ou retirer des cartes kanban de la ligne de fabrication afin de rendre ces systèmes plus réactifs aux changements de l'environnement et donc performants sur le long terme. Malgré les diverses publications sur ce sujet, l'adaptation dynamique du nombre de cartes kanban dans un environnement de production dynamique et changeant reste encore un défi scientifique.

Nous avons présenté, dans le premier chapitre, plusieurs travaux de la littérature visant à étudier le problème d'adaptation dynamique du nombre de cartes kanban dans les systèmes en flux tirés. Nous avons constaté que certains travaux supposent, la plupart du temps, la disponibilité d'un horizon temporel, un plan de production et même une visibilité future sur les variations de la demande clients. Cela leur permet d'utiliser des modèles mathématiques et des méthodes d'optimisation classiques pour déterminer quand et dans quelle étape de la ligne de fabrication adapter le nombre de cartes kanban pour la prochaine période. Or, des prévisions adéquates ne sont pas toujours disponibles dans le contexte actuel des systèmes de production qui est caractérisé, dans la plupart du temps, par l'incertitude et l'imprévisibilité des variations de la demande clients.

Nous avons pu aussi distinguer les approches de pilotage adaptatif qui sont davantage sujettes à la complexité et l'imprévisibilité de l'évolution dynamique et changeante de l'environnement de production. Par conséquent, elles prennent des décisions en temps réel et dépendantes de l'état actuel du système de production pour déterminer quand et dans quelle étape de la ligne de fabrication changer le nombre de cartes kanban pour s'adapter aux changements.

Nous avons également identifié dans la littérature que la simulation a été largement utilisée. Ceci est dû à sa capacité à reproduire la dynamique des systèmes de production ainsi qu'à prendre en compte leur complexité. En effet, la simulation a été montrée très utile dès que la réalisation d'un modèle mathématique du système de production est trop simplificatrice de la réalité, de sorte que le modèle obtenu ne permet plus d'appréhender les informations utiles sur le système à étudier (Erard et Déguénon, 1996; Fleury, 2006). Généralement, la simulation est largement utilisée pour étudier des systèmes de production complexes.

Elle permet de modéliser des entités (machines, opérateurs, kanbans, etc.), leurs attributs (temps opératoire, nombre de kanbans, etc.) et la façon dont elles interagissent dans le but de voir comment le système se comporte sur une période de temps. La simulation nous permet d'étudier le comportement dynamique d'un modèle réel à partir des hypothèses à tester. Elle permet d'analyser des scénarii et donc de répondre à des questions du type « Que se passerait-il si...? » sans pour autant optimiser le système étudié. Cela rend intéressant son couplage à un algorithme d'optimisation. Pour plus de détails sur l'optimisation via simulation, se référer à (Sarker, 1989; Law et McComas, 2002; Long-Fei et Le-Yuan, 2013; Nguyen, Reiter et Rigo, 2014; Pedrielli, Alfieri et Matta, 2015; Pierreval et al., 2015).

Nous avons aussi constaté que toutes les approches de pilotage adaptatif en temps réel, proposées dans la littérature, changent, dynamiquement, le nombre de cartes kanban dont l'objectif est d'améliorer les mesures de performances du système de production qui exprime globalement les coûts. Ces deniers sont généralement un compromis entre les coûts de stockage (encours et stock final) et un coût lié à la satisfaction du client (commandes non satisfaites). Néanmoins, on n'accorde pas suffisamment d'attention à l'impact des changements excessifs du nombre de cartes kanban effectués sur l'environnement de production.

En effet, changer le nombre de cartes kanban est nécessaire pour ajuster la tension de flux dans un environnement de production dynamique et changeant. Cependant, changer trop souvent le nombre de cartes kanban peut impliquer des coûts supplémentaires influant négativement sur le système de production. Ces coûts sont souvent difficiles à quantifier mais devraient être pris en considération. Tout d'abord, dans certaines industries, la carte kanban peut être attachée à un conteneur. Par conséquent, changer le nombre de cartes kanban revient à changer le nombre de conteneurs. Ce changement prend du temps et, généralement, au moins un opérateur sera requis pour l'effectuer. Cela peut engendrer différents coûts supplémentaires : des coûts de transport (par exemple, si les conteneurs sont stockés dans des aires de stockages lointaines), des coûts de production liés à des perturbations ou encore des difficultés humaines (par exemple, le stress induit, la fatigue, la dégradation de la qualité du travail, la résistance au changement), etc... Des demandes de changement excessives du nombre de cartes kanban peuvent, par exemple, induire le mécontentement des opérateurs et, par conséquence, une résistance au changement. On parle ici de nervosité du système. Bien que ce problème de nervosité soit largement étudié dans la littérature dans d'autres domaines (comme le MRP), il n'a jamais attiré l'attention des chercheurs dans le contexte du pilotage adaptatif des systèmes en flux tirés malgré son importance.

En fait, pour répondre aux besoins des clients rapidement tout en maintenant un faible niveau d'encours, il faut effectuer suffisamment de changements dans le nombre de cartes kanban. Cependant, pour limiter la nervosité, elles ne doivent être appliquées que si cela est vraiment nécessaire. À cet égard, il ne suffit pas d'utiliser les mesures de performance classiques pour le pilotage adaptatif des systèmes en flux tirés. Il est nécessaire d'étudier le compromis entre les coûts de production et le nombre de changement effectué.

Pour cela, nous proposerons, dans ce chapitre, une approche d'optimisation multi-objectif via simulation qui permet de déterminer quand et dans quelle étape de la ligne de fabrication changer le nombre de cartes kanban dans un système en flux tiré afin d'améliorer les performances du système de production tout en minimisant la nervosité. Cette approche, nous l'avons déjà publiée dans (Azouz et al. 2018).

Ce chapitre est organisé comme suit : Dans un premier lieu, nous fournirons les principes généraux de notre approche. Puis nous introduirons et définirons le problème de la nervosité dans les approches de pilotage adaptatif des systèmes en flux tirés. L'approche de résolution sera donc introduite par la suite dans sa forme la plus générale. Une fois les bases posées, nous choisirons une méthode d'optimisation multi-objectif adéquate et nous présenterons le codage d'une solution en accord avec la méthode choisie. Enfin, afin de montrer la faisabilité et la pertinence de l'approche proposée, nous l'illustrerons sur un système Kanban et nous comparerons nos résultats avec une approche de pilotage adaptatif qui a été largement utilisée dans la littérature pour des fins comparatives.

## **2. Principes généraux de l'approche**

Nous nous intéressons à trouver une stratégie de pilotage capable de déterminer quand et dans quelle étape de la ligne de fabrication changer le nombre de cartes kanban dynamiquement dans un environnement de production dynamique et incertain. Contrairement à la plupart des travaux dans la littérature, nous ne nous restreignons pas à l'amélioration des performances du système de production mais notre objectif est de prendre aussi en considération le problème de la nervosité. Afin d'identifier le moment où le système doit être remis en cause, nous devons surveiller son évolution régulièrement (voire à tout instant).

L'interaction entre le système de production et la stratégie de pilotage est décrite dans la Figure 9. En effet, nous décomposons le système de production en deux sous-systèmes complémentaires : la composante physique et la composante de pilotage.



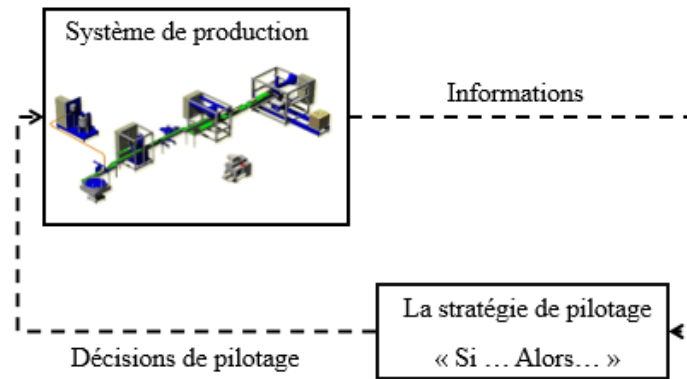


Figure 9: Interaction entre la composante physique et le module décisionnel

En effet, la composante physique représente la ligne de production avec les machines, les produits, les ressources ainsi que les flux des cartes kanban. D'autre part, la composante de pilotage contrôle la précédente en lui communiquant les décisions de pilotages prises (ajout ou retrait des cartes kanban) relativement à son état. C'est cette dernière composante qui permet d'être réactif puisqu'elle correspond à la stratégie de pilotage qui nous intéresse.

A cet égard, nous nous intéressons à trouver hors ligne une stratégie de pilotage qui soit pertinente pour un système de production donné en utilisant une approche d'optimisation multi-objectif via simulation.

L'utilisation d'un modèle de simulation présente l'avantage de pouvoir gérer les comportements dynamiques, stochastiques et complexes de manière réaliste et d'éviter les hypothèses restrictives (Erard and Déguéon 1996; Fleury 2006; A. M. Law 2015; Li et al. 2010; R. Zhang, Ong, and Nee 2015). Cela nous permet de prendre en considération les fluctuations de la demande des clients. Ainsi, le taux moyen de la demande client peut être modélisé pour évoluer dans le temps en fonction de différents types de signaux (patterns) (voir Figure 10), qui peuvent être simulés, en plus du comportement stochastique de l'arrivée des commandes (Gaury, Kleijnen, et Pierreval 2001). Ceci nous permet de résoudre notre problème de pilotage adaptatif dans des conditions où nous n'avons aucune information sur l'arrivée des demandes clients et qu'aucune prévision n'est utilisée.

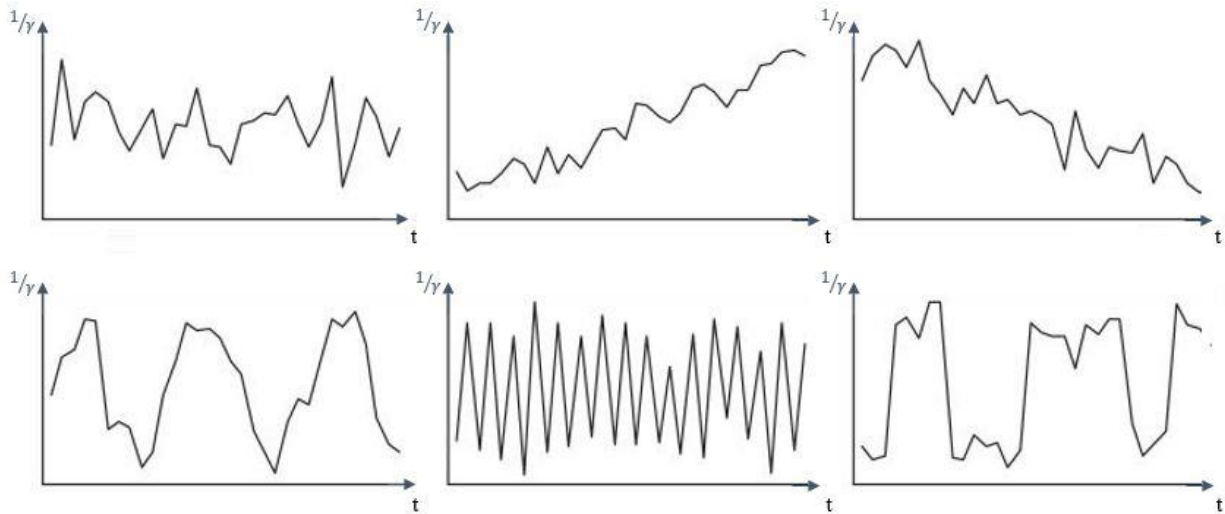


Figure 10: Exemples de signaux des variations de la demande clients

Ces signaux peuvent être fournis en entrée au modèle (simulation) pour indiquer comment la demande moyenne évoluera dans le temps durant toute la simulation.

Afin d'évaluer la performance de notre stratégie de pilotage, nous proposons d'utiliser une méthode d'optimisation multi-objectif. Notre approche est composée alors de deux modules complémentaires : Le module de la simulation et le module de l'optimisation multi-objectif. La Figure 11 montre l'interaction entre ces deux modules. Notons que nous donnerons d'abord les explications de façon assez générique sans faire référence à la méthode d'optimisation multi-objectif utilisée, ni à la stratégie de pilotage adoptée dans cette thèse. Cela montre que rien n'empêche d'en utiliser d'autres en suivant les mêmes principes.

En effet, le module de l'optimisation multi-objectif propose au module de la simulation une solution candidate qui représente, dans notre cas, les paramètres de la stratégie de pilotage adoptée. Ainsi, le module de la simulation évalue la stratégie de pilotage basée sur la solution proposée et lui renvoie les performances demandées. Pour ce faire, durant la simulation, à chaque fois qu'une décision doit être prise, le système de production envoie des informations sur son état à la composante de pilotage. Cette dernière contient la stratégie de pilotage qui va déterminer les éventuelles décisions à prendre, qui seront retournées au système. Les modifications correspondantes, s'il y en a, sont ensuite appliquées sur le système simulé. Puis

la simulation suit son cours jusqu'au besoin d'une nouvelle décision relançant ce processus, et ainsi de suite jusqu'à la fin de la simulation. Nous obtenons alors les performances souhaitées.

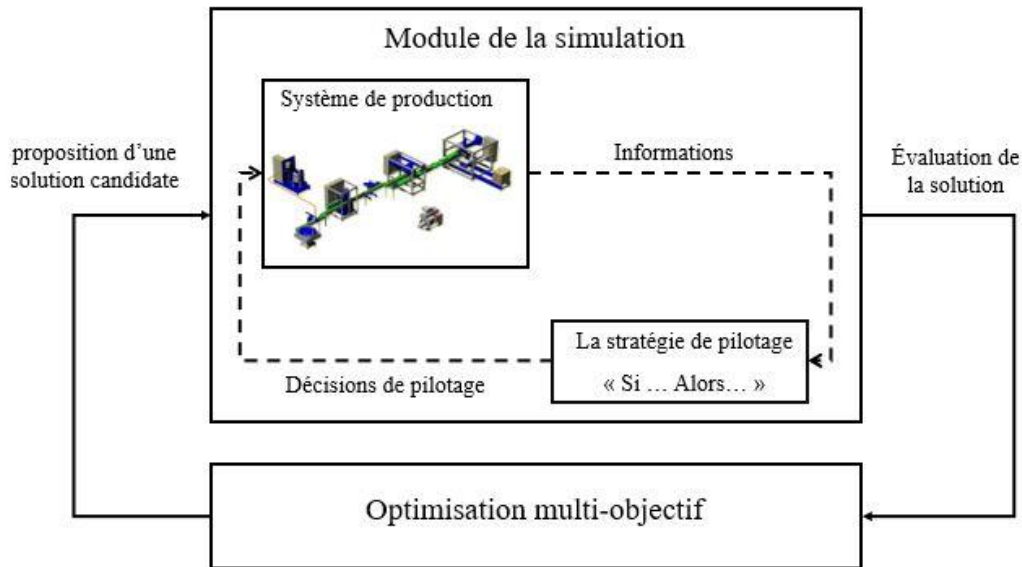


Figure 11: Couplage de la simulation et de l'optimisation multi-objectif

### 3. Le problème de la nervosité

#### 3.1. La nervosité dans les systèmes de production

Le problème de la nervosité a été largement étudié dans plusieurs domaines scientifiques (les systèmes MRP, les systèmes multi-agents (MAC) et les systèmes holoniques (HMS)) et de différents points de vue comme décrit dans le Tableau 1 : du réajustement fréquent au manque de coordination entre les agents ou les holons.

En effet, la nervosité a d'abord été utilisée pour décrire l'instabilité des plans de production dans les systèmes MRP. Elle décrit les réajustements continuels dans les calendriers de production. (Blackburn, Kropp et Millen, 1985) définissent la nervosité dans les systèmes MRP comme suit : « l'instabilité et le réajustement fréquent des ordres de fabrication en termes de temps et quantité ». Dans les systèmes multi-agent, la nervosité peut être engendrée suite à la coordination entre les différents agents. Par exemple, selon (Hadeli et al. 2007), la nervosité est causée par la fréquence des changements d'intention des agents pour trouver une nouvelle

solution. Si cette fréquence est trop élevée, cela indique automatiquement que l'agent est incapable de contrôler son désir de passer à une nouvelle solution, ce qui peut provoquer une instabilité dans le système. Récemment, dans le domaine des systèmes HMS, l'introduction de propriétés d'auto-organisation dans les entités autonomes (holon) et dans le système global peut entraîner une nervosité systémique si le holon change trop souvent d'intention (Barbosa et al. 2012).

Tableau 1: Les différentes causes de la nervosité.

Contexte	Causes de la nervosité
MRP	Réajustements continuels dans les calendriers de production
MAC	La fréquence de changement d'intention des agents
HMS	La fréquence de changement d'intention des holons

Pour résumer, le fait de donner de l'autonomie à un système ou à des entités, afin de faciliter l'adaptation et améliorer la réactivité du système de production, peut conduire à un effet pervers où les entités cherchent continuellement à adapter leur comportement. Cela peut contribuer à l'émergence d'une instabilité et de comportements chaotiques (Barbosa et al. 2015; Borangiu et al. 2016).

### **3.2. La nervosité dans les approches de pilotages des systèmes en flux tirés**

Nous définissons la nervosité dans le contexte des systèmes en flux tirés comme suit : « les modifications excessives du nombre de cartes kanban, en réponse à des stimuli résultant de l'évolution dynamique et imprévisible de la demande des clients ». Naturellement, un système est nerveux s'il modifie trop souvent le nombre de cartes kanban, ce qui peut conduire à un phénomène de « sur-correction », et inversement, un système est stable s'il modifie peu ce nombre.

Si nous analysons les raisons qui peuvent conduire à une nervosité dans les approches du pilotage adaptatif des systèmes en flux tirés, nous pouvons souligner trois causes principales :

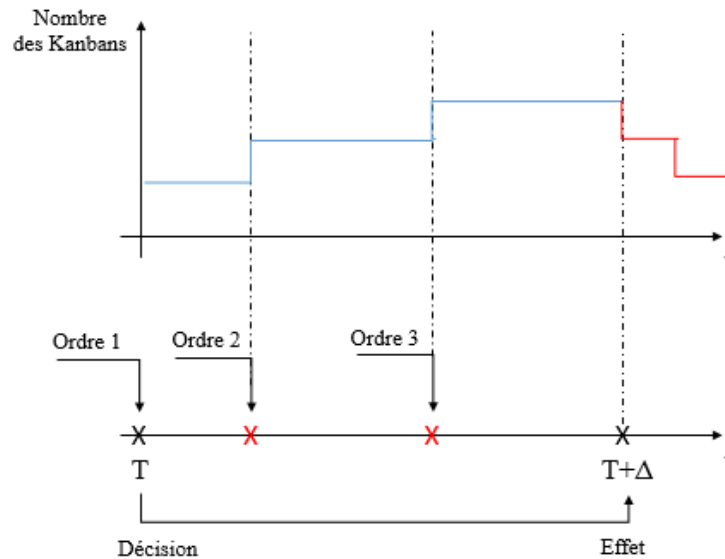
**Cause 1.** Délai de prise en compte d'une modification dans le nombre de cartes kanban

Figure 12: La conséquence du délai entre une décision et son effet

La modification du nombre de cartes kanban consiste à ajouter ou retirer un/des cartes kanban de la ligne de production. Cependant, la décision d'ajouter ou retirer une carte nécessite un temps pour qu'elle produise l'effet attendu. Ce délai, entre la décision et son effet, dépend du temps de traitement, du nombre de machines, etc. Typiquement, comme on peut le voir dans la Figure 12, après l'arrivée de la commande (ordre 1) à l'instant  $T$ , si nous décidons d'ajouter une carte, le délai entre l'action (ajouter une carte) et la conséquence attendue (par exemple, un stock plus important) à l'instant  $(T+\Delta)$  dépend fortement des caractéristiques et de l'état du système de production. De plus, puisque l'ajout d'une carte à l'instant  $T$  n'a pas encore produit l'effet escompté, si un nouvel événement (ordre 2) arrive à l'instant  $(T+\epsilon)$  juste après (ordre 1), avec  $\epsilon \ll \Delta$ , le système a tendance à ajouter un nouveau kanban, qui peut ne pas être nécessaire. Le même phénomène peut également apparaître si un nouvel événement (ordre 3) arrive juste après (ordre 2). Par conséquent, à l'instant  $T+\Delta$ , ce phénomène (c'est-à-dire permettre d'ajouter un kanban à chaque arrivé de commande) peut conduire à une correction excessive (ce qui incitera le système à retirer une ou plusieurs cartes par la suite, comme le montre la Figure 12). Cette sur-correction peut induire à une nervosité du système.

## **Cause 2.** Les paramètres de la stratégie du pilotage adaptatif

Une autre cause possible de la nervosité est la stratégie de pilotage adaptatif utilisée pour déterminer quand et dans quelle étape de la ligne de fabrication changer le nombre de cartes kanban. Comme nous l'avons vu dans le chapitre 1, lorsque ces décisions sont prises de façon dynamique et en temps réel, certains chercheurs utilisent des règles de pilotages basées sur des seuils, comme celles utilisées dans (Tardif et Maaseidvaag 2001; Sivakumar et Shahabudeen 2009; Sivakumar et Shahabudeen 2008). Ces seuils déterminent, dans la mesure du possible, si le nombre de cartes kanban doit être changé ou pas. Si ces seuils sont réputés avoir un impact important sur les coûts de production (voir par exemple (Belisário, Azouz, and Pierreval 2016; Framinan, González, and Ruiz-Usano 2006; González et al. 2006; Pierreval et al. 2015), ils peuvent néanmoins conduire à un comportement nerveux du système de production. En fait, l'intervalle entre les paramètres utilisés (seuils) pour décider quand ajouter ou retirer une carte kanban a une grande influence sur la tension des flux et le comportement du système. Un intervalle court peut améliorer les performances du système global. Cependant, cela pourrait conduire à des changements excessifs, ce qui pourrait engendrer une nervosité dans le système de production. Comme ces paramètres dépendent fortement du type de système considéré et de son comportement (stochastique, variation de la demande client, etc.), déterminer ces paramètres tout en prenant en considération la nervosité est un problème difficile.

## **Cause 3.** Le choix des mesures de performance

La dernière cause de la nervosité est le choix des mesures de performance dans les approches de pilotage adaptatif des systèmes en flux tiré et, plus particulièrement, les approches de pilotage en temps réel. En effet, la plupart de ces approches cherchent à déterminer quand et dans quelle étape de la ligne de fabrication changer le nombre de cartes kanban dans le but de réduire les coûts de stockage (encours et produits finis) et améliorer la satisfaction des clients. Cependant, aucun critère de performance complémentaire n'est utilisé pour gérer le nombre de changement afin de minimiser la nervosité. Par conséquent, étudier le compromis entre les coûts de production et le nombre de changement effectué peut être intéressant.

Pour résumer, nous avons identifié trois causes principales qui peuvent rendre trop nerveux un système de production en flux tiré. Afin de les prendre en considération, nous proposons une approche d'optimisation multi-objectif via simulation qui permet de déterminer quand et dans quelle étape de la ligne de fabrication changer le nombre de cartes kanban dans un environnement dynamique et changeant tout en prenant en considération la nervosité du système de production.

## **4. Résolution par hybridation de l'optimisation Multi-objectif et de la simulation**

### **4.1. Prise en compte de la nervosité : période gelée adaptative**

Plusieurs techniques ont été proposées dans la littérature pour maîtriser et réduire la nervosité dans d'autres contextes (Blackburn, Kropp et Millen, 1985; Kadipasaoglu et Sridharan, 1997; Kazan, Nagi et Rump, 2000; Hadeli et al., 2007; Tunc et al., 2013). Citons à titre d'exemple, Barbosa et al. (2012b) qui propose la mise en place d'une commande PID (pour Proportionnel, Intégral et Dérivé) bien connue en automatique. Une autre façon pour répondre au problème de nervosité consiste à utiliser les périodes gelées (Rees et al., 1987; Hopp et Roof, 1998). De façon basique, geler des périodes consiste à fixer le nombre de kanbans en circulation pendant une durée de temps fixe, les modifications de ce nombre sont donc interdites. Revenons à notre exemple, cette technique permet d'éviter les sur-corrrections en interdisant de réagir et prendre de nouvelles décisions suite à l'arrivée des ordres 2 et 3 (voir Figure 13).

Cependant, malgré la simplicité et l'efficacité des périodes gelées, l'utilisation de cette stratégie pose un problème potentiel dans la détermination de la longueur de la période à geler. Cette longueur dépend de l'état du système et influe considérablement sur les performances du système dans son ensemble. En effet, une période gelée trop large permettrait d'éviter des modifications dans le nombre de kanbans qui pourraient être utiles. Au contraire, une période trop courte permettrait des changements trop fréquents, ce qui peut augmenter la nervosité. Par conséquent, cette mesure devrait être adaptée aux caractéristiques du système de production et à sa dynamique. Pour cette raison, nous proposons de prendre en considération cette durée et la déterminer par optimisation comme nous le verrons en détail dans les sections suivantes.

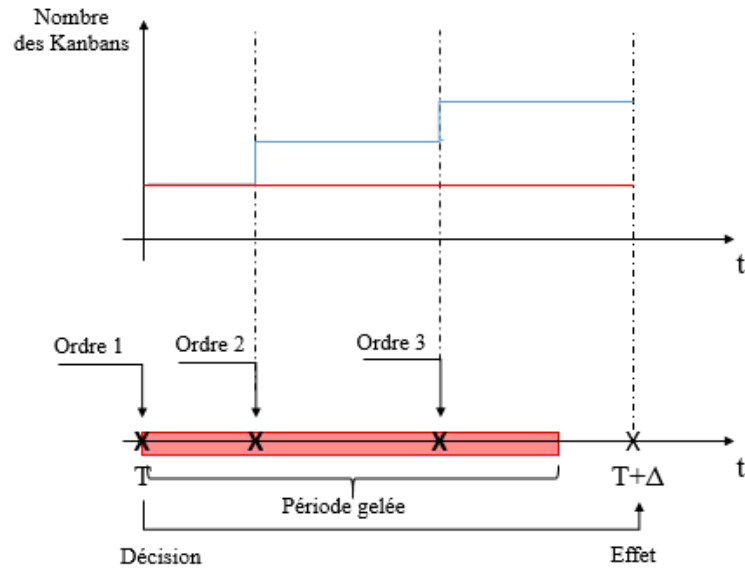


Figure 13: Effet de la période gelée

## 4.2. Stratégie de pilotage

Afin de piloter dynamiquement le système de production, nous proposons une nouvelle stratégie de pilotage qui permet de contrôler et déterminer quand il faut ajouter/retirer une carte kanban ou ne rien faire en fonction d'une période gelée adaptative FI.

Nous considérons un système en flux tirés mono-produit composé de  $N$  stations (voir Figure 14). Nous supposons l'existence d'un stock infini de matières premières de sorte qu'il n'y aura pas une rupture de stock au début de la ligne de production. Ainsi, les produits sont fabriqués un par un. Une fois fabriqués dans une station, les produits sont placés dans un stock  $PF_j$  ( $j = 1, \dots, N - 1$ ) en attente d'une autorisation (un kanban) pour passer à la station suivante. Dès la réception d'un kanban, le produit est livré à la station suivante et son kanban est placé dans le stock des cartes kanban de la station précédente  $D_{j-1}$ . Les produits finis sont livrés par la dernière station  $N$  et placés dans un stock de produit finis  $PF_N$  où ils sont disponibles pour répondre aux demandes des clients. Les demandes arrivent une par une. Si le stock des produits finis est vide, les demandes sont mises en attente dans un stock  $D_N$ .



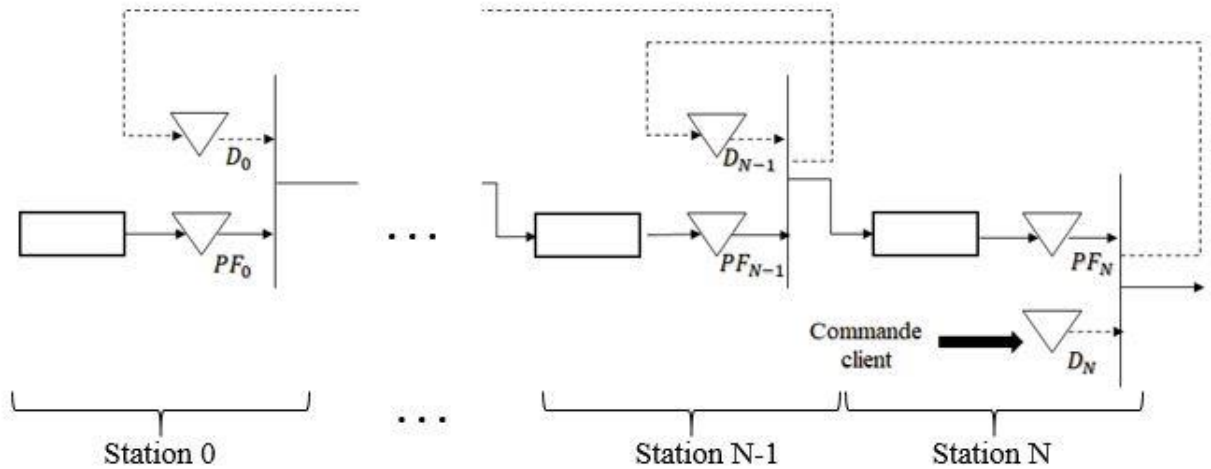


Figure 14: Un système en flux tiré à plusieurs stations

Notre stratégie de pilotage est basée sur une période gelée durant laquelle les modifications du nombre de cartes kanban, s'il y en a, sont interdites. Ensuite, si cette période prend fin et un évènement déclencheur apparaît à l'instant  $t$ , le niveau de stock  $I_j(t)$  de chaque station  $j$  est vérifié. Dans la dernière station  $N$ , le niveau de stock  $I_N(t)$  est donné par la quantité de produits finis dans le stock  $PF_N$  moins le nombre des clients non satisfaits dans le stock  $D_N$ . Tandis que dans les autres stations  $j$  ( $j=1, \dots, N-1$ ), le niveau de stock  $I_j(t)$  est donné par le nombre total de produits venant d'être fabriqués à la station  $j$  et qui sont placés dans le stock  $PF_j$  moins le nombre total de cartes kanban non attachées à des produits dans la station  $j$  et placées dans le stock  $D_j$ . Les notations suivantes sont utilisées pour la description de notre stratégie de pilotage :

- $K_j$  : Nombre total des cartes kanban fixes disponibles dans la station  $j$  ;
- $E_j$  : Nombre total des cartes supplémentaires disponibles dans la station  $j$  ;
- $R_j$  : Seuil minimum au-dessous duquel nous ajoutons un kanban dans la station  $j$  ;
- $C_j$  : Seuil maximum au-dessus duquel nous retirons un kanban dans la station  $j$  ;
- $I_j(t)$  : Représente le niveau de stock dans la station  $j$  ;
- $FI = [t, t + \alpha]$  : La durée de la période gelée ;
- $E_j(t)$  : Nombre des cartes supplémentaires non encore utilisées, à l'instant  $t$ , dans la station  $j$  ;
- $PF_j$  : Nombre total des produits (ou produits finis) dans la station  $j$  ;

- $D_j$  : Nombre total des demandes non satisfaites (ou des carte kanban libres) dans la station  $j$  ;
- $ET(t)$  : La durée entre la dernière modification dans le nombre de kanban et l'instant  $t$  ;
- $LMT$  : l'instant  $t$  de la dernière modification dans le nombre de cartes kanban.

Nous pouvons donc résumer la stratégie de pilotage proposée comme suit :

A l'apparition d'un évènement déclencheur à l'instant  $t$ , nous calculons  $ET(t)$  :

- i. Si l'évènement déclencheur apparaît durant la période gelée ( $ET(t) < FI$ ), alors aucune modification n'est autorisée. Ainsi, la demande client est satisfaite s'il y a un produit fini disponible ou mise en attente dans le stock  $D_N$  ;
- ii. Si l'évènement déclencheur apparaît après la fin de la période gelée ( $ET(t) \geq FI$ ), alors nous vérifions l'état de chaque station  $j$  afin de vérifier s'il est nécessaire ou pas de changer le nombre de cartes kanban comme suit :

Pour chaque station  $j$ , nous calculons le niveau de stock  $I_j(t) = PF_j - D_j$  ;

- a. Si le niveau de stock est inférieur au seuil minimum ( $I_j(t) \leq R_j$ ) et une carte supplémentaire est disponible ( $E_j(t) > 0$ ), alors nous décidons d'ajouter un nouveau kanban à la station  $j$  et nous mettons à jour le LMT ;
- b. Si le niveau de stock est supérieur au seuil maximum ( $I_j(t) > C_j$ ) et il existe encore une carte supplémentaire non utilisée ( $E_j(t) < E_j$ ), alors nous décidons de retirer un kanban de la station  $j$  et nous mettons à jour le LMT ;
- c. Sinon le système est gardé tel qu'il est ;
- d. Après les vérifications, si un produit fini est disponible, la demande client sera satisfaite sinon elle sera mise en attente.

Notons que des contraintes sont nécessaires pour le bon fonctionnement de notre stratégie de pilotage. Elles peuvent être formalisées comme suit :

$$\left\{ \begin{array}{ll} C_j < R_j & \text{Contrainte. 1} \\ C_j \leq K_j + 1 & \text{Contrainte. 2} \\ K_j, E_j, R_j, C_j, FI > 0 & \text{Contrainte. 3} \end{array} \right. \quad \forall j \in \{1, \dots, N\}$$

La contrainte (1) garantit que le nombre de cartes kanban supplémentaires capturé ne dépasse pas le nombre de cartes kanban supplémentaires libéré dans la ligne de production. Par conséquent, on ne peut ajouter ou retirer qu'une carte kanban parmi les cartes kanban supplémentaires.

La contrainte (2) garantit que le nombre de cartes kanban ne peut jamais être réduit en dessous du niveau de base du nombre de cartes.

La difficulté de l'utilisation de cette stratégie de pilotage revient à la détermination des paramètres  $FI, K_j, E_j, R_j$  et  $C_j$  (pour chaque station  $j=1, \dots, N$ ), car ils ont une grande influence sur la performance du système de production ainsi que la nervosité. En effet, un intervalle  $[R, C]$  trop large peut réduire le nombre de changement des cartes kanban et par conséquent éviter le problème de la nervosité. En revanche, ceci pourra augmenter le niveau d'encours et détériorer la satisfaction des clients. En conséquence, ces paramètres doivent être adaptés aux caractéristiques du système de production. Pour cette raison, nous proposons d'optimiser ces paramètres.

Dans ce qui suit, nous nous intéressons à optimiser le vecteur des paramètres de notre stratégie de pilotage  $\vec{P}$  afin d'optimiser les performances du système de production tout en minimisant la nervosité. Avec  $\vec{P} = (K_j, E_j, R_j, C_j, FI) \forall j \in \{1, \dots, N\}$ .

### **4.3. Formulation mathématique du problème : Optimisation Multi-objectif**

Afin de mieux caractériser et résoudre le problème d'adaptation dynamique du nombre de cartes kanban tout en prenant en considération la nervosité, une formalisation mathématique s'avère intéressante.

Nous nous intéressons à trouver le paramétrage adéquat  $\vec{P}^* \in \Omega$  de notre stratégie de pilotage, décrite ci-dessus, pour adapter dynamiquement le nombre de cartes kanban dans un système en flux tiré adaptatif afin d'améliorer les performances du système de production  $F_1$

tout en minimisant la nervosité  $F_2$ . En effet, la règle de pilotage  $S$  permet de déterminer une action  $A_{li}(t)$  selon l'état courant du système  $\vec{ST}(t) = \{ST_1(t), ST_2(t), \dots, ST_j(t)\}$  à l'instant  $t$  et des paramètres  $\vec{P} = \{P_1, P_2, \dots, P_k\}$ . Ceci peut être formalisé comme suit :

$$\text{Trouver } \vec{P}^* \in \Omega \text{ tel que } F(\vec{P}^*) = \text{Min}_{s \in \Omega} \begin{cases} F_1(\vec{P}) \\ F_2(\vec{P}) \end{cases} \quad (\text{Equ.1})$$

$$\text{Avec } S: (\vec{ST}(t), \vec{P}) \rightarrow A_{li}(t), \text{ où } l \in \{1,2,3\} \text{ et } i \in \{1,2, \dots, j\} \quad (\text{Equ.2})$$

Où la notation utilisée peut être résumée comme suit :

- $t$  : Indice de temps qui représente l'instant de la prise de décision (suite à l'apparition d'un événement déclencheur).
- $J$  : Représente le nombre total des variables d'états utilisés.
- $K$  : Représente le nombre total des paramètres utilisés.
- $L$  : Représente le nombre total des actions possibles.
- $\Omega$  : Ensemble de tous les paramètres possibles et applicables au problème à résoudre.
- $S$  : Représente la règle de pilotage adoptée.
- $\vec{P} = \{P_1, P_2, \dots, P_k\}$  : Représente le vecteur de l'ensemble des paramètres de la règle de pilotage.
- $\vec{ST}(t) = \{ST_1(t), ST_2(t), \dots, ST_j(t)\}$  : Représente le vecteur de l'état du système à l'instant  $t$  et  $ST_i(t)$  l'état de la station  $i$ .
- $A_{li}(t)$  : Représente l'action de pilotage à appliquer à la station  $i$  à l'instant  $t$ .
- $F$  : Représente l'ensemble des mesures de performance à optimiser.

A chaque instant  $t$  de la prise de décision, le système de production envoie au module décisionnel un vecteur  $\vec{ST}(t)$  qui décrit l'état du système à l'instant  $t$ , par exemple : la quantité d'encours, la quantité du stock de produits finis, le nombre de clients en attente, etc. Ainsi, la règle de pilotage prend la décision en se basant sur le vecteur  $\vec{ST}(t)$  et ses paramètres  $\vec{P}$ , et renvoie l'action  $A_{li}(t) = \{A_1(t), A_2(t), A_3(t)\}$ , à appliquer à la station  $i$ .

Avec :

- $A_1(t)$  : Ajouter un Kanban.
- $A_2(t)$  : Retirer un Kanban.
- $A_3(t)$  : Consiste à ne rien changer.

Pour résumer, la règle de pilotage dépend de deux types de variables pour piloter le système de production, des variables d'états qui varient dans le temps et dépendent du système à étudier et des paramètres fixes. Ces paramètres représentent un facteur-clé de la performance de la règle de pilotage adopté et influent sur les performances du système de production.

Pour cela, nous nous intéressons à optimiser ces paramètres afin d'améliorer les performances du système de production tout en minimisant la nervosité. Notre problème d'optimisation multi-objectif peut être alors formalisé comme suit :

$$\text{Min}_{\vec{P} \in \Omega} \begin{cases} F_1(\vec{P}, \varepsilon) = \bar{f}_1 + \bar{f}_2 + \dots + \bar{f}_j \\ F_2(\vec{P}, \varepsilon) = Ner \end{cases} \quad (\text{Equ.3})$$

Avec :

- $\varepsilon$  : Représente la composante stochastique de la simulation.
- $F_1(\vec{P}, \varepsilon) = \bar{f}_1 + \bar{f}_2 + \dots + \bar{f}_j$  : Représente l'estimation de la moyenne des performances du système de production, tout en étant soumis à des effets stochastiques  $\varepsilon$ .
- $\bar{f}_j$  : Représente la moyenne des coûts de production générés dans chaque station j et qui peut être formalisée comme suit :

$$\bar{f}_j = \overline{encours_j} + \overline{PdtFinis_j} + b_j * \overline{DemandeEnRetards_j}$$

Avec :

- $\overline{PdtFinis_j}$  : correspond au stock moyen de produits finis pour chaque station j ;
- $\overline{encours_j}$  : correspond aux encours moyens pour chaque station j ;
- $\overline{DemandeEnRetards_j}$  : correspond à la moyenne des demandes en retard pour chaque station j ;

- $b_j$  : correspond aux coefficients de pénalités pour chaque demande non satisfaite dans la station  $j$ .
- $N_{er} = \frac{\text{Nombre de changement de kanbans}}{\Delta}$  : Représente la fréquence des changements de nombre de cartes kanban durant la période de la simulation  $\Delta$ .

#### 4.4. Usage de la simulation à évènement discret

Puisque nous ne disposons pas d'une fonction explicite ( $F_1$  et  $F_2$ ) pouvant être calculée analytiquement, nous proposons d'évaluer ces performances en utilisant une simulation et, plus particulièrement, une simulation à évènements discrets.

Bien que le système de production change dynamiquement, nous ne nous intéressons à prendre des décisions de pilotage qu'à un certain instant précis. Ainsi, représenter la dynamique du système consiste à représenter les instants auxquels le système change d'état. Un changement d'état du système se produit à l'occurrence des événements particuliers appelés événements déclencheurs. Ces événements peuvent se produire lorsqu'une entité (machine, ressource, kanban, unité de stockage, etc.) change d'état, ce qui correspond, par exemple, à l'arrivée d'un nouvel ordre de fabrication, l'indisponibilité d'une machine suite à une panne, la libération d'un kanban, la fin d'une opération sur une machine, l'apparition d'une tendance, etc.

Pour ce faire, la simulation va recevoir en entrée le vecteur des paramètres  $\vec{P}$  et après  $R$  répliques, fournir en sortie les mesures de performances  $F_1(\vec{P}, \varepsilon)$  et  $F_2(\vec{P}, \varepsilon)$  (voir Figure 15).

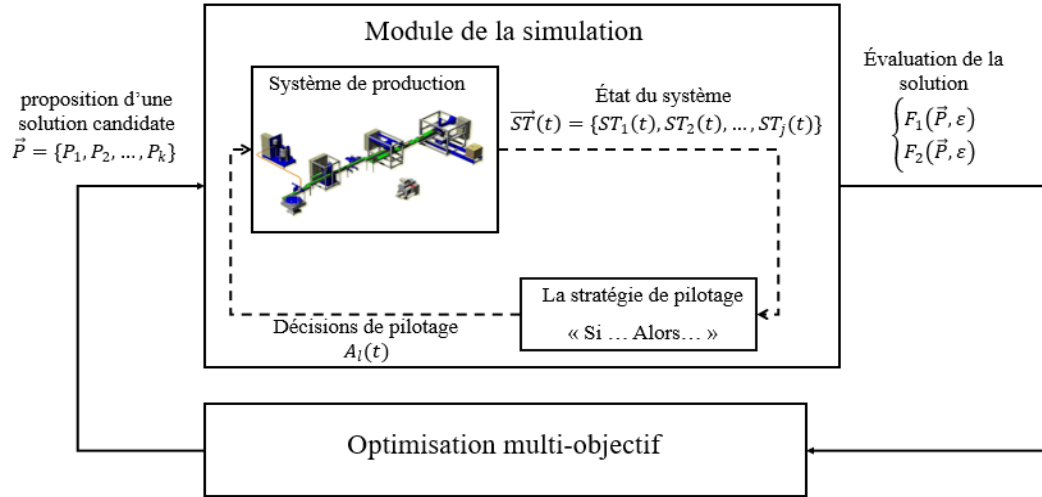


Figure 15: Optimisation multi-objectif des paramètres de pilotage et évaluation via simulation

Pour ce faire, le module de l'optimisation multi-objectif propose au module de la simulation une solution candidate qui représente, dans notre cas, le vecteur des paramètres  $\vec{P}$ . Durant la simulation, à chaque fois qu'une décision doit être prise, le système de production envoie des informations sur son état à la composante de pilotage. Cette dernière contient la stratégie de pilotage qui va déterminer les éventuelles décisions à prendre, qui seront retournées au système. Les modifications correspondantes, s'il y en a, sont ensuite appliquées sur le système simulé. Puis la simulation suit son cours jusqu'au besoin d'une nouvelle décision relançant à ce processus, et ainsi de suite jusqu'à la fin de la simulation. Ainsi, le module de la simulation évalue la stratégie de pilotage basée sur la solution proposée et après  $R$  répliques, lui renvoie les mesures de performances  $F_1(\vec{P}, \varepsilon)$  et  $F_2(\vec{P}, \varepsilon)$ .

## 5. Implémentation

Pour résoudre le problème d'optimisation multi-objectif, de nombreux algorithmes génétiques peuvent être utilisés. Parmi les plus connus dans la littérature figurent : NSGA-II (Deb, Anand et Joshi, 2002), SPEA-II (Zitzler, Laumanns et Thiele, 2001), IBEA,  $\varepsilon$ -MOEA, etc. Dans cette thèse, notre choix s'est porté sur le NSGA-II mais d'autres algorithmes génétiques peuvent être utilisés (Kalyanmoy et Others, 2001; Konak, Coit et Smith, 2006).

### 5.1. NSGAII : vocabulaire et principe de fonctionnement

NSGA-II (Non-dominated Sorting Genetic Algorithm) est une amélioration de l'algorithme NSGA (Srinivas et Deb, 1994) dotée de nouveaux mécanismes qui ont prouvé leur efficacité sur des benchmarks de la littérature (Deb, Anand et Joshi, 2002; Khare, Yao et Deb, 2003). En effet, NSGA-II est basé sur un mélange d'élitisme et de nichage dans leur étape de sélection. L'élitisme consiste à sauvegarder les meilleures solutions trouvées lors des générations précédentes (les moins dominées). Le nichage préserve de la diversité dans les solutions en favorisant celles qui sont les plus éloignées d'autres solutions dans l'espace des objectifs.

Dans cet algorithme, à chaque génération  $t$ , une population des parents  $P_t$  et une population des enfants  $Q_t$  sont réunies pour former la population mixte  $R_t$  de taille  $2S$  ( $|P_t|=|Q_t|=S$ ), comme indiqué sur la Figure 16. À chaque génération, les nouveaux individus doivent être évalués. Cette population est ensuite triée selon le principe de la dominance de Pareto et répartie en plusieurs fronts. Ce tri associe un rang de dominance à chaque individu de la population. Les individus non dominés ont un rang 1 et constituent le front  $FR_1$ . Les autres fronts  $FR_i$  sont ensuite définis récursivement en ignorant les solutions des fronts précédemment constitués. Par la suite, une nouvelle population des parents  $P_{t+1}$  de taille  $S$  est remplie par les individus appartenant aux meilleurs fronts. Lorsque le dernier front autorisé est pris en compte, il peut contenir plus de solutions que les emplacements disponibles restants dans  $P_{t+1}$ . Pour choisir les individus qui vont survivre du front et intégrer la nouvelle population, une mesure de la densité (appelée aussi mesure de surpeuplement) des individus est utilisée afin de favoriser les solutions dans les régions les moins peuplées du front et améliorer la diversité des solutions. Le processus se répète d'une génération à une autre jusqu'à la satisfaction d'un critère d'arrêt.

Comme pour tout algorithme évolutionniste, le NSGA-II nécessite un certain nombre de paramètres à définir pour assurer son bon fonctionnement :

- Une valeur de fitness doit être attribuée à chaque individu qui sera dans notre cas les coûts de production et la nervosité (à minimiser) ;
- Un ou plusieurs critères d'arrêt doivent être spécifiés, le plus commun étant un nombre maximum de générations ;
- La taille de la population et les opérateurs de croisement et mutation doivent également être précisés.



Afin d'adapter le NSGA-II à notre problème d'adaptation dynamique du nombre de cartes kanban, nous présentons dans ce qui suit les différents éléments nécessaires pour constituer une solution : le codage d'une solution, l'initialisation et les opérateurs génétiques de reproduction.

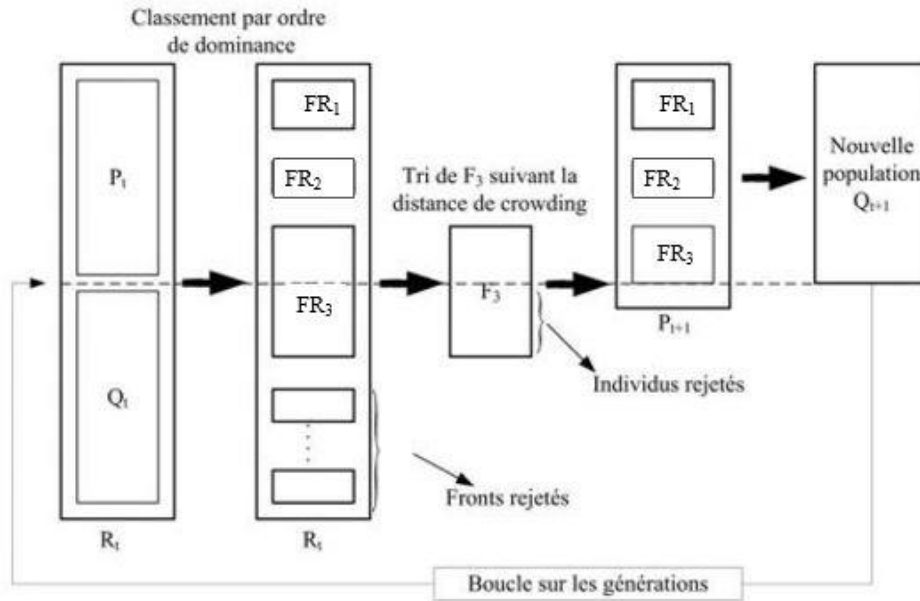


Figure 16: Schéma de l'évolution de l'algorithme NSGA-II

## 5.2. Application de NSGA-II au problème d'adaptation dynamique

### 5.2.1. Représentation vectorielle des solutions

Dans les algorithmes génétiques, chaque individu de la population est codé par un chromosome. Chaque chromosome représente un point de l'espace de recherche. L'efficacité de l'algorithme génétique va donc dépendre du choix du codage d'un chromosome. Afin de résoudre notre problème, nous optons pour un codage mixte entier et réel. En effet, le codage que nous utilisons est de longueur égale au nombre total des paramètres à optimiser et représente une configuration de la règle de pilotage. Ainsi, notre chromosome est composé de  $k=4N + 1$  gènes (où  $N$  représente le nombre de stations de la ligne de production). Chaque station  $j$  ( $j=1 \dots N$ ) contient quatre gènes entiers représentant le nombre de cartes kanban fixes  $K_j$ , le nombre de cartes supplémentaires  $E_j$ , le seuil minimum  $R_j$  et le seuil maximum  $C_j$ . Le dernier gène réel

représente la longueur de la durée de gelée  $FI$  pour l'ensemble du système. La Figure 17 représente un exemple de paramétrage de la règle de pilotage pour un système de production composé de 3 stations.

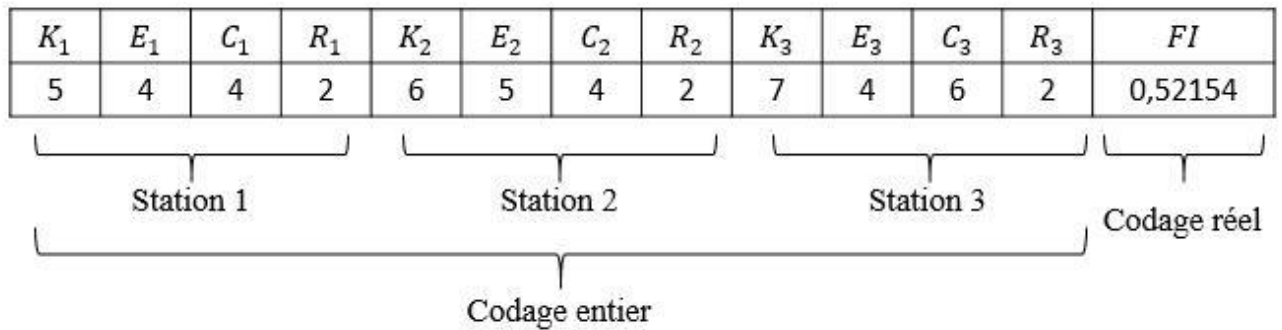


Figure 17: Représentation génétique d'un chromosome

### 5.2.2. Initialisation

La première étape de l'implantation d'un algorithme génétique est de créer une population d'individus initiaux. Plusieurs mécanismes de génération de la population initiale sont utilisés dans la littérature (Rahnamayan, Tizhoosh et Salama, 2007). De manière simple, la population initiale  $P_0$  de solutions réalisables  $S$  est générée aléatoirement.

Afin de prendre en compte les contraintes 2, 3 et 4, pour chaque solution  $S_i$  ( $i = 1 \dots S$ ), une procédure dédiée est utilisée pour préserver la faisabilité de la solution générée. En fait, la procédure consiste à sélectionner de manière aléatoire une valeur pour chaque variable dans son domaine spécifique. Afin de prendre en compte les contraintes 2, 3 et 4, les valeurs minimums qui peuvent prendre les variables  $K, E, R$  et  $C$ , respectivement, sont 1, 1, 1, 2. Si nous supposons que le nombre maximum de cartes kanban fixes et de cartes supplémentaires est respectivement  $Kmax_j$  et  $Emax_j$ , les étapes suivantes seront répétées pour chaque boucle  $j$  ( $j = 1, \dots, N$ ), indépendamment :

- 
- étape 1. Sélectionner aléatoirement une valeur  $K_j$  dans l'intervalle  $[1; Kmax_j]$
  - étape 2. Sélectionner aléatoirement une valeur  $E_j$  dans l'intervalle  $[1; Emaj_j]$
  - étape 3. Sélectionner aléatoirement une valeur  $C_j$  dans l'intervalle  $[2; K_j + 1]$
-

---

étape 4. Sélectionner aléatoirement une valeur  $R_j$  dans l'intervalle  $[1; C_j - 1]$

---

Les solutions générées sont ensuite évaluées à l'aide du modèle de la simulation afin d'estimer les valeurs des fonctions de fitness F1 et F2.

### 5.2.3. Croisement

Le croisement consiste à mélanger de manière stochastique le génotype de deux parents  $\vec{P}_1$  et  $\vec{P}_2$  afin d'obtenir deux descendants  $\vec{P}_{D1}$  et  $\vec{P}_{D2}$ . Plusieurs techniques ont été proposées dans la littérature comme : croisement 1-point, croisement 2-point, SBX, uniforme, etc. Nous choisissons dans cette thèse d'utiliser le croisement uniforme proposé par (Syswerda 1989). La procédure consiste à sélectionner de manière aléatoire, avec une probabilité  $P_c$ , les individus de la population parent qui participeront à l'opération de croisement. Par la suite, une chaîne de bits de taille  $k$  est générée aléatoirement et est utilisée pour obtenir les descendants  $\vec{P}_{D1}$  et  $\vec{P}_{D2}$ . Pour ce faire, considérons les parents suivants :

$$\vec{P}_1 = (P_{11}, P_{12}, \dots, P_{1k});$$

$$\vec{P}_2 = (P_{21}, P_{22}, \dots, P_{2k});$$

Pour chaque position  $p \in \{1, \dots, k\}$ , le choix aléatoire indique quel parent déterminera la valeur des descendants. Un exemple d'un croisement uniforme est illustré dans la Figure 18.

Par exemple, pour une position  $p$  :

- Si le bit est égal à 1, le descendant  $\vec{P}_{D1}$  prend la valeur du parent  $\vec{P}_1$  et  $\vec{P}_{D2}$  prend la valeur du parent  $\vec{P}_2$  ;
- Si le bit est égal à 0, le descendant  $\vec{P}_{D1}$  prend la valeur du parent  $\vec{P}_2$  et  $\vec{P}_{D2}$  prend la valeur du parent  $\vec{P}_1$  ;

	$K_1$	$E_1$	$C_1$	$R_1$	$K_2$	$E_2$	$C_2$	$R_2$	$K_3$	$E_3$	$C_3$	$R_3$	$FI$
Parent 1	5	4	4	2	6	5	4	2	7	4	6	2	0,52154
Parent 2	7	2	3	5	5	3	2	3	6	5	3	4	0,12548
Chaîne de bits	0	1	0	0	1	0	1	1	1	0	0	1	1
Descendant 1	7	4	3	5	6	3	4	2	7	5	3	2	0,52154
Descendant 2	5	2	4	2	5	5	2	3	6	4	6	4	0,12548

Figure 18: Un exemple de croisement uniforme

#### 5.2.4. Mutation

Dans le cas d'un codage entier, selon (Mitchell 1998), l'opérateur de mutation le plus efficace est le « creep mutation ». La mutation consiste à générer un descendant  $\vec{P}_D$  en modifiant les gènes d'un parent  $\vec{P}$ . La procédure consiste à sélectionner de manière aléatoire, avec une probabilité  $P_M$ , les individus de la population parent qui participeront à l'opération de mutation. Considérons le parent suivant :

$$\vec{P} = (P_1, P_2, \dots, P_k) ;$$

Un gène est sélectionné aléatoirement, avec une probabilité  $C_r$ , et remplacé par une valeur choisie aléatoirement dans son domaine de définition.

#### 5.2.5. Procédure de réparation

Après l'opération de croisement et de mutations, des solutions non réalisables peuvent être générées, qui ne satisfont pas les contraintes 4 et 5. Plusieurs techniques ont été proposées dans la littérature pour prendre en considération les contraintes, telles que le rejet ou la réparation de la solution, la modification de l'opérateur génétique et l'application de stratégies de pénalité. Ici, une procédure de réparation est appliquée pour convertir une solution non réalisable en une solution réalisable (voir Figure 19). Cette procédure est appliquée pour chaque station  $j \in \{1, \dots, N\}$  indépendamment comme suit:

---

étape 1. Si la contrainte (3) est non satisfaite :

---

Sélectionner aléatoirement une valeur  $C_j$  dans l'intervalle  $[2 ; K_j + 1]$  ;

étape 2. Si la contrainte (2) est non satisfaite :

Sélectionner aléatoirement une valeur  $R_j$  dans l'intervalle  $[1 ; C_j - 1]$ .

---

Avant réparation	$K_1$	$E_1$	$C_1$	$R_1$	$K_2$	$E_2$	$C_2$	$R_2$	$K_3$	$E_3$	$C_3$	$R_3$	$FI$
	5	8	4	2	5	7	8	2	7	4	6	5	0,52154

↓

Après réparation	5	8	4	2	5	7	8	2	7	4	6	5	0,52154
---------------------	---	---	---	---	---	---	---	---	---	---	---	---	---------

Figure 19: Exemple d'application de la procédure de réparation

## 6. Application de l'approche à un système Kanban adaptatif

Afin de montrer la faisabilité de notre approche, décrite dans les sections précédentes, et illustrer son potentiel, nous allons l'appliquer à un exemple d'un système Kanban adaptatif extrait de la littérature. Dans ce qui suit, nous utilisons le principe de la décomposition du système en deux sous-systèmes : physique et décisionnel pour expliquer le modèle étudié. Par la suite, nous appliquons notre approche sur ce modèle. Enfin, nous présentons et analysons les résultats numériques obtenus.

### 6.1. Description du système

Le système de production sélectionné ici, comme cas d'étude, est extrait de (Sivakumar et Shahabudeen, 2008). C'est un exemple largement repris dans la littérature sur les systèmes de production en flux tiré adaptatifs à des fins comparatives.

C'est un système composé de trois stations ( $N=3$ ) en série et produisant un seul type de produit. Chaque station de travail est dotée de 10 machines en parallèles et un air de stockage en sortie  $PF_i$  pour tout  $i \in \{1,2,3\}$ . Le délai de production est distribué selon une loi exponentielle de moyenne  $1/6$  ( $\mu=1/6$ ). Les demandes arrivent une par une selon un processus

de Poisson et aucune information n'est disponible sur la demande future ni de prévision possible. Si le stock des produits finis (ou semi-finis)  $PF_i$  est vide, les demandes sont mises en attente dans un stock  $D_i$  et le système encourt un coût de retard  $b_i$ . Ce coût est égal à 500, 800 et 1000 respectivement pour les stations 1, 2 et 3.

Ce système est basé sur quelques hypothèses, qui sont (Sivakumar et Shahabudeen, 2008) :

- Il n'existe pas de blocage au niveau des matières premières ;
- Il n'y a pas de pannes (les machines sont performantes) ;
- La stratégie FIFO (premier arrivé, premier servi) est utilisée pour l'ordonnancement des processus ;
- Le temps de transport est négligeable.

Cependant, pour qu'un système adaptatif soit performant et réactif, il doit s'adapter à différentes situations. Pour cela, nous utilisons deux versions de ce système qui diffèrent par le pattern utilisé (les deux patterns sont représentés dans la Figure 20).

- Dans la première version, les demandes arrivent selon un processus de poisson de moyenne  $\lambda$  qui évolue selon un pattern dit stable où la moyenne reste constante et égale à 8 (ou  $\lambda = 1/8$ ) tout au long de la période d'étude.
- Dans la deuxième version, les demandes arrivent selon un processus de poisson de moyenne  $\lambda$  qui évolue selon un pattern dit cyclique où la moyenne change à chaque 25 unités de temps, passant de 2 à 4 puis à 6 et finalement à 8, d'où elle recommence à 2 et ainsi de suite pour toute la durée de la période d'étude.

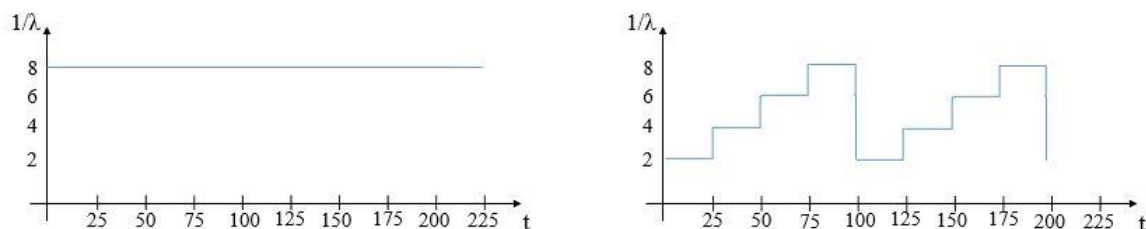


Figure 20: Patterns utilisés pour la variation de la demande dans les deux systèmes étudiés

## 6.2. Module décisionnel : fonctionnement de la stratégie de pilotage du système étudié

Sivakumar et Shahabudeen (2008) définissent l'état initial du système avant l'arrivée de toute demande. Ils partent de l'hypothèse que le stock des produits finis  $PF_3$  est plein et en attente pour être livré aux clients. Toutes les stations sont en arrêt et il n'y a aucune demande en attente.

Le fonctionnement de leur stratégie de pilotage peut être décrit comme suit :

Dès l'arrivée d'une demande client (ce qui correspond à l'évènement déclencheur de notre stratégie) à l'instant  $t$  :

- i. S'il y a un produit fini disponible dans le stock  $PF_3$ , alors la demande est satisfaite et un produit fini peut être livré au client. Sinon la demande est mise en attente dans le stock  $D_3$  ;
- ii. Dans le cas où la demande est satisfaite, le kanban attaché au produit à livrer est suspendu jusqu'à la vérification de l'état de chaque station. Les vérifications se déroulent comme suit :
  - a. Pour chaque station  $j$ , calculer le niveau de stock  $I_j(t) = PF_j - D_j$  ;
  - b. Si le niveau de stock est inférieur au seuil minimum ( $I_j(t) \leq R_j$ ) et une carte supplémentaire est disponible ( $E_j(t) > 0$ ), alors une carte supplémentaire est ajoutée à la station  $j$  et le kanban suspendu est envoyé à la station en amont pour autoriser la production d'un nouveau produit ;
  - c. Si le niveau de stock est supérieur au seuil maximum ( $I_j(t) > C_j$ ) et il existe encore une carte supplémentaire non utilisée ( $E_j(t) < E_j$ ), alors le kanban suspendu est retiré de la station  $j$  ;
  - d. Sinon le système est gardé tel qu'il est.

Les auteurs dans (Sivakumar et Shahabudeen, 2008) cherchent à optimiser le vecteur des paramètres de leur stratégie de pilotage qui est :  $\vec{P} = (K_j, E_j, R_j, C_j)$  pour tout  $j \in \{1, \dots, N\}$ ,

dont l'objectif est de minimiser un seul objectif qui est la somme des coûts générés dans chaque station  $j$  et qui peut être formalisée comme suit :

$$\text{Minimiser } F1(P) = \sum_{j=1}^N F1_j$$

$$F1_j = \overline{encours_j} + \overline{PdtFinis_j} + b_j * \overline{DemandeEnRetards_j}$$

Avec :

- $\overline{PdtFinis_j}$  : correspond au stock moyen de produits finis pour chaque station  $j$  ;
- $\overline{encours_j}$  : correspond aux encours moyens pour chaque station  $j$  ;
- $\overline{DemandeEnRetards_j}$  : correspond à la moyenne des demandes en retard pour chaque station  $j$  ;
- $b_j$  : correspond aux coefficients de pénalités pour chaque demande non satisfaite dans la station  $j$ .

Puisque nous ne disposons pas d'une fonction explicite  $F1_j$  pouvant être calculée analytiquement, nous proposons d'évaluer cette performance en utilisant une simulation et, plus particulièrement, une simulation à événements discrets.

### 6.3. Conditions des expérimentations

Tous les modèles de simulation décrits dans les sections précédentes ont été construits avec le logiciel de simulation Arena, version 14.7. Toutes les expérimentations ont été effectuées sur un PC équipé d'un processeur Intel Core i7-4770 (4 coeurs) à 3,40 GHz, de 16 Go de RAM et fonctionnant sous Windows 7 Professionnel 64 bits.

Notons que les expérimentations ont été réalisées en deux versions :

- Expérimentation 1 : Nous nous comparons à l'approche proposée par Sivakumar et Shahabudeen (2008), sans prendre en considération le problème de la nervosité, puisqu'il n'est pas pris en compte par ces auteurs. Dans le but de rendre notre comparaison aussi équitable que possible, nous utilisons la même fonction fitness ( $F1$ )



que ces auteurs et nous maintenons constante notre deuxième fonction (F2). Dans ce cas, notre problème d'optimisation multi-objectif peut être formalisé comme suit :

$$\text{Min}_{\vec{P} \in \Omega} \begin{cases} F_1(\vec{P}, \varepsilon) = \sum_{j=1}^N F1_j \\ F_2(\vec{P}, \varepsilon) = \text{Constante} \end{cases} \quad \forall j \in \{1, \dots, N\}$$

$$\text{Avec : } F1_j = \overline{\text{encours}_j} + \overline{\text{PdtFinis}_j} + b_j * \overline{\text{DemandeEnRetards}_j}$$

- Expérimentation 2 : Nous nous comparons aux mêmes auteurs tout en prenant en considération la nervosité. Dans ce cas, notre problème d'optimisation multi-objectif peut être formalisé comme suit :

$$\text{Min}_{\vec{P} \in \Omega} \begin{cases} F_1(\vec{P}, \varepsilon) = \sum_{j=1}^N F1_j \\ F_2(\vec{P}, \varepsilon) = \text{Ner} \end{cases} \quad \forall j \in \{1, \dots, N\}$$

$$\text{Avec : } F1_j = \overline{\text{encours}_j} + \overline{\text{PdtFinis}_j} + b_j * \overline{\text{DemandeEnRetards}_j}$$

Afin de réaliser ces expérimentations, nous optimisons notre vecteur de paramètres  $\vec{P} = (K_j, E_j, R_j, C_j, FI)$  avec NSGA II. Les paramètres de la configuration de NSGA-II ont été définis empiriquement et ils sont reportés dans le Tableau 2.

Tableau 2: Les valeurs des paramètres de NSGA-II.

Paramètres	Valeurs
Nombre de génération	500
Probabilité de croisement $P_c$	0.7

Probabilité de mutation $P_M$	0.3
Taux de creep $C_r$	0.05
Nombre d'exécution	5

---

Afin d'obtenir la meilleure performance possible de l'approche proposée par Sivakumar et Shahabudeen (2008), nous avons déterminé la valeur du vecteur de paramètre  $\vec{P} = (K_j, E_j, R_j, C_j) \forall j \in \{1, \dots, N\}$  avec l'optimisation via simulation par l'optimiseur OptQuest intégré au logiciel Arena. Pour plus de détails sur OptQuest, se référer à (Bradley 2007; Eskandari et al. 2011).

La signification de la différence des résultats obtenus durant nos expérimentations est testée avec le test statistique T de Student, avec un seuil de signification égal à 5%.

Pour que les résultats puissent être comparés statistiquement, nous avons défini un nombre de réplifications  $R$ , une durée de simulation  $Ds$  (unité de temps) (la même utilisée dans (Sivakumar et Shahabudeen 2008)) ainsi qu'une période de simulation transitoire  $Tr$  (unité de temps). Les valeurs de ces paramètres sont données dans le Tableau 3.

Tableau 3: Les paramètres utilisés dans les expérimentations

---

<b>Paramètres</b>	<b>Valeurs</b>
nombre de réplifications $R$	30
durée de simulation $Ds$	1000000
période transitoire $Tr$	1000

---

## 6.4. Résultats

Comme nous l'avons expliqué précédemment, notre expérimentation s'est déroulée en deux versions afin de montrer la faisabilité et le potentiel de notre approche.

Pour montrer la faisabilité de notre approche, nous nous comparons à l'approche proposée dans (Sivakumar et Shahabudeen, 2008) sans prendre en compte la nervosité et dans les mêmes conditions des variations de la demande client. Pour cela, nous utilisons le système décrit dans la section 4.1 avec des patterns stables. Par la suite, afin d'illustrer le potentiel de notre approche, nous nous comparons à la même approche mais dans des conditions différentes (des patterns cycliques) et en prenant en considération la nervosité.

Dans ce qui suit, nous faisons référence à ces deux versions de notre expérimentation par le terme « scénario ». Chaque scénario correspond à des objectifs particuliers et des conditions particulières.

- Scénario 1 : Pattern stable sans prise en compte de la nervosité ;
- Scénario 2 : Pattern cyclique avec prise en compte de la nervosité.

### 6.4.1. Scénario 1

L'évolution de la meilleure solution par rapport à la mesure des coûts générés par le système (F1) est illustrée dans la Figure 21. La courbe en rouge montre la convergence de la meilleure solution obtenue en appliquant l'approche de Sivakumar et Shahabudeen (2008), alors que la courbe en bleu illustre la convergence de la meilleure solution obtenue par notre approche durant l'optimisation.

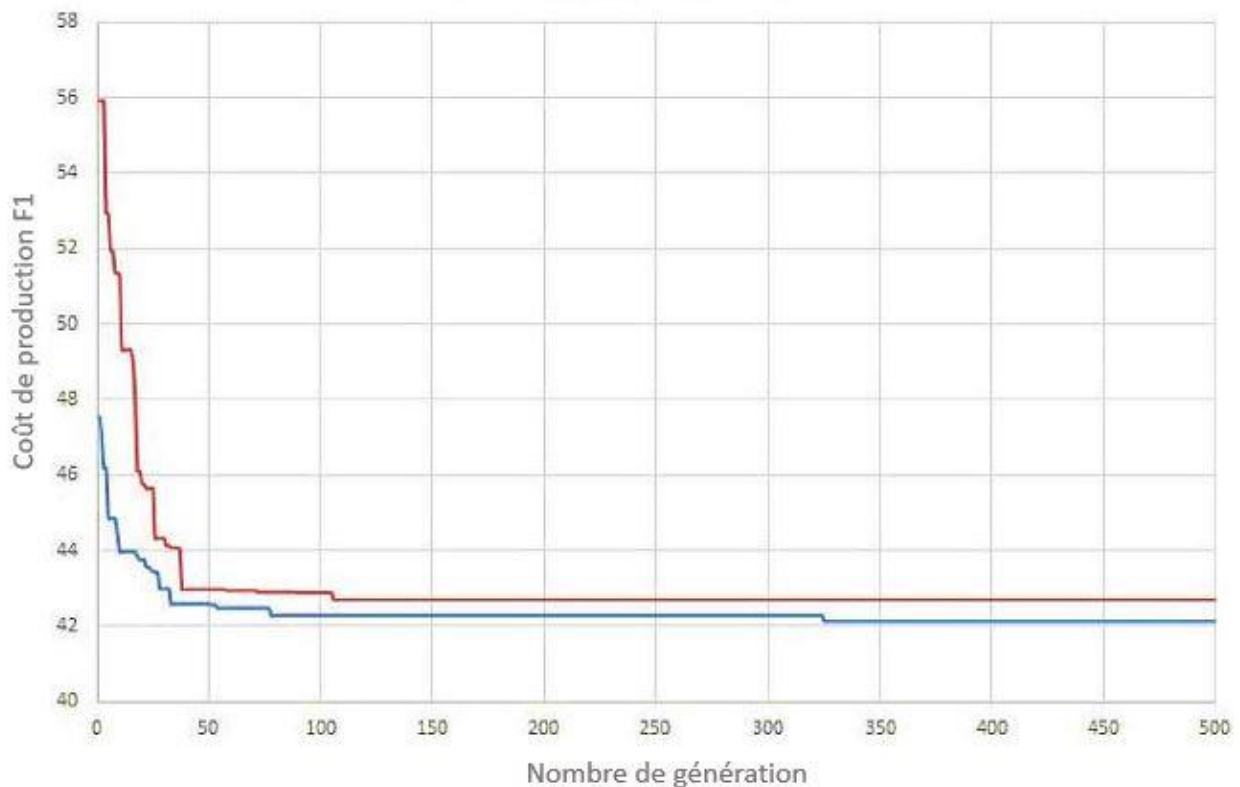


Figure 21: La convergence de la mesure de performance des coûts de production durant l'optimisation pour notre approche et l'approche proposée dans (Sivakumar et Shahabudeen, 2008)

Nous commençons par constater que les deux courbes d'optimisation montrent une bonne convergence, les résultats étant rapidement améliorés en quelques générations. Nous pouvons également constater que la convergence s'est arrêtée avant l'atteinte du nombre maximum de générations, ce qui montre qu'on a obtenu de bonnes performances. De plus, à la fin de la convergence, notre approche a obtenu un coût moyen de production de 42,3, significativement

plus bas (donc meilleur) que le coût de production moyen obtenu par l'approche de Sivakumar et Shahabudeen (2008), qui est de 42.7. Les résultats obtenus montrent la faisabilité de notre approche et son efficacité à s'adapter à un environnement dynamique et des changements dus uniquement au hasard. Ces résultats confirment aussi que, même dans la résolution du problème de façon mono-objective lorsque la nervosité n'est pas prise en compte, notre approche permet effectivement d'obtenir de bons résultats.

#### 6.4.2. Scénario 2

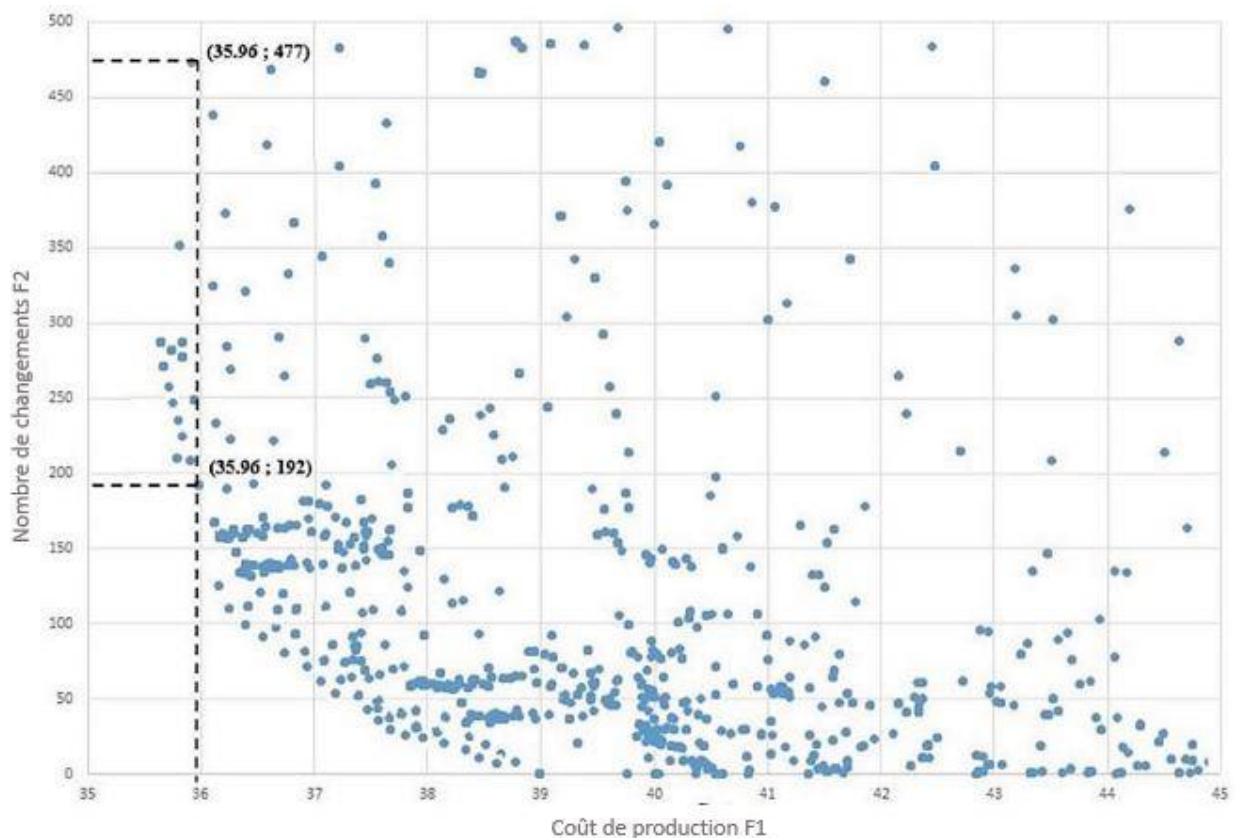


Figure 22: Une vue d'ensemble durant l'évolution de NSGA-II

Pour notre deuxième scénario, nous nous sommes intéressés à étudier l'adaptation du nombre de cartes kanban dans un contexte dynamique et changeant tout en prenant en considération le problème de la nervosité.

Une vue globale des solutions trouvées durant l'évolution de l'optimisation multi-objectif via NSGA-II est illustrée dans la Figure 22. En analysant les différentes mesures de performance ( $F1$ ,  $F2$ ) obtenues par NSGA-II, nous constatons que le même coût de production moyen peut être atteint avec des niveaux de nervosité très différents. Par exemple, nous remarquons dans la Figure 22 que le même coût de production moyen (35.96) peut être obtenu avec différents niveaux de nervosité (le nombre de changements varie de 477 à 192). Par conséquent, réduire les coûts de production tout en limitant le nombre de changements des cartes kanban est possible. Ces résultats prouvent l'importance de notre formulation multi-objectif du problème d'adaptation dynamique du nombre des cartes kanban.

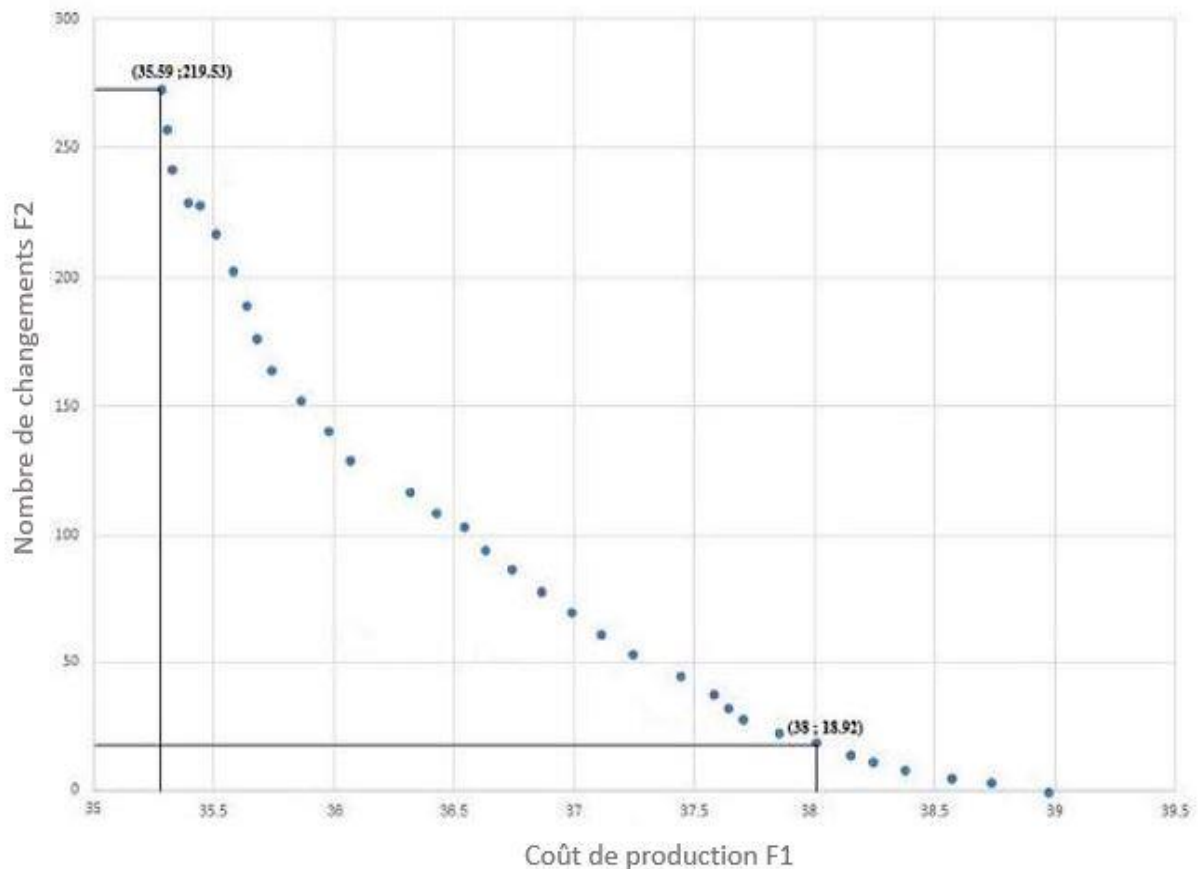


Figure 23: Le front de Pareto obtenu par notre approche

Les meilleures solutions obtenues par notre approche dans le scénario 2 sont illustrées dans la Figure 23. Elle représente un ensemble de solutions de compromis (front de Pareto) entre les

deux objectifs à optimiser qui sont : la réduction du coût moyen généré par le système (F1) et la réduction de la nervosité (F2).

En effet, dans un environnement de production dynamique et changeant, les décideurs sont confrontés à plusieurs difficultés et doivent prendre des décisions en respectant plusieurs objectifs contradictoires. Ils peuvent considérer que la réduction des coûts générés par le système (F1) est aussi importante que la réduction de la nervosité (F2), ou que la réduction des coûts est plus importante que la nervosité. Pour cette raison, le front de Pareto se présente comme un véritable outil d'aide à la décision aux décideurs. Grâce au front Pareto, les décideurs peuvent choisir la solution de compromis qui semble la mieux adaptée à leur cas particulier. Ce choix dépend fortement de la spécificité du système de production. Par exemple, si les cartes kanban peuvent être ajoutés ou retirés sans engendrer des problèmes organisationnels dans l'atelier, les critères du coût de production seront alors privilégiés. Dans un tel cas, si nous ne nous intéressons qu'à la minimisation des performances de coût, nous pouvons constater sur la Figure 23 que pour obtenir les meilleures performances du coût de production avec une moyenne de 35.59, le décideur devra modifier le nombre de cartes kanban en moyenne de 219.53 fois. Cependant, si l'ajustement du nombre de cartes kanban peut avoir un impact négatif sur l'organisation et la stabilité du système de production, une solution de compromis sera nécessaire. Nous remarquons sur la Figure 23 que le décideur peut sacrifier un peu du coût moyen de la production (de 35.59 à 38), au profil d'un gain significativement important en termes de nervosité (de 219.53 à 18.92). Cela montre qu'un petit sacrifice des coûts de production avec un gain significatif de nervosité peut être possible.

Notons que les algorithmes évolutionnaires multi-objectif peuvent générer plusieurs solutions de compromis, ce qui rend le choix de la meilleure solution finale très difficile pour le décideur. Afin de résoudre ce problème, plusieurs approches de décision multicritères ont été proposées dans la littérature pour aider le décideur à choisir la meilleure solution par rapport à ses préférences (par exemple, Fuzzy, TOPSIS, AHP, etc.) (Triantaphyllou, 2000; Velasquez et Hester, 2013).

Ces résultats montrent qu'être nerveux (changer le nombre de cartes kanban) est utile pour le système afin de minimiser les coûts de production. Cependant, dépasser un certain niveau (changer trop souvent le nombre de cartes kanban) n'a pas nécessairement un impact considérable sur le coût de la performance. En conséquence, il est important de connaître les implications pratiques de ces modifications afin de décider à quel moment il est vraiment nécessaire de modifier le nombre de Kanban. Cela prouve qu'utiliser uniquement les mesures de performances relatives aux coûts de production pour l'adaptation dynamique du nombre de cartes kanban, dans un système en flux tirés adaptatif, n'est pas suffisant. Ceci montre l'importance du front de Pareto obtenu par notre approche, qui offre au gestionnaire la possibilité de réduire les coûts de production au prix d'un nombre limité de modifications, ce qui peut réduire les risques de nervosité.

## 7. Conclusion

Dans ce chapitre, nous avons proposé une approche de pilotage adaptatif destinée aux entreprises qui ne peuvent pas émerger immédiatement vers l'industrie 4.0. Cette approche est basée sur l'optimisation multi-objectif via simulation. Nous avons aussi présenté et discuté un exemple de problème organisationnel engendré par les changements excessifs du nombre de cartes kanban, qui est le problème de la nervosité. Par la suite, Nous avons pu appliquer cette approche à un cas d'étude issu de la littérature avec différents scénarii. Ceci a permis d'illustrer l'efficacité et l'intérêt de notre approche dans un environnement de production dynamique et imprévisible. Les résultats obtenus ont montré l'intérêt du couplage de la simulation et de l'optimisation multi-objectif.

En effet, notre approche a été capable de s'adapter dans un environnement dynamique et changeant tout en prenant en considération le problème de la nervosité. Nous avons pu montrer qu'il est parfois possible d'améliorer les performances des coûts de production au prix d'un nombre limité de changements dans le nombre de cartes kanban, ce qui réduit le problème de la nervosité. De plus, la détermination d'un front de Pareto offre des solutions de compromis possibles, qui peuvent aider les gestionnaires à mieux concevoir le système de production en flux tiré adaptatif tout en prenant en considération le problème de la nervosité.



## Chapitre 4.

Apprentissage à partir du jumeau numérique et de l'optimisation multi-objectif pour le pilotage adaptatif dans l'industrie 4.0

## 1. Introduction

Aujourd'hui avec l'émergence de l'Industrie 4.0, une large communauté de chercheurs et de responsables de production s'accordent à dire que les entreprises devraient bénéficier dans les années à venir des nouvelles opportunités offertes par cette nouvelle révolution. Nous pouvons citer parmi celles-ci, l'interconnexion des machines et des systèmes au sein des sites de production, mais aussi entre eux et l'extérieur (clients, partenaires, autres sites de production). La possibilité de connaître à tout moment et à chaque étape de la ligne de fabrication le contexte de production : l'emplacement, le statut et ce qui est arrivé aux produits et aux ressources. La collecte, le stockage et la communication d'informations précises pour chacun des produits en temps réel. Ainsi, l'industrie 4.0 est une industrie productrice de données et d'informations plus variées. Pour accéder aux données et les gérer, l'usine possède différents outils de gestion, de validation ou de suivi, dirigés par des méthodes de l'Intelligence Artificielle, comme par exemple le jumeau numérique. De tels outils peuvent être exploités par des stratégies intelligentes de pilotage de flux afin d'adapter dynamiquement le nombre de cartes Kanban. Mais, comment peut-on alors les exploiter pour mieux ajuster les flux dans un environnement de production en flux tiré dynamique et incertain ?

Une solution consiste à faire appel à des techniques de l'intelligence artificielle et plus particulièrement aux méthodes d'apprentissage automatique. Cependant, dans la plupart des approches d'apprentissage classiques, l'objectif est généralement d'extraire des connaissances à partir d'exemples et/ou d'observations. Or, dans le contexte des systèmes de production en flux tirés réactifs, il est difficile de trouver de bons exemples indiquant comment adapter efficacement le nombre de cartes kanban. En plus, Les experts n'ont pas toutes les connaissances requises et ne sont donc pas en mesure de fournir des exemples ou des cas (suffisamment représentatifs) à partir desquels il serait possible d'apprendre. Ceci revient à l'état changeant dynamiquement au cours du temps de ces types de systèmes. Comment peut-on alors apprendre sans avoir à disposition un ensemble d'apprentissage de bonne qualité ?

Pour répondre à cette question, nous proposons, dans ce chapitre, une nouvelle approche de pilotage adaptatif capable d'exploiter au mieux les données disponibles sur l'évolution du

contexte de production afin de décider à quel moment et à quelle station le nombre de cartes kanban doit être modifié. Notre objectif est d'exploiter de nouveaux usages de la simulation dans le contexte de l'industrie 4.0 (on parle ici du jumeau numérique), en particulier la simulation à événements discrets, couplée à des outils d'optimisation multi-objectif et à l'intelligence artificielle afin d'aider à la décision pour la conception et le pilotage des systèmes en flux tirés adaptatifs. Pour cela, nous proposons une approche basée sur un réseau de neurones (NN) dont l'apprentissage est réalisé hors ligne, à l'aide d'un modèle de simulation couplé à l'optimisation multi-objectif (Azouz and Pierreval 2019). Contrairement aux approches classiques d'apprentissage, aucune base d'apprentissage n'est nécessaire, ce qui est important dans notre cas, car il est très difficile de fournir suffisamment d'exemples pertinents.

Ce chapitre s'organise comme suit. Dans un premier temps, nous présenterons le problème d'apprentissage dans le contexte des systèmes en flux tiré adaptatifs ainsi que les nouvelles opportunités offertes par l'industrie 4.0. Par la suite, nous fournirons les principes généraux de l'approche proposée. Une fois le problème est formalisé, nous présenterons la méthode de résolution choisie. Enfin, nous illustrerons notre approche sur différents cas d'études issus de la littérature afin de montrer son efficacité.

## **2. Apprentissage à partir d'un jumeau numérique**

L'apprentissage se présente comme un moyen possible pour aider à déterminer quand et dans quelle étape de la ligne de fabrication il faut changer le nombre de cartes kanban pour que les systèmes de production en flux tiré soient capables de répondre aux divers changements et variations de l'environnement et être réactifs afin d'améliorer les performances et satisfaire les clients rapidement. Pour cela, il nous faut pouvoir associer les décisions que l'on souhaite assister à des mesures de performance. Cependant, la difficulté de ces approches d'apprentissage réside dans l'élicitation des connaissances dont doivent être dotés ces systèmes afin de piloter efficacement les systèmes de production. En effet, la recherche des connaissances pertinentes et de bons exemples qui reflètent au mieux le comportement dynamique et incertain du système de production constitue, jusqu'à aujourd'hui, un écueil majeur pour leur mise en œuvre.

Comme identifié dans (Belisário 2015; Mouelhi-Chibani 2009), il existe plusieurs travaux qui sont basés sur l'apprentissage à partir de la simulation (voir Figure 24), appliqués aux systèmes de production puisque celle-ci est capable de fournir des informations extrêmement utiles sur leur comportement. Ceci revient à la capacité de la simulation à modéliser le comportement dynamique du système de production. En effet, la simulation permet de représenter et ainsi de prédire le comportement dynamique d'un système dans différentes situations (y compris des situations inhabituelles). Il est alors possible d'apprendre à partir des expériences menées à l'aide de modèles de simulation. Cependant, la plupart de ces travaux utilisent la simulation pour la génération d'exemples pour l'apprentissage. Pour cela, ils ont associé des stratégies de décision aux performances afin de générer des exemples d'apprentissage et extraire les connaissances nécessaires et donc l'expertise recherchée. Or, dans le contexte des systèmes de production, il est difficile de générer des exemples d'apprentissage distinguant les bonnes décisions des mauvaises décisions au regard d'un objectif de performance. Ceci devient encore plus difficile lorsque l'on cherche à prendre des décisions en temps réel.

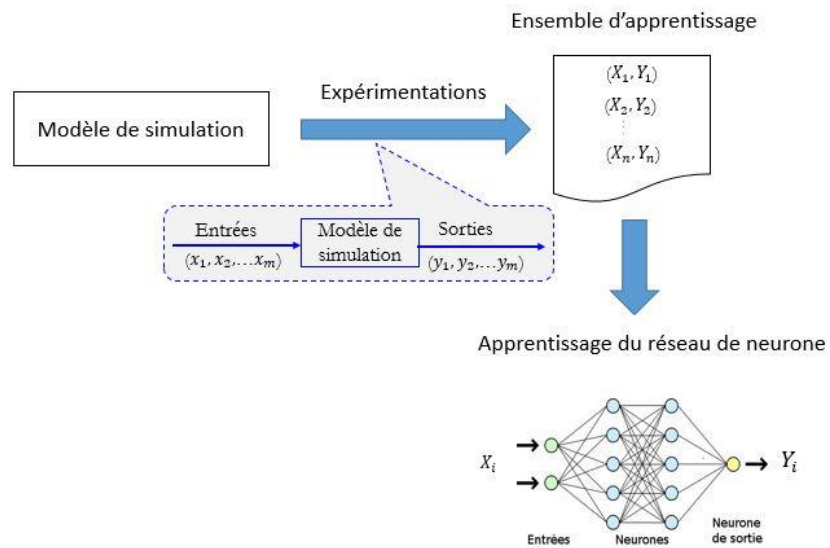


Figure 24: Apprentissage du réseau de neurones à partir d'ensemble généré par simulation

En effet, les méthodes d'apprentissage sont très sensibles à la qualité de la base d'apprentissage : les résultats obtenus restent très dépendants des données à partir desquelles les connaissances sont extraites. Ainsi, leur usage nécessite que l'on soit capable non seulement de générer des exemples (ce qui peut effectivement être fait par simulation), mais surtout d'en générer des bons. Or, dans le contexte des systèmes de production dynamiques et incertains, les conséquences d'une décision ne sont pas facilement visibles et mesurables une fois celle-ci mise en place. Comment savoir sur-le-champ si ajouter une carte kanban dans un système en flux tiré est une bonne décision ?

En effet, l'état du système de production évolue dynamiquement et de façon incertaine avec le temps tout comme l'environnement dans lequel il opère. De plus, les décisions du pilotage sont dépendantes de cette évolution. Par conséquent, l'impact des décisions prises en temps réel est difficile à évaluer immédiatement ou dans le très court terme, puisque ces dernières ne sont pas uniques mais s'enchaînent au cours du temps et sont dépendantes des anciennes décisions ainsi qu'aux décisions qui vont être prises dans le futur. C'est cet enchaînement de décisions qui s'avère être performant ou non, et non pas chaque décision à part. En effet, l'aspect dynamique du système de production rend difficile la tâche d'évaluation intrinsèque à la génération d'exemples pour l'apprentissage.

Il nous faut donc être capable de générer des connaissances relatives au fonctionnement d'un système de production dynamique et imprévisible où aucun ensemble d'apprentissage ne peut être conçu. Mais comment peut-on le faire ?

C'est ici où le jumeau numérique peut servir à faire apprendre aux systèmes comment s'adapter aux fluctuations de l'environnement et par conséquence, créer un outil d'aide à la décision performante dans un milieu virtuel et l'utiliser par la suite dans le monde réel.

Comme défini dans le Chapitre 2, les jumeaux numériques désignent une représentation numérique virtuelle d'un système physique. Ils utilisent les technologies de l'IoT pour suivre et piloter le fonctionnement de la ligne de production. En effet, l'IoT permet, via des systèmes d'identification électronique et des dispositifs mobiles sans fil, de suivre les flux de production en temps réel et ainsi de pouvoir récupérer et stocker toutes les informations de l'environnement

de production, par exemple, l'état des produits, les comportements des clients, les défaillances, etc. De plus, le jumeau numérique permet de simuler le dynamique du système afin d'en optimiser son fonctionnement hors ligne. Il est alors possible de s'appuyer sur le jumeau numérique pour apprendre au système à s'adapter.

Nous nous intéressons donc à comment apprendre au système la façon dont il doit adapter dynamiquement le nombre de cartes kanban dans les systèmes en flux tirés adaptatifs, sans la disponibilité d'une base d'apprentissage, afin d'améliorer les performances du système de production tout en minimisant le problème de la nervosité.

### **3. Approche proposée : Apprentissage à partir du jumeau numérique et l'optimisation Multi-objectif**

#### **3.1. Principe général de l'approche**

Afin de décider à quel moment et à quelle station le nombre de cartes kanban doit être modifié, nous proposons une approche basée sur un réseau de neurones multicouches (MLP-NN) dont l'apprentissage est réalisé hors ligne, à l'aide d'un modèle de simulation. Contrairement aux approches classiques, aucune base d'apprentissage n'est nécessaire, ce qui est important dans notre cas.

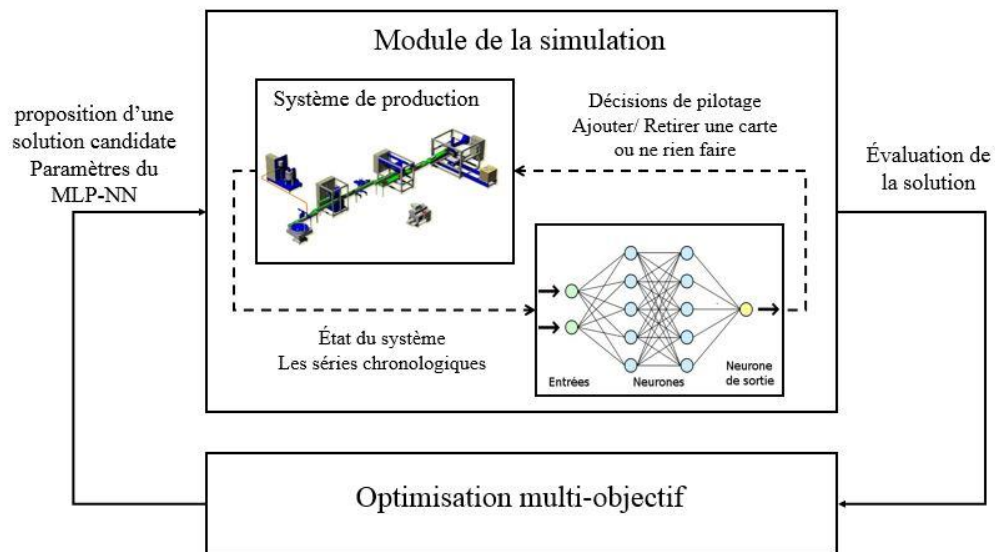


Figure 25: L'interaction entre le MLP-NN, la simulation et l'optimisation multi-objectif

L'idée est d'apprendre à partir d'un modèle et plus précisément d'un modèle de simulation à évènement discret. En effet, comme décrit dans la Figure 25, le modèle de simulation fournit au MLP-NN des informations sur l'état de l'atelier ainsi que la série chronologique des arrivées de la demande client. En se basant sur ses informations, le MLP-NN lui renvoie les décisions à prendre afin d'ajuster les flux. D'autre part, le modèle de simulation permet d'évaluer si la décision prise par le MLP-NN est bonne ou pas par rapport aux performances qu'on cherche à optimiser. Un diagramme de séquence décrivant l'interaction entre la simulation, l'optimisation et le MLP-NN est présenté dans la Figure 26.

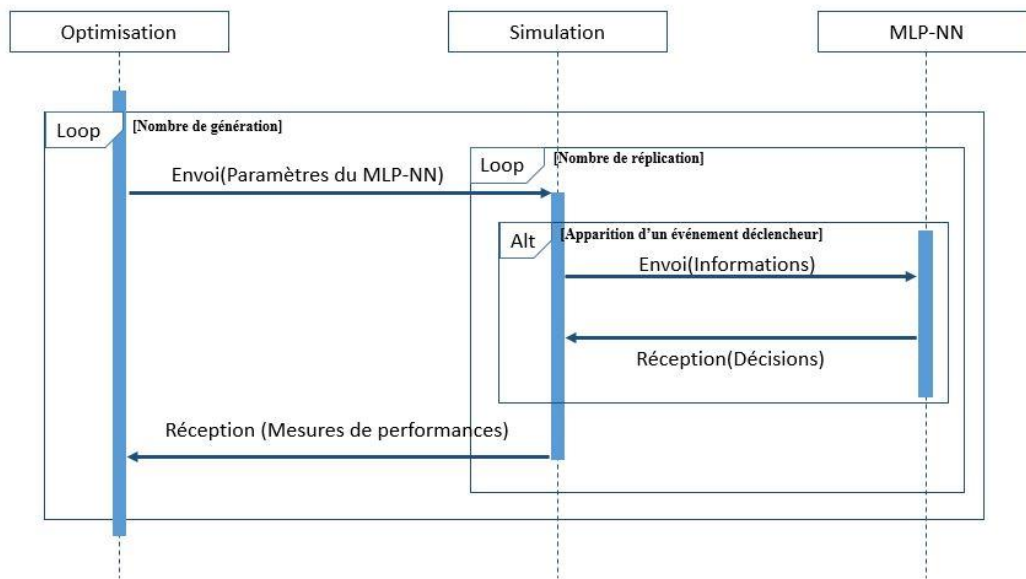


Figure 26: Diagramme de séquence décrivant les interactions dans notre approche

Au départ, le MLP-NN prend de mauvaises décisions car il n'est pas encore formé. Par conséquent, la simulation obtient de mauvaises performances. Pour cela, nous devons déterminer le vecteur de paramètres (par exemple, les poids, les paramètres sigmoïdes, etc.) pouvant permettre au MLP-NN de prendre de bonnes décisions. Pour cette raison, nous suggérons de déterminer ces paramètres à l'aide de la technique d'optimisation multi-objectif via simulation.

Pour cela, afin d'obtenir les bonnes décisions et optimiser les performances du système de production, nous avons formalisé notre problème d'apprentissage du réseau de neurones comme un problème d'optimisation multi-objectif via simulation.

### 3.2. Apprentissage du réseau de neurones par simulation et optimisation multi-objectif

En effet, nous cherchons à apprendre au MLP-NN à déterminer une action  $A_1(t)$  selon l'état courant du système  $\vec{ST}(t)$  à l'instant  $t$  et la série chronologique des arrivées de la demande clients SC, comme décrit dans la Figure 27.



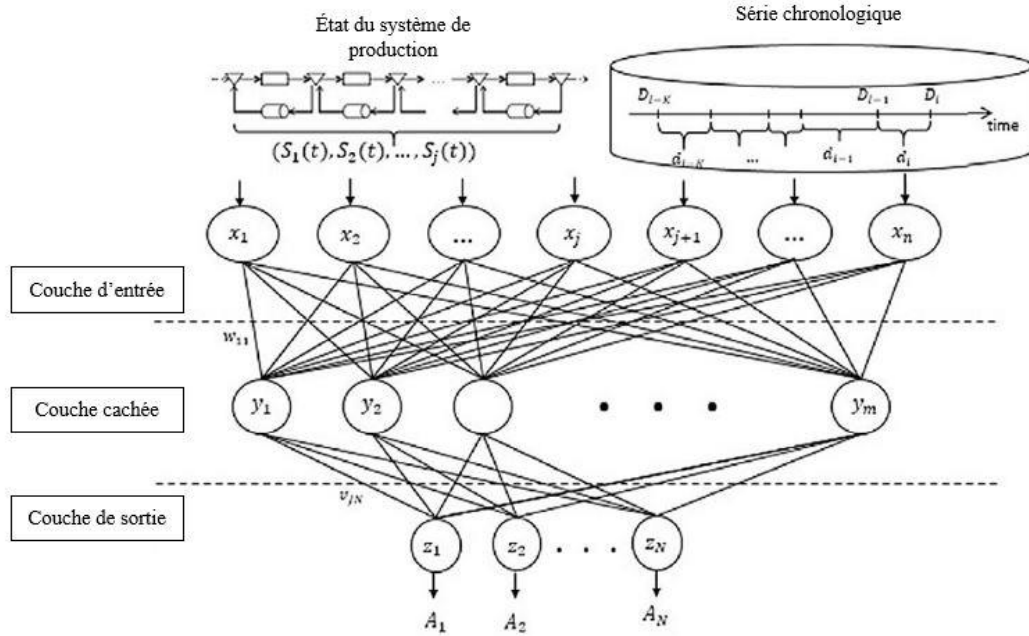


Figure 27: La transmission des informations dans le MLP-NN pour la prise de décision

Pour cela, nous avons formalisé notre problème d'apprentissage du réseau de neurones à partir d'un modèle de simulation à évènement discret comme suit :

Nous avons considéré le vecteur  $\vec{ST}(t)$  qui décrit l'état courant du système à chaque instant  $t$  de prise de décision avec :

$$\vec{ST}(t) = \{SC, ST_1(t), ST_2(t), \dots, ST_j(t)\}$$

Afin de prendre en considération l'évolution de la variation de la demande client, le vecteur  $\vec{ST}(t)$  inclut aussi le vecteur SC qui décrit la série chronologique des arrivées des demandes clients (l'historique) :

$$SC = \{d_i, d_{i-1}, \dots, d_{i-k}\}$$

Nous définissons la série chronologique (ou temporelle) comme un ensemble d'observations à des dates différentes. Avec :

- $d_i$  représente l'intervalle de temps entre deux arrivées successives des demandes clients  $D_i$  et  $D_{i-1}$ ;
- $k$  la taille de la série chronologique.

En fonction des informations reçues, le MLP-NN calcule la valeur  $y_j$  et  $z_l$  de chaque neurone dans la couche cachée et dans la couche de sortie, respectivement, comme suit :

$$y_j = f\left(\sum_{i=1}^E W_{ij}X_j\right)$$

$$z_l = f\left(\sum_{j=1}^m V_{jl}y_j\right)$$

La sortie du MLP-NN  $z_l$  correspond à l'action de pilotage  $A_l(t) = \{A_1(t), A_2(t), A_3(t)\}$  à appliquer, à l'instant  $t$ , à la station  $l$  de la ligne de production.

Où la notation utilisée peut être résumée comme suit :

- $y_j$  : La valeur de sortie du nœud  $j$  ( $j = 1, 2, \dots, m$ ) dans la couche cachée ;
- $z_l$  : La valeur de sortie du nœud  $l$  ( $l = 1, 2, \dots, N$ ) dans la couche de sortie;
- $w_{ij}$ : Le poids de l'arc reliant le nœud de la couche d'entrée  $i$  ( $i = 1, 2, \dots, E$ ) avec le nœud  $j$  ( $j = 1, 2, \dots, m$ ) de la couche cachée;
- $v_{jl}$ : Le poids de l'arc reliant le nœud de la couche cachée  $j$  avec le nœud  $l$  ( $l = 1, 2, \dots, N$ ) de la couche de sortie ;
- $m$  : Le nombre total des nœuds dans la couche cachée ;
- $E$  : Le nombre total des nœuds dans la couche d'entrée ;
- $N$  : Le nombre total des nœuds dans la couche de sortie ;

- $f$ : Représente la fonction d'activation ;
- $A_1(t)$  : Ajouter un ou plusieurs Kanban(s) ;
- $A_2(t)$  : Retirer un ou plusieurs Kanban(s) ;
- $A_3(t)$  : Consiste à ne rien changer.

Les décisions prises sont basées sur une logique qui est le raisonnement du MLP-NN. Ce raisonnement dépend fortement de ses caractéristiques : sa topologie (nombre de couche cachée, nombre de neurones dans chaque couche, etc.), ses poids et ses paramètres (le type des fonctions d'activation). Ce sont eux qui déterminent l'évolution et la performance du réseau en fonction de ses informations d'entrée.

Pour cela, nous nous intéressons à optimiser le vecteur des paramètres  $\vec{W}$  du MLP-NN, afin d'améliorer les performances du système de production  $F_1$  tout en minimisant la nervosité  $F_2$ . Notre problème d'optimisation multi-objectif peut être alors formalisé comme suit :

$$\underset{\vec{W} \in \Omega}{\text{Min}} \begin{cases} F_1(\vec{W}, \varepsilon) = \bar{f}_1 + \bar{f}_2 + \dots + \bar{f}_c \\ F_2(\vec{W}, \varepsilon) = Ner \end{cases}$$

Avec  $\vec{W} = \{w_{ij}, v_{jl}; i (i = 1, 2, \dots, E), j (j = 1, 2, \dots, m), l (l = 1, 2, \dots, N)\}$ , représente le vecteur des coefficients synaptiques (poids) du MLP-NN.

## 4. Implémentation

### 4.1. Optimisation Multi-objectif

Afin de résoudre le problème d'optimisation multi-objectif, nous utilisons l'algorithme évolutionnaire NSGA-II qui a été décrit dans le Chapitre 3. Pour adapter le NSGA-II à notre problème d'apprentissage du MLP-NN, nous présentons dans ce qui suit les différents éléments nécessaires pour constituer une solution : le codage d'une solution, l'initialisation et les opérateurs génétiques de reproduction.

### 4.1.1. Représentation vectorielle des solutions

Nous choisissons un codage réel pour optimiser les paramètres du MLP-NN. En effet, le codage que nous utilisons est de longueur égale au nombre total des poids à optimiser et représente une configuration du MLP-NN. Ainsi, notre chromosome est composée de  $S$  gènes, avec :

$$S = (E + 1) * m$$

La Figure 28 représente un exemple de configuration MLP-NN avec trois couches (deux neurones dans la couche d'entrée, deux neurones dans la couche cachée et un neurone dans la couche de sortie).

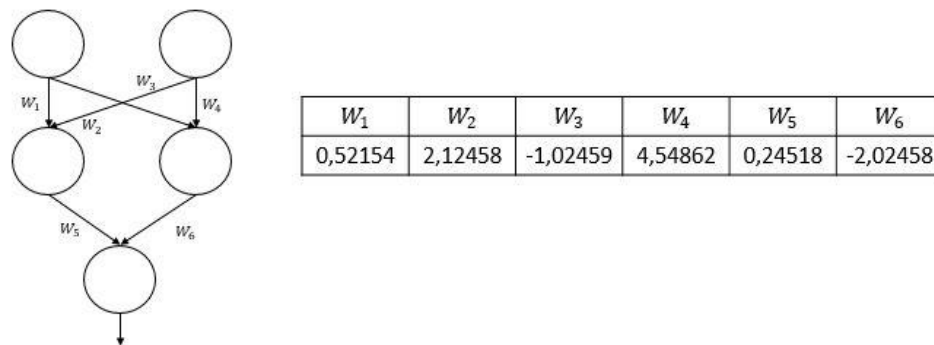


Figure 28: Un exemple d'une représentation vectorielle d'un MLP-NN

### 4.1.2. Initialisation

La génération de la population initiale a un grand impact sur la convergence de l'algorithme évolutionnaire. En effet, si la population initiale contient de bonnes solutions, la convergence sera améliorée. Pour le souligner, nous utilisons deux stratégies pour générer la population initiale :

- Stratégie S1 : Consiste à générer aléatoirement la population initiale ;
- Stratégie S2 : Consiste à utiliser un ensemble d'apprentissage construit avec des règles de pilotage adaptatif issues de la littérature. Pour cela, un apprentissage supervisé hors ligne est effectué à l'aide des ensembles d'apprentissage générés lors des expériences de simulation qui sont construits comme suit :

Au cours d'une simulation, chaque fois que la règle de pilotage adaptatif est appliquée, l'état du système au moment de la décision ainsi que la décision prise par la règle de pilotage sont enregistrés en tant qu'observation, de manière à constituer la base d'apprentissage. L'ensemble obtenu est utilisé pour former le MLP-NN en utilisant l'algorithme de rétro-propagation et les poids résultants forment la population initiale  $P_0$ .

### 4.1.3. Opérateurs génétiques de reproduction

#### Opérateur de croisement

Nous avons utilisé deux types de croisements qui sont largement connus dans la littérature : le croisement un-point et deux-points. Un exemple de croisement un-point est illustré dans la Figure 29. Le croisement à un point sélectionne de manière aléatoire le point de croisement dans les deux parents, puis permute les segments correspondants, y compris les points de croisement pour obtenir les descendants.

Point de croisement

Parent 1	-0,25638	1,02544	2,25146	-3,63598	5,78165	0,21546
Parent 2	0,52154	2,12458	-1,02459	4,54862	0,24518	-2,02458
Descendant 1	-0,25638	1,02544	2,25146	4,54862	0,24518	-2,02458
Descendant 2	0,52154	2,12458	-1,02459	-3,63598	5,78165	0,21546

Figure 29: Un exemple de croisement un-point

## Opérateur de mutation

Plusieurs opérateurs de mutation peuvent être utilisés dans notre cas. Nous avons opté pour la mutation gaussienne, appliquée avec une probabilité  $P_m$ , qui a été largement utilisée dans la littérature et a été prouvée performante (comme rapporté par exemple dans (Deb et Deb, 2014; Dai, Yuan et Wu, 2017)). Pour un parent choisi au hasard, chaque gène  $x_i = (x_1, \dots, x_n)$  est modifié avec une probabilité  $P_r$ , tandis que  $x_i$  est mis à jour par une étape de mutation gaussienne. Dans l'étape de mutation gaussienne, une valeur mutée  $x_{new,i}$  est générée par une fonction gaussienne  $N(x_i, \sigma)$  de moyenne  $x_i$  et de variance  $\sigma$ . Ensuite,  $x_i$  est mis à jour comme suit:

$$x_{new,i} = \text{Min}(\text{Max}(N(x_i, \sigma), a), b)$$

Ainsi, la valeur mutée  $x_{new,i}$  est bornée dans  $[a, b]$ , où  $a$  et  $b$  sont les bornes supérieure et inférieure des variables de décision de notre problème d'optimisation multi-objectif.

### 4.2. Implémentation logicielle de l'approche

Afin d'implémenter notre approche et assurer l'interconnexion entre la simulation, l'optimisation et le MLP-NN, nous nous sommes servis de deux logiciels :

- Arena, version 14.7 : Simulation à événements discrets ;
- Interfaces C++ sous Visual Studio 2015 : Développement de NSGA-II et MLP-NN.

En effet, Arena possède plusieurs fonctionnalités, par exemple :

- Communiquer avec des programmes et des logiciels externes (Excel, etc.) ;
- Lire et écrire des données dans des fichiers ;
- Utiliser des fichiers DLL permettant l'ajout de code utilisateur externe sous forme de bibliothèque dynamique ;
- Etc.

De plus, le logiciel peut être utilisé de façon « externe » en une ligne de commande (application console) ou à travers un programme qui y fait appel.

Nous avons alors tiré profit de ces fonctionnalités offertes pour assurer l'échange d'informations entre les différents modules de notre approche, comme illustré dans la Figure 30.

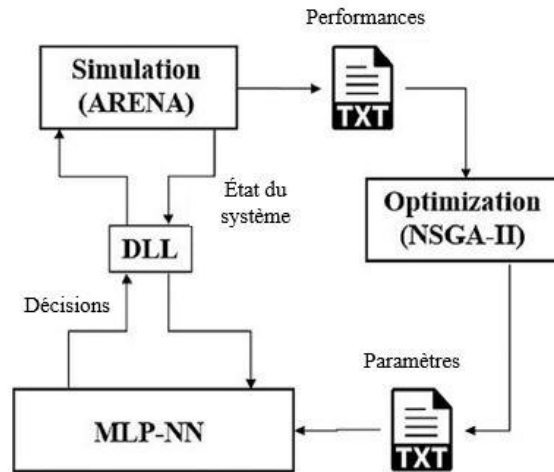


Figure 30 : La connexion entre Arena, NSGA-II et MLP-NN

Les fichiers DLL permet l'exécution d'un certain nombre de calculs externes qui sont effectués pendant la simulation et qui peuvent accéder aux variables et attributs déclarés dans le modèle de la simulation. Pour cela, le fichier DLL contient notre MLP-NN qui communique directement avec la simulation via des fonctions prédéfinies (par exemple : `cevent ()`, `get ()`, `set ()`, etc.). Il reçoit l'état du système à partir de la simulation et renvoie la décision prise par le MLP-NN.

D'autre part, le logiciel peut être utilisé de façon « externe » en une ligne de commande (application console) ou à travers un programme qui y fait appel. Pour ce faire, il faut compiler le modèle de simulation initialement au format « .doe », ce qui va générer automatiquement un fichier exécutable par Siman au format « .p ». Ainsi, ce fichier est appelé par l'algorithme NSGA-II, développé en C++, en ligne de code.

## 5. Exemples d'application

Afin de montrer la faisabilité de notre approche, décrite dans les sections précédentes et illustrer son potentiel, nous allons l'appliquer sur un système ConWIP et un système Kanban. Puis, nous allons les comparer à différentes approches de pilotages adaptatifs des systèmes en flux tirés bien reportées dans la littérature (Tardif et Maaseidvaag, 2001; Sivakumar et Shahabudeen, 2008; Xanthopoulos, Koulouriotis et Gasteratos, 2017), ainsi qu'aux systèmes Kanban et ConWIP traditionnel (où le nombre de Kanban reste fixe). Dans ce qui suit, nous présentons le modèle étudié pour chaque système. Par la suite, nous appliquons notre approche sur ces modèles. Enfin, nous présentons et analysons les résultats numériques obtenus.

## **5.1. Cas du système ConWIP**

### **5.1.1. Description du système**

Le système de production considéré ici, comme cas d'étude, est décrit dans (Tardif et Maaseidvaag, 2001). C'est un exemple largement repris dans la littérature sur les systèmes de production en flux tirés adaptatifs à des fins comparatives.

C'est un système mono-produit composé d'une seule station de travail ( $N=1$ ) qui est doté de 10 machines en parallèle, comme illustré dans la Figure 31. La station est suivie d'un air de stockage en sortie  $PF$ . Le délai de production est distribué selon une loi exponentielle de moyenne  $1/6$  ( $\mu=1/6$ ). Les demandes arrivent une par une selon un processus de Poisson et aucune information n'est disponible sur la demande future ni de prévision possible. Si le stock des produits finis  $PF$  est vide, les demandes sont mises en attente dans un stock  $D$  et le système encourt un coût de retard  $b$  égal à 1000 fois plus élevé que celui des encours (stock final compris).



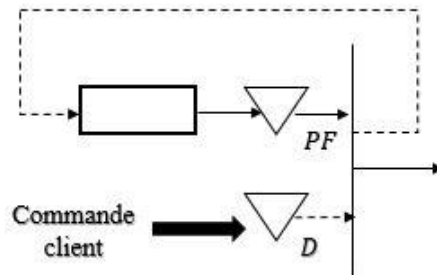


Figure 31: Le modèle ConWIP étudié (Tardif et Maaseidvaag, 2001)

Les demandes arrivent selon un processus de poisson de moyenne  $\lambda$  qui évolue selon un pattern dit triangulaire (voir Figure 32) qui exprime la variation de la demande des clients où la moyenne change à chaque 75 unités de temps, passant de 3 à 7 puis redescend à 3 et ainsi de suite pour toute la durée de la période d'étude.

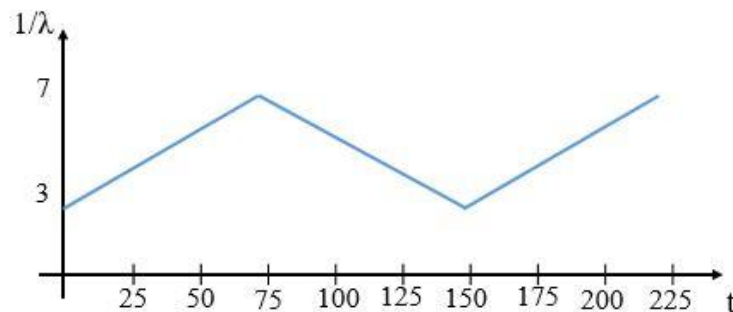


Figure 32: Patterns triangulaires utilisés pour la variation de la demande clients

### 5.1.2. Configuration du MLP-NN

Dans le cas du système ConWIP, nous utilisons un MLP-NN composé de trois couches. La couche d'entrée contient 7 nœuds ( $E=7$ ), qui représentent l'état actuel du système de production à l'instant  $t$  de la prise de décision  $\vec{ST}(t)$  et la série chronologique des arrivées de la demande clients  $SC$ , comme décrit dans la Figure 33.

L'état du système inclut la quantité d'encours WIP ( $t$ ) dans la ligne de production à l'instant  $t$ , la quantité de produits finis FP ( $t$ ) et le nombre de demande clients en attente B ( $t$ ). La série

chronologique SC est composée de quatre derniers intervalles enregistrés ( $K=4$ ). La couche de sortie nous donne la décision à appliquer au système de production (ajouter/retirer un kanban, ne rien faire). Afin de réduire le nombre de poids à optimiser, nous avons utilisé une seule couche cachée dont le nombre de nœuds est déterminé empiriquement.

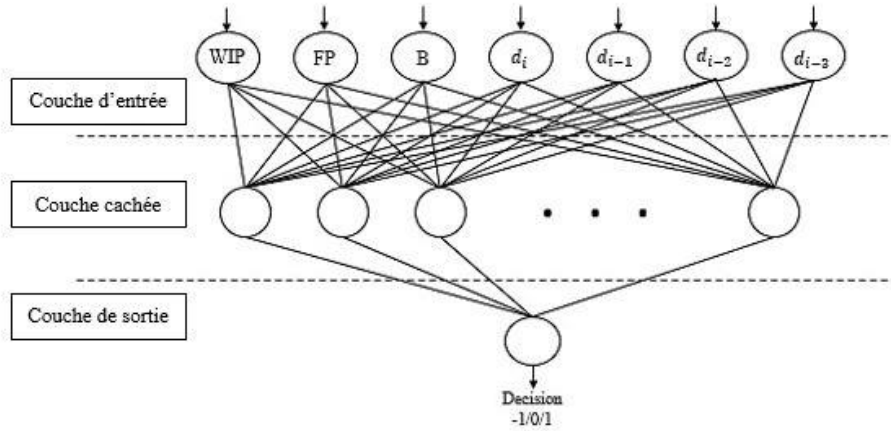


Figure 33: La configuration du MLP-NN dans le cas d'un système ConWIP

### 5.1.3. Conditions des expérimentations

Afin de mieux évaluer nos résultats, nous comparons notre approche d'une part à deux approches de pilotage adaptatifs d'un système ConWIP (Tardif et Maaseidvaag, 2001; Xanthopoulos, Koulouriotis et Gasteratos, 2017), et d'autre part à un système ConWIP traditionnel.

Dans le but de rendre notre comparaison aussi équitable que possible, nous procédons au même principe d'expérimentation utilisée dans le Chapitre 3.

Dans un premier temps, nous nous comparons aux différentes approches, sans prendre en considération le problème de la nervosité puisqu'il n'est pas pris en compte par ces auteurs. Pour cela, nous utilisons la même fonction coût (F1) que ces auteurs et nous maintenons constante notre deuxième fonction (F2).

Puis, dans un deuxième temps, Nous nous comparons aux mêmes auteurs tout en prenant en considération la nervosité. Dans ce cas, notre problème d'optimisation multi-objectif peut être formalisé comme suit :

$$\underset{\vec{W} \in \Omega}{\text{Min}} \begin{cases} F_1(\vec{W}, \varepsilon) = \overline{encours} + \overline{PdtFinus} + b * \overline{DemandeEnRetards} \\ F_2(\vec{W}, \varepsilon) = Ner \end{cases}$$

Afin de réaliser ces expérimentations, nous optimisons notre vecteur de poids  $\vec{W}$  avec NSGA II. Les paramètres de la configuration de NSGA-II ont été définis empiriquement et ils sont reportés dans le Tableau 4.

Tableau 4: Les valeurs des paramètres de NSGA-II pour le problème d'apprentissage

Paramètres	Valeurs
Nombre de génération	500
Probabilité de croisement $P_c$	0.6-1.0
Probabilité de mutation $P_M$	0.2
Indice de distribution pour croisement	10.0
Indice de distribution pour mutation	20.0

Dans le cas du ConWIP traditionnel, l'optimisation du système consiste à trouver le nombre de kanban qui minimise les coûts générés par le système F1, et ce sans aucune carte supplémentaire. D'autre part, dans le cas du ConWIP adaptatif, les approches de pilotage proposées par (Tardif et Maaseidvaag, 2001; Xanthopoulos, Koulouriotis et Gasteratos, 2017) dépendent des paramètres spécifiques qui sont résumés dans le Tableau 5.

Tableau 5: Les paramètres des approches de comparaisons

Approches	Acronymes	Paramètres
(Tardif et Maaseidvaag, 2001)	ACS (Adaptive ConWIP System)	A, E, R et C
(Xanthopoulos, Koulouriotis et Gasteratos, 2017)	AEK (Adaptive Extended Kanban)	B, A1 et X
ConWIP Traditionnel	TCS (Traditionnal ConWIP System)	K

Afin d'obtenir la meilleure performance possible de ces approches (ACS, AEK et TCS), nous déterminons la valeur de ses paramètres avec l'optimisation via simulation par l'optimiseur OptQuest intégré au logiciel Arena.

La signification de la différence des résultats obtenus durant nos expérimentations est testée avec le test statistique T de Student, avec un seuil de signification égal à 5%.

Pour que les résultats puissent être comparés statistiquement, nous avons défini un nombre de réplifications  $R$ , une durée de simulation  $Ds$  (unité de temps) (la même utilisée dans (Tardif et Maaseidvaag, 2001)) ainsi qu'une période de simulation transitoire  $Tr$  (unité de temps). Les valeurs de ces paramètres sont les mêmes que nos expérimentations dans le Chapitre 3 (Voir Tableau 3).

#### 5.1.4. Résultats

Le tableau 6 présente les meilleures performances en termes de coûts de production obtenues par les approches TCS, ACS, AEK et notre approche d'apprentissage via simulation et optimisation multi-objectif (NN-MOSO). Dans ce tableau, nous pouvons voir que le résultat obtenu par notre approche NN-MOSO est significativement meilleur que les résultats obtenus par ACS, AEK et TCS.

Ces résultats montrent que notre réseau de neurones MLP-NN a acquis de très bonnes connaissances dans un environnement dynamique et complexe, grâce à l'optimisation multi-objectif via simulation.

Tableau 6: Résultats des meilleures performances obtenues pour le cas d'un système ConWIP

---

	TCS	ACS	AEK	NN-MOSO
Performance des coûts générés par le système (F1)	7.1839	5.9987	6.4215	5.6681

---

La convergence de la mesure de la performance des coûts de production au cours du processus d'apprentissage de notre approche NN-MOSO est illustrée dans la Figure 34.

La courbe rouge représente la convergence des performances basée sur la stratégie S2 (stratégie d'initialisation décrite dans la section 4.1.2), tandis que la seconde (la courbe bleue) représente la convergence du coût basée sur la stratégie S1 (la stratégie d'initialisation décrite dans la section 4.1.2). Comme prévu, pour la stratégie S1, nous constatons que l'apprentissage est un peu lent. Cependant, pour la stratégie S2, les performances obtenues durant les premières générations sont plus raisonnables et les résultats sont rapidement améliorés par la suite.

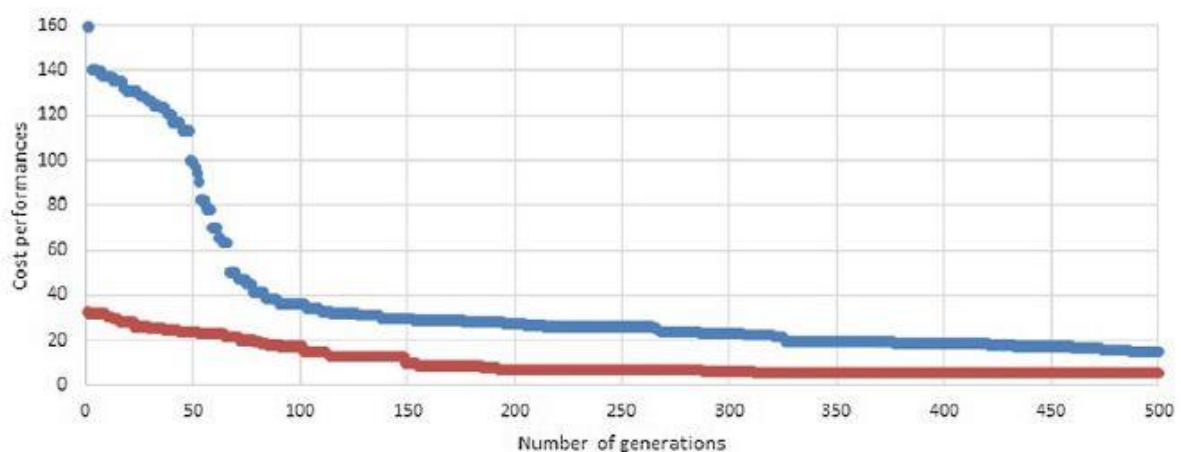


Figure 34: Les courbes de la convergence du coût de production durant l'apprentissage selon les deux stratégies d'initialisation dans le cas d'un système ConWIP

Passons maintenant à la deuxième phase d'expérimentation où la nervosité a été prise en considération. Dans ce cas, l'ensemble des solutions de compromis (non dominés) obtenues par notre approche est présenté dans la Figure 35.

Nous constatons, dans un premier lieu, que les résultats obtenus par notre approche de pilotage adaptatif sont meilleurs que le résultat obtenu par le système ConWIP traditionnel (entouré en vert). Par la suite, nous analysons les résultats obtenus par rapport aux résultats obtenus par les approches de pilotage adaptatives (ACS et AEK). Comme on peut le voir sur la Figure 34, les résultats obtenus par NN-MOSO sont meilleurs que la solution obtenue par ACS (entourée en rouge) et par AEK (entourée en bleu). En fait, ACS obtient la meilleure performance en terme du coût de production avec une moyenne de 35.59, avec une moyenne de 2329,58 changements dans le nombre de Kanban, tandis que AEK considère que 6,4215 est la meilleure performance du coût moyen, avec une moyenne de 2927,52 changements dans le nombre de Kanban. Notre approche obtient à nouveau une meilleure performance du coût à 5,6681, avec une moyenne de 1769,24 changements dans le nombre de Kanban.

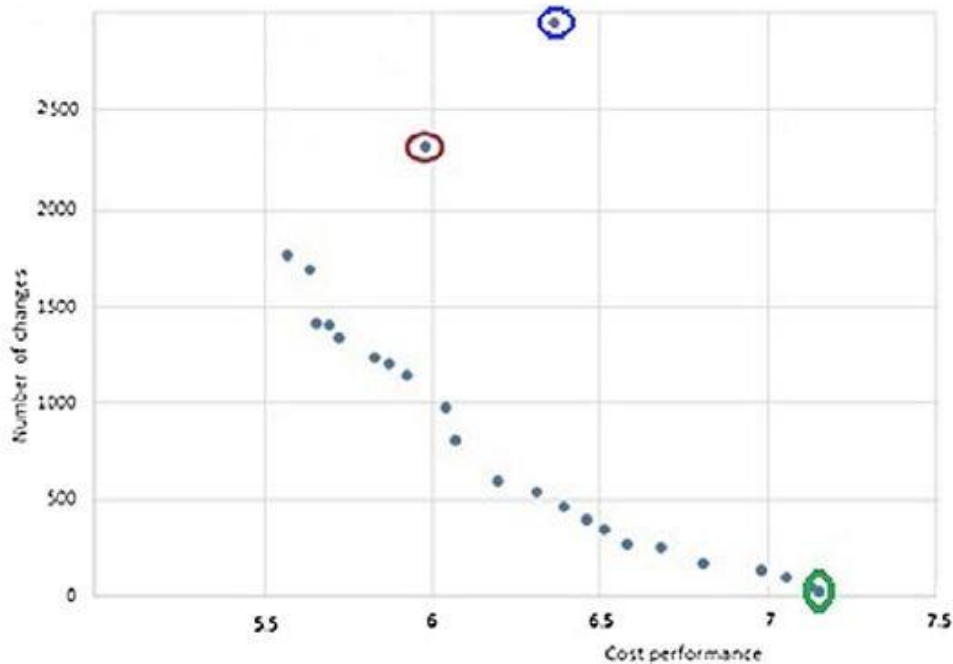


Figure 35: Le front de Pareto obtenu par notre approche dans le cas du système ConWIP

---

## 5.2. Cas du système Kanban

### 5.2.1. Description du système

Le système de production sélectionné ici, comme cas d'étude, est extrait de (Sivakumar et Shahabudeen, 2008) (voir Figure 36). Ce système a été décrit dans la Section 5.1 du Chapitre 3. Puisque nous l'avons utilisé comme cas d'étude pour notre première approche.

Dans ce système, les demandes arrivent une par une selon un processus de Poisson de moyenne  $\lambda$  qui évolue selon un pattern dit triangulaire qui exprime la variation de la demande des clients. Par conséquent, aucune information n'est disponible sur la demande future et aucune prévision n'est possible.

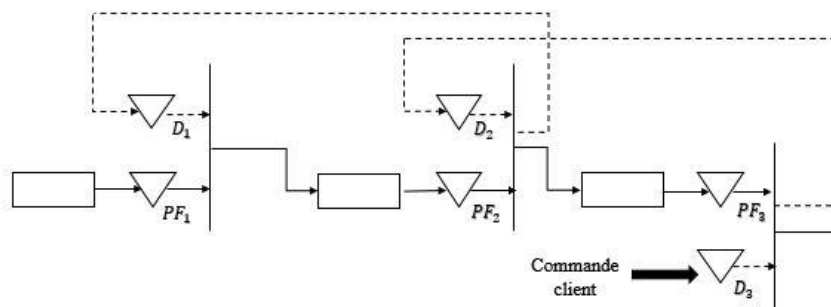


Figure 36: Le modèle Kanban étudié (Sivakumar et Shahabudeen, 2008)

### 5.2.2. Configuration du MLP-NN

Dans le cas du système Kanban, nous utilisons un MLP-NN composé de trois couches. La couche d'entrée contient 17 nœuds ( $E=17$ ), qui représentent l'état actuel du système de production à l'instant  $t$  de la prise de décision  $\overrightarrow{ST}(t)$  et la série chronologique des arrivées de la demande clients  $SC$ , comme décrit dans la Figure 37.

L'état du système inclut la quantité d'encours  $WIP_i(t)$  dans la station  $i$  à l'instant  $t$ , pour tout  $i \in \{1, \dots, 3\}$ , la quantité de produits finis  $FP_i(t)$  et le nombre de demande clients en attente  $B_i(t)$ . La série chronologique  $SC$  est composée des huit derniers intervalles enregistrés ( $K=8$ ). La couche de sortie est composée de 3 nœuds dont chacune d'elles nous donne la décision à

appliquer dans chaque station du système de production (ajouter/retirer un kanban, ne rien faire). Afin de réduire le nombre de poids à optimiser, nous avons utilisé une seule couche cachée dont le nombre de nœuds est déterminé empiriquement.

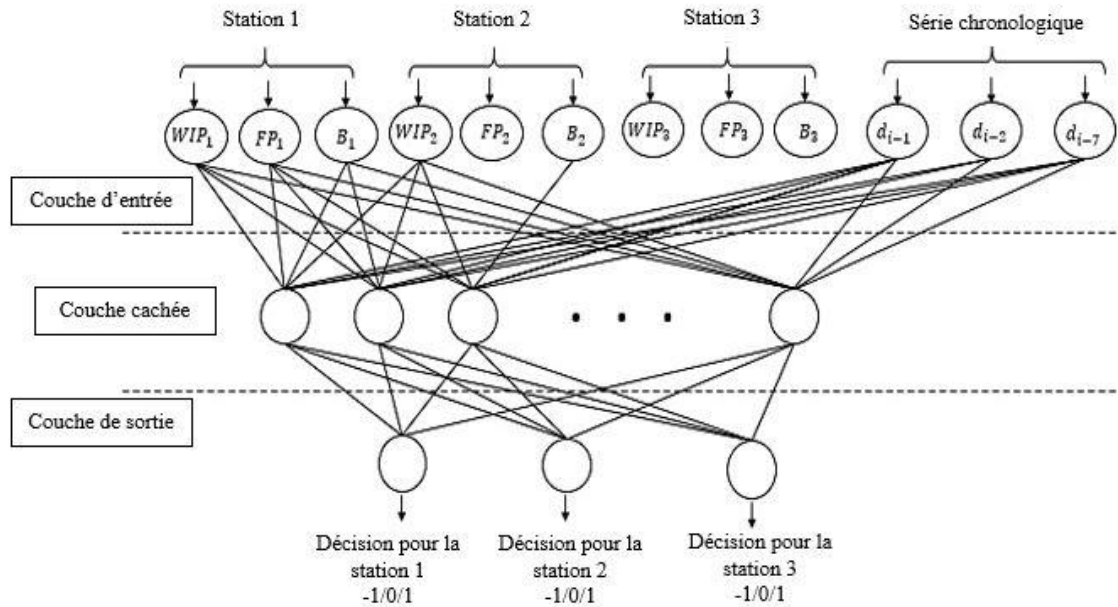


Figure 37: La configuration du MLP-NN dans le cas d'un système Kanban

### 5.2.3. Conditions des expérimentations

Dans le cas du système Kanban, nous comparons notre approche, d'une part, à l'approche du pilotage adaptatif proposée dans (Sivakumar et Shahabudeen, 2008) et d'autre part au système Kanban traditionnel.

Dans le cas du Kanban traditionnel, l'optimisation du système consiste à trouver le nombre de kanban  $K_i$  ( $\forall i \in \{1, \dots, N\}$ ) pour chaque station  $i$  qui minimise les coûts générés par le système F1, et ce sans aucune carte supplémentaire. D'autre part, dans le cas de l'approche proposée par (Sivakumar et Shahabudeen, 2008), l'optimisation consiste à trouver le vecteur des paramètres  $\vec{P} = (K_i, E_i, R_i \text{ et } C_i)$  (Voir Tableau 7).





Tableau 7: Les paramètres des systèmes Kanban à optimiser

Approches	Acronymes	Paramètres
(Sivakumar et Shahabudeen, 2008)	AKS (Adaptive ConWIP System)	$K_i, E_i, R_i$ et $C_i$ $\forall i \in \{1, \dots, N\}$
Kanban Traditionnel	TKS (Traditionnal Kanban System)	$K_i \forall i \in \{1, \dots, N\}$

Afin de rendre notre comparaison aussi équitable que possible, nous utilisons le même principe d'expérimentation expliqué dans la section 5.1.3 pour le cas du système ConWIP.

#### 5.2.4. Résultats

Le Tableau 8 présente les meilleures performances en termes de coûts de production obtenues par les approches TKS, AKS et notre approche NN-MOSO. Dans ce tableau, nous constatons que le résultat obtenu par notre approche NN-MOSO est significativement meilleur que les résultats obtenus par TKS et AKS.

Tableau 8: Résultats des meilleures performances obtenues pour le cas d'un système Kanban

	TKS	AKS	NN-MOSO
Performance des coûts générés par le système (F1)	44.3876	41.9324	38.6852

Les courbes de convergence de la mesure de la performance des coûts de production sont illustrées dans la Figure 38. Comme dans le cas du système ConWIP, nous constatons que l'apprentissage est un peu lent avec la stratégie S1. Alors que, avec la stratégie S2, l'apprentissage commence avec des solutions plus raisonnables et les résultats sont rapidement améliorés par la suite.

Examinons maintenant les résultats liés au système Kanban adaptatif. Le résultat obtenu par AKS est entouré d'un cercle rouge sur la Figure 39. À partir de cette figure, nous constatons que notre approche NN-MOSO a obtenu des résultats significativement meilleurs que les résultats obtenus par AKS pour les deux mesures de performance.

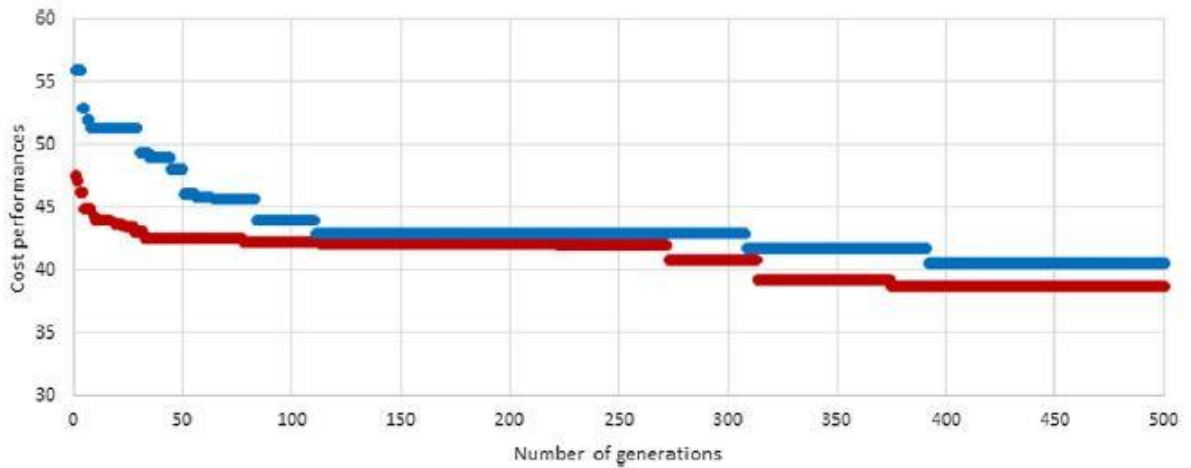


Figure 38: Les courbes de la convergence du coût de production durant l'apprentissage selon les deux stratégies d'initialisation dans le cas d'un système Kanban

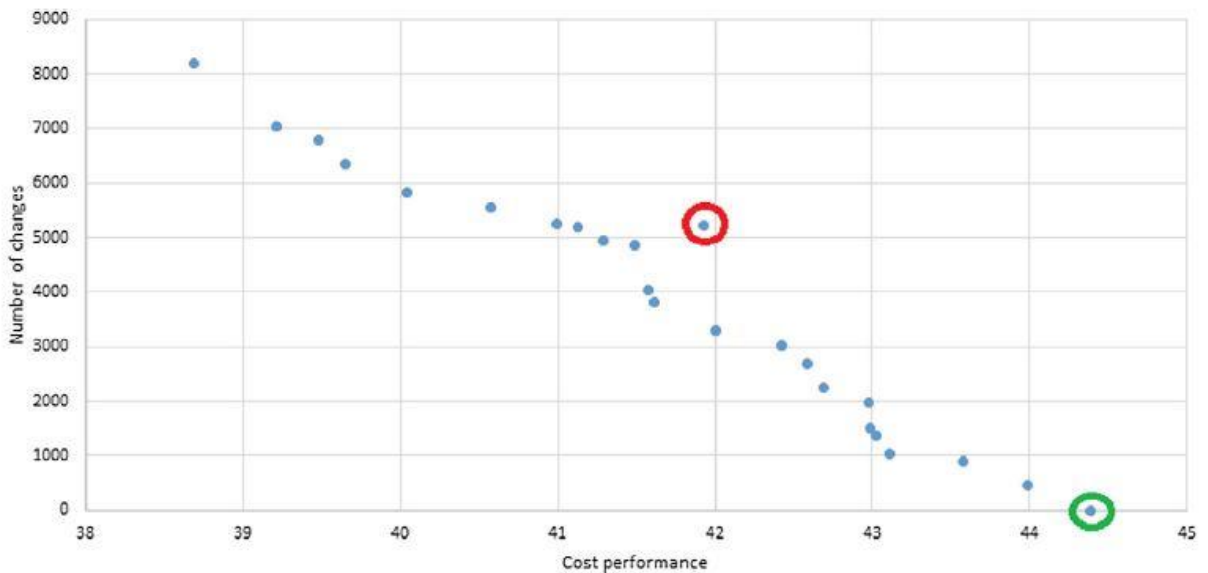


Figure 39: Le front de pareto dans le cas d'un système Kanban

## 6. Conclusion

Dans ce chapitre, nous avons proposé une nouvelle approche de pilotage adaptatif des systèmes en flux tirés dans le contexte de l'industrie 4.0. Notre approche est basée sur un réseau de neurones capable d'exploiter au mieux les données disponibles sur l'état du contexte de production en temps réel et les données historiques sur l'évolution de la demande client afin de décider à quel moment et à quelle station le nombre de cartes kanban doit être modifié. Pour ce faire, nous avons exploité de nouveaux usages de la simulation couplé à une méthode d'optimisation multi-objectif afin d'apprendre notre réseau de neurones hors ligne. Notons que l'usage de la simulation pour l'apprentissage du MLP-NN nous a permis de :

- Prendre en compte des phénomènes aléatoires ;
- Prendre en considération diverses évolutions possibles de la demande clients ;
- Utiliser différents types de systèmes en flux tirés ;
- Modéliser des systèmes de production complexes sans hypothèses restrictives (par exemple, possibilité de prendre en compte des processus de Poisson ou des temps de traitement exponentiels, etc.).

De plus, nous avons appliqué notre approche sur différents cas d'études. Ceci a montré que, grâce à l'utilisation de données appropriées sur le contexte de production, le MLP-NN a obtenu de meilleures performances, en terme de coûts de production, que les approches de pilotage adaptatif publiés souvent cités dans la littérature (Tardif et Maaseidvaag, 2001; Sivakumar et Shahabudeen, 2008; Xanthopoulos, Koulouriotis et Gasteratos, 2017), avec l'avantage important de réduire considérablement le nombre de changements nécessaires dans le nombre des cartes kanban. De plus, nous avons ainsi vu que notre approche était capable de fournir des connaissances pertinentes qui ont été générées artificiellement et qui permettent de prendre d'excellentes décisions.

En effet, notre approche fournit comme livrable un outil intelligent d'aide à la décision qui pourra être relié au système de suivi de production de l'entreprise afin de mieux concevoir le système de production en flux tiré adaptatif tout en prenant en considération les problèmes organisationnels. De plus, fournir un front de Pareto permet aux décideurs de choisir la solution

la mieux adaptée à leurs besoins en ce qui concerne plusieurs types de considérations opérationnelles pratiques. Si les fronts de Pareto deviennent trop grands, une méthode de décision multicritères pourrait alors être utilisée pour aider le décideur à faire un choix final (par exemple, Topsis, AHP, etc.).

# Conclusions et perspectives

De nos jours, pour rester compétitif, les systèmes de production doivent mettre en place des modes de pilotage intelligents permettant et être plus flexibles et réactifs afin de s'adapter aux besoins des clients ainsi qu'aux aléas de fabrication. Au niveau stratégique, ils sont aussi appelés à tirer profit des opportunités prometteuses offertes par l'industrie 4.0. Cependant, nous avons vu qu'aujourd'hui nous pourrions distinguer deux types d'industries :

- Des industries qui ont besoin encore du temps pour moderniser leur processus et émerger vers l'industrie 4.0. Dans ce cas, notre objectif, dans ces travaux de thèse, est de trouver l'approche la plus adéquate pour qu'elles réussissent à mieux ajuster leurs flux et qu'elle préserve ses avantages concurrentiels.
- Des industries 4.0 dont les objectifs sont d'intégrer les nouvelles technologies dans leurs usines et de profiter des nouvelles opportunités offertes par l'industrie 4.0 afin de mieux piloter leurs flux dans un environnement dynamique et incertain.

Notre objectif dans ces travaux était alors de proposer des approches de pilotage adaptatif, capables de piloter dynamiquement le nombre de cartes kanban dans un environnement de production en flux tiré, tout en prenant en considération les variations des demandes clients ainsi que les aléas de fabrication.

Pour cela, une analyse de la littérature a été effectuée dans le premier chapitre et nous a permis de dresser diverses constatations à partir des approches existantes pour le pilotage adaptatif des systèmes en flux tirés, parmi lesquelles, nous avons constaté que plusieurs auteurs utilisent des règles de pilotages basées sur des seuils qui déterminent, dans la mesure du possible, si le nombre de cartes kanban doit être changé ou pas. Leur objectif commun est d'améliorer les mesures de performances du système de production qui exprime globalement les coûts. Ces coûts sont généralement un compromis entre les coûts de stockage (encours et stock final) et un coût lié à la satisfaction du client (commandes non satisfaites). Par conséquent, nous avons constaté qu'on n'accorde pas suffisamment d'attention à l'impact des changements

excessifs du nombre de cartes kanban effectuées sur l'environnement de production. Pour cela, nous avons présenté et discuté, dans le chapitre 3, un nouveau problème qui peut être engendré par les changements excessifs du nombre de cartes kanban, qui est le problème de la nervosité et nous avons proposé de le prendre en considération lors du pilotage adaptatif en temps réel. De plus, le problème majeur de ces règles de contrôle réside dans la détermination des seuils qui dépendent largement du système de production et influent sur sa performance. Pour résoudre ce problème, nous avons alors proposé notre première approche d'optimisation multi-objectif via simulation afin de déterminer ces seuils tout en prenant en considération les variations dynamiques de l'environnement ainsi que les aléas de fabrication. L'objectif de cette approche est d'améliorer les performances du système de production, tout en réduisant la nervosité.

La simulation joue un rôle crucial dans notre approche car elle est capable de représenter, de façon assez réaliste, le comportement dynamique et complexe d'un système et éviter les hypothèses restrictives, tout en prenant en considération des données incertaines et/ou aléatoires. De plus, elle permet de modéliser différents types de systèmes en flux tirés et évaluer leurs performances et d'intégrer des variations de la demande des clients.

Dans le chapitre 3, nous avons exploité les avantages de la simulation couplée à un algorithme d'optimisation multi-objectif afin de déterminer quand et dans quelle étape de la ligne de fabrication il faut changer le nombre de cartes kanban dans un environnement dynamique et incertain, tout en prenant en considération le problème de la nervosité. Pour cela, nous avons introduit les principes de notre approche, proposé une formulation du problème puis décrit l'approche en détail. Par la suite, Nous avons appliqué cette approche à un cas d'étude issu de la littérature avec différents scénarii, et nous l'avons comparée à une approche de pilotage adaptatif publiée, souvent citée dans la littérature. Notre approche a obtenu de meilleures performances, en terme de coûts de production, que celles de l'approche publiée, avec l'avantage important de réduire considérablement le nombre de changements nécessaires dans le nombre des cartes kanban. Nous avons, également, pu montrer qu'il est parfois possible d'améliorer les performances des coûts de production au prix d'un nombre limité de changements dans le nombre de cartes kanban, ce qui réduit le problème de la nervosité. De plus, la détermination d'un front de Pareto offre des solutions de compromis possibles, qui peuvent aider les gestionnaires à mieux concevoir le système de production en flux tiré adaptatif,

tout en prenant en considération les caractéristiques de leurs systèmes ainsi que leurs préférences.

Dans le chapitre 2, nous avons étudié les opportunités offertes par l'industrie 4.0 et plus particulièrement l'intérêt de l'intégration des technologies d'identification automatique dans les systèmes en flux tirés. En effet, disposer en temps réel des données sur le contexte de production : l'emplacement, le statut et ce qui est arrivé aux produits et aux ressources ainsi que des données historiques sur l'évolution de la production et des demandes clients, peuvent être exploités par des stratégies intelligentes de pilotage de flux afin d'adapter dynamiquement le nombre de cartes kanban.

Bien que, dans notre première approche, décrite dans le chapitre 3, nous ayons montré les avantages majeurs de la simulation, son potentiel reste encore sous exploité. Notre deuxième contribution, présentée dans le chapitre 4, a justement cherché à approfondir l'étude de ce potentiel. Pour cela, nous avons proposé une approche d'apprentissage à partir d'un jumeau numérique et optimisation multi-objectif. Notre approche est basée sur un réseau de neurones capable d'exploiter au mieux les données disponibles sur l'état du contexte de production ainsi que les données historiques sur l'évolution de la demande client afin de décider à quel moment et à quelle station le nombre de cartes kanban doit être modifié. Pour ce faire, nous avons exploité de nouveaux usages de la simulation couplée à une méthode d'optimisation multi-objectif afin d'apprendre notre réseau de neurones hors ligne sans utiliser une base d'apprentissage. Par la suite, nous avons appliqué notre approche sur différents cas d'études. Ceci a montré que, notre approche peut être appliquée à un large spectre de système en flux tirés. D'ailleurs, dans tous les cas étudiés, notre approche a obtenu de meilleures performances, en terme de coûts de production, que les meilleures approches de pilotage adaptatif souvent citées dans la littérature, avec l'avantage important de réduire considérablement le nombre de changements nécessaires dans le nombre des cartes kanban.

Ces travaux de thèse ouvrent de nombreuses perspectives de recherche, liées à notre première approche d'optimisation multi-objectif via simulation ou à l'apprentissage automatique à partir d'un jumeau numérique.



Une première perspective consiste à mieux prendre en compte le problème de la nervosité au sein des systèmes en flux tirés. Dans cette thèse, nous avons étudié le problème dans une optique d'optimisation multi-objectif. Cependant, une alternative à exploiter sera d'utiliser un régulateur PID ou un PID flou pour contrôler le problème de la nervosité.

Dans une industrie, donner des consignes claires et simples aux gestionnaires afin de piloter dynamiquement les flux représente un outil d'aide à la décision plus pratique et plus simple à utiliser dans la réalité. Pour cela, une alternative aux règles de contrôle utilisées dans notre première approche sera d'utiliser des règles floues permettant de piloter les flux dynamiquement de façon plus intuitive.

Plusieurs autres perspectives concernent, cette fois-ci, l'apprentissage automatique. En effet, une des conditions essentielles pour l'applicabilité de notre approche d'apprentissage à partir d'un jumeau numérique est que la configuration du système de production reste stable. Si des changements majeurs apparaissent, comme l'ajout d'une machine ou la suppression d'une boucle de carte kanban, alors le réseau de neurones en vigueur ne serait plus adéquat. Dans ce cas, une mise à jour du modèle de la simulation, un changement de topologie ainsi qu'un réapprentissage du réseau de neurones seront nécessaires. Une perspective consiste à utiliser, par exemple, des réseaux de neurones à topologie dynamique pour prendre en compte les changements de l'environnement de production.

Une autre perspective concerne l'amélioration du temps de calcul nécessaire à notre approche qui représente la limite principale de ce type d'approche. Pour ce faire, nous envisageons l'adaptation de notre méthode pour qu'elle supporte des calculs parallèles et/ou distribués. De plus, un meilleur choix des opérateurs génétiques peut améliorer la convergence de l'apprentissage par optimisation multi-objectif et simulation.

Une autre perspective intéressante se présente aussi et consiste à appliquer nos différentes approches sur d'autres systèmes en flux tirés, citons par exemple : COBACABANA, ou à d'autres problèmes comme l'ajustement des flux dans un contexte des systèmes en flux tirés multi-produits ou avec double cartes (carte kanban de production et de transport), ou encore l'ajustement des tailles de lots.

Une autre perspective concerne la prise en considération des perturbations internes, comme les pannes, lors du pilotage adaptatif des systèmes en flux tirés.

Bien que, dans tous les cas étudiés, nous ayons pu montrer que notre réseau de neurone était capable de fournir une aide effective à la prise de décision, nous avons vu qu'il a également vocation à fournir des connaissances pertinentes. Une dernière perspective consiste à exploiter l'apprentissage profond (Deep learning) pour le pilotage adaptatif des systèmes en flux tirés.

Les contributions scientifiques de cette thèse ont été publiées lors de congrès nationaux et internationaux et ont fait l'objet de deux articles dans des revues très renommées. Ils sont cités en référence dans ce document.

# Les publications issues de la thèse

Azouz, N., & Pierreval, H. (2019). Adaptive smart card-based pull control systems in context-aware manufacturing systems: Training a neural network through multiobjective simulation optimization. *Applied Soft Computing*, 75, 46-57. IF: 4.873 (2018)

Azouz, N., Silva Belisário, L., Ammar, A., & Pierreval, H. (2018). Addressing overcorrection in adaptive card-based pull control systems. *International Journal of Computer Integrated Manufacturing*, 31(12), 1189-1204. IF:2.090 (2018)

Azouz, N., (2016, May). Systèmes Kanban Électroniques: Une analyse de la littérature. In 22èmes journées STP (Sciences et Techniques de la Production) du GdR MACS

Belisário, L. S., Azouz, N. & Pierreval, H. "Adaptive ConWIP: Analyzing the impact of changing the number of cards," 2015 International Conference on Industrial Engineering and Systems Management (IESM), Seville, 2015, pp. 930-937. doi: 10.1109/IESM.2015.7380266

# Bibliographie

- Ajorlou, Saeede, Issac Shams, and Mirbahador G Aryanezhad. 2011. "Optimization of a Multiproduct CONWIP-Based Manufacturing System Using Artificial Bee Colony." *Proceedings of the International MultiConference of Engineers and Computer Scientists 2011 Vol II. II*: 16–20.
- Ansari, A., and B. Modarress. 1995a. "Wireless Kanban." *Production and Inventory Management Journal* 36(1): 60–64.
- Ansari, A, and Batoul Modarress. 1995b. "Wireless Kanban." *Production and Inventory Management Journal* 36(1): 60.
- Azouz, Nesrine et al. 2018. "Addressing Over-Correction in Adaptive Card-Based Pull Control Systems." *International Journal of Computer Integrated Manufacturing* 00(00): 1–16. <https://doi.org/10.1080/0951192X.2018.1509132>.
- Azouz, Nesrine, and Henri Pierreval. 2019. "Adaptive Smart Card-Based Pull Control Systems in Context-Aware Manufacturing Systems : Training a Neural Network through Multi-Objective Simulation Optimization." *Applied Soft Computing Journal* 75: 46–57. <https://doi.org/10.1016/j.asoc.2018.10.051>.
- Barbosa, José, Paulo Leitão, Emmanuel Adam, and Damien Trentesaux. 2012. "Nervousness in Dynamic Self-Organized Holonic Multi-Agent Systems." In *Advances in Intelligent and Soft Computing*, , 9–17.
- . 2015. "Dynamic Self-Organization in Holonic Multi-Agent Manufacturing Systems: The ADACOR Evolution." *Computers in Industry* 66: 99–111.
- Baudin, Michel, and Arun Rao. 2005. "RFID Applications in Manufacturing." *RFID applications in manufacturing*: 1–12. [http://www.mmt-inst.com/rfid\\_applications\\_in\\_manufacturing\\_draft\\_7\\_.pdf](http://www.mmt-inst.com/rfid_applications_in_manufacturing_draft_7_.pdf).
- Belisário, Lorena Silva. 2015. "Contribution de l'apprentissage Par Simulation à l'auto-Adaptation Des Systèmes de Production." Université Blaise Pascal - Clermont II.

- Belisário, Lorena Silva, Nesrine Azouz, and Henri Pierreval. 2016. “Adaptive ConWIP: Analyzing the Impact of Changing the Number of Cards.” In *Proceedings of 2015 International Conference on Industrial Engineering and Systems Management, IEEE IESM 2015*, , 930–37.
- Belisário, Lorena Silva, and Henri Pierreval. 2015. “Using Genetic Programming and Simulation to Learn How to Dynamically Adapt the Number of Cards in Reactive Pull Systems.” *Expert Systems with Applications* 42(6): 3129–41.
- Bell, Steve. 2005. *33 Lean Enterprise Systems: Using IT for Continuous Improvement*. John Wiley & Sons.
- Berkley, Blair J. 1992. “A Review of the Kanban Production Control Research Literature.” *Production and operations management* 1(4): 393--411.
- Blackburn, Joseph D, Dean H. Kropp, and Robert A. Millen. 1985. “MRP System Nervousness: Causes and Cures.” *Engineering Costs and Production Economics* 9(1–3): 141–46.
- Borangiu, Theodor, Damien Trentesaux, André Thomas, and Duncan McFarlane. 2016. “Service Orientation in Holonic and Multi-Agent Manufacturing.” In *Studies in Computational Intelligence*, , 199–209.
- Boschert, Stefan, and Roland Rosen. 2016. *Digital Twin — The Simulation Aspect*. Springer.
- Bradley, Allen. 2007. *OptQuest for Arena User’s Guide*. Rockwell Software.
- Bril El-Haouzi, Hind. 2017. *Contribution à La Conception et à l’évaluation Des Architectures de Pilotage Des Systèmes de Production Adaptables : Vers Une Approche Anthropocentrée Pour La Simulation et Le Pilotage*.
- Brintrup, Alexandra, Damith Ranasinghe, and Duncan McFarlane. 2010. “RFID Opportunity Analysis for Leaner Manufacturing.” *International Journal of Production Research* 48(9): 2745–64.
- Buzacott, J A. 1989. “QUEUEING MODELS OF KANBAN AND MRP CONTROLLED PRODUCTION SYSTEMS.” 17: 3–20.
- Cardin, Olivier. 2016. “Contribution à La Conception, l’évaluation et l’implémentation de

Systèmes de Production Cyber-Physiques.” Université de Nantes.

