
DEFINITION ET EVALUATION DES ALGORITHMES GENERATIFS DE GAMMES ENVELOPPES

4.1 Introduction

4.2.Contexte

4.3.Formulation du problème

4.4.Méthodologie

4.5.Résultats

La tendance pousse les industriels à personnaliser leurs produits pour répondre à la demande des clients. Dans ce contexte, le ConWip fournit un système de contrôle de la production durable, efficace et adaptatif. Le présent chapitre traite des questions clés liées à la mise en œuvre du ConWip dans un environnement caractérisé par un mix produit élevé et/ou un mix de gammes opératoires élevé. Quatre algorithmes qui permettent, à partir d'un jeu de gammes opératoires, de générer une gamme « moyenne » appelée ici « gamme enveloppe », sont présentés. Ces gammes enveloppes sont ensuite implémentées dans Wipsim, un outil d'ingénierie utilisé dans les projets pour concevoir et améliorer les lignes d'assemblages pilotées par un ConWip, ce qui permet de calculer les paramètres ConWip optimisés pour chacune de ces gammes. Un échantillon de données, issues d'un cas industriel, est utilisé pour tester les quatre modèles obtenus. Nous les comparons sur le plan de leur sensibilité aux variations de mix produit et à l'introduction de différents produits atypiques. Nos expériences montrent d'une part que nos algorithmes génèrent des gammes enveloppes viables et d'autres parts fournissent des éléments permettant de choisir entre les quatre solutions proposées en fonction du contexte.

4.1. INTRODUCTION

La possibilité de choisir le système de contrôle de la production (PCS) le plus adapté représente un problème majeur qui influe fortement sur l'efficacité d'un système de production dans notre environnement Vulnérable, Incertain, Complexe et Ambigu (VUCA). Une illustration appropriée est donnée par le contexte stimulant du développement durable qui traite de l'empreinte écologique et des objectifs de réduction de la consommation d'énergie.

Un PCS adapté doit être capable de contrôler correctement le niveau de stock et, en même temps, de fournir une réponse pertinente à une personnalisation élevée et une réponse réactive aux demandes changeantes des clients. Dans un tel contexte, le ConWip fait partie des bons candidats PCS pour fournir la flexibilité et l'efficacité attendue en supportant plusieurs produits et faisant l'objet d'une demande fluctuante. La pierre angulaire du ConWip consiste à contrôler la quantité totale de travail sur la ligne de production en la maintenant constante. Cette capacité est enracinée dans la loi construite par Little (1961), communément appelée « Little's Law » :

$$\text{Work in Process} = \text{Throughput} * \text{Makespan}$$

Depuis son introduction par Spermán et al. (1990), plusieurs travaux ont étudié la différence entre le ConWip et le Kanban développé par Toyota (Spermán et al. (1990), Takahashi et al 2002, Takahashi et al 2005, Pettersen et al. Jing-Wen Li 2010). (Jaegler et al., 2017) ont mis à jour la dernière revue de la littérature au sujet du ConWip (Framinan, Gonzalez et Ruiz-Usano 2003) réalisée en 2003 et suggéré plusieurs pistes de recherche non résolues le concernant. Ces axes de recherche portent sur la mise en œuvre et l'optimisation du ConWip, l'environnement dans lequel il est implémenté, la méthodologie de comparaison du ConWip avec d'autres PCS, les outils de simulation de son fonctionnement et les retours d'expériences industrielles impliquant le ConWip.

Nous avons décidé de nous concentrer sur des pistes de recherche liées à l'environnement de production dans lequel est implémenté le ConWip. Trois thèmes liés aux contextes productifs sont mis en évidence. L'un d'eux concerne les caractéristiques du mix produit. Les questions clés sont liées à un mix produit élevé et/ou à un mix gammes opératoires élevées. Même si (Prakash et Chin 2015 a) expliquent que le ConWip a été « initialement conçu pour la fabrication dans un environnement caractérisé par des gammes de produits constantes, des temps opératoires similaires, des temps de changement de série réduits et des ateliers de type flowshop » (Spearman et Zazanis 1992) ont suggéré depuis 1992 d'utiliser la notion de gammes pour optimiser le système ConWip. Le lien entre les gammes et le dimensionnement du ConWip est le point central de leur théorie. (Ziengs, Riezebos et Germs 2012) s'intéressent également à ce concept et proposent de construire un ConWip basé sur la charge de travail.

En plus de cela (Khojasteh-Ghamari 2009) dont le modèle est basé sur la production d'un produit unique, affirme que l'étude d'un environnement plus complexe (en particulier avec un enjeu multiproduit) représente l'axe de recherche futur le plus intéressant. Sur la même ligne (Parvin et al., 2012) proposent également d'explorer le sujet du ConWip dans des environnements de fabrication complexes. (Takahashi et Nakamura 2002) étudient aussi le ConWip dans les chaînes d'approvisionnement complexes. (Satyam et Krishnamurthy 2013) ajoutent que la compréhension des interactions complexes dans un environnement multiproduit est cruciale. En conclusion, l'idée consiste à trouver le nombre optimal de tickets et la meilleure taille de lot pour les gammes considérées afin d'optimiser le ConWip (Eng et Sin 2013), (Prakash et Chin 2015b).

Un intérêt croissant pour l'étude du ConWip dans un environnement multiproduit s'est clairement concrétisé dans la littérature. Par exemple, Herer et Masin 1997 ont développé un modèle de programmation mathématique pour un système de flux de production ConWip multiproduit. L'objectif affiché consiste à déterminer la séquence de production optimale, mais ils ne sont néanmoins pas en mesure de proposer un algorithme permettant de résoudre complètement la problématique. Ryan et al. 2000 commencent par étudier (en mettant l'accent sur le service client) un environnement de travail piloté par un ConWip dans lequel deux produits avec des gammes distinctes utilisent le même ensemble de ressources, puis un système multiproduit piloté par un CONWIP est modélisé sur la base d'un réseau fermé de files d'attente multichaîne à plusieurs classes (Ryan et Vorasayan 2005). Satyam et al. 2008 proposent une nouvelle approche basée sur la décomposition paramétrique pour modéliser analytiquement un système de fabrication multiproduit piloté grâce à un ConWip à l'aide de réseaux fermés de files d'attente et une synchronisation des postes de travail. Une autre illustration est donnée par le travail de Park et Lee 2013, ils ont étudié un système de production multiproduit qui permet d'assembler un produit fini pour répondre à une commande client. Le produit fini est composé d'un kit de différents composants fabriqués par le même système. Ils supposent que les commandes arrivent conformément à une distribution de Poisson et que les temps de traitement de chaque station suivent une distribution de Coxian.

Pour synthétiser les éléments précédents, il semble que l'optimisation d'un ConWip dans un environnement multiproduit caractérisé par un mix produit élevé et/ou un mix gamme élevée n'a pas été résolue. En plus de cela, cette optimisation représente un réel besoin dans les environnements de fabrication. Par conséquent, ce chapitre propose une solution à ce problème dans une direction majeure qui consiste à définir un ConWip générique basé sur la gamme enveloppe des produits mis en jeu et à utiliser cet outil pour résoudre le problème d'optimisation qui a été énoncé. Pour définir un tel ConWip, une gamme unique doit être définie pour représenter les gammes des différents produits. Grâce à quatre algorithmes, cette section étudie la question de la définition d'une telle gamme représentative, appelée « gamme enveloppe ». Un échantillon numérique dérivé de données industrielles est ensuite utilisé. Les quatre algorithmes sont testés avec cet échantillon selon différents scénarios. Ce chapitre est organisé comme suit : la méthodologie est d'abord détaillée puis les résultats obtenus sont analysés et discutés, enfin de futures pistes de recherche sont proposées.

4.2. CONTEXTE

La motivation de cette étude réside dans le fait que la théorie du ConWip reste à compléter et qu'elle représente un réel intérêt pour l'industrie. Pettersen et Segerstedt 2009 expliquent que l'absence de directives d'installation de ConWip constitue un obstacle à l'utilisation du système. En pratique, ceci signifie qu'il n'est pas trivial d'installer un ConWip et que la plupart des systèmes ERP (Enterprise Resource Planning) existants ne contiennent pas de fonctionnalités pour la mise en œuvre d'un ConWip, en particulier dans le contexte multiproduit.

Plusieurs contributions montrent comment dimensionner un ConWip (nombre de tickets et taille de lot) centré sur une gamme de fabrication unique (par exemple (Hopp et Spearman 1996) pour le nombre de tickets et (Zhang et Chen 2001) pour la taille de lot). Néanmoins, dans la pratique, une chaîne de production fabrique des produits différents avec des gammes différentes.

Dans ce cas, le corps de la littérature n'aide guère sur la façon d'optimiser le dimensionnement d'un ConWip. En effet, la littérature au sujet du ConWip contient des outils d'optimisation, le premier a été donné par Hopp et Spearman (1996). Tous basent leur calcul sur la gamme unique d'un produit unique. Dans le cas où un environnement multiproduit doit être considéré, les solutions potentielles pourraient être ancrées dans un outil de simulation qui « joue » la future séquence de production prévue afin d'optimiser le ConWip grâce à une recherche par dichotomie comme le montrent Belisario et al. 2015. Ces solutions potentielles sont pertinentes, mais elles impliquent de simuler la séquence de production future pour chaque changement non planifié. Cela ne constitue pas nécessairement un obstacle du point de vue de l'outil théorique ou informatique, mais cette solution serait difficile à mettre en œuvre d'un point de vue opérationnel en raison des multiples changements du nombre de tickets et de la taille de lot qui seraient occasionnés. En plus de cela, l'effet de chaque changement pourrait potentiellement être partiellement annulé par l'effet du suivant, ce qui entraînerait un processus d'exploitation très instable (Jaegler et al., 2017).

Cela signifie qu'il sera difficile pour un praticien de calculer les meilleurs ou même les bons paramètres pour dimensionner son ConWip. En effet, même le plus simple des modèles mathématiques de dimensionnement du ConWip est souvent NP-complet et donc difficile à optimiser pour un cas réel qui implique un grand nombre de produits et de machines. Il n'aura alors pas d'autre choix que d'utiliser l'expérience ou de procéder par tests successifs sur les différents paramètres.

Cet état de fait est illustré par (Jaegler et al., 2017) qui expliquent que la littérature met en lumière de nombreuses pistes qui devraient être explorées pour répondre efficacement aux questions liées aux PCS et en relation avec la notion de prise de décision. L'adaptation du ConWip à un environnement spécifique, tel qu'une industrie très complexe caractérisée par une fabrication à la commande, un mix produit élevé, des ateliers de type jobshop et un ensemble hétérogène de gammes opératoires, constitue un défi de recherche important. En plus de cela, ils proposent de prioriser deux voies de recherche ouvertes au sein de la riche littérature au sujet du ConWip.

La première concerne la définition d'un ConWip basée sur une gamme enveloppe pour résoudre les problèmes de mix produit élevé. Dans le cas d'un environnement en mouvement, l'optimisation basée sur l'expérience ou sur des tests successifs prendrait trop de temps pour constituer une réelle solution. Dans ce contexte, la question à laquelle ce chapitre va tenter de répondre est la suivante : comment peut-on déterminer quels paramètres utiliser pour dimensionner un système ConWip dans un environnement multi produits en mouvement ?

4.3. FORMULATION DU PROBLEME

Le système de production multiproduit considéré dans cette section consiste en une ligne de production avec plusieurs postes de travail. Chaque poste de travail peut traiter plusieurs opérations. La ligne de production fabrique divers produits et chaque produit a sa propre gamme opératoire. Chaque gamme comprend plusieurs opérations à traiter à l'aide d'un ensemble de postes de travail. Nous supposons que la ligne est sous un contrôle de production ConWip. Il existe un temps opératoire et un temps de changement de série correspondant à chaque paire (produit, opération).

Le modèle de programmation mathématique utilisé pour le système de production ci-dessus vise à optimiser la gamme enveloppe en utilisant le nombre minimum d'opérations pour la construire et en respectant les conditions de chronologie entre les postes à l'intérieur de cette gamme (aucun retour en arrière autorisé pour aucun des produits).

Les notations suivantes sont utilisées dans notre modèle mathématique.

4.3.1. Paramètres

W_l : le numéro du poste de travail l de l'atelier ou de la ligne de production, $l = 1, \dots, o$

n : le nombre de produits

m : le nombre d'opérations

i : l'indice du produit, $i = 1..n$

j : l'indice de l'opération, $i = 1..m$

m_i : le nombre d'opérations dans la gamme du produit i

T_{ij} : le temps d'installation nécessaire à l'opération j pour le produit i

P_{ij} : le temps de traitement de l'opération j pour le produit i

r_0 : le nombre maximal de postes de travail autorisés à être utilisés pour former la gamme enveloppe sur la chaîne de fabrication

Soit:

- $a_{jl}^i = \begin{cases} 1 & \text{si la } j^{eme} \text{ opération sur le } i^{eme} \text{ produit est faite sur le poste de travail } W_l \\ 0 & \text{sinon} \end{cases}$
 $\sum_{j=1}^{m_i} \sum_{l=1}^o a_{jl}^i = 1, \forall i = 1..n$

4.3.2. Variables de décisions

$$X_{l,p} = \begin{cases} 1 & \text{si le poste de travail } W_l \text{ est assigné à la position } p \\ 0 & \text{sinon} \end{cases}$$

$$Y_{i,j,p} = \begin{cases} 1 & \text{si la } j^{eme} \text{ opération sur le } i^{eme} \text{ produit est faite sur le poste} \\ & \text{de travail assigné à la position } p \\ 0 & \text{sinon} \end{cases}$$

En considérant ces notations et hypothèses, le problème est modélisé comme suit :

$$\min \sum_l \sum_p X_{l,p} \quad (1)$$

Avec :

$$\sum_p X_{l,p} \geq 1, \forall l = 1..o \quad (2)$$

$$\sum_l X_{l,p} = 1, \forall p = 1..r_0 \quad (3)$$

$$\sum_p Y_{i,j,p} X_{l,p} = a_{jl}^i, \forall l = 1..o, \forall i = 1..n, \forall j = 1..m \quad (4)$$

$$\sum_{i,j} Y_{i,j,p} \geq 1, \forall p = 1..r_0 \quad (5)$$

$$\sum_p Y_{i,j,p} = 1, \forall i = 1..n, \forall j = 1..m_i \quad (6)$$

$$\sum_p p Y_{i,j,p} \leq \sum_k k Y_{i,j,k}, \forall j = 1..m_i, j < l, \forall i = 1..n \quad (7)$$

$$X_{l,p} \in \{0,1\} \forall l = 1..o, \forall p = 1..r_0 \quad (8)$$

$$Y_{i,j,p} \in \{0,1\} \forall i = 1..n, \forall j = 1..m_i, \forall p = 1..r_0 \quad (9)$$

4.3.3. Fonction objectif

La fonction objectif (1) consiste à minimiser le nombre de postes de travail utilisés pour former le gamme enveloppe sur la chaîne de fabrication. Les contraintes sont les suivantes :

(2) : Chaque poste de travail doit être affecté à au moins une position dans la gamme enveloppe

(3) : Chaque position dans la gamme enveloppe est affectée à un unique poste de travail

(4) : Chaque tâche j sur le produit i est affectée à un unique poste de travail

(5) : Au moins une opération est effectuée dans chaque poste de travail dans la gamme enveloppe

(6) : Chaque tâche j sur le produit i est affectée à une unique position dans la gamme enveloppe

(7) : Les chronologies des opérations de toutes les gammes opératoires sont respectées par la gamme enveloppe

(8), (9) : Définition des variables de décision

Ce problème est NP complet, l'utilisation d'une méthode dite exacte pour sa résolution est donc possible pour les instances de petite taille, mais pour les plus grandes, nous devons appliquer une procédure heuristique ou métaheuristique par exemple : simulation de recuit, recherche tabou, algorithme génétique ... même si ces dernières ne garantissent pas le caractère optimal de la solution.

4.3.4. Approche de solution

Pour répondre à la question sur l'optimisation du ConWip dans un environnement multiproduit, nous proposons de déterminer une gamme opératoire pouvant être considérée comme représentative de l'ensemble des gammes de fabrication. Cette gamme représentative est appelée gamme enveloppe. Nous l'utiliserons comme la gamme de l'environnement simplifié caractérisé par un produit « virtuel » unique. Cet environnement simplifié simulera l'environnement multiproduit auquel nous nous intéressons. Cette approche simplificatrice permet de réduire la complexité du problème initial afin de lui apporter une réponse théorique.

En outre, dans le cas où la solution s'avère acceptable, elle représente également une option qui permet de résoudre le problème de dimensionnement du ConWip directement dans l'atelier par des opérationnels. Dans ce cadre, le présent chapitre montre comment définir une gamme enveloppe répondant aux deux contraintes suivantes :

- la séquence des opérations dans la gamme enveloppe n'engendrera aucun retour en arrière pour aucun produit ; ceci afin de garantir un pilotage efficace de l'atelier,
- afin d'éviter la redondance des opérations, la gamme enveloppe la plus courte sera recherchée.

Dans ce chapitre, quatre algorithmes capables de construire une gamme enveloppe sont proposés. Après cela, l'objectif poursuivi consiste à qualifier les performances des différents algorithmes afin d'apporter une valeur ajoutée pour les chercheurs et/ou les industriels qui souhaiteraient les utiliser et donc choisir parmi eux.

Pour ce faire, nous avons choisi d'utiliser l'angle de vue d'un industriel qui doit faire face à un environnement multiproduit caractérisé par des instabilités de la demande. Dans un tel cas, il lui est nécessaire d'évaluer l'impact d'un changement (variation du mix modèle, changement de produit, produit personnalisé...) sur la performance de l'atelier. La réponse que nous essayons d'apporter n'est pas directement formulée en matière de performance (temps de traitement, efficacité, délais de livraison), car nous ne cherchons pas à simuler le comportement complet d'un atelier déterminé. Nous essayons plutôt de donner des éléments permettant de faire le meilleur choix entre les algorithmes que nous avons construits. Dans ce cadre, nous avons décidé de comparer les gammes enveloppes induites par les quatre algorithmes à travers l'impact de différents types de variations sur leurs paramètres théoriques optimisés. Si chaque variation implique un changement significatif des paramètres optimisés, cela signifie qu'une perte de performance se produirait dans ce cas particulier.

4.4. METHODOLOGIE

4.4.1. Processus de recherche

Le processus de recherche est le suivant. Nous concevons quatre algorithmes dans le but de définir une gamme enveloppe. Ces gammes enveloppes sont implémentées dans un outil de simulation capable de calculer les paramètres optimisés du ConWip : nombre de tickets et taille de lot.

Les algorithmes construisent des gammes enveloppes sur la base d'un échantillon de données industrielles. Ensuite, nous effectuons une analyse de sensibilité à la fois pour les variations du mix modèle et la séquence des produits. De plus, nous testons la robustesse des résultats des algorithmes dans différents scénarii qui consistent à intégrer un produit atypique dans le mix modèle. Pour simuler ce scénario, nous ajoutons d'abord un temps opératoire élevé dans la gamme de l'un des produits existants. L'augmentation du temps opératoire est intégrée au début de gamme, puis à la fin. Ensuite, nous simulons ce scénario en intégrant un produit atypique dans le mix modèle.

Considérant le problème générique de construction de gamme, nous choisissons ici d'intégrer un produit avec un nombre élevé d'opérations. Pour ce faire, nous convertissons le premier produit de l'échantillon en un produit atypique. Enfin, nous analysons les résultats de l'ensemble des scénarii et concluons.

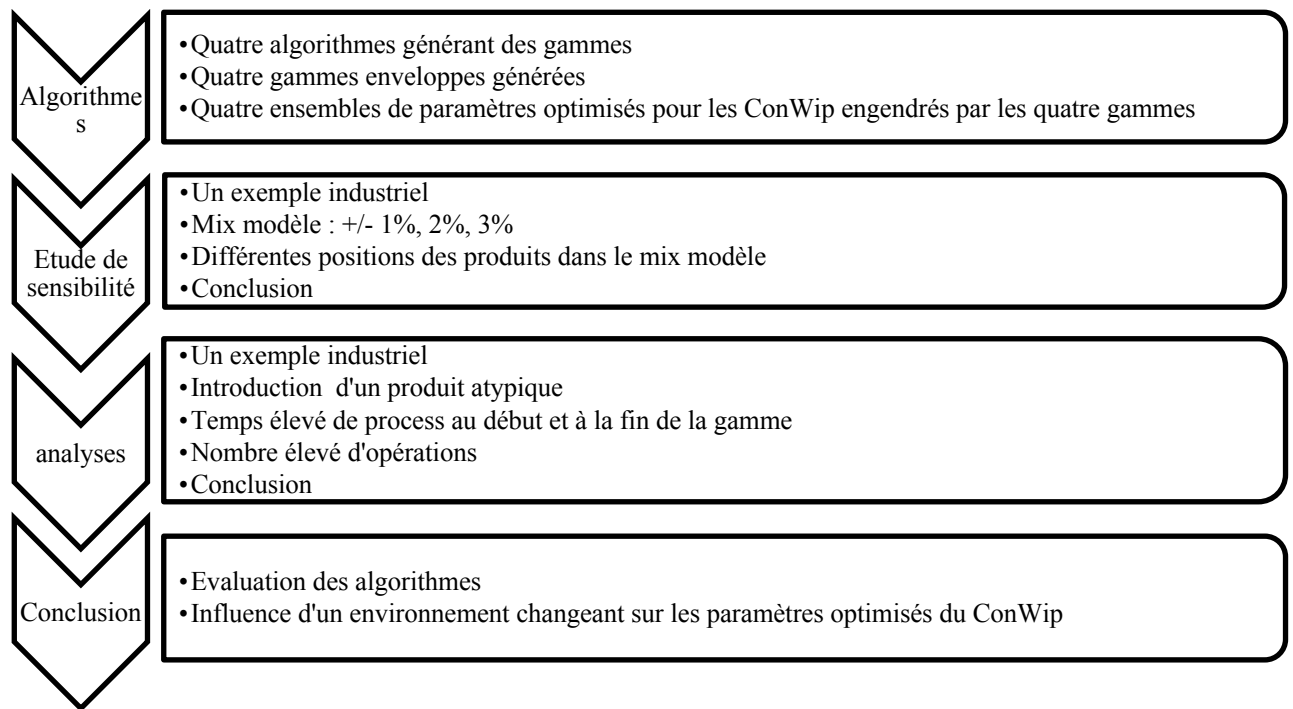


Figure 18. Processus de recherche

4.4.2. Algorithmes

La première étape consiste à concevoir les quatre algorithmes pour déterminer une gamme enveloppe pour une combinaison de gammes données. Nous présentons les quatre algorithmes dans cette sous-section. Pour une meilleure compréhension, il est pertinent de spécifier ici des éléments de vocabulaire :

- la gamme enveloppe générée est appelée "gamme enveloppe finale",
- toutes les gammes successives calculées par les algorithmes pour obtenir finalement la "gamme enveloppe finale" sont appelées "gammes enveloppes intermédiaires". Elles sont nécessaires pour le calcul, mais n'ont pas de valeur pour le résultat final.

Les paramètres suivants sont utilisés :

$mIGRO_k$: le nombre d'opérations dans la gamme enveloppe intermédiaire k

$mGRO$: le nombre d'opérations dans la gamme enveloppe finale

O_{ij} : opération j de la gamme du produit i

$W(O_{ij})$ opération j de la gamme du produit i

β_i poids en pourcentage associé au produit i dans le mix

Ce qui implique :

$$\sum_{i=1}^n \beta_i = 1$$

$IGRT_{kj}$ le temps de réglage de l'opération j pour la gamme enveloppe intermédiaire k

$IGRP_{kj}$ le temps de process de l'opération j pour la gamme enveloppe intermédiaire k

$IGRO_k(W_l, IGRT_j, IGRP_j)$ l'opération j de la gamme enveloppe intermédiaire k

GRT_j le temps de réglage de l'opération j pour la gamme enveloppe finale

GRP_j le temps de process de l'opération j pour la gamme enveloppe finale

$GRO_j(W_l, GRT_j, GRP_j)$ l'opération j de la gamme enveloppe finale

4.4.2.1. Algorithme “deux à deux”

Cet algorithme « additionne » les gammes deux par deux. Cela signifie que deux premières gammes de produits sont considérées, la première opération de la première gamme est comparée avec la première opération de la deuxième gamme, et la deuxième opération de la première gamme est comparée avec la deuxième opération de la deuxième gamme. Nous sommions les opérations si elles sont identiques (ce qui signifie effectuées sur la même machine ou poste de travail). Sinon nous gardons la première. Nous appliquons ce processus à l'ensemble des opérations des deux gammes. Nous obtenons une nouvelle gamme, appelé première gamme enveloppe intermédiaire. Ensuite, nous ajoutons cette nouvelle gamme à la troisième gamme produit pour obtenir la deuxième gamme enveloppe intermédiaire. Le processus se poursuit jusqu'à ce que toutes les gammes des produits aient été ajoutées. L'ensemble de ce processus aboutit à la gamme enveloppe finale. Les opérations j de la gamme enveloppe intermédiaire k sont définies comme suit :

For $k \neq 1$:

$$(1) IGRO_{kj} = \left\{ \begin{array}{l} \beta_{k+1} O_{k+1j} + IGRO_{k-1j} + \beta_{k+1} O_{k+1j+1} + IGRO_{k-1j+1} \\ \text{if } W(O_{k+1j}) = W(IGRO_{k-1j}) = W(O_{k+1j+1}) = W(IGRO_{k-1j+1}) \end{array} \right\}$$

$$(2) IGRO_{kj}$$

$$= \left\{ \begin{array}{l} \beta_{k+1} O_{k+1j} + IGRO_{k-1j} + \beta_{k+1} O_{k+1j+1} \\ \text{if } W(O_{k+1j}) = W(IGRO_{k-1j}) = W(O_{k+1j+1}), \text{ and } <> W(IGRO_{k-1j+1}) \end{array} \right\}$$

$$(3) IGRO_{kj}$$

$$= \left\{ \begin{array}{l} \beta_{k+1} O_{k+1j} + IGRO_{k-1j} + IGRO_{k-1j+1} \\ \text{if } W(O_{k+1j}) = W(IGRO_{k-1j}) = W(IGRO_{k-1j+1}), \text{ and } <> W(O_{k+1j+1}) \end{array} \right\}$$

$$(4) IGRO_{kj}$$

$$= \left\{ \begin{array}{l} \beta_{k+1} O_{k+1j} + \beta_{k+1} O_{k+1j+1} + IGRO_{k-1j+1} \\ \text{if } W(O_{k+1j}) = W(O_{k+1j+1}) = W(IGRO_{k-1j+1}), \text{ and } <> W(IGRO_{k-1j}) \end{array} \right\}$$

$$(5)$$

$$IGRO_{kj} =$$

$$\left\{ \begin{array}{l} \beta_{k+1} O_{k+1j} + IGRO_{k-1j} \\ \text{if } W(O_{k+1j}) = W(IGRO_{k-1j}), \text{ and } <> W(O_{k+1j+1}), \text{ and } <> W(IGRO_{k-1j+1}) \end{array} \right\}$$

$$(6)$$

$$IGRO_{kj} =$$

$$\left\{ \begin{array}{l} \beta_{k+1} O_{k+1j} + \beta_{k+1} O_{k+1j+1} \\ \text{if } W(O_{k+1j}) = W(O_{k+1j+1}), \text{ and } <> W(IGRO_{k-1j}), \text{ and } <> W(IGRO_{k-1j+1}) \end{array} \right\}$$

$$(7)$$

$$IGRO_{kj} =$$

$$\left\{ \begin{array}{l} \beta_{k+1} O_{k+1j} + IGRO_{k-1j+1} \\ \text{if } W(O_{k+1j}) = W(IGRO_{k-1j+1}), \text{ and } <> W(O_{k+1j+1}), \text{ and } <> W(IGRO_{k-1j}) \end{array} \right\}$$

$$(8)$$

$$IGRO_{kj} =$$

$$\left\{ \begin{array}{l} \beta_{k+1} O_{k+1j} \\ \text{if } W(O_{k+1j}) <> W(IGRO_{k-1j+1}), \text{ and } <> W(O_{k+1j+1}), \text{ and } <> W(IGRO_{k-1j}) \end{array} \right\}$$

For $k = 1$, le process est le même en utilisant $\beta_1, O_{1j}, \beta_2, O_{2j}$ au lieu de $\beta_{k+1}, O_{k+1j}, O_{k+1j+1}, IGRO_{k-1}, IGRO_{k-1j+1}$ sauf dans les cas de figure (4) et (7) qui sont alors interdits dans la mesure où ils conduiraient à un retour en arrière.

4.4.2.2. Algorithme de “la plus grande occurrence”

Nous considérons ici toutes les premières opérations de tous les produits. Le poste de travail ayant l'occurrence la plus élevée est alors défini comme le poste de travail ($W(GRO_1)$) de la première opération (GRO_1) de la gamme enveloppe (parmi les $W(O_{1j})$, $j = 1, \dots, m$). Nous calculons le temps opératoire GRP_1 de la première opération de gamme enveloppe en ajoutant les temps de traitement O_{1j} qui correspondent au poste de travail ($W(GRO_1)$) multipliés par leurs poids respectifs dans le mix. Le même calcul est effectué pour le temps de changement de série. Si toutes les opérations sont différentes et affichent par conséquent chacune une occurrence de 1, celle du premier produit est conservée pour la gamme enveloppe. Toutes les opérations inutilisées sont ensuite décalées d'une étape vers l'aval. Nous réappliquons le processus jusqu'à la dernière opération. Pour définir la gamme enveloppe, nous calculons :

C_l les occurrences des $W(O_{jl})$, $l = 1, \dots, o$

Ensuite, nous sélectionnons z vérifiant : $C_z = \text{Max}(C_l)$, $l = 1, \dots, o$

Ensuite, nous calculons :

$$\left\{ \sum_{i=1}^n \beta_i T_{ij} \text{ with } i / W(O_{ij}) = W(O_{iz}) \right\}$$

$$GRP_j = \left\{ \sum_{i=1}^n \beta_i P_{ij} \text{ with } i / W(O_{ij}) = W(O_{iz}) \right\}$$

Nous définissons pour finir : $GRO_j(W(O_{jz}), GRT_j, GRP_j)$

4.4.2.3. Algorithme du “plus grand temps de process”

Nous considérons toutes les premières opérations de tous les produits. Nous considérons ensuite les sommes, pour chaque poste de travail, des temps opératoires issus de ces premières opérations. Nous définissons le poste de travail ($W(GRO_1)$) de la première opération (GRO_1) de la gamme enveloppe comme le poste de travail (parmi les $W(O_{1j})$, $j = 1, \dots, m$) qui est le plus chargé lorsqu'on considère les sommes évoquées ci-dessus. Nous calculons le temps de traitement GRP_1 de la première opération de la gamme enveloppe en ajoutant les temps de traitement des O_{1j} correspondant au poste de travail ($W(GRO_1)$) multipliés par leurs poids respectifs dans le mix. Le même calcul est effectué pour le temps de changement de série.

Si toutes les opérations sont différentes et que plusieurs d'entre elles sont chargées de manière égale, l'opération du premier produit considéré est conservée. Toutes les opérations inutilisées sont décalées vers l'aval. Nous réappliquons le processus jusqu'à la dernière opération. Pour définir la gamme enveloppe, nous recherchons :

$$z / \sum_{i=1}^n P_{iz} = \text{Max} \left(\sum_{i=1}^n \sum_{j=1}^m P_{ij} \right)$$

Puis nous calculons :

$$GRT_j = \sum_{i=1}^n \beta_i T_{ij} \text{ with } i / W(O_{ij}) = W(O_{iz})$$

$$GRP_j = \sum_{i=1}^n \beta_i P_{ij} \text{ with } i / W(O_{ij}) = W(O_{iz})$$

Enfin, nous définissons : $GRO_j(W(O_{jz}), GRT_j, GRP_j)$

4.4.2.4. Algorithme accordéon

Pour cet algorithme, nous considérons toutes les gammes des différents produits dans un même temps. Nous considérons le poste de travail de la première opération du premier produit ($W(O_{11})$). Si le poste de travail de la première opération du deuxième produit $W(O_{21})$ est le même, il reste dans cette position. Si les deux postes de travail sont différents, il passe en position deux dans la gamme enveloppe, ce qui signifie que (O_{21}) devient (O_{22}) et (O_{21}) est remplacé par une opération vide (pas de poste de travail, temps de changement de série égal à 0 et temps de traitement égal à 0). Nous répétons ce processus pour les premières opérations de tous les produits et nous ajoutons toutes les premières opérations restantes. La deuxième étape consiste à passer par le même processus pour la deuxième opération de tous les produits, et ainsi de suite, jusqu'à la dernière opération du dernier produit. Si nécessaire, une opération du premier produit peut également être sautée et remplacée par une opération vide. L'ensemble du processus est illustré par un exemple schématique :

La convention de notation suivante est utilisée dans l'exemple ci-dessous :

Product 1

A		B		C		D	
0,5	2	1	3	0,5	4	1	4

Product 2

B		A		E		D	
1	2	0,5	3	0	0,5	1	3

Product 3

A		C		E	
0,5	2	1	3	0	1

Products	Mix
1	20%
2	40%
3	20%

Workstation	
Setup time	Operating Time

Product 1

A		B		C	
0,5	2	1	3	0,5	4

D	
1	4

Product 2

B	
1	2

A		E		D	
0,5	3	0	0,5	1	3

Product 3

A	
0,5	2

C	
1	3

E	
0	1

Generic Routing

A		B		C		A		E		D	
0,2	0,8	0,6	1,4	0,3	1,4	0,2	1,2	0	0,4	0,6	2

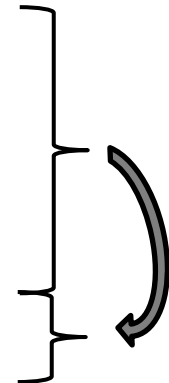


Figure 19. Ensemble du processus

4.5. RESULTATS

Nous testons maintenant ces quatre algorithmes avec un échantillon numérique, dérivé d'un cas industriel. Le but de cette section est de qualifier les quatre algorithmes et d'évaluer leur pertinence.

4.5.1. Exemple numérique

Dans cet exemple simplifié, nous considérons un atelier de production qui fabrique six produits (1 à 6) en utilisant 6 postes de travail (A à F) : deux machines (A et C), trois postes de fraisage (B, D, E) et un poste de travail de contrôle (F). Un temps de process (TP) et un temps de réglages (TR) sont associés à chaque opération.

Tableau 11. Poids dans le mix modèle associés à chaque produit

Produits		1	2	3	4	5	6
Poids dans le mix		17%	17%	17%	17%	17%	17%
Opération 1	nom	A	C	A	A	A	B
	TR	0,5	0,5	0,5	0,5	0,5	0,5
	TP	2	3	2	3	2	2
Opération 2	nom	B	D	B	C	C	C
	TR	1	0,5	1	0,5	1	1
	TP	3	3	3	3	3	3
Opération 3	nom	C	E	C	B	D	D
	TR	0,5	1	0,5	1	0,5	0,5
	TP	4	3	4	3	4	4
Opération 4	nom	D	F	F	F	E	E
	TR	1	0	0	0	1	1
	TP	4	1	0,5	1	4	4
Opération 5	nom	F				F	F
	TR	0				0	0
	TP	0,5				0,5	0,5

Le modèle mathématique présenté dans la section "Formulation du problème", pour un problème de taille moyenne (Tableau 12), est codé par un solveur général standard, ILOG CPLEX, sur un PC typique (Intel Core i7 2600 @ 3.4 GHz, 8 Go Ram). CPLEX, qui résout le modèle en utilisant la méthode Branch and Bound, a besoin de quelques minutes pour trouver la solution optimale.

Le mix modèle de cet échantillon est parfaitement homogène. Les gammes enveloppes obtenues grâce aux quatre algorithmes sont les suivantes.

Tableau 12. Gammes enveloppes obtenues

Algorithme		Deux à deux	Plus grande occurrence	Plus grand temps de process	Accordéon
Poste 1	nom	B	A	A	A
	TR	0,08	0,25	0,33	0,33
	TP	0,3	1,5	1,5	1,5
Poste 2	nom	A	B	C	B
	TR	0,33	0,375	0,33	0,4
	TP	1,5	1,2	1,5	1,33
Poste 3	nom	C	C	B	C
	TR	0,6	0,65	0,58	0,67
	TP	2,67	3,3	1,83	3,3
Poste 4	nom	D	D	C	D
	TR	0,17	0,4	0,33	0,42
	TP	1,3	2,4	1,83	2,5
Poste 5	nom	E	E	D	E
	TR	0,17	0,5	0,42	0,5
	TP	0,7	1,8	2,5	1,83
Poste 6	nom	B	F	E	B
	TR	0,5	0	0,5	0,2
	TP	1,5	0,5	1,83	0,5
Poste 7	nom	F	B	F	F
	TR	0	0,2	0	0
	TP	0,4	0,6	0,67	0,7
Poste 8	nom	E	F		
	TR	0,17	0		
	TP	0,7	0,2		
Poste 7	nom	D			
	TR	0,1			
	TP	0,5			
Poste 8	nom	C			
	TR	0,1			
	TP	0,7			
Poste 9	nom	E			
	TR	0,17			
	TP	1			
Poste 10	nom	D			
	TR	0,2			
	TP	0,67			
Poste 11	nom	F			
	TR	0			
	TP	0,25			

Les paramètres optimisés des ConWip liés à chacune des 4 gammes enveloppes sont calculés grâce à Wipsim qui est un outil d'ingénierie utilisé dans les projets de conception et d'amélioration des lignes ConWip.

Tableau 13. Paramètres optimisés du ConWip

Modèle	Tickets ConWip	Tailles de lot
Deux à deux	11 to 14	2
Plus grande occurrence	6 to 8	2
Plus grand temps de process	7 to 9	3
Accordéon	6 to 7	3

Les valeurs de ces paramètres fournissent des résultats de référence. Nous pouvons remarquer que les trois derniers algorithmes induisent des paramètres optimisés similaires alors que le premier algorithme propose un nombre significativement plus élevé de tickets ConWip. Dans l'étape suivante, nous testerons la sensibilité des quatre algorithmes au poids de chacun des produits présents dans le mix.

4.5.2. Analyse de sensibilité

Une étude de sensibilité à la variation du mix modèle est clairement pertinente pour qualifier les quatre algorithmes. En effet, prenons le cas où un industriel utilise une gamme enveloppe puis les paramètres optimisés du ConWip associé. Après quoi la boucle ConWip est dimensionnée sur la ligne d'assemblage grâce aux paramètres calculés. Il devient pertinent de savoir ce qui se passera dans le cas où le mix produit évoluerait. Par exemple, cela peut se produire dans le cas d'une demande instable ou dans le cas de prévision de vente inexacte ou même dans le cas d'une commande annulée. Dans tous ces cas, une faible sensibilité à la variation du mix modèle constituerait un atout majeur pour un industriel.

4.5.2.1. Sensibilité aux variations du mix produit

Dans ce cadre, nous décidons, pour chaque produit, de simuler trois changements de leurs poids respectifs dans le mix modèle : + -1%; + -2%; + -3%. L'amplitude de ces changements a été dimensionnée pour couvrir les cas d'une variation du mix modèle considérée comme habituelle.

Dans tous les cas, si un changement majeur venait à se produire, nous supposons que les paramètres ConWip doivent quoi qu'il en soit être recalculés et que la boucle ConWip doit être redimensionnée afin de maintenir l'efficacité de la chaîne d'assemblage.

Tableau 14. Résultats de la sensibilité aux variations du mix produit

Algorithmes		Référence	+/-1%	+/-2%	+/-3%
Deux à deux	Variation Mix	0	0	4,52	9,36
	Tickets ConWip	11 to 14	11 to 14	11 to 14	11 to 14
	Taille de lot	2	2	2	2
Plus grande occurrence	Variation Mix	0	2,31	4,81	7,52
	Tickets ConWip	6 to 8	6 to 8	6 to 8	6 to 8
	Taille de lot	2	2	2	2
Plus grand temps de process	Variation Mix	0	1,17	2,38	3,63
	Tickets ConWip	7 to 9	7 to 9	7 to 9	7 to 9
	Taille de lot	2 to 3	2 to 3	2 to 3	2 to 3
Accordéon	Variation Mix	0	2,04	4,23	6,6
	Tickets ConWip	6 to 7	6 to 7	6 to 7	6 to 7
	Taille de lot	2	2	2	2

Les résultats montrent qu'il n'y a pas de changement en matière de paramètres optimisés ConWip. Les quatre algorithmes ont une faible sensibilité à une variation de mix modèle considérée comme habituelle.

4.5.2.2. Sensibilité à l'ordre des produits pour les algorithmes asymétriques

Les algorithmes deux à deux gammes et accordéon sont asymétriques. Cela signifie qu'un mix modèle donné peut conduire à différentes gammes enveloppes en fonction de l'ordre dans lequel les produits sont « injectés » dans l'algorithme. Par exemple, le choix du premier produit peut modifier la gamme enveloppe finale.

Dans ce cadre, une analyse de sensibilité à la séquence de produits utilisée pour construire la gamme enveloppe s'avère nécessaire. Pour effectuer cette analyse, nous modifions l'emplacement des produits dans la séquence de produits pour conclure sur l'influence de l'ordre en question. Les différentes séquences sont présentées dans le tableau 15.

En blanc, les produits avec 4 opérations, en gris clair avec 5 opérations, et en gris soutenu avec des séquences d'opérations différentes. Les produits 1, 5 et 6 ont cinq opérations. Les produits 2, 3 et 4 ont quatre opérations. Le produit 2 est différent des autres, car sa séquence d'opérations est atypique. Les 5 scénarios choisis mettent en évidence les éléments suivants :

- Scénario 1 : scénario de référence
- Scénarios 2 et 3 : le produit 2 est au début de la séquence. Tout d'abord, les produits sont choisis au hasard (scénario 2) puis les deux premiers produits sont choisis avec quatre opérations.
- Scénario 4 : les trois premiers produits ont cinq opérations. Nous concevons un scénario avec le produit atypique en avant-dernière position.
- Scénario 5 : les deux premiers produits ont quatre opérations. Nous concevons un scénario avec le produit atypique en milieu de séquence.

Tableau 15. Scénarii des différentes séquences de produits

Scénarios	Séquence des produits					
scénario 1: reference	1	2	3	4	5	6
scénario 2	2	3	1	4	5	6
scénario 3	2	3	4	1	5	6
scénario 4	1	5	6	4	2	3
scénario 5	3	4	2	1	5	6

Nous injectons ces cinq scénarii dans les algorithmes « accordéon » et « deux à deux ». Les résultats sont synthétisés dans le tableau 16. Le nombre d'opérations représente la longueur de la gamme enveloppe définie par chaque algorithme, les tickets ConWip et la taille du lot représentent les paramètres optimisés calculés par Wipsim pour les gammes enveloppes données.

Tableau 16. Résultats de la sensibilité aux variations de séquences des produits

	Nombre d'opérations	Ticket ConWip	Taille de lot
Deux à deux			
scénario 1: reference	11	11 to 14	2
scénario 2	16	16 to 17	2
scénario 3	10	9 to 12	2
scénario 4	14	15 to 16	2
scénario 5	14	15 to 16	2
Accordéon			
scénario 1: reference	6	6 to 7	2
scénario 2	9	11 to 12	2
scénario 3	9	11 to 12	2
scénario 4	6	6 to 7	2
scénario 5	6	6 to 7	2

Il apparaît que l'algorithme « deux à deux » est relativement sensible à la position des produits dans la séquence. Le nombre d'opérations dans les gammes enveloppes est fluctuant avec une amplitude de 10 à 16 et le nombre optimal de tickets ConWip est compris entre 9 et 16. Les paramètres de gestion visuelle et de ConWip sont donc clairement et significativement influencés par le choix de la séquence. L'algorithme « accordéon » est plus stable. Nous obtenons deux types de résultats : les scénarios 1, 4 et 5 apportent les mêmes paramètres optimisés alors que les scénarios 2 et 3 apportent des paramètres significativement différents. Le produit 2 est à la première place dans les scénarios 2 et 3. Ce produit a une gamme significativement différente par rapport aux autres produits. En effet, les postes de travail A et B sont absents de sa gamme opératoire. Cet échantillon de données a tendance à montrer qu'il est recommandé de construire une séquence de produits qui débute avec les gammes les plus typiques. En effet, par construction de l'algorithme, les premiers produits injectés ont une influence prégnante sur le profil de la gamme enveloppe. Pour favoriser une faible sensibilité pour cet algorithme, le produit atypique doit par conséquent être situé à la fin de la séquence de produits.

4.5.3. Analyse des scénarios

Après l'analyse de sensibilité qui a montré que les quatre algorithmes ont un taux de sensibilité réduit aux variations dites faibles du mix modèle et que l'algorithme « deux à deux » souffre d'une sensibilité significative à l'ordre des produits dans la séquence prise en compte, nous testerons les quatre algorithmes dans un contexte hétérogène. A cet effet, deux types de tests seront effectués pour qualifier le rendu des différents algorithmes dans un contexte hétérogène : le premier test cible une hétérogénéité des temps opératoires, le second test une hétérogénéité du nombre d'opérations.

4.5.3.1. Temps opératoires élevés

Le premier test concerne l'augmentation significative du temps opératoire dans un produit existant. Ce test vise à vérifier l'évolution des paramètres optimisés de chaque gamme enveloppe dans le cas typique d'un ordre de fabrication personnalisé qui impliquerait une augmentation du temps opératoire sur une machine donnée.

Pour tester la robustesse des résultats de référence, nous changeons un temps opératoire standard en un temps de traitement significativement plus élevé (15 h au lieu d'une moyenne de 3 h). Nous décidons d'introduire ce changement à différents endroits dans la gamme. En effet, compte tenu du mode de fonctionnement des quatre algorithmes, nous souhaitons également vérifier si cet emplacement a une influence sur le résultat final. Une des vocations de ce test consiste à orienter le choix entre deux algorithmes pour un industriel ou un chercheur qui aurait à faire face à un problème de produits personnalisés par exemple.

Le poids du produit atypique dans le mix modèle est de 5%. Le poids des autres produits est de 19% chacun. Nous simulons 6 scénarii : pour les produits 1, 3 et 6, nous intégrons une opération avec un temps de traitement élevé d'abord au début de la gamme opératoire, puis à la fin.

Tableau 17. Résultats des temps opératoires élevés

Modèle		Référenc e	Produit 1		Produit 3		Produit 6		Valeurs partagée s
Produit atypique		0	Début	Fin	Début	fin	Début	fin	
Deux à deux	ticket		13 to 14	15 to 15	15 to 16	13 to 15	14 to 16	14 to 16	13 to 15
	ConWip								
	taille de lot	2	2	2	2	2	2	2	2
Plus grande occurrence	ticket	6 to 8	7 to 8	7 to 9	7 to 10	5 to 7	7 to 8	6 to 8	7 to 8
	ConWip								
	taille de lot	2	2	2	2	2	2	2	2
Plus grand temps de process	ticket						10 to 13		
	ConWip	7 to 9	8 to 9	8 to 10	8 to 11	7 to 10		7 to 11	N/A
	taille de lot	2	2	2	2	2	2	1 to 2	2
Accordéon	ticket	6 to 7	7 to 9	6 to 7	7 to 8	5 to 7	6 to 9	7 to 8	7
	ConWip								
	taille de lot	2	2	2	2	2	2	2	2

L'algorithme « deux à deux » est plus sensible à l'introduction d'un temps de fonctionnement élevé que les algorithmes de « la plus grande occurrence » et « accordéon ». L'algorithme du « temps opératoire le plus élevé » est plus sensible si le temps est ajouté à la fin de la gamme opératoire du produit que s'il est ajouté au début.

La notion de "valeurs partagées" montre que pour l'algorithme du « temps opératoire le plus élevé », il n'y a pas de valeur partagée pour les tickets ConWip entre tous les scénarii. La « plus grande occurrence » et « accordéon » sont les algorithmes les plus stables.

4.5.3.2. Nombre élevé d'opérations

Une autre hétérogénéité peut provenir de l'introduction d'un produit atypique et par exemple d'un produit complexe qui nécessite plus d'opérations (ici 10 au lieu de 5) avec par exemple des aller-retours au fraisage et à l'usinage. Le poids de ce produit est de 5%. Les autres produits pèsent 19% chacun. Le produit 1 est modifié conformément à ces spécifications.

Tableau 18. Résultats du test avec un nombre élevé d'opérations

Algorithmes		Référence	Nombre élevé d'opérations
Deux à deux	tickets ConWip	11 to 14	21 to 22
	taille de lot	2	2
Plus grande occurrence	ticket ConWip	6 to 8	13 to 16
	taille de lot	2	2
Plus grand temps de process	ticket ConWip	7 to 9	13 to 16
	taille de lot	2 to 3	2
Accordéon	ticket ConWip	6 to 7	12 to 14
	taille de lot	2 to 3	2

Tous les algorithmes sont sensibles à ce changement. Cette conclusion montre l'ampleur de l'impact de la nouvelle gamme allongée d'un produit sur la gamme enveloppe de l'ensemble de produits. Quel que soit l'algorithme, les gammes enveloppes recalculées induisent des paramètres optimisés avec deux fois plus de tickets que la gamme enveloppe de référence.

Synthèse du chapitre 4

Tableau 19. Synthèse

Algorithme	Variation du mix modèle	Localisation de la séquence	Temps de process élevé	Nombre élevé d'opérations
Deux à deux	Sensibilité faible	Sensibilité élevée	Sensibilité élevée avec localisation à la fin de la gamme	Sensibilité
Plus grande occurrence	Sensibilité faible	N/A	Faible sensibilité	Sensibilité
Plus grand temps de process	Sensibilité faible	N/A	Sensibilité élevée avec localisation au début de la gamme	Sensibilité
Accordéon	Sensibilité faible	Sensibilité si le produit avec une séquence différente est au début	Faible sensibilité	Sensibilité

Le chapitre 4 propose une approche de l'optimisation du ConWip dans un environnement multiproduit. L'objectif consiste à fournir aux industriels des outils « conviviaux » et simples pour leur permettre d'optimiser un atelier. Dans la mesure où la méthode pour optimiser un ConWip dans un environnement monoproduit est connue, notre approche consiste à construire une gamme enveloppe pour réduire la complexité du problème multiproduit. Les quatre algorithmes proposés pour construire une gamme enveloppe sont la première contribution présentée ici.

Dans un second temps, le présent chapitre propose une analyse visant à qualifier la pertinence des quatre algorithmes et ainsi aider un décideur à choisir lequel sera le plus pertinent dans le cadre de son besoin spécifique ou du moins à comprendre les forces et les faiblesses des quatre alternatives proposées ici. Dans cette partie, différents résultats sont obtenus grâce à l'utilisation d'un échantillon de données industrielles :

- les quatre algorithmes sont peu sensibles à des variations limitées du mix modèle,
- les algorithmes asymétriques nécessitent une attention particulière à la séquence de produits qui leur est injectée, car celle-ci peut affecter la nature de la gamme enveloppe qui en découlera,