RAPPORT DE MODAL : RECHERCHE OPÉRATIONNELLE

Modélisation d'un atelier de production

15 mai 2023

Oussama Eddaoudi, Simon Setti, Ghada Ben Slama, tuteur : Frédéric Meunier

École Polytechnique
IP Paris

TABLE DES MATIÈRES

ABSTRACT

In a world where delivery speed and efficiency have become important criteria in the choice of a product or service, knowledge and optimization of lead time (the time required to produce and deliver a product to a customer) are essential to ensure the competitiveness and profitability of a manufacturing company. In this study, different methods have been tested to minimize the processing time of a product in a manufacturing chain, such as assigning an employee to the largest waiting list or allowing the employee to choose which product to start with in a machine's waiting list. The results show that several assignment rules stand out. Our presentation concludes with a theoretical result on the choice of assignment rule in the particular case of two employees with three machines.

REMERCIEMENTS

Nous tenons à exprimer notre gratitude envers toutes les personnes qui ont contribué à la réalisation de notre projet MODAL.

Tout d'abord nous tenons à remercier Frederic Meunier, notre enseignant et tuteur, pour son soutien sans faille tout au long du projet. Il a été présent pour nous rencontrer toutes les semaines et consacrer du temps à nous aider dans les parties informatiques et théoriques de notre projet. Sa contribution a été essentielle pour la réussite de notre projet.

Nous sommes également reconnaissants envers l'École Polytechnique pour nous avoir offert l'opportunité de réaliser un projet de haut niveau avec des enseignants de qualité. Nous sommes convaincus que cette expérience nous a permis de développer nos compétences et d'acquérir une expérience précieuse pour notre avenir professionnel.

Vos encouragements ont été une source de motivation pour nous. Merci à tous pour votre soutien et votre contribution à la réussite de notre MODAL.

1 INTRODUCTION

1.1 Contexte

Dans notre monde en constante évolution, les attentes des clients en matière de délai de livraison sont de plus en plus élevées. La rapidité et l'efficacité de la livraison sont désormais des critères primordiaux dans le choix d'un produit ou d'un service.

Imaginons un client qui souhaite acheter une voiture. Lorsqu'on envisage l'achat d'une voiture, on cherche généralement celle qui sera disponible dans les plus brefs délais. Ainsi, si l'on a le choix entre une voiture moins chère mais qui ne sera disponible que dans deux ans et une autre plus chère mais prête à être livrée dans dix jours, on choisira probablement la deuxième option. C'est cette réalité qui nous a motivés à explorer le sujet et à chercher à optimiser le temps nécessaire pour produire et livrer dans un atelier de fabrication, que l'on appelle le lead time.

Le lead time est un facteur crucial dans l'industrie de la fabrication. Il représente le temps nécessaire pour produire et livrer un produit à un client. En étudiant et en optimisant le leadtime, les entreprises peuvent améliorer l'efficacité de leur chaîne de production, réduire les coûts et augmenter la satisfaction des clients. La connaissance du leadtime permet également de planifier efficacement les opérations de production, de minimiser les délais de livraison et d'optimiser les niveaux de stock. L'étude du leadtime est un élément clé pour assurer la compétitivité et la rentabilité d'une entreprise de fabrication. C'est essentiel pour garantir la réussite d'une entreprise de fabrication dans un marché en constante évolution.

1.2 Contribution

Aprés avoir testé plusieurs façons afin de minimiser le temps de traitement d'un produit dans une chaîne de fabrication, nous avons parvenu à identifier quelques méthodes qui peuvent être utiles pour les instances que nous avons testés. Par exemple, affecter un employé à la liste d'attente la plus grande est une bonne méthode. En prévoyant le futur et en regardant quelle machine va être la prochaine machine dans le parcours d'un produit et en affectant un employé dessus est une bonne façon pour minimiser le temps de traitement sans oublier de prendre aussi la machine qui a le moins d'employés capables de l'utiliser.

Nous avons aussi réfléchi à rajouter un degré de liberté dans notre chaîne de fabrication en donnant la possibilité à l'employé de choisir avec quel produit commencer dans la liste d'attente d'une machine ce qui permettra d'accélerer le travail. Nous avons trouvé que c'était trés utile de choisir le produit qui a le temps de service et le temps d'attente à la prochaine machine le plus petit.

2
FORMALISATION DU PROBLÈME

Notre problème consiste en une modélisation informatique et mathématique d'un atelier doté d'une capacité de production, à savoir des machines et des employés capables de gérer et de servir l'ensemble des produits qui entrent dans le système. En ce qui concerne les employés, ils sont tous distingués par leurs propres qualifications en termes des différentes machines sur lesquelles ils sont capables d'effectuer des tâches. Ils peuvent notamment se déplacer librement d'une machine à l'autre ou en salle d'attente s'ils ne sont pas occupés. On suppose que les machines sont suffisamment proches les unes des autres pour pouvoir négliger le temps de déplacement des employés. Des produits de différents types arrivent dans l'atelier selon un processus de Poisson et avec une intensité d'arrivée qui varie d'un type à l'autre. Les produits n'ont pas nécessairement le même parcours dans l'atelier (cela dépend bien entendu de la nature et donc du type du produit en question). Le temps de service d'un produit sur chaque machine de son parcours dépend uniquement du type de ce produit, ce temps est tiré suivant une loi uniforme sur des intervalles donnés.

2.1 Mise en équation du problème

Dans un premier temps, nous avons cherché à définir un certain nombre de variables mathématiques afin de modéliser notre atelier. Nous avons supposé que notre atelier était capable de gérer plusieurs types de produits, notés de 1 à P, où P est le nombre total de types de produits. Par la suite, nous avons associé à chaque type de produit p un ensemble de machines par lesquelles il doit passer, formant ainsi son parcours sur l'atelier. Ces ensembles de machines sont représentés par $\left(n_{p_1}, n_{p_2}, \ldots, n_{p_{M_p}}\right)$, où M_p est le nombre de machines pour le type de produit p.

De plus, nous avons pris en compte le nombre d'employés dans notre atelier, noté W, en désignant chaque employé par un indice w allant de 1 à W. Chaque employé w possède un ensemble de compétences, noté S_w, qui définit les machines sur lesquelles il est qualifié pour travailler. Ainsi, S_w est un sous-ensemble de l'ensemble des machines $\{1, 2, \ldots, M\}$.

Dans notre modélisation, nous avons également attribué à chaque machine m dans le parcours d'un produit de type p une durée de service T_{pm}. Cette durée de service est tirée de manière aléatoire selon une distribution uniforme dans l'intervalle $[t_{min}^{pm}, t_{max}^{pm}]$. Cependant, il est important de noter que cette durée de service T_{pm} est définie uniquement si la machine m appartient à l'ensemble des machines $n_{pj} \mid 1 \leqslant j \leqslant M_p$ du parcours du produit p.

Enfin, nous avons formulé une hypothèse selon laquelle les intervalles de temps séparant les arrivées successives de deux produits du même type p suivent un processus de Poisson avec un paramètre moyen λ_p représentant la fréquence moyenne d'arrivée de ces produits.

2.2 La fonction objectif

L'objectif de notre projet sera d'étudier et de minimiser une grandeur précise qui est la moyenne du temps passé par chaque produit dans l'atelier (appelée aussi le *Lead Time*). Cette grandeur reflète l'efficacité de notre atelier et c'est donc ce que nous allons essayer d'optimiser tout au long de notre projet. Pour ce faire, la démarche que nous avons adoptée est telle que si un employé finit une tâche, alors nous déterminons, parmi toutes les machines qui lui sont disponibles (c'est-à-dire qui disposent d'une liste d'attente non vide et n'ont aucun employé qui travaille dessus), la prochaine destination de l'employé. Nous appelons ce choix de l'employé une règle d'affectation. Le but sera donc de chercher des règles d'affectation pour lesquelles nous avons un *Lead Time* le plus petit possible. Toutefois, le *Lead Time* est une grandeur assez connue dans les modèles mathématiques qui abordent la question de la Supply Chain, et la loi de Little est un moyen qui permet d'exprimer cette grandeur en fonction du nombre moyen de produits présents dans l'atelier (qu'on appelle l'en-cours) et du taux de production. En effet, le *Lead Time* est égal à l'en-cours multiplié par le taux de production.

Nous avons notamment pu associer au *Lead Time* une expression mathématiques de la manière suivante :

$$\begin{aligned}
\textit{Lead Time} &= \mathbb{E}[\textit{temps de présence du produit dans l'atelier}] \\
&= \mathbb{E}[\mathbb{E}[\textit{Lead Time} \mid \textit{produit tiré}]] \\
&= \sum_{p=1}^{P} \mathbb{E}[\textit{Lead Time} \mid \textit{produit tiré} = p] \cdot \mathbb{P}(\textit{produit tiré} = p) \\
&= \lim_{N \to +\infty} \frac{1}{N} \sum_{i=1}^{N} (\textit{temps de présence de } i)
\end{aligned}$$

Cette dernière égalité est justifiée par l'hypothèse ergodique (On en parlera dans la partie expérimentation). Pour la probabilité $\mathbb{P}(\textit{produit tiré} = p)$, on pourra l'exprimer en fonction des moyennes d'arrivées de chaque type de produits de la manière suivante :

$$\mathbb{P}(\textit{produit tiré} = p) = \frac{\lambda_p}{\sum_{p' \neq p} \lambda_{p'}}$$

3 Règles de décision

Le choix des règles d'affectation est crucial dans ce projet, car elles visent à minimiser le temps de traitement des produits : le *Lead Time*. Nous présentons ci-dessous quelques règles auxquelles nous avons pu réfléchir et leur justification pour optimiser le processus. Nous allons ensuite testé ces différentes règles sur les différentes instances pour déterminer la meilleure.

Nous aurons donc besoin de trois types de règles : La première qui sert à affecter un employé venant de finir sa tâche à une machine disponible ou à la salle d'attente. Une deuxième pour choisir le produit à traiter dans la liste d'attente d'une machine, ainsi qu'une dernière règle pour affecter un employé à une machine bien spécifique.

3.1 Affectation aux machines

Pour chaque produit i, on note par p_i son type et donc sur une machine j, la quantité $T_{p_i j}$ représente le temps de service de i sur j. On désigne également par L_j la liste qui contient les produits en attente pour la machine j.

Ensuite, pour chaque machine j, $T(j)$ retourne le nombre des employés qui sont capables d'utiliser la machine j et B est la fonction telle que $B(j)$ qui vaut 1 si j est occupé par un employé et 0 sinon. Une autre fonction dont on aura besoin pour introduire nos règles d'affections étant f telle que pour tout produit $i \in L_j$, $f(i)$ retourne la machine suivante dans le parcours de i. Nous proposons 4 règles d'affectations :

— La machine la moins occupée : La règle la plus simple et intuitive consiste à choisir la machine qui minimise la somme des temps de traitement des produits présents sur sa liste d'attente.
Mathématiquement, on doit chercher parmi toutes les machines disponibles, la machine j qui minimise la quantité $\sum_{i \in L_j} T_{p_i j}$. Cette règle permet de réduire les queues d'attente, ce qui diminue le temps de traitement des produits et donc le *Lead Time*.

— Compétences employé : Cette règle se base sur la disponibilité des employés plutôt que sur les produits. En effet, chaque machine est associée à un nombre d'employés capables de l'utiliser. Il faut donc sélectionner la machine ayant le moins d'employés capables de l'utiliser afin qu'elle soit vidée immédiatement plutôt qu'à un moment où les employés disponibles seraient incapables de l'utiliser.
Mathématiquement, on choisit la machine qui minimise $T(j)$ sachant que $B(j) = 0$, c'est-à-dire qu'elle n'est occupé par aucun employé.

— Attente premier produit : Considérons i le premier produit dans la liste d'attente de la machine j. On doit chercher parmi toutes les machines disponibles, la machine j qui minimise la quantité $\sum_{k \in L_{f(i)}} T_{p_k L_{f(i)}}$. Cette règle permet de choisir la machine tel que le premier produit qui sera traité sur cette machine aura le temps d'attente le plus faible à la prochaine machine de son parcours.

— Service premier produit : Considérons i le premier produit dans la liste d'attente de la machine j. On doit chercher parmi toutes les machines disponibles, la machine j qui minimise la quantité $T_{p_i j}$.

— L'état de la liste d'attente des machines : Attribuons un score S_j à chaque machine. $S_j = \sum_{i \in L_j} 1$. Nous choisissons la machine j qui a S_j le plus grand sachant que $B(j) = 0$.

— Les produits affectés à la prochaine étape : Attribuons un score S_j à chaque machine. Pour chaque produit i en service, on rajoute 1 à $S_f(i)$. Nous choisissons la machine j qui a S_j le plus grand sachant que $B(j) = 0$.

— L'état de la liste d'attente des machines à la prochaine étape : Cette règle vise à limiter les queues d'attente en sélectionnant la machine qui aura la plus grande queue à la prochaine étape plutôt que celle ayant la plus grande queue à l'étape actuelle.
Mathématiquement, Attribuons un score S_j à chaque machine. Pour chaque produit i en service, on rajoute 1 à $S_f(i)$. Puis, on rajoute à S_j la quantité $\sum_{i \in L_j} 1$ qui représente le nombre des produits en attente à la machine j. Nous choisissons la machine j qui a le score S_j le plus grand sachant que $B(j) = 0$.

— La machine la plus demandée : Cette règle consiste à sélectionner la machine la plus demandée en choisissant celle qui est à la fin du parcours d'un produit le plus souvent possible. Mathématiquement, nous établissons une fonction h comptabilisant le nombre de fois où une machine se trouve à la fin du parcours d'un produit donné. La machine j choisie sera celle qui maximise $h(j)$ sous la condition $B(j) = 0$.

3.2 Choix du prochain produit à traiter

En ce qui concerne le choix d'un produit à traiter dans la liste d'attente de la machine, on propose les règles d'affectations suivantes :

— First in First Out : Le premier produit entré dans la liste d'attente de la machine est le premier traité.

— Service Time : Choisir le produit qui a le temps de service le plus petit.

— Temps d'attente sur la prochaine machine : Choisir le produit qui aura un temps d'attente sur la prochaine machine dans le parcours le plus faible.

— Service Time + Temps d'Attente : Choisir le produit qui aura la somme du temps de service et d'attente sur la prochaine machine dans le parcours le plus faible.

— Temps d'arrivée du produit : Choisir le produit qui est arrivé le premier dans la chaîne de fabrication.

3.3 Choix de l'employé en attente

Dans ce cas, la règle d'affectation que nous avons décidé de choisir pour notre simulateur consiste à affecter la machine disponible à l'employé qui a le moins de compétences.

3.4 Combinaison de quelques règles d'affectation

Dans cette partie, nous allons essayer de combiner quelques règles parmi celles qu'on a déjà introduit afin de former de nouvelles règles d'affectation qu'on va tester :

— COMPÉTENCES EMPLOYÉS + PRODUITS PROCHAINE ÉTAPE + SERVICE TIME : Cette règle consiste à attribuer un score S_j à chaque machine j sachant que $B(j) = 0$. Soit k le premier produit dans la liste d'attente des machines. On rajoute à S_j la quantité $T(j)+T_{p_k j}$. Pour chaque produit i en service, on rajoute 1 à $S_f(i)$. Nous choisissons la machine j qui a S_j le plus petit.

— TEMPS ATTENTE PROCHAINE MACHINE + SERVICE TIME : Cette règle consiste à attribuer un score S_j à chaque machine j sachant que $B(j) = 0$. Considérons i le premier produit dans la liste d'attente de la machine j. $S_j = \sum_{k \in L_f(i)} T_{p_k L_f(i)} + T_{p_i j}$.

— TEMPS ATTENTE PROCHAINE MACHINE + SERVICE TIME + COMPÉTENCES EMPLOYÉS : Cette règle consiste à attribuer un score S_j à chaque machine j sachant que $B(j) = 0$. Considérons i le premier produit dans la liste d'attente de la machine j. $S_j = \sum_{k \in L_f(i)} T_{p_k L_f(i)} + T_{p_i j} + T(j)$.

4
EXPÉRIMENTATION

4.1 Données et conditions expérimentales

4.1.1 • Hypothèse ergodique

Afin d'évaluer le lead time et pouvoir l'optimiser, nous avons adopté l'hypothèse ergodique. L'hypothèse ergodique suppose que si l'on suit l'évolution d'un système au fil du temps, en observant son comportement à différents moments, alors les propriétés statistiques du système peuvent être déterminées en étudiant une seule trajectoire aléatoire, plutôt que de prendre en compte l'ensemble des trajectoires possibles. Cette hypothèse généralise la loi des grands nombres. On stipule que pour une fonction f(X) arbitraire, la moyenne temporelle de f(X(t)) sur une trajectoire du système (c'est-à-dire, une suite de valeurs successives de X(t) déterminée par les équations du système) est égale à la moyenne spatiale de f(X) sur l'ensemble des états possibles du système. Autrement dit :

$$\lim_{T \to \infty} \frac{1}{T} \int_0^T f(X(t))\, dt = \frac{1}{\mu(X)} \int f(X)\, dX$$

avec μ une mesure sur l'espace des phases.

L'application de l'hypothèse ergodique dans notre cas sera tel que le *Lead Time* soit égale à la moyenne des temps de séjours de plusieurs produits au régime permanent.

$Lead\ Time = \lim_{N \to +\infty} \frac{1}{N} \sum_{i=1}^{N} (temps\ de\ présence\ de\ i)$

4.1.2 • Simulateur

Nous avons construit un simulateur codé en Python qui permet de simuler la dynamique d'une chaîne de fabrication. Dans ce programme, nous avons d'abord fixé les différents paramètres et l'instance à utiliser. Nous avons également défini des fonctions qui représentent les différentes règles d'affectation à utiliser, lesquelles seront appelées dans le programme pour effectuer les différents tests. Afin de représenter les produits, nous avons crée une classe "Produit" qui associe à chaque produit son identifiant, son temps de départ et d'arrivée, son type ainsi que son parcours. En ce qui concerne le système, nous avons défini une classe "Système" qui contient l'échancier représentant la pile des différentes actions à effectuer, la liste des produits déjà traités et les files d'attente et de service de chaque machine. Les évènements qui sont dans l'échancier sont représenté par une classe "Event" et ses sous classes "Event_end", "Event_arrival" ainsi que "Event_end_service" qui désignent respectivement la fin de la simulation, l'arrivée d'un nouveau produit et la fin de service d'une machine. Finalement, notre simulateur permet d'afficher l'évolution dynamique du système mais auusi de calculer le *Lead Time*.

```
Le produit 0 de type 3  arrive à 0.79
Le produit 0 de type 3  sera traite par l'employe 1 à la machine 4 à 0.79
Le produit 1 de type 2  arrive à 1.16
Le produit 1 de type 2  sera traite par l'employe 0 à la machine 2 à 1.16
Le produit 0 repart à 1.23 de la machine 4
Le produit 0 de type 3  sera traite par l'employe 3 à la machine 5 à 1.23
Le produit 1 repart à 1.69 de la machine 2
Le produit 1 de type 2  sera traite par l'employe 2 à la machine 4 à 1.69
Le produit 2 de type 2  arrive à 1.87
Le produit 2 de type 2  sera traite par l'employe 0 à la machine 2 à 1.87
Le produit 0 repart à 1.87 de la machine 5
Le produit 0 de type 3  sera traite par l'employe 3 à la machine 6 à 1.87
Le produit 1 repart à 2.18 de la machine 4
Le produit 2 repart à 2.34 de la machine 2
Le produit 1 de type 2  sera traite par l'employe 0 à la machine 0 à 2.34
Le produit 2 de type 2  sera traite par l'employe 1 à la machine 4 à 2.34
Le produit 3 de type 3  arrive à 2.49
Le produit 3 de type 3  sera en liste d'attente à la machine 4 à 2.49
Le produit 0 repart à 2.71 de la machine 6
```

Figure 1 – Une partie de l'évolution du système

4.1.3 • Instances utilisées

Nous avons testés notre algorithme sur plusieurs instances pour déterminer les règles d'affectations les plus appropriée (qui minimisent le *Lead Time*).

