Université de Nantes — UFR Sciences et Techniques Master informatique parcours "optimisation en recherche opérationnelle (ORO)" Année académique 2019-2020

Métaheuristiques

Consignes du devoir maison n°2

Xavier Gandibleux

Métaheuristique reactiveGRASP sur le "Set Packing Problem (SPP)"

Ce devoir maison conduit à l'élaboration d'un solveur reposant sur une métaheuristique pour le Set Packing Problem (SPP). Pour une instance numérique disponible représentant un SPP quelconque :

- 1. Mettre en place (1) GRASP et ensuite (2) la variante reactiveGRASP.
- 2. A l'aide de votre algorithme, et pour un budget de calcul donné (exemple 60 secondes), mener une expérimentation numérique (résoudre par exemple 10 fois sur une même machine) avec les 10 instances déjà étudiées dans le précédent devoir et rapporter vos résultats ($\hat{z}_{min}, \hat{z}_{max}, \hat{z}_{moy}$). Etudier l'influence de α pour GRASP, les valeurs saillantes de α pour ReactiveGRASP.
- 3. Bonus : mettre en place une stratégie de path-relinking ou une stratégie Destroy-Repair. Evaluer la contribution de ces stratégies à l'aide de vos 10 instances de référence et rapporter vos résultats dans votre rapport.
 - Aussi, outre les variantes algorithmiques que connait GRASP, la métaheuristique se prête particulièrement bien au calcul distribué. On peut ainsi envisager d'utiliser tous les coeurs d'un processeur, songer à la parallélisation de l'algorithme ou encore s'orienter sur le calcul sur GP-GPU. Ces aspects dépassent les notions abordées dans ce cours, mais les facilités d'un langage comme Julia sur ces mécanismes ouvrent des portes aux curieux et avertis d'entre vous qui voudraient expérimenter sur ce volet dans le cadre de ce travail.
- 4. Rapporter avec rigueur vos résultats dans un document LATEX en complétant le compterendu produit à l'issue du DM1.
- 5. Votre solution logicielle reservira dans le devoir maison n°3.

Travail à engager sans attente et à remettre sur madoc le 09 octobre 2019 au plus tard.

Remarques (rappel) :

- Vous trouverez sur madoc les documents scientifiques complémentaires.
- Le codage des algorithