

---

# Apprentissage automatique pour l'ordonnancement dynamique d'un atelier de production hybride basé sur une métaheuristique

Nasser Mebarki<sup>\*1</sup>, Wassim Bouazza<sup>\*2</sup>, Abdelhakim Ghiles Hamiti<sup>\*2</sup>, Arnaud Laurent<sup>\*2</sup>,  
and Mohamed Kenani<sup>3</sup>

<sup>1</sup>Laboratoire des Sciences du Numérique de Nantes (LS2N) – Université de Nantes : UMR6004 – IUT de Nantes - Département QLIO 2 avenue du Professeur Jean Rouxel 44475 Carquefou Cedex, France

<sup>2</sup>Laboratoire des Sciences du Numérique de Nantes – Université de Nantes – France

<sup>3</sup>Laboratoire des Sciences du Numérique de Nantes – Université de Nantes – France

## Résumé

Optimiser les stratégies de planification rencontre des défis notables dans les environnements de production dynamiques, en particulier dans les systèmes de fabrication flexibles tels que les flowshops hybrides (HFS). Cette étude se concentre sur l'allocation des ressources dans un contexte dynamique. Notre approche intègre l'Algorithme Génétique (AG) et l'apprentissage automatique (ou ML pour Machine Learning) pour générer des données et entraîner des modèles. À travers l'expérimentation de divers scénarios, la performance des modèles de ML est évaluée pour l'allocation des ressources. De plus, des travaux futurs visent à évaluer l'impact sur le temps de parcours moyen ou Mean Flowtime, afin de mieux éprouver l'efficacité de la méthode proposée. Les résultats obtenus mettent en lumière le potentiel des techniques de ML pour améliorer la prise de décision dans les systèmes de fabrication dynamiques et flexibles, contribuant ainsi aux efforts visant à améliorer les stratégies de planification réactive.

Le problème complexe d'ordonnancement d'atelier de type flowshop implique la gestion simultanée de ressources et de tâches tout en prenant en considération un certain nombre de contraintes. On désigne par job une séquence d'opérations liée par des contraintes de précedence exécutée par une ressource donnée. L'objectif est de trouver le mean flow time minimale, souvent en tenant compte des dates de disponibilité et d'échéance. Les méthodes traditionnelles telles que les règles de répartition, en anglais Dispatching Rules (DR) sont parfois complétées par des approches intégrant du ML, en particulier le renforcement profond (DRL), pour une adaptation dynamique du système de pilotage à l'évolution du contexte. Les avancées récentes explorent des techniques telles que les hyper-heuristiques, les algorithmes génétique, et les stratégies de fusion. Une approche novatrice utilise l'algorithme génétique et la simulation pour générer des données afin d'entraîner des modèles de ML, réduisant ainsi la dépendance à la disponibilité des données en quantité importante. Cette évolution marque un progrès dans la recherche sur l'ordonnancement dans des environnements dynamiques (Ochoa and Özcan, 2010, Bouazza et al., 2021, Braune et al., 2022, Liu et al., 2023, Zhao et al., 2022).

L'approche se divise en trois phases : la génération de données via un algorithme génétique,

---

\*Intervenant

le pré-traitement de ces données pour ne conserver que l'information pertinente, et enfin, l'entraînement de modèles ML. Ceux-ci incluent les algorithmes de Random Forests, Bagging Classifier, Gradient Boosting Classifier, Dense Neural Network dont une variante avec Optimisation Bayésienne.

Phase de Génération de Données : Un algorithme génétique est utilisé "hors ligne" comme un oracle, pour générer des données d'entraînement. Il prend en entrée des scénarios d'arrivées de produits et retourne les meilleures allocations possibles en connaissant à l'avance les dates d'arrivée de chaque produit (qui ne sont pas connues dans le contexte d'un ordonnancement dynamique). Pour chaque choix d'allocation de l'AG on extrait les informations des ressources et cellules du flowshop, on obtient ainsi les données d'apprentissage que l'on va tenter de prédire.

Phase de prétraitement des Données : Les données sont pré-traitées en sélectionnant des co-variables pertinentes pour l'apprentissage. Les variables conservées incluent les temps d'arrivée, les familles de produits, les charges actuelles des ressources, et d'autres paramètres spécifiques à l'attribution.

Phase d'entraînement des Modèles ML : Différents modèles ML sont entraînés pour prédire les décisions d'allocation. Les performances des modèles sont évaluées sur trois étapes du flowshop, avec des résultats présentés en termes d'exactitude.

Instances de Scénarios :

L'AG a été exécuté sur 200 scénarios partageant toutes les statistiques suivantes :

Le nombre de produits entre 90 et 110 suivant une distribution uniforme.

Le temps d'arrivée des produits entre la période 1 et 600 suivant une distribution uniforme.

Distribution uniforme des familles de produits allant de 1 à 5.

Résultats et discussion

Chaque étape de la production ou cellule du flowshop est associée à un module dédié pour les décisions d'allocation de ressources, pour un total de trois modèles différents pour nos instances.

Les résultats présentent des variations significatives entre les différentes étapes du flowshop. À l'étape 1, caractérisée par deux ressources, les modèles atteignent des degrés d'exactitude satisfaisants, avec le Dense NN + Bayesian Optimization montrant une performance notablement élevée. Cependant, avec l'augmentation de ressources à l'étape 2, les exactitudes diminuent légèrement, mettant en évidence une difficulté accrue due à un environnement plus complexe. L'étape 3, caractérisée par deux ressources, révèle des performances exceptionnelles pour tous les modèles, avec des exactitudes dépassant 95