

Recherche Opérationnelle :
TP 2-3 : Algorithme de branch-and-Bound pour résoudre le
problème du sac à dos

Sandra U. Ngueveu, Corentin Boennec, Arthur Clavière, Aloïs Duguet
ngueveu@laas.fr

2021

Table des matières

1	Prise en main	1
2	Travail à faire	2
3	Rapport de TP attendu : Instructions	2
4	BONUS	2

Le but est de résoudre le problème du sac à dos à l'aide d'un algorithme de branch-and-bound (*procédure de séparation et évaluation (PSE)*) que vous aurez implémenté.

Rendus attendus (à soumettre sur moodle à l'emplacement dédié) : archive .zip contenant :

- le notebook en julia permettant de résoudre les instances fournies et d'autres de votre choix
- un rapport synthétique (3 pages max) au format pdf s'il n'est pas déjà inclu dans le notebook

1 Prise en main

Télécharger de Moodle le zip contenant :

- un dossier instancesETU contenant les jeux de données disponibles pour vos évaluations numériques (vous pouvez aussi en ajouter d'autres pour illustrer des cas particuliers)
- un notebook-exemple fourni, qui ne fonctionne que pour l'instance test.opb.txt correspondant à l'exemple du cours, et utilise un solveur de LP nommé Clp pour résoudre les relaxations linéaires

2 Travail à faire

Votre objectif sera de prendre en main le notebook-exemple, supprimer tous les appels à JuMP, supprimer tous les appels à un quelconque solveur (et donc à Clp), remplacer les calculs de bornes par des algorithmes que vous aurez implémenté (pas de solveur) et refaire fonctionner l'algorithme de branch-and-bound, d'abord pour l'instance test, puis pour toutes les instances du dossier instancesETU.

Attention, supprimer la section dédiée à Clp supprimera non seulement le modèle mathématique model2, mais aussi le vecteur des variables x , et la liste des variables binaires par ordre de priorité pour branchement varsshouldbinary. A vous de corriger le code complet en conséquence pour que votre algorithme fonctionne.

3 Rapport de TP attendu : Instructions

Ce rapport de 3 pages maximum pourra être inclus dans un notebook. Il participera à la note globale de TP du cours de Recherche Opérationnelle. Il est à rendre en binôme du même groupe de TP.

Le fichier sera à rendre au plus tard le **vendredi 17 décembre 23h55** sur moodle.

Donner, par exemple :

- Quelques détails sur les points clés et non triviaux de votre implémentation
- Une courte argumentation de l'adéquation du résultat avec l'instance résolue
- Quelques éléments d'analyse, par exemple :
 - Analyser l'impact des différents paramètres de la PSE sur la performance en termes de vitesse/délais (en temps cpu et nombre de noeuds explorés) pour l'obtention d'une solution réalisable, pour l'obtention de la meilleure solution, pour l'obtention de bornes supérieures de qualité et pour la complétion de l'algorithme

4 BONUS

Comparer votre PSE avec GLPK : une comparaison en temps ne serait pas forcément pertinente au vue des différences de langages de programmation, mais une comparaison avec le nombre de noeuds de l'arborescence serait intéressante à faire, en particulier avec différents paramètres de choix de stratégies d'exploration de GLPK