
TP - Méta-heuristiques 4-IR - Informatique

MJ. HUGUET
homepages.laas.fr/huguet

Objectifs

L'objectif des séances de TP est de résoudre un problème d'optimisation combinatoire en utilisant des méta-heuristiques. Pour cela vous allez mobiliser vos compétences en :

- méta-heuristique et optimisation combinatoire
- algorithmique, structure de données, graphes, complexité
- développement
- bon sens

Vous travaillez en binômes et devez développer deux méthodes différentes. Il y a 5 séances de TP encadrées. L'évaluation s'effectue sur la base d'un rapport expliquant à la fois les méthodes utilisées et les résultats obtenus.

1 Problème étudié

Le problème considéré est un problème d'ordonnancement appelé "Job Shop Flexible". La première étape consiste à comprendre le problème posé.

1.1 Ordonnancement

Un problème d'ordonnancement consiste à organiser dans le temps la réalisation d'un ensemble de d'activités, compte tenu de contraintes temporelles (délais, enchaînements, ...) et de contraintes portant sur l'utilisation et la disponibilité des ressources requises. Un ordonnancement est ainsi caractérisé par :

- Un ensemble d'activités
- un ensemble de ressources
- des contraintes sur les activités et les ressources
- un (ou plusieurs) critères d'optimisation

Une solution d'un problème d'ordonnancement définit la meilleure stratégie de réalisation temporelle des activités compte tenu des critères d'optimisation : par exemple dates de début des activités, séquençement des activités,).

Un problème d'ordonnancement est dit *flexible*, si les ressources nécessaires à la réalisation des activités n'est pas fixée a priori mais doit être établi lors du calcul de la solution. Résoudre un problème d'ordonnancement flexible nécessite alors de déterminer à la fois une affectation des ressources aux activités et un positionnement temporel des activités.

Compréhension. Pouvez-vous donner un exemple de problème d'ordonnancement et d'un problème d'ordonnancement flexible ? Quelles contraintes pouvez-vous imaginer sur les activités et sur les ressources ? Quels critères d'optimisation pouvez-vous envisager ?

1.2 Job Shop Flexible

1.2.1 Définition du problème de Job Shop

Un problème de Job Shop est un problème d'ordonnancement défini comme suit :

- un ensemble de n travaux (jobs) devant être exécutés sur une ensemble de m machines
- chaque job est composé d'au plus m activités¹ (ayant une durée de réalisation connue) devant être réalisé dans un ordre donné
- chaque activité nécessite une machine spécifique et on suppose généralement que chaque machine n'est demandée que par une seule activité de chaque job
- chaque machine ne peut réaliser qu'une activité à la fois

Un critère d'optimisation classique est la minimisation de la durée totale de réalisation de l'ensemble des jobs (ie. des activités) aussi appelé C_{\max} . Ce problème est connu pour être NP-difficile.

Exemple. On considère 3 jobs notés J_1 , J_2 et J_3 et 3 machines M_1 , M_2 et M_3 . Chaque job est composé d'au plus 3 activités (opérations) consécutives que l'on note $o_{i,j}$ où i représente le rang de l'activité dans le job et j le numéro de job. Elle a une durée notée $p_{i,j}$. La machine utilisée par l'activité $o_{i,j}$ est notée $M_{i,j}$.

1. aussi appelées opérations

J_j	J_1			J_2			J_3	
$o_{i,j}$	$o_{1,1}$	$o_{2,1}$	$o_{3,1}$	$o_{1,2}$	$o_{2,2}$	$o_{3,2}$	$o_{1,3}$	$o_{1,3}$
$M_{i,j}$	M_1	M_2	M_3	M_2	M_1	M_3	M_3	M_2
$p_{i,j}$	3	2	5	4	2	2	2	3

TABLE 1 – Exemple Job Shop

Graphe potentiels-tâches. Il est possible de représenter ce problème en utilisant un graphe (appelé initialement graphe potentiels-tâches) dans lequel les sommets représentent les dates de début de chaque activité. Ce graphe est complété par un sommet origine relié aux premières activités de chaque job et un sommet fin auquel sont reliées les dernières activités de chaque job. Un arc entre deux sommets x et y valué par $l_{x,y}$ représente la relation $y \geq x + l_{x,y}$ c'est à dire $y - x \geq l_{x,y}$. Ces arcs permettent de représenter les contraintes temporelles reliant les activités entre elles.