- Chai, Linton L S. 2008. “E-Based Inter-Enterprise Supply Chain Kanban for Demand and Order Fulfilment Management.” In *Emerging Technologies and Factory Automation, 2008. ETFA 2008. IEEE International Conference On*, , 33–35.
- Chang, T. M., and Y. Yih. 1994a. “Determining the Number of Kanbans and Lotsizes in a Generic Kanban System: A Simulated Annealing Approach.” *International Journal of Production Research* 32(8): 1991–2004.
- Chang, T M, and Y Yih. 1994b. “Generic Kanban Systems for Dynamic Environments.” *International Journal of Production Research* 32(4): 889–902.
- Chow, Harry K.H., K. L. Choy, and W. B. Lee. 2007. “A Dynamic Logistics Process Knowledge-Based System - An RFID Multi-Agent Approach.” *Knowledge-Based Systems* 20(4): 357–72.
- Cullen, Thomas James. 2002. “Toyota Speeds Parts Delivery with E-Kanban.” *Automotive News* 77(6006): 22B.
- Dai, Xiangshan, Xiaofang Yuan, and Lianghong Wu. 2017. “A Novel Harmony Search Algorithm with Gaussian Mutation for Multi-Objective Optimization.” *Soft Computing* 21(6): 1549–67.
- Dallery, Wes, and George Liberopoulos. 2000. “Extended Kanban Control System: Combining Kanban and Base Stock.” *IIE Transactions (Institute of Industrial Engineers)* 32(4): 369–86.
- Deb, Kalyanmoy, Ashish Anand, and Dhiraj Joshi. 2002. “A Computationally Efficient Evolutionary Algorithm for Real-Parameter Optimization.” *Evolutionary Computation* 10(4): 371–95. <http://www.mitpressjournals.org/doi/10.1162/106365602760972767>.
- Deb, Kalyanmoy, and ayan Deb. 2014. “Analysing Mutation Schemes for Real-Parameter Genetic Algorithms.” *International Journal of Artificial Intelligence and Soft Computing* 4(1): 1. <http://www.inderscience.com/link.php?id=59280>.
- Duri, C, Yannick Frein, and Maria Di Mascolo. 2000. “Comparison among Three Pull Control Policies: Kanban, Base Stock, and Generalized Kanban.” *Annals of Operations Research*

93(1–4): 41--69.

- Duri, C, Maria Di Mascolo, Yannick Frein, and A Hodac. 1997. “Sur l’intérêt de La Gestion Kanban Généralisé.” *Journal européen des systèmes automatisés* 31(8): 1311--1337.
- Erard, Pierre-Jean, and Pontien Déguénon. 1996. *Simulation Par Événements Discrets*. PPUR presses polytechniques.
- Eriksson, Stefan. 2010. “Call-Off Production, Triggered by the Traditional Kanban Card or by Electronic Kanban: A Case Study at Ericsson.” In *Production and Management Operation Society 2010*,.
- Eskandari, Hamidreza, Ehsan Mahmoodi, Hamed Fallah, and Christopher D Geiger. 2011. “Performance Analysis of Comercial Simulation-Based Optimization Packages: OptQuest and Witness Optimizer.” In *Proceedings of the Winter Simulation Conference*, , 2363--2373.
- Fleury, Gerard. 2006. “Simulation à Événements Discrets.” *Eyrolles*.
- Framinan, Jose M, Pedro L Gonza lez, and Rafael Ruiz-Usano. 2006. “Dynamic Card Controlling in a Conwip System.” 99: 102–16.
- Framinan, Jose M, Pedro L González, and Rafael Ruiz-Usano. 2003. “The CONWIP Production Control System: Review and Research Issues.” *Production Planning and Control* 14(3): 255–65.
- Framinan, Jose M, and Henri Pierreval. 2012. “Special Issue on Pull Strategies in Manufacturing Systems and Supply Chains: Recent Advances.” *Journal of Intelligent Manufacturing* 23(1): 1–3.
- Frein, Yannick, Maria Di Mascolo, and Yves Dallery. 1995. “On the Design of Generalized Kanban Control Systems.” *International Journal of Operations \& Production Management* 15(9): 158--184.
- Gaury, E. G.A., J. P.C. Kleijnen, and H. Pierreval. 2001. “A Methodology to Customize Pull Control Systems.” *Journal of the Operational Research Society* 52(7): 789–99.
- González-R, Pedro L., Jose M Framinan, and Rafael Ruiz Usano. 2011. “A Response Surface Methodology for Parameter Setting in a Dynamic Conwip Production Control System.”

*International Journal of Manufacturing Technology and Management* 23(1/2): 16.  
<http://www.inderscience.com/link.php?id=42106>.

- González, Pedro L, Jose M Framinan, Andreas Dopfer, and Rafael Ruiz-usano. 2006. "Optimization Customized Token-Based Production Control Systems Using Cross-Entropy." *Digital Enterprise Technology*: 123–31. [http://dx.doi.org/10.1007/978-0-387-49864-5\\_14](http://dx.doi.org/10.1007/978-0-387-49864-5_14).
- Guion, Rémy, Hind El Haouzi, and André Thomas. 2011. "Etude de La Pertinence d'un Kanban Adaptatif Avec Des Contraintes Multicritères: Cas d'une Cellule de Découpe." *4èmes journées doctorales/journées nationales MACS, JD-JN-MACS*: CDROM. <https://hal.archives-ouvertes.fr/hal-00583986>.
- Gupta, Surendra M, and Yousef AY Al-Turki. 1997. "An Algorithm to Dynamically Adjust the Number of Kanbans in Stochastic Processing Times and Variable Demand Environment." *Production Planning & Control* 8(2): 133–41.
- Hachicha, Wafik, and Ahmed Ghorbel. 2012. "A Survey of Control-Chart Pattern-Recognition Literature (1991-2010) Based on a New Conceptual Classification Scheme." *Computers and Industrial Engineering* 63(1): 204–22. <http://dx.doi.org/10.1016/j.cie.2012.03.002>.
- Hadeli et al. 2007. "Towards the Design of Autonomic Nervousness Handling in Holonic Manufacturing Execution Systems." In *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, , 2883–88.
- El Haouzi, Hind. 2008. "Approche Méthodologique Pour l'intégration Des Systèmes Contrôlés Par Le Produit Dans Un Environnement de Juste-à-Temps: Application à l'entreprise TRANE." UNIVERSITE HENRI POINCARÉ (DIRECTEUR.
- Henderson, Bruce D. 1986. "THE LOGIC OF KANBAN." *Journal of Business Strategy* 6(3): 6–12. <http://www.emeraldinsight.com/doi/10.1108/eb039115>.
- Hopp, Wallace J, and ML Roof. 1998. "Setting WIP Levels with Statistical Throughput Control (STC) in CONWIP Production Lines." *International Journal of Production Research* 36(4): 867–82.
- Huang, Chun Che, and Andrew Kusiak. 1996. "Overview of Kanban Systems." *International*



*Journal of Computer Integrated Manufacturing* 9(3): 169–89.

- Huang, George Q, Y F Zhang, and P Y Jiang. 2008. “RFID-Based Wireless Manufacturing for Real-Time Management of Job Shop WIP Inventories.” *The International Journal of Advanced Manufacturing Technology* 36(7–8): 752–64.
- Jaegler, Yann et al. 2017. “The ConWip Production Control System: A Systematic Review and Classification.” *International Journal of Production Research* 7543(September): 1–22. <https://www.tandfonline.com/doi/full/10.1080/00207543.2017.1380325>.
- Jiang, Pingyu, Yingbin Fu, and Mei Zheng. 2010. “Tracking and Visualizing RFID-Driven Material Flows for Multistage Machining Processes.” *2010 International Conference on Manufacturing Automation*: 186–90.  
<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5695177>.
- Kagermann, Henning, Johannes Helbig, and Wolfgang Wahlster. 2013. “Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0: Securing the Future of German Manufacturing Industry.” (final report of the Industrie 4.0 Working Group).
- Kalyanmoy, Deb, and Others. 2001. *Multi Objective Optimization Using Evolutionary Algorithms*. John Wiley and Sons.
- Khare, Vineet, Xin Yao, and Kalyanmoy Deb. 2003. “Performance Scaling of Multi-Objective Evolutionary Algorithms.” In *International Conference on Evolutionary Multi-Criterion Optimization*, , 376–90.
- Kochan, Anna. 2006. “BMW Innovates at New Leipzig Assembly Plant.” *Assembly Automation* 26(2): 111–14.  
<http://www.emeraldinsight.com/doi/abs/10.1108/01445150610658086>.
- Kolberg, Dennis, and Detlef Zuhlke. 2015. “Lean Automation Enabled by Industry 4.0 Technologies.” In *IFAC-PapersOnLine*, Elsevier, 1870–75.
- Konak, Abdullah, David W Coit, and Alice E Smith. 2006. “Multi-Objective Optimization Using Genetic Algorithms: A Tutorial.” *Reliability Engineering & System Safety* 91(9): 992--1007.
- Kotani, S. 2007. “Optimal Method for Changing the Number of Kanbans in the E-Kanban

- System and Its Applications.” *International Journal of Production Research* 45(24): 5789–5809.
- Kumar, C. Sendil, and R. Panneerselvam. 2007. “Literature Review of JIT-KANBAN System.” *International Journal of Advanced Manufacturing Technology* 32(3–4): 393–408.
- Lage Junior, Muris, and Moacir Godinho Filho. 2010. “Variations of the Kanban System: Literature Review and Classification.” *International Journal of Production Economics* 125(1): 13–21.
- Land, Martin J. 2009. “Cobacabana ( Control of Balance by Card-Based Navigation ): A Card-Based System for Job Shop Control.” 117: 97–103.
- Law, A. M. 2015. *Simulation Modeling and Analysis*. 5 edition. ed. McGraw-Hill.
- Law, Averill M, and Michael G. McComas. 2002. 1 Proceedings of the 2002 Winter Simulation Conference *Simulation-Based Optimization*.  
<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1172866>.
- Lee-Mortimer, Andrew. 2008. “A Continuing Lean Journey: An Electronic Manufacturer’s Adopting of Kanban.” *Assembly Automation* 28(2): 103–12.
- Li, Y. F., S. H. Ng, M. Xie, and T. N. Goh. 2010. “A Systematic Comparison of Metamodeling Techniques for Simulation Optimization in Decision Support Systems.” *Applied Soft Computing Journal* 10(4): 1257–73.
- Liu, Qing, and Dao Huang. 2009. “Dynamic Card Number Adjusting Strategy in Card-Based Production System.” *International Journal of Production Research* 47(21): 6037–50.
- Long-Fei, Wang, and Shi Le-Yuan. 2013. “Simulation Optimization: A Review on Theory and Applications.” *Acta Automatica Sinica* 39(11): 1957–68.  
[http://dx.doi.org/10.1016/S1874-1029\(13\)60081-6](http://dx.doi.org/10.1016/S1874-1029(13)60081-6).
- Lu, Yang. 2017. “Industry 4.0: A Survey on Technologies, Applications and Open Research Issues.” *Journal of Industrial Information Integration* 6: 1–10.  
<http://dx.doi.org/10.1016/j.jii.2017.04.005>.
- Luo, Hao, Ji Fang, and George Q Huang. 2015. “Real-Time Scheduling for Hybrid Flowshop

- in Ubiquitous Manufacturing Environment.” *Computers and Industrial Engineering* 84: 12–23. <http://dx.doi.org/10.1016/j.cie.2014.09.019>.
- Maria, Di Mascolo. 1990. “Modélisation et Évaluation de Performances de Systèmes de Production Gérés En Kanban.” Grenoble INPG.
- Martin, Andrew D, Te-Min Chang, Yeuhwern Yih, and Rex K Kincaid. 1998. “Using Tabu Search to Determine the Number of Kanbans and Lotsizes in a Generic Kanban System.” *Annals of Operations Research* 78: 201–17. <http://dx.doi.org/10.1023/A:1018950016849>.
- Mitchell, Melanie. 1998. *An Introduction to Genetic Algorithms*. MIT press.
- Moattar Hussein, S. M., C. O’Brien, and S T Hosseini. 2006. “A Method to Enhance Volume Flexibility in JIT Production Control.” *International Journal of Production Economics* 104(2): 653–65.
- Mouelhi-Chibani, Wiem. 2009. “Apprentissage Autonome de Réseaux de Neurones Pour Le Pilotage En Temps Réel Des Systèmes de Production Basé Sur l’optimisation via Simulation.” <http://tel.archives-ouvertes.fr/tel-00725259>.
- Ng, Irene C L, and Susan Y L Wakenshaw. 2016. “The Internet of Things: Review and Research Directions.” *International Journal of Research in Marketing*. <http://dx.doi.org/10.1016/j.ijresmar.2016.11.003>.
- Nguyen, Anh-tuan, Sigrid Reiter, and Philippe Rigo. 2014. “A Review on Simulation-Based Optimization Methods Applied to Building Performance Analysis.” *Applied Energy* 113: 1043–58. <http://dx.doi.org/10.1016/j.apenergy.2013.08.061>.
- Paolo, Fontanot, Haase Antje, and Sala Fabio. 2014. “Methods and Software for Controlling Just in Time Item Production via Kanban Cards.” 2(12).
- Papadopoulou, TC, and A Mousavi. 2007. “Control of Constant Work-in-Progress in Dynamic Lean Job Shops Using a Multi-Agent System Approach.” *International Journal of Agile Manufacturing* 10(2): 19–28.
- Pedrielli, Giulia, Arianna Alfieri, and Andrea Matta. 2015. “Integrated Simulation-Optimisation of Pull Control Systems.” *International Journal of Production Research* 53(14): 4317–36.

- Peshkin, Michael A et al. 2001. "Cobot Architecture." *IEEE Transactions on Robotics and Automation* 17(4): 377--390.
- Pierreval, Henri et al. 2015. "A Simulation Optimization Approach for Reactive ConWIP Systems." In *Proceedings - 8th EUROSIM Congress on Modelling and Simulation, EUROSIM 2013*, , 415–20.
- Ponche, Rémi. 2013. Thèse de Doctorat "Méthodologie de Conception Pour La Fabrication Additive, Application à La Projection de Poudres." <https://tel.archives-ouvertes.fr/tel-00916534>.
- Powell, Daryl. 2013. "ERP Systems in Lean Production: New Insights from a Review of Lean and ERP Literature." *International Journal of Operations & Production Management* 33(11): 1490–1510. <http://www.emeraldinsight.com/10.1108/IJOPM-07-2010-0195>.
- Preiss, Martin. 2016. "Kanban Control Cycle System." 2(12).
- Qi, Zhang, Cheng Lu, and Boutaba Raouf. 2010. "Cloud Computing: State of the Art and Research Challenges." *Journal of internet services and applications* 1(1): 7--18.
- Rahnamayan, Shahryar, Hamid R Tizhoosh, and Magdy MA Salama. 2007. "A Novel Population Initialization Method for Accelerating Evolutionary Algorithms." *Computers & Mathematics with Applications* 53(10): 1605--1614.
- Rees, Loren P, Patrick R Philipoom, Bernard W Taylor Iii, and Philip Y Huang. 1987. "Dynamically Adjusting the Number of Kanbans in a Just-in-Time Production System Using Estimated Values of Leadtime." *IIE transactions* 19(2): 199–207.
- Renna, Paolo. 2015. "A Fuzzy Control System to Adjust the Number of Cards in a CONWIP-Based Manufacturing System." *International Journal of Services and Operations Management* 20(2): 188. <http://www.scopus.com/inward/record.url?eid=2-s2.0-84923132556&partnerID=tZOtx3y1>.
- Renna, Paolo, Luciano Magrino, and Renato Zaffina. 2013. "Dynamic Card Control Strategy in Pull Manufacturing Systems." *International Journal of Computer Integrated Manufacturing* 26(9): 881–94.
- Sarker, Bhaba R. 1989. "Simulating a Just-in-Time Production System." *Computers and*

*Industrial Engineering* 16(1): 127–37.

Saucedo-Martínez, Jania Astrid et al. 2018. “Industry 4.0 Framework for Management and Operations: A Review.” *Journal of Ambient Intelligence and Humanized Computing* 9(3): 789–801.

Shahabudeen, P, and G. D. Sivakumar. 2008. “Algorithm for the Design of Single-Stage Adaptive Kanban System.” *Computers and Industrial Engineering* 54(4): 800–820.

Simonis, K, YS Gloy, and T Gries. 2016. “INDUSTRIE 4.0—automation in Weft Knitting Technology.” In *IOP Conference Series: Materials Science and Engineering*, , 012014.

Sivakumar, G D, and P Shahabudeen. 2008. “Design of Multi-Stage Adaptive Kanban System.” *International Journal of Advanced Manufacturing Technology* 38: 321–36.

———. 2009. “Algorithms for the Design of a Multi-Stage Adaptive Kanban System.” *International Journal of Production Research* 47(23): 6707–38.

Von Solms, Rossouw, and Johan Van Niekerk. 2013. “From Information Security to Cyber Security.” *Computers & security* 38: 97–102.

Spearman Mark L, Woodruff David L, Hopp Wallace J. 1990. “CONWIP: A Pull Alternative to Kanban.” *International Journal of Production Research* 28(5): 879--894.

Srinivas, Nidamarthi, and Kalyanmoy Deb. 1994. “Multi-Objective Optimization Using Nondominated Sorting in Genetic Algorithms.” *Evolutionary computation* 2(3): 221–48.

Sutherland, Ivan E. 1965. “The Ultimate Display.” *Multimedia: From Wagner to virtual reality*: 506--508.

Svirčević, Vasa V, Dragan S Simić, and Vladimir A Ilin. 2013. “Advantages of E-Kanban System Compared to Classic Kanban Serving Production Line.” In *Proc. 1st Logistics International Conference*, , 161–65.

Syswerda, Gilbert. 1989. “Uniform Crossover in Genetic Algorithms.” In *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers, 2-9.

Takahashi, Katsuhiko, Katsumi Morikawa, and Nobuto Nakamura. 2004. “Reactive JIT

- Ordering System for Changes in the Mean and Variance of Demand.” *International Journal of Production Economics* 92(2): 181–96.
- Takahashi, Katsuhiko, and Nobuto Nakamura. 1999a. “Applying a Neural Network to the Adaptive Control for JIT Production Systems.” In *Proceedings of the 1999 IEEE International Conference on Control Applications*, , 1648–53.  
[http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=801219&url=http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=801219](http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=801219&url=http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=801219).
- . 1999b. “Reacting JIT Ordering Systems to the Unstable Changes in Demand.” *International Journal of Production Research* 37(10): 2293--2313.
- . 2002. “Decentralized Reactive Kanban System.” *European Journal of Operational Research* 139(2): 262–76.
- Talibi, Zakia, Hind Bril El Haouzi, and André Thomas. 2013. “The Relevance Study of Adaptive Kanban in a Multicriteria Constraints Context Using Data-Driven Simulation Method.” In *International Conference on Industrial Engineering and Systems Management (IEEE-IESM'13)*, , 169–75.
- Tardif, Valerie, and Lars Maaseidvaag. 2001. “An Adaptive Approach to Controlling Kanban Systems.” *European Journal of Operational Research* 132(2): 411–24.
- Thürer, Matthias, Martin J Land, and Mark Stevenson. 2014. “Card-Based Workload Control for Job Shops : Improving COBACABANA.” *Intern. Journal of Production Economics* 147: 180–88. <http://dx.doi.org/10.1016/j.ijpe.2013.09.015>.
- Thürer, Matthias, Mark Stevenson, and Charles W Protzman. 2016. “Card-Based Production Control: A Review of the Control Mechanisms Underpinning Kanban, ConWIP, POLCA and COBACABANA Systems.” *Production Planning and Control* 27(14): 1143–57.
- Triantaphyllou, Evangelos. 2000. “Multi-Criteria Decision Making Methods.” In *Multi-Criteria Decision Making Methods: A Comparative Study*, Springer.
- Velasquez, Mark, and Patrick T Hester. 2013. “An Analysis of Multi-Criteria Decision Making Methods.” *International Journal of Operations Research* 10(2): 56--66.
- Vercraene, Samuel et al. 2011. “Control of a Two-Stage Production / Inventory System with

Products Returns To Cite This Version : Production / Inventory System with Products.” (October 2014).

Wan, Hung Da, and F Frank Chen. 2008. “A Web-Based Kanban System for Job Dispatching, Tracking, and Performance Monitoring.” *International Journal of Advanced Manufacturing Technology* 38(9–10): 995–1005.

Wolter, Marc Ingo et al. 2015. *Industrie 4.0 Und Die Folgen Fur Arbeitsmarkt Und Wirtschaft: Szenario-Rechnungen Im Rahmen Der BIBB-IAB-Qualifikations-Und Berufsfeldprojektionen.*

Womack J, Jones Daniel T, Roos Daniel. 1990. “The Machine That Changed the World: Based on the Massachusetts Institute of Technology \$5 Million 5-Year Study on the Future of the Automobile.” *New York, Rawson.*

Xanthopoulos, A S, D E Koulouriotis, and A Gasteratos. 2017. “Adaptive Card-Based Production Control Policies.” *Computers and Industrial Engineering* 103(November): 131–44. <http://dx.doi.org/10.1016/j.cie.2016.11.019>.

Yasuhiro, Monden. 1983. “Toyota Production Systems Industrial Engineering and Management Press.” *Norcross, Ga.*

Zarbo, Richard J. et al. 2009. “The Henry Ford Production System Reduction of Surgical Pathology In-Process Misidentification Defects by Bar Code-Specified Work Process Standardization.” *American Journal of Clinical Pathology* 131(4): 468–77.

Zhang, Rui, S. K. Ong, and A. Y.C. Nee. 2015. “A Simulation-Based Genetic Algorithm Approach for Remanufacturing Process Planning and Scheduling.” *Applied Soft Computing Journal* 37: 521–32. <http://dx.doi.org/10.1016/j.asoc.2015.08.051>.

Zhang, Yingfeng, Pingyu Jiang, and George Huang. 2008. “RFID-Based Smart Kanbans for Just-In-Time Manufacturing.” *International Journal of Materials and Product Technology* 33(1/2): 170. <http://www.scopus.com/inward/record.url?eid=2-s2.0-49149120257&partnerID=tZOtx3y1>.

Zitzler, Eckart, Marco Laumanns, and Lothar Thiele. 2001. “SPEA2: Improving the Strength Pareto Evolutionary Algorithm.” *TIK-report* 103.