Pour ce faire, nous avons considéré un atelier à 8 machines et 4 types de produits (T_1, T_2, T_3, T_4). Les produits arrivent dans la chaîne de fabrication suivant un processus de Poisson de paramètres différents. Ces paramètres sont représentés dans le tableau suivant :

Types	Paramètre
T_1	0.29
T_2	0.32
T_3	0.47
T_4	0.38

En ce qui concerne le temps de service T_{pm} d'un produit de type p sur une machine m dans son parcours, il sera tiré selon une loi uniforme dans l'intervalle $[t^{pm}_{min}, t^{pm}_{max}]$. Les intervalles sont présentés sur le tableau suivant :

Machines		M1	M2	M3	M4	M5	M6	M7	M8
Types	T1	[0.58, 0.78]	[0.23, 0.56]	[0.81, 0.93]	[0.12, 0.39]				[0.82, 1.04]
	T2		[0.59, 0.68]		[0.74, 0.77]			[0.30, 0.55]	
	T3	[0.57, 0.64]		[0.37, 0.54]		[0.35, 0.63]			
	T4					[0.36, 0.51]	[0.61, 0.70]	[0.78, 0.85]	[0.18, 0.37]

FIGURE 2 – Temps de traitement des différents produits

Les instances I_1 et I_2 disposent de 4 employés, tandis que les instances I_3 et I_4 en ont 6. Les compétences des employés sont représentés par une matrice dont chaque ligne est associé à un employé et chaque colonne représente une machine de l'atelier. Si on note Q la matrice des compétences, alors $Q_{i,j} = 1$ signifie que l'employé i est capable de travailler sur la machine j, tandis que $Q_{i,j} = 0$ signifie que l'employé i n'est pas qualifié pour travailler sur la machine j.

$$Q(\mathrm{I1}) = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \qquad Q(\mathrm{I2}) = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

$$Q(\mathrm{I3}) = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \qquad Q(\mathrm{I4}) = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$$

FIGURE 3 – Matrices de compétences pour les différentes instances

Chaque type de produit a son propre parcours de traitement. Ainsi les produits passent dans l'ordre par :

$$T_1 : M_1, M_2, M_3, M_4, M_8$$

$$T_2 : M_2, M_4, M_7$$

$$T_3 : M_1, M_3, M_5$$

$$T_4 : M_5, M_6, M_7, M_8$$

Nous allons tester plusieurs règles d'affectation sur les 4 instances avec 20 runs et une durée totale de simulation de 1000s. Afin de calculer le *Lead Time* au régime permanent, nous avons estimé que la phase transitoire est atteinte au bout de 200s.

A la fin, nous allons présenter les meilleurs règles d'affectation pour le problème de 3 machines et 2 employés et un seul type de produits. Les produits ont le même parcours et ils passent par la machine 0 puis 1 puis 2. Les employés sont tous qualifiés pour travailler sur tout les machines.

4.2 Résultats

4.2.1 • Instance I_1

Nous représentons ci-dessous les résultats des tests de trois règles d'affectation pertinentes pour l'instance I_1.

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	5.9179	(5.6030, 6.2329)
Service Time	5.0718	(4.8544, 5.2892)
Temps d'attente sur la prochaine machine	5.4703	(5.2163, 5.7243)
Service Time + Temps d'attente sur la prochaine machine	5.1650	(4.9361, 5.3939)
Temps d'arrivée du produit	5.6755	(5.3855, 5.9654)

Table 1 – La règle d'affectation : L'état de la liste d'attente des machines

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	5.7969	(5.4849, 6.1089)
Service Time	5.2097	(4.9756, 5.4438)
Temps d'attente sur la prochaine machine	5.5457	(5.2746, 5.8167)
Service Time + Temps d'attente sur la prochaine machine	5.2505	(5.0051, 5.4959)
Temps d'arrivée du produit	5.7084	(5.4029, 6.0138)

Table 2 – La règle d'affectation : Les produits affectés à la prochaine étape

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	5.9982	(5.6707, 6.3256)
Service Time	5.1841	(4.9656, 5.4027)
Temps d'attente sur la prochaine machine	5.5674	(5.2988, 5.8360)
Service Time + Temps d'attente sur la prochaine machine	5.2342	(5.0084, 5.4599)
Temps d'arrivée du produit	5.7812	(5.4908, 6.0716)

Table 3 – La règle d'affectation : L'état de la liste d'attente des machines à la prochaine étape

Nous nous sommes donc menés à conclure en remarquant que la meilleure stratégie consiste à affecter l'employé à la machine qui a la liste d'attente la plus grande en commençant par taiter le produit qui a le temps de service le plus petit. Nous avons trouvé en appliquant cette règle un *Lead Time* de l'ordre de **5.0718**.

4.2.2 • Instance I_2

Nous avons obtenu un *Lead Time* de **3.9233** avec un intervalle de confiance à 95% de [3.8117, 4.0349] avec la règle d'affectation : L'état de la liste d'attente des machines.

Nous avons obtenu un *Lead Time* de **4.0408** avec un intervalle de confiance à 95% de [3.8920, 4.1896] avec la règle d'affectation : Les produits affectés à la prochaine étape.

Ci-dessous les résultats en appliquant la règle d'affectation : L'état de la liste d'attente des machines à la prochaine étape.

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	3.8470	(3.7356, 3.9585)
Service Time	3.7558	(3.6526, 3.8590)
Temps d'attente sur la prochaine machine	3.8025	(3.7002, 3.9048)
Service Time + Temps d'attente	3.7468	(3.6478, 3.8458)
Temps d'arrivée du produit	3.8305	(3.7214, 3.9395)

Table 4 – La règle d'affectation : L'état de la liste d'attente des machines à la prochaine étape

Nous pouvons ainsi en déduire que la meilleure règle d'affectation pour l'instance I_2 est : **L'état de la liste d'attente des machines à la prochaine étape** avec **ST+Attente**. On trouve donc un *Lead Time* de **3.7468**.

4.2.3 • Instance I3

Nous avons trouvé pour la règle d'affectation **Attente premier produit** un *Lead Time* de **3.7829** avec la règle First In First Out. Tandis que pour la règle **Compétences employé** un *Lead Time* de **3.9024** avec First In First Out. Enfin, pour la règle **Les produits affectés à la prochaine étape** un *Lead Time* de **3.8863** avec First In First Out.

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	3.8470	(3.7356, 3.9585)
Service Time + Temps d'attente	3.6847	(3.5821, 3.7874)

Table 5 – La règle d'affectation : Compétences employés + produits prochaine étape + Service Time

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	3.9058	(3.7808, 4.0308)
Service Time + Temps d'attente	3.6989	(3.5973, 3.8006)

Table 6 – La règle d'affectation : Temps attente prochaine machine + Service Time

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	3.7829	(3.6652, 3.9007)
Service Time + Temps d'attente	3.6505	(3.5519, 3.7491)

Table 7 – La règle d'affectation : Temps attente prochaine machine + Service Time + compétences employés

On conclut donc que la meilleure stratégie pour l'instance I_3 est : Temps attente prochaine machine + Service Time + compétences employés avec Service Time + Temps d'attente sur la prochaine machine. On trouve donc pour ces règles combinées un Lead Time de **3.6505**.

4.2.4 • Instance I_4

Nous avons trouvé les mêmes règles d'affectation pertinentes que pour l'instance I_3. Nous représentons ci-dessous les résultats pour les 3 règles d'affectations :

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	2.7852	(2.7587, 2.8117)
Service Time + Temps d'attente	2.7511	(2.7256, 2.7766)

Table 8 – La règle d'affectation : Compétences employés + produits prochaine étape + Service Time

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	2.7791	(2.7538, 2.8044)
Service Time + Temps d'attente	2.7424	(2.7188, 2.7661)

Table 9 – La règle d'affectation : Temps attente prochaine machine + Service Time

Méthode	lead time	Intervalle de confiance à 95%
Firt In First Out	2.7767	(2.7517, 2.8017)
Service Time + Temps d'attente	2.7415	(2.7166, 2.7665)

Table 10 – La règle d'affectation : Temps attente prochaine machine + Service Time + compétences employés

On remarque que les *Lead Time* sont très proches pour les différentes méthodes. On en déduit que la stratégie qui permet d'avoir un *Lead Time* relativement faible est Temps attente prochaine machine + Service Time + compétences employés avec Service Time + Temps d'attente sur la prochaine machine. On trouve donc un leadtime de **2.7415**.

4.2.5 • Problème de 3 machines et 2 employés

Nous nous plaçons dans le cas où on prend le premier produit dans la liste d'attente (First In First Out) et nous testons différentes règles d'affectation d'un employé à des machines. Nous avons trouvé 3 règles d'affectation pertinentes.

En appliquant la règle d'affectation **Attente Premier Produit**. On trouve un *Lead Time* de **2.1023** avec un intervalle de confiance à 95% de (2.0853, 2.1193).

En appliquant la règle d'affectation **Les produits affectés à la prochaine étape**. On trouve un *Lead Time* de **2.1023** avec un intervalle de confiance à 95% de (2.0853, 2.1193).

En appliquant la règle d'affectation **La machine la plus demandée**. On trouve aussi un *Lead Time* de **2.1023** avec un intervalle de confiance à 95% de (2.0853, 2.1193).

4.3 Commentaires

Les meilleurs règles d'affectation qui donnent un *Lead Time* qui est bon pour la majorité des instances sont les suivantes :

— L'état de la liste d'attente des machines

— L'état de la liste d'attente des machines à la prochaine étape

— Temps attente prochaine machine + Service Time + compétences employés

— La machine la plus demandée

— Attente Premier Produit

— Les produits affectés à la prochaine étape

Pour les produits, la meilleure méthode en générale est Service Time et Temps d'attente sur la prochaine machine.

5 COMPLÉMENT THÉORIQUE SUR LA VERSION OFFLINE DU PROBLÈME

Dans cette section, nous allons essayer de théoriser notre problème et ce en essayant de déterminer la stratégie (règle d'affectation) la plus optimale pour le problème déterministe suivant :

On considère 3 machines (M_1, M_2 et M_3), 2 employés (ω_1 et ω_2) *full skills* c'est à dire qu'ils ont des compétences sur toutes les machines et 3 produits ($p_1, p_2 et p_3$). On suppose que les 3 produits sont du même type et que p_i est le seul produit présent sur la liste d'attente de la machine M_i avec $i \in \{1, 2, 3\}$. De plus, le fait que les produits soient du même type nous permet ainsi de faire l'hypothèse que le temps de service sur chaque machine ne dépend que de la machine en question. Autrement dit, sur la machine M_i, le temps de service du produit p_j ne dépend que de i et on le notera s_i. Ainsi, avec les notations introduites dans la section **Formalisation du problème**, nous avons le problème suivant :

— P=1

— W=2

— M=3

— $\forall i \in \{1,2\} : S_{\omega_i} \subset \{M_1, M_2, M_3\}$
— $\forall i \in \{1,2,3\} : Parcours(p_i) = (M_1, M_2, M_3)$
— $\forall i, j \in \{1,2,3\} : T_{p_i M_j} = s_j$

Pour simplifier, supposons que $s_1 \leqslant s_2 \leqslant s_3$. Étant donné que les deux employés sont compétents à part entière (*full skills*), les stratégies suivantes sont équivalentes : "À $t = 0$, envoyer ω_1 sur M_1 et ω_2 sur M_2" et "À $t = 0$, envoyer ω_2 sur M_1 et ω_1 sur M_2", ainsi que toutes les autres stratégies de ce type sont équivalentes. Cela ne nous laisse que 3 stratégies à étudier afin de les comparer l'une par rapport à l'autre pour pouvoir déterminer la plus optimale :

— Stratégie 1 (S_1) : À $t = 0$, envoyer ω_1 sur M_1 et ω_2 sur M_2
— Stratégie 2 (S_2) : À $t = 0$, envoyer ω_1 sur M_1 et ω_2 sur M_3
— Stratégie 3 (S_3) : À $t = 0$, envoyer ω_1 sur M_2 et ω_2 sur M_3

On associe à chaque stratégie son *Lead Time*, noté $LT(S_i)$ avec $i \in \{1,2,3\}$

Proposition 1. *Le Lead Time minimal vaut* $2s_3$ *et il est atteint pour la stratégie* S_3 *mais aussi pour la stratégie* S_2 *à condition que* $s_1 + s_2 <= s_3$*. La stratégie optimale dans ce cas est donc la startégie* S_3*.*

Démonstration. Pour prouver ce résultat, on doit exprimer le *Lead Time* pour chaque stratégie en fonction des données du problèmes afin de les comparer après.

— **Expression de $LT(S_1)$**
Nous allons dans un premier temps exprimer le temps de séjour pour chaque produit. Pour p_1, on a :

$$\text{Temps de séjour}(p_1) = \text{Temps d'attente}(p_1) + \text{Temps de service}(p_1)$$

Or :

$$\text{Temps de service}(p_1) = s_1 + s_2 + s_3$$

Avec les notations :
— Temps d'attente(p_1) $= T_{p_1}^{att}$
— Temps d'attente de p_1 sur $M_1 = T_{p_1}^{M_1}$
— Temps d'attente de p_1 sur $M_2 = T_{p_1}^{M_2}$
— Temps d'attente de p_1 sur $M_3 = T_{p_1}^{M_3}$

On a :

$$T_{p_1}^{att} = T_{p_1}^{M_1} + T_{p_1}^{M_2} + T_{p_1}^{M_3}$$

Or pour la stratégie S_1 $T_{p_1}^{M_1} = 0$ et $T_{p_1}^{M_2} = s_3 - s_1$. En effet, p_1 arrive sur M_2 à $t = s_1$ et sera servi à $t = s_2$ (L'instant où p_2 quitte M_2).

Enfin :

$$T_{p_1}^{M_3} = s_1 + 2s_3 - 2s_2$$

Car, p_1 arrive sur M_3 à $t = 2s_2$ (L'instant où p_1 quitte M_2) et p_2 quitte M_3 à $t = s_1 + 2s_3$ (Vu que p_3 quitte M_3 à $t = s_1 + s_3$)

D'où :

$$\text{Temps de séjour}(p_1) = s_1 + 3s_3$$

Avec ces mêmes notations, on peut calculer le temps de séjour du produit p_2 pour cette première stratégie

on a :

$$\text{Temps de service}(p_2) = s_2 + s_3$$

et :

$$T_{p_2}^{att} = T_{p_2}^{M_2} + T_{p_2}^{M_3}$$

Or $T_{p_2}^{M_2} = 0$ et $T_{p_2}^{M_3} = s_1 + s_3 - s_2$. En effet, p_2 arrive en M_3 à $t = s_2$ et sera servi à $t = s_1 + s_3$

Ainsi :

$$T_{p_2}^{att} = s_1 + s_3 - s_2$$

Donc :

$$\text{Temps de séjour}(p_2) = s_1 + 2s_3$$

Finalement on exprime le temps de séjour du produit p_3

On a :

$$\text{Temps de service}(p_3) = s_3$$

Or $T_{p_3}^{att} = s_1$. En effet, p_3 attend la fin de service du produit p_1 sur la machine M_1 pour que l'employé ω_1 passe en M_3.

Ainsi :

$$\text{Temps de séjour}(p_3) = s_1 + s_3$$

Enfin :

$$LT(S_1) = s_1 + 2s_3$$

— **Expression de $LT(S_2)$**
Pour le produit p_1, on a toujours :

$$\text{Temps de service}(p_1) = s_1 + s_2 + s_3$$

avec $T_{p_1}^{M_1} = 0$ et $T_{p_1}^{M_2} = s_2$. En effet, ω_1 traite p_1 sur M_1 et passe en M_2 pour traiter p_2, ainsi p_1 attend à ce que p_2 soit servi sur M_2 par ω_1.

Ensuite :

$$T_{p_1}^{M_3} = (s_1 + s_2 + (s_3 - (s_1 + s_2))^+) + s_3 - (s_1 + 2s_2)$$

La raison de cette dernière égalité étant que p_2 part de M_3 à $t = s_1 + s_2 + (s_3 - (s_1 + s_2))^+) + s_3$ et p_1 arrive sur cette même machine à $t = s_1 + 2s_2$ avec $a^+ = a$ si $a \geqslant 0$ et 0 sinon D'où :

$$\text{Temps de séjour}(p_1) = s_1 + s_2 + 2s_3 + (s_3 - (s_1 + s_2))^+$$

Pour p_2 maintenant, on a :

$$\text{Temps de service}(p_2) = s_2 + s_3$$

avec $T_{p_2}^{M_2} = s_1$ et $T_{p_2}^{M_3} = (s_3 - (s_1 + s_2))^+$. En effet, p_2 passe en M_3 à $t = s_1 + s_2$ et p_3 quitte M_3 à $t = s_3$.

Donc :

$$\text{Temps de séjour}(p_2) = s_1 + s_2 + s_3 + (s_3 - (s_1 + s_2))^+$$

Enfin pour le produit p_3 :

$$\text{Temps de séjour}(p_3) = s_3$$

Finalement :

$$LT(S_2) = \frac{4}{3}s_3 + \frac{2}{3}(s_1 + s_2 + (s_3 - (s_1 + s_2))^+)$$

— **Expression de** $LT(S_3)$
Pour p_1, on a encore une fois :

$$\text{Temps de service}(p_1) = s_1 + s_2 + s_3$$

avec $T_{p_1}^{M_1} = s_2$ et $T_{p_1}^{M_2} = 0$. En effet, p_1 attend à ce que le produit p_2 part de M_2 vers M_3 à $t = s_2$ et ce sera donc l'employé ω_1 qui va le servir sur M_1 mais aussi sur M_2.

Ensuite :

$$T_{p_1}^{M_3} = 2s_3 - (s_1 + 2S_2)$$

Car p_1 arrive à la machine M_3 à $t = s_1 + 2s_2$ et p_2 la quitte à $t = 2s_3$ Ainsi :

$$\text{Temps de séjour}(p_1) = 3s_3$$

En ce qui concerne p_2 : On a $T_{p_2}^{M_2} = 0$ et $T_{p_2}^{M_3} = s_3 - s_2$ D'où :

$$\text{Temps de séjour}(p_2) = 2s_3$$

Finalement, pour p_3 : On a

$$\text{Temps de séjour}(p_3) = s_3$$

Par conséquent :

$$LT(S_3) = 2s_3$$

Nous pouvons ainsi conclure le calcul précedent avec les expressions suivantes pour les *Lead Time* des 3 stratégies S_1, S_2 et S_3 :

— $LT(S_1) = s_1 + 2s_3$
— $LT(S_2) = \frac{4}{3}s_3 + \frac{2}{3}(s_1 + s_2 + (s_3 - (s_1 + s_2))^+)$
— $LT(S_3) = 2s_3$

On constate clairement que la stratégie S_3 est bien meilleure que la stratégie S_1. Par contre, la stratégie S_2 est un peu spéciale, car il faut distinguer les cas selon la valeur de $s_1 + s_2$ pour pouvoir estimer à quel point elle est efficace. En effet, si $s_1 + s_2 \leqslant s_3$ alors $LT(S_2) = 2s_3 = LT(S_3)$ et dans ce cas on peut faire un choix arbitraire entre S_2 et S_3, puisqu'elles sont toutes les deux meilleures que

S_1. Sinon, nous aurons l'inéglité stricte $LT(S_2) > 2s_3 = LT(S_3)$ ce qui nous permettra de dire que le *Lead Time* minimal vaut $2s_3$ et il est atteint pour la stratégie S_3 mais aussi pour la stratégie S_2 sous la condition $s_1 + s_2 \leqslant s_3$. Ces résultats permettent de justifier le bon fonctionnement de certaines règles d'affectations introduites dans ce rapport, notamment celle pour laquelle on choisit d'affecter l'employé libéré à la machine qui a la liste d'attente la plus grande tout en servant en premier le produit qui a le temps de service le plus grand (ce qui peut être interprété par le fait que $s_3 \geqslant s_2 \geqslant s_1$). $\square$

CONCLUSION

En optimisant le lead time, les entreprises peuvent améliorer l'efficacité de leur chaîne de production, réduire les coûts et augmenter la satisfaction des clients. Cette étude montre que différentes méthodes peuvent être utilisées pour minimiser le temps de traitement d'un produit dans une chaîne de fabrication, mais que certaines sont à moins efficace dans le cadre particulier de 3 machines et 2 employés. Cela peut être utile pour les entreprises de fabrication qui cherchent à améliorer leur compétitivité et leur rentabilité dans un marché en constante évolution.