Compréhension et rappel des cours de graphes de 3MIC. Donnez le graphe correspondant à l'exemple ci-dessus en supposant que les dates de début au plus tôt de chaque job sont nulles. Quelles sont les caractéristiques de ce graphe ? Comment utiliser ce graphe pour déterminer la durée minimale de réalisation de l'ensemble des activités (ie. minimiser le critère C_{\max}) ? Quelle est la méthode permettant de réaliser ce calcul ? Quelle est la complexité de ce problème ?

Déterminez la solution et représentez là sur un diagramme de Gantt. Lors du calcul de la solution, vous avez déterminé également les dates de début au plus tôt de chaque activité. Vous souvenez-vous comment déterminer les dates de début au plus tard ?

1.2.2 Contraintes temporelles

Le modèle de graphe précédent peut être étendu à la prise en compte de diverses contraintes temporelles.

- Comment représenter le fait que la date de début au plus tôt du job 2 vaut 1 ?
- Comment représenter le fait que la date de début au plus tôt de l'opération $o_{3,1}$ vaut 6 ?
- Comment représenter le fait que la date de début au plus tard du job 3 vaut 4 ?
- Comment représenter des contraintes de chevauchement temporelle telles que l'activité x peut débiter à partir de la fin de l'activité $y + 3$? Ces contraintes sont appelées "time-lag" minimal.
- Comment représenter des contraintes de chevauchement temporelle telles que l'activité x peut débiter au maximum après le début de l'activité $y + 3$? Ces contraintes sont appelées "time-lag" maximal.

Compréhension. La méthode précédente peut-elle être utilisée en présence de contraintes de dates de début au plus tôt ? Peut-elle être utilisée en présence de contraintes de dates de début au plus tard ? Peut-elle être utilisée en présence de contraintes de "time-lag" minimal ? Peut-elle être utilisée en présence de contraintes de "time-lag" maximal ?

1.2.3 Contraintes de ressources

Les contraintes de ressources du problème de Job Shop exprime le fait que deux activités nécessitant la même ressource doivent être réalisés séquentiellement. La ressource est dite disjonctive.

Pour représenter ces contraintes de ressources, le graphe potentiels-tâches doit être étendu à la prise en compte d'un second type d'arc. Pour chaque paire d'activités x et y en conflit par rapport à l'utilisant d'une ressource, le graphe comporte deux arcs : un arc dans le sens x vers y et traduisant le fait que x est réalisée avant y et un arc dans le sens y vers x traduisant le fait que y est réalisée avant x .

Compréhension. Quelles sont les paires de tâches en conflit dans l'exemple précédent ? Représentez les arcs correspondant sur le graphe (notez les arcs d'une couleur différente ou en pointillé pour les distinguer des arcs représentant les contraintes temporelles initiales). Quelles sont les valeurs des différents arcs associés ?

Une solution d'un problème de Job Shop doit donc choisir un seul arc pour chaque paire d'activités en conflit. Lorsque tous les conflits sont résolus, on peut déterminer une solution minimisant le critère C_{\max} et les dates de début au plus tôt de chaque activité.

Exemple 1 de solution. Considérez une solution pour laquelle l'ordre de réalisation des opérations sur les machines est le suivant :

- sur M_1 : $o_{1,1} \prec o_{2,2}$
- sur M_2 : $o_{1,2} \prec o_{2,1} \prec o_{2,3}$
- sur M_3 : $o_{1,3} \prec o_{3,2} \prec o_{3,1}$

Représentez la solution sur le graphe. Quelles sont les caractéristiques de ce graphe ? Comment utiliser ce graphe pour déterminer la durée minimale de réalisation de l'ensemble des activités (ie. minimiser le critère C_{\max}) ? Quelle est la complexité de ce problème ?

Déterminez la solution et représentez là sur un diagramme de Gantt.

Exemple 2 de solution. Considérez une solution pour laquelle l'ordre de réalisation des opérations sur les machines est le suivant :

- sur M_1 : $o_{2,2} \prec o_{1,1}$
- sur M_2 : $o_{2,1} \prec o_{1,2} \prec o_{2,3}$
- sur M_3 : $o_{1,3} \prec o_{3,1} \prec o_{3,2}$

Représentez la solution sur le graphe. Quelles sont les caractéristiques de ce graphe ?

Compréhension. Pourquoi cette solution n'est-elle pas réalisable ?

1.2.4 Job Shop avec flexibilité des ressources

Un problème de Job Shop Flexible est un problème de Job Shop dans lequel la machine nécessaire à la réalisation de chaque activité n'est pas fixée mais peut être choisie parmi un ensemble de machines. La durée de réalisation de l'activité peut dépendre ou non de la machine affectée.

Exemple. Soit le problème de Job Shop précédent avec flexibilité de ressources. La durée d'une opération $o_{i,j}$ sur une machine k est notée $p_{i,j,k}$.

J_j	J_1			J_2			J_3	
$o_{i,j}$	$o_{1,1}$	$o_{2,1}$	$o_{3,1}$	$o_{1,2}$	$o_{2,2}$	$o_{3,2}$	$o_{1,3}$	$o_{1,3}$
$M_{i,j}$	M_1	$\{M_2, M_3\}$	$\{M_2, M_3\}$	M_2	M_1	$\{M_1, M_2, M_3\}$	$\{M_1, M_3\}$	$\{M_2, M_3\}$
$p_{i,j,k}$	3	{2,4}	{5,5}	4	2	{2,4,2}	{2,2}	{3,5}

TABLE 2 – Exemple Job Shop Flexible

Un critère d'optimisation classique est la minimisation de la durée totale de réalisation de l'ensemble des jobs (C_{\max}). Ce problème est connu pour être NP-difficile.

Sa résolution nécessite de déterminer une affectation des machines aux activités et un séquençement des activités en conflit.

2 Méthodes de résolution du Job Shop Flexible

Le Job Shop Flexible est un problème largement étudié et il existe des méthodes variées pour le résoudre. Lors de ces TP, votre but est de le résoudre en utilisant deux méta-heuristiques différentes :

- une méthode de recherche locale
- une méthode à population de solutions

Un travail de recherche récent faisant une synthèse de l'état de l'art vous est communiqué afin de vous aider dans votre démarche.

Des jeux de données, pour évaluer vos méthodes entre elles et avec les méthodes de l'état de l'art, sont accessibles sur le site <http://people.idsia.ch/~monaldo/fjsp.html>.

Selon les méta-heuristiques choisies, vous pouvez trouver des bibliothèques pour vous aider à les développer.

3 Des conseils ...

Il y a diverses étapes à réaliser : compréhension du problème, lecture des jeux de données, représentation des solutions, processus d'intensification et de diversification, développement des méta-heuristiques, vérification de l'admissibilité des solutions, tests unitaires, tests de paramétrage, tests de performance, une visualisation des solutions peut vous aider... Organisez vous et utilisez les outils qui sont là pour vous aider (comme git).

- Réfléchir avant d'agir, c'est la qualité de votre travail qui est importante
- Le langage de programmation est celui qui vous convient
- Vous pouvez implémenter une méthode déjà proposée pour le Job Shop Flexible
- Pensez aux tests unitaires
- Vous devez concevoir et implémenter une méthode pour vérifier que les solutions obtenues sont bien des solutions valides (ça serait dommage de proposer des solutions irréalisables ...)
- Les paramètres d'une méta-heuristique sont souvent nombreux, prévoyez une phase préliminaire de tests pour vous aider dans le choix de ces paramètres
- Vérifiez que votre méthode se comporte comme attendu
- Une méthodologie rigoureuse d'évaluation des performances devra être mise en place.
- Le contenu de votre rapport doit permettre aux futurs étudiants de 4-IR de ré-implémenter vos méthodes et de retrouver les mêmes résultats que vous.
- Il est fortement conseillé d'utiliser \LaTeX pour la rédaction du rapport (par exemple le site overleaf vous permet de travailler en mode collaboratif).