Contributions à l'Ordonnancement des Interventions en Chirurgie Ambulatoire : Q-learning et Flow-Shop Hybride

L. Bouchlaghem[1], A. Guessoum[2], F. Ghedjati[1]

[1] Université de Reims Champagne-Ardenne, Reims, France
[2] DataImpact by NielsenIQ, Paris, France

Résumé

Les avancées récentes en intelligence artificielle (IA), notamment en matière d'apprentissage par renforcement, ont suscité l'intérêt croissant des chercheurs occupés par des problèmes d'ordonnancement et d'optimisation. Cette étude explore l'applicabilité de ces techniques de résolution au problème bi-objectif d'ordonnancement des interventions en chirurgie ambulatoire. Les objectifs sont à la fois la minimisation de la durée totale d'hospitalisation des patients (makespan) et de leur temps d'attente moyen, simultanément. Plus précisément, en comparant notre approche à des méthodes de résolution établies, nous évaluons l'efficacité de l'algorithme Q-learning pour la résolution du problème considéré. Ce problème est modélisé comme un problème de type Flow Shop hybride à trois étapes. Nous commençons par décrire l'approche de l'algorithme Q-learning que nous utilisons, puis nous présentons les résultats qui démontrent que notre approche Q-learning fournit de meilleurs résultats que ceux obtenus avec un algorithme génétique (GA) bi-objectif, une recherche de voisinage variable (VNS) bi-objectif et un algorithme de tri non-dominé (NSGA-II). Notre approche permet d'améliorer à la fois le makespan et le temps d'attente moyen. Ces résultats encouragent à poursuivre l'exploration des techniques d'apprentissage par renforcement dans nos recherches.

Mots-clés

Q-learning, Chirurgies Ambulatoires, Flow Shop Hybride, Ordonnancement, Métaheuristiques.

Abstract

Recent advancements in artificial intelligence (AI) have brought machine learning and reinforcement learning techniques to the attention of researchers interested in scheduling and optimisation problems. This study explores the applicability of such resolution techniques to the BI-objective outpatient surgery scheduling problem, considering both the reduction of the total hospitalization durations for patients (makespan) and the minimization of their waiting time simultaneously (waiting time). Specifically, through a comparative approach with established resolution methods, we assess the effectiveness of the Q-learning algorithm in addressing our outpatient surgery scheduling problem, formulated as a three-stage Hybrid Flow Shop problem. We, first, describe the Q-learning algorithm approach used; and then, present results that show that our Q-learning approach surpasses the results obtained with a BI-objective Genetic Algorithm (GA), a BI-objective Variable Neighbourhood Search (VNS) and a non-dominated sorting algorithm (NSGA-II). Our approach improves both the makespan and mean waiting time. The results encourage further exploration of reinforcement learning techniques in our research.

Keywords

Q-learning, Outpatient Surgery, Hybrid Flow Shop, Scheduling, Metaheuristics.

1 Introduction

La part grandissante de l'ambulatoire dans le secteur du soin et de la santé s'explique principalement par les réponses qu'elle offre aux défis contemporains auxquels ce dernier fait face (budgétaire, technologique, politique). La transition vers ce mode de prise en charge témoigne d'une tendance de fond centrée sur son efficacité autant que sur le bien-être des patients. En effet, la mise en œuvre réussie de la prise en charge ambulatoire dépend principalement de l'efficacité des pratiques de planification afin d'assurer la circulation fluide des patients ainsi que de l'utilisation optimale des ressources matérielles et humaines disponibles. Notre article traite principalement de l'ordonnancement des prises en charge des patients en chirurgie ambulatoire. Les défis spécifiques liés à ce mode de prise en charge pour la chirurgie sont par exemple l'optimisation du taux d'utilisation des salles d'opération, la réduction des temps d'attente des patients et la coordination efficace des différentes ressources, en particulier étant donné les différences de procédures entre spécialités chirurgicales.

L'ordonnancement et la planification des blocs opératoire en chirurgie ambulatoire ont fait l'objet de nombreuses études dans la littérature [1], ce qui témoigne de l'importance de ces questions. Plusieurs travaux ont examiné les complexités liées aux exigences des différentes spécialités chirurgicales, à la gestion des flux de patients et à la réduction des temps d'attente, tout en veillant à la sécurité et à la satisfaction des patients [2]. De nombreux modèles de planification proposés dans la littérature intègrent des facteurs tels que la disponibilité du personnel médical [3], les contraintes liées à la disponibilité des équipements et à l'organisation de l'établissement et les préférences des patients. Différentes techniques de résolution ont été utilisées, telles que l'optimisation mathématique [4], les méthodes de simulation [5] ou les méthodes heuristiques/métaheuristiques [6], afin de fournir des plannings répondant à plusieurs objectifs simultanément.

D'autre part, les méthodes d'apprentissage par renforcement (RL) se sont avérées prometteuses pour résoudre des problèmes d'optimisation complexes [7]. Dans cet article, nous examinons l'efficacité de cette approche pour résoudre le problème

d'ordonnancement des interventions en chirurgie ambulatoire. Dans le cas ambulatoire, le parcours des patients est organisé en trois étapes distinctes préopératoire, intra-opératoire et postopératoire. Notre objectif est de minimiser simultanément le temps d'attente des patients et la durée d'hospitalisation. Cette organisation du parcours des patients correspond à ceıle de l'Unité de Chirurgie Ambulatoire (UCA) du CHU de Reims. Cette structure pluridisciplinaire dispose de 4 salles d'intervention utilisées par divers services médicaux, ainsi que d'une salle de réanimation partagée avec la chirurgie conventionnelle, offrant une capacité de 16 lits. Elle propose des soins ambulatoires pour des actes chirurgicaux sous diverses formes d'anesthésies, avec une surveillance postopératoire permettant une sortie le jour même. Cette organisation en trois phases nous a conduit à appréhender le problème, comme un Flow Shop Hybride (FSH) à trois étages, classé NP-difficile [8]. Dans cette étude, l'algorithme d'apprentissage testé est le Q-learning qui dispose de propriétés intéressantes : 1) une certaine efficacité pour ce type de problèmes, selon des études similaires ; 2) un fonctionnement simple, bien connu et facilement transposable, en principe comme dans son implémentation, par rapport aux autres méthodes de résolution disponibles dans la littérature et auquel on cherche à le comparer. L'utilisation de la méthode d'apprentissage par renforcement choisi implique la formulation du problème d'ordonnancement sous la forme d'un processus de décision markovien (MDP) pour en justifier l'usage.

L'article est organisé comme suit : la section 2 passe en revue la littérature existante sur les problèmes d'apprentissage par renforcement et d'ordonnancement. Dans la section 3, nous définissons le problème considéré. La section 4 est dédiée à notre approche basée sur le Q-learning et la section 5 présente nos expérimentations et nos résultats. Enfin, nous concluons dans la section 6.

2 Travaux connexes

L'apprentissage par renforcement est un processus d'apprentissage automatique dans lequel un agent apprend à effectuer des actions à partir d'expériences, dans le but d'optimiser une récompense quantitative au fil du temps. L'agent est placé dans un environnement et prend des décisions en fonction de son état actuel. En retour, l'environnement fournit une récompense, qui peut être positive ou négative. Au fur et à mesure des expériences, l'agent cherche à trouver un comportement décisionnel (« policy ») optimal, c'est-à-dire qu'il maximise la somme des récompenses.

Au cours des dernières années, l'apprentissage par renforcement a attiré l'attention en raison de son succès dans la résolution de problèmes complexes de prise de décisions séquentielles dans différents domaines tels que la robotique, les jeux, la finance ou encore l'organisation et la planification logistique des soins pour le domaine de la santé. L'un des principaux avantages de l'apprentissage par renforcement réside dans sa capacité à traiter efficacement les environnements dynamiques et incertains [9]. Dans le domaine de la planification chirurgicale, l'étude de Lee et Lee [10] porte sur la tâche complexe de l'ordonnancement des patients dans un service d'urgence hospitalier. Cette étude propose l'utilisation de l'apprentissage par renforcement profond pour la planification des patients en situation d'urgence. Les chercheurs introduisent un modèle mathématique et un MDP, puis ils développent un réseau neuronal profond Q-networks (DQN) pour établir une politique de planification optimale. Les résultats de l'évaluation de l'approche proposée montrent sa supériorité par rapport aux règles de répartition en termes de temps d'attente et de pénalités pour les patients en situation d'urgence dans les scénarios suggérés.

Dans une autre étude menée par Ribino et al. [11], la planification quotidienne des chirurgies est formulée comme un jeu de Markov coopératif. Les chercheurs introduisent un algorithme Q-learning multi-agents, où chaque agent est responsable de la gestion d'une chirurgie distincte. En favorisant l'apprentissage collaboratif entre ces agents, l'objectif principal était d'atteindre un objectif collectif, à savoir obtenir un planning optimal pour les chirurgies quotidiennes. L'intégration de l'apprentissage collaboratif met en évidence les avantages de la coopération dans le processus d'apprentissage. Les premiers résultats indiquent des améliorations dans la planification des chirurgies, surpassant les méthodes traditionnelles (programmation linéaire mixte en nombres entiers par exemple). Ces deux exemples de travaux récents impliquant des techniques avancées d'apprentissage par renforcement, basées sur le Q-learning, suggèrent que ces approches s'adaptent efficacement aux problèmes d'ordonnancement. Elles peuvent entraîner des gains d'efficacité significatifs dans les processus de prise de décisions en environnements complexes et / ou dynamiques (industrie manufacturière, hôpitaux ou les usines d'assemblage).

Cependant, nous avons pu constater, d'après la revue de littérature réalisée jusqu'à présent, un certain manque de travaux comparant directement l'efficacité des approches RL par rapport à l'existant (heuristiques, métaheuristiques, …) dans la résolution des problèmes d'ordonnancement en général, et de type Flow Shop Hybride (FSH) en particulier. Une étude menée par Han et al. [12] aborde cette question dans le cadre spécifique de l'ordonnancement des sorties des avions embarqués, modélisé comme un FSH. Les résultats indiquent que l'approche Q-learning surpasse à la fois un algorithme génétique simple et un algorithme de système immunitaire artificiel (AIS) en termes de qualité de solution et de temps d'exécution. Notre article contribue dans cette même direction en comparant l'approche Q-learning à des algorithmes plus avancés qui ont démontré leur efficacité dans la résolution des problèmes d'ordonnancement de type FSH. Dans la section suivante, nous présenterons l'approche basée sur le Q-learning que nous proposons pour résoudre le problème d'ordonnancement des interventions en chirurgie ambulatoire.

3 Définition du problème

Au sein d'un centre de chirurgie ambulatoire, les soins suivent un processus spécifique qui se décompose en trois phases distinctes : la phase préopératoire ou de préparation qui débute dès l'arrivée du patient au centre ambulatoire et se termine lors de son transfert vers la salle d'opération ; la phase intra-opératoire ou d'intervention qui englobe à la fois l'anesthésie et l'acte chirurgical lui-même et finalement la phase post-opératoire ou de récupération qui couvre la période qui s'étend depuis l'entrée du patient dans la salle de récupération jusqu'à la fin des soins de suivi dispensés par le chirurgien.

Les patients arrivent à l'heure prévue, et leur nombre est préalablement fixé en fonction des ressources disponibles. Ils

sont répartis par spécialité chirurgicale, avec des durées de service variables à chaque étape du processus. Les phases suivent une politique de "premier arrivé, premier servi", à l'exception de la première étape qui suit un planning préétabli pour l'admission des patients. Cette organisation des soins définie en 3 étapes nous a permis de modéliser le problème en Flow Shop Hybride (FSH) à 3 étages où chacun des étages correspond à une des phases décrites précédemment. Typiquement, un FSH implique des étages de production notées $S = \{1,2, \dots, s\}$. Chaque étage $k \in S$ dispose d'au moins $m_k \geq 1$ machines identiques parallèles, avec au moins un étage ayant plus de deux machines($m_k \geq 2$). Un ensemble de n « jobs » doit être traité séquentiellement dans le même ordre, en passant de l'étage 1 à l'étage s. Chaque « job » $j \in J$ se compose de *O* opérations distinctes : $\{O_{1,j}, O_{2,j}, \dots, O_{s,j}\}$. Chaque opération $O_{k,j}$ correspond au à la réalisation du « job » j à l'étage k avec un temps de traitement déterministe et ininterrompu, noté $p_{k,j}$. Ainsi, dans notre problème le premier étage est dénoté S_{prep} et est composé de m lits de préparation identiques, le second S_{oper} englobe m salles opératoires organisées a priori par spécialité chirurgicale, et le dernier S_{recup} composé de m lits de récupération identiques. Chaque patient j est caractérisé par : une durée de préparation $(PT_{1,j})$ sur l'étage 1, une durée d'intervention $(PT_{2,j})$ sur l'étage 2 et une durée de récupération $(PT_{3,j})$ sur l'étage 3, ainsi que sa spécialité chirurgicale (SS_i). Un patient j peut être affecté à n'importe quel lit $i \in 1,2, \dots, m_k$aux étages 1 et 3. Cependant, à l'étage 2, il ne peut être affecté qu'à des salles d'opération correspondant à sa spécialité chirurgicale. Tous les patients $J_j(o\grave{u} j = \{1,2, \dots, n\})$ doivent séquentiellement passer par les trois étages en commençant par l'étage 1, puis en passant à l'étage 2 et en terminant à l'étage 3. Un patient ne peut occuper qu'une seule ressource à la fois. Aucune ressource ne peut être attribuée à plusieurs patients simultanément et la préemption n'est pas autorisée. Les patients peuvent attendre entre les différentes étapes, ce qui peut entraîner un blocage. Un blocage signifie qu'une ressource reste occupée par un patient malgré la fin de l'étape actuelle car les ressources de la prochaine étape ne sont pas disponibles. Les temps de préparation des ressources et les temps de déplacement des patients sont inclus dans la durée de chaque étape. L'objectif principal est de réduire le makespan qui représente la durée maximale d'hospitalisation (Eq. 1) et le temps d'attente moyen des patients (Eq. 2).

$$minC_{max} = minmax\left(C_{j,l}\right) j = 1 \grave{a} n, l = 3 (1)$$

$$minmean(Wt) = \frac{1}{n}\sum_{l=1}^{3}\sum_{j=1}^{n} wt_{j,l} (2)$$

où : $C_{j,l}$représente le temps de fin traitement du patient j à l'étage l, et $Wt_{j,l}$ indique le temps d'attente du patient j à l'étage l, n est le nombre de patients.

4 Approche proposée

Un processus de décision de Markov (MDP) peut être utilisé pour modéliser des problèmes de prise de décision dans lesquels se succèdent des états distincts vers lesquels on transitionne de manière probabiliste et en tant que la probabilité de transition vers tel ou tel état futur ne dépend conditionnellement que de l'état actuel, et non pas des états passés (Propriété de Markov). L'état actuel contient ainsi toute l'information nécessaire pour que « l'agent apprenant » (dans notre contexte) prenne des décisions de manière autonome (absence de mémoire). Dans un Flow Shop Hybride, cette propriété est vérifiée, car la transition vers une position future (machine à l'étage suivant) d'un « job » dépend uniquement de sa position actuelle (étage et machine actuels), sans nécessité de tenir compte de l'historique des positions antérieures pour décider la transition.

Un MDP est défini par le tuple noté (S, A, P, R), S désigne l'ensemble de tous les états où l'agent peut se trouver, A est l'ensemble des actions que l'agent peut entreprendre, $P(s' \vee s, a)$ représente la probabilité de transition d'un état s à un état s', tandis que $R(s, a, s')$ attribue une récompense à chaque combinaison état-action.

Dans ce contexte théorique, le Q-Learning est l'un des algorithmes les plus utilisés en apprentissage par renforcement et repose sur les concepts clés définis précédemment : les états (s) et les actions (a) représentent les différentes situations disponibles ; les récompenses (r) fournissent un feedback ; les valeurs (Q(s, a)) anticipent les récompenses futures et guident la sélection des actions ; la politique π mappe les paires état-action en fonction des probabilités d'action ; des paramètres tels que le taux d'apprentissage (α) et le facteur d'actualisation (γ) influencent la dynamique d'apprentissage. L'algorithme Q-learning apprend itérativement une politique optimale en commençant par une stratégie aléatoire et en initialisant les valeurs Q à zéro. La sélection des actions est basée sur une stratégie ε-greedy qui vise à équilibrer l'exploration et l'exploitation. Les récompenses obtenues sont utilisées pour mettre à jour les valeurs Q en utilisant l'équation de Bellman (Eq.3), améliorant ainsi les capacités de prise de décision de l'agent.

$$Q[s, a] \coloneqq (1 - \alpha)Q[s, a] + \alpha(r + \gamma max_{a'} Q[s', a']) (3)$$

4.1 Représentation de l'état et de l'action

Un état dans notre problème est représenté par le tuple (patient, étage), qui décrit la position du patient dans le centre de chirurgie ambulatoire à l'instant "t". Pour représenter le début (l'arrivée des patients), un étage (étage 0) fictif est ajouté. Le passage d'un étage à un autre correspond au changement de l'état. Ce passage nécessite l'exécution d'une action. À chaque état est attribué un temps d'arrivée et un temps de sortie.

Une action consiste à affecter une ressource à un patient à chaque étage. Par exemple, pour permettre au patient de passer de l'étage 1 à l'étage 2, une salle d'opération est affectée. Ainsi, sur chacun des étages, le nombre d'actions à entreprendre correspond au nombre de ressources disponibles sur cet étage.

4.2 Représentation de la récompense composite

L'objectif d'utiliser une récompense composite est de créer un modèle de récompense global en agrégeant des récompenses provenant de diverses sources ou critères. Cette approche vise à offrir à l'agent une rétroaction plus complète et équilibrée, dans le but de le former de manière plus efficace.

Quand l'agent prend une action a l'instant t, la récompense composite (R_{t+1}) est une somme pondérée formulée de la manière suivante :

$$R_{t+1} = \sum_{i=1}^{N} w_i \cdot R_{t+1}^{i}, \sum_{i=1}^{N} w_i = 1 (4)$$

Où N est le nombre de fonctions objectifs considérées.

Dans notre modèle cette fonction composite prend en compte à

la fois la minimisation de la durée d'hospitalisation et le temps d'attente moyen des patients, en attribuant des poids appropriés à chaque objectif en fonction de leur importance relative. Étant donné qu'il s'agit d'un problème de minimisation, la fonction de récompense est inversement corrélée avec les objectifs considérés. La récompense cumulative est donc calculée de la manière suivante :

$$R^{cumul} = w_1 \cdot R^{cmax} + w_2 R^{wt} (5)$$

Où w_1 et w_2 sont deux poids reflétant l'importance attribuée aux objectifs.

4.3 L'exploration et l'exploitation

En apprentissage par renforcement, la balance entre exploitation (des meilleures récompenses obtenues) et exploration (de meilleures récompenses pour une nouvelle action) est essentielle pour qu'un agent apprenne à interagir efficacement avec son environnement. Plusieurs méthodes sont utilisées à cette fin tels que : l'approche dite « epsilon-greedy » est communément utilisée pour déterminer l'arbitrage entre exploration et exploitation dans ce contexte. L'agent sélectionne l'action optimale (exploitation) avec une probabilité élevée $(1-\varepsilon)$ où $\varepsilon \in [0,1]$ et explore en sélectionnant une action par hasard parmi les actions disponibles avec une probabilité ε. Cependant, avec cette approche, on comprend que l'agent continue d'explorer avec la même probabilité dans les premiers et derniers temps de l'apprentissage. Il est intuitivement plus pertinent d'explorer les options disponibles plus fortement au début de l'apprentissage puis d'exploiter les meilleures options après un certain temps, dans notre contexte

Ainsi, dans ce travail, nous avons adopté la méthode "Epsilon Decay" pour équilibrer l'exploration et l'exploitation au fil du temps. Cette approche consiste à réduire progressivement la valeur d'epsilon au cours des épisodes. Initialement, cela permet à l'agent d'explorer davantage l'environnement. Cependant, à mesure que l'agent acquiert plus d'informations, il se concentre de plus en plus sur l'exploitation des connaissances acquises. Pour notre étude, nous avons choisi une décroissance linéaire (Eq.6), une approche simple qui réduit progressivement epsilon d'une valeur initiale à une valeur finale sur un nombre spécifié d'étapes ou d'épisodes.

$$\varepsilon = \varepsilon_{initial} - \frac{step}{decay_{steps}} \times (\varepsilon_{final} - \varepsilon_{initial}) (6)$$

Où $\varepsilon_{initial}$ est la valeur initiale d'epsilon, ε_{final} est la valeur finale d'epsilon, *step* est l'étape ou l'épisode en cours, *decay_steps* est le nombre d'étapes ou d'épisodes au cours desquels epsilon diminue de sa valeur initiale à sa valeur finale.

4.4 Q-learning pour l'ordonnancement des interventions en chirurgies ambulatoire

L'algorithme que nous présentons (algorithme 1) détaille notre approche Q-Learning pour résoudre le problème d'ordonnancement des interventions en chirurgies ambulatoires. Le fonctionnement de l'algorithme est organisé en épisodes. Il commence par initialiser les valeurs Q de manière aléatoire pour toutes les paires d'état-action. En début de chaque épisode, les états ainsi que les divers paramètres pertinents pour le parcours du patient sont initialisés. Par la suite, il procède à la sélection d'une action pour chaque patient, consistant à affecter une ressource en fonction des ressources disponibles à l'étage suivant. Le choix de l'action est basé sur une stratégie d'exploration. Après l'exécution de chaque action, la récompense correspondante est observée et utilisée pour mettre à jour la valeur Q de la paire état-action actuelle en utilisant l'équation de Bellman. Ce processus se répète jusqu'à ce que tous les patients soient transférés vers l'état suivant. Une fois le transfert de tous les patients effectué, les patients sont triés selon l'ordre décroissant de leur temps de sortie. Ce processus se répète pour chaque état jusqu'à ce que l'état final soit atteint. L'objectif de l'algorithme est d'apprendre une politique de planification optimale en mettant à jour de manière itérative les valeurs Q au cours des interactions entre l'agent et l'environnement à travers les épisodes.

Algorithme 1 : Q-learning pour l'ordonnancement des interventions en chirurgie ambulatoire

Entrée : facteur de remise γ, facteur d'apprentissage α, probabilité d'exploration initiale ε
Début
  Initialiser Q arbitrairement //($Q(s,a) = 0$; $\forall s \in S$; $\forall a \in A$)
  Pour chaque épisode **faire** :
    Initialiser s // *(état initial)* ;
    Pour chaque état *s* **faire** :
      Pour chaque patient P_i **faire** :
        Choisir une action $a \in A(s)$ étant donné ε ;
        Effectuer l'action a
        Observer le nouvel état s' et recevoir la récompense r
        Mettre à jour Q (Eq.3)
      Fin pour // *tous les patient sont transféré à l'état s'*
      Trier les patients par ordre croissant de leur temps de sortie.
    Fin pour// *s' est un état final*
  Fin pour // *dernier épisode*
Fin

5 Résultats et comparaisons

Dans cette section, nous présentons les résultats et leur comparaison avec ceux donnés par d'autres approches utilisées pour résoudre le problème d'ordonnancement en chirurgie ambulatoire considéré. Nous avons utilisé des données simulées en créant 100 instances avec un nombre variable de patients. Nous avons considéré différents types de durée à chaque étage. Nous avons également inclus six spécialités chirurgicales, avec trois salles d'opération attribuées à chaque spécialité. Aux stades préopératoire et postopératoire, nous avons considéré 20 lits. Les durées à chaque étape ont été générées aléatoirement selon une distribution normale, comme le montre le tableau 1.

étage	Préopératoire	Intra-opératoire					Post-opératoire		
	Petite	Petite	Moyenne	Grande	Extra-grande	Spéciale	Petite	Moyenne	Grande
durée (minutes)	μ = 8	μ = 33	μ = 86	μ = 153	μ = 213	μ = 316	μ = 59	μ = 183	μ = 210
	σ = 2	σ = 15	σ = 15	σ = 17	σ = 17	σ = 62	σ=16	σ = 17	σ = 41

Tableau 1 Paramètres utilisés pour la génération de données

Nous avons comparé notre approche avec différentes métaheuristiques, à savoir un algorithme génétique bi-critères simple (BI-GA) [6] avec une somme pondérée des fonctions objectives, une recherche à voisinage variable bi-critères (BI-VNS) [13] et un algorithme génétique de tri non dominé (NSGA-II) [14] [15]. Ces approches sont développées en Python, sous le système d'exploitation : Microsoft Windows 10 (64 bits) sur un processeur Intel(R) Core (TM) i7-1185G7 de 11e génération avec 32 Go de RAM. Chaque instance a été exécutée 20 fois. Le tableau 2 présente le gain moyen calculé à l'aide de l'Eq. 7.

$$GAIN = \frac{ApprocheA - ApprocheB}{ApprocheB} \times 100 (7)$$

Les résultats montrent que le Q-learning présente un avantage distinct en apprenant des politiques optimales et devance les performances des métaheuristiques utilisées en termes de

makespan et de temps d'attente moyen, comme le montre le Tableau 2. Le Q-learning améliore constamment le makespan avec un gain moyen de 0,13 par rapport à BI-GA, de 0,20 par rapport à NSGA-II, et de 0,053 par rapport à BI-VNS. De même, en ce qui concerne la réduction du temps d'attente moyen, le Q-learning présente un gain moyen de 1,31 par rapport à BI-GA, de 1,60 par rapport à NSGA-II, et de 0,97 par rapport à BI-VNS.

	Moyenne GAIN (1)[1]		*Moyenne GAIN (2)*[2]		*Moyenne GAIN (3)*[3]	
Instance	C_{max}[4]	*WT*[5]	C_{max}	*WT*	C_{max}	*WT*
C-10	0,003	2,60	0,01	2,82	0,00	1,46
C-15	0,08	1,78	0,08	2,59	0,04	2,30
C-20	0,13	1,34	0,23	1,80	0,08	0,81
C-30	0,12	1,23	0,27	1,73	0,04	0,75
C-40	0,16	1,12	0,28	1,30	0,17	0,72
C-50	0,13	1,04	0,22	1,23	-0,01	0,67
C-60	0,17	0,97	0,24	1,08	0,03	0,68
C-80	0,22	0,88	0,27	0,95	0,09	0,69
C-100	0,19	0,87	0,24	0,91	0,04	0,70
Moyenne	**0,13**	**1,31**	**0,20**	**1,60**	**0,053**	**0,97**

Tableau 2 Comparaison de Q-learning, BI-GA, BI-VNS, NSGA-II
[1]***Gain 1****: (BI-GA / Q-learning),* [2]***Gain 2****: (NSGA-II/Q-learning),* [3]***Gain 3****: (BI-VNS/Q-learning) ,* [4]C_{max}*: la plus grande durée d'hospitalisation,* [5] ***WT****: temps d'attente moyen*.

6 Conclusion et perspectives

Dans cet article, nous avons abordé le problème d'ordonnancement des interventions en chirurgie ambulatoire en développant une approche basée sur l'apprentissage par renforcement utilisant l'algorithme Q-learning. À travers la génération de données simulées, nous avons réalisé une analyse comparative impliquant des métaheuristiques connues, à savoir BI-GA, BI-VNS et NSGA-II. Cette analyse a fourni des informations sur l'efficacité de notre approche proposée, qui a constamment devancé ces métaheuristiques en minimisant le makespan et le temps d'attente moyen simultanément. Ces résultats relèvent l'adaptabilité et l'efficacité du Q-learning dans la résolution de problèmes d'ordonnancement complexes dans le domaine hospitalier.

En se basant sur les résultats présentés dans cet article, il est important de relever plusieurs perspectives qui méritent une exploration. Nous suggérons les directions suivantes : l'intégration des imprévus dans les délais des différentes phases est essentielle pour développer une approche robuste qui permet de faire face aux imprévus avec résilience et efficacité. l'étude de techniques de Q-learning avancées, telles que les réseaux Q profonds (DQN) ou le double Q-learning, qui pourraient davantage améliorer les performances et l'adaptabilité ; l'exploration d'approches hybrides en combinant l'apprentissage par renforcement avec des métaheuristiques ; et enfin, la validation dans le monde réel en collaborant avec des structures hospitalières pour mettre en œuvre et évaluer les algorithmes et fournir un feedback sur la satisfaction des patients, l'utilisation des ressources et l'efficacité opérationnelle. En poursuivant ces pistes de recherche, nous pouvons faire progresser l'état de l'art dans la planification et l'ordonnancement de la chirurgie ambulatoire.

7 Références

[1] L. Wang, E. Demeulemeester, N. Vansteenkiste, et F. E. Rademakers, « Operating room planning and scheduling for outpatients and inpatients: A review and future research », Operations Research for Health Care, vol. 31, p. 100323, déc. 2021, doi: 10.1016/j.orhc.2021.100323.

[2] C. Bandi et D. Gupta, « Operating Room Staffing and Scheduling », M&SOM, vol. 22, no 5, p. 958-974, sept. 2020, doi: 10.1287/msom.2019.0781.

[3] S. Ghasemi, R. Tavakkoli-Moghaddam, et M. Hamid, « Operating room scheduling by emphasising human factors and dynamic decision-making styles: a constraint programming method », International Journal of Systems Science: Operations & Logistics, vol. 10, no 1, p. 2224509, déc. 2023, doi: 10.1080/23302674.2023.2224509.

[4] K. Wang, H. Qin, Y. Huang, M. Luo, et L. Zhou, « Surgery scheduling in outpatient procedure centre with re-entrant patient flow and fuzzy service times », Omega, vol. 102, p. 102350, juill. 2021, doi: 10.1016/j.omega.2020.102350.

[5] J. R. Munavalli, S. V. Rao, A. Srinivasan, et G. van Merode, « Integral patient scheduling in outpatient clinics under demand uncertainty to minimize patient waiting times », Health Informatics J, vol. 26, no 1, p. 435-448, mars 2020, doi: 10.1177/1460458219832044.

[6] S. Gul, B. T. Denton, J. W. Fowler, and T. Huschka, "Bi-Criteria Scheduling of Surgical Services for an Outpatient Procedure Center," Production and Operations Management, vol. 20, no. 3, pp. 406–417, Feb. 2011, doi: https://doi.org/10.1111/j.1937-5956.2011.01232.x.

[7] N. Mazyavkina, S. Sviridov, S. Ivanov, et E. Burnaev, « Reinforcement learning for combinatorial optimization: A survey », Computers & Operations Research, vol. 134, p. 105400, oct. 2021, doi: 10.1016/j.cor.2021.105400.

[8] J. N. D. Gupta, A. M. A. Hariri, et C. N. Potts, « Scheduling a two-stage hybrid flow shop with parallel machines at the first stage », Annals of Operations Research, vol. 69, no 0, p. 171-191, janv. 1997, doi: 10.1023/A:1018976827443.

[9] R. S. Sutton and A. G. Barto, "Reinforcement Learning: An Introduction," in IEEE Transactions on Neural Networks, vol. 9, no. 5, pp. 1054-1054, Sept. 1998, doi:10.1109/TNN.1998.712192.

[10] S. Lee et Y. H. Lee, « Improving Emergency Department Efficiency by Patient Scheduling Using Deep Reinforcement Learning », Healthcare, vol. 8, no 2, p. 77, mars 2020, doi: 10.3390/healthcare8020077.

[11] P. Ribino, C. Di Napoli, et L. Serino, « A Multi-Agent RL Algorithm for Single-Day Operating Room Scheduling », in Ambient Intelligence and Smart Environments, H. H. Alvarez Valera et M. Luštrek, Éd., IOS Press, 2022. doi: 10.3233/AISE220053.

[12] W. Han, F. Guo, et X. Su, « A Reinforcement Learning Method for a Hybrid Flow-Shop Scheduling Problem », Algorithms, vol. 12, no 11, p. 222, oct. 2019, doi: 10.3390/a12110222.

[13] E. Claudio, Rafael, and A. de, "Multi-objective Variable Neighborhood Search Algorithms for a Single Machine Scheduling Problem with Distinct due Windows," Electronic notes in theoretical computer science, vol. 281, pp. 5–19, Dec. 2011, doi: https://doi.org/10.1016/j.entcs.2011.11.022.

[14] Q. Lu, X. Zhu, D. Wei, K. Bai, J. Gao, and R. Zhang, "Multi-phase and Integrated Multi-objective Cyclic Operating Room Scheduling Based on an Improved NSGA-Ⅱ Approach," Symmetry, vol. 11, no. 5, p. 599, Apr. 2019, doi: https://doi.org/10.3390/sym11050599.

[15] L. Bouchlaghem, F. Ghedjati, Résolution d'un problème BI-objectif d'ordonnancement d'un bloc opératoire en chirurgie ambulatoire. Workshop du Groupe ROSa du GDR RO sur la Recherche Opérationnelle et Santé, Juin 2023, Amiens, France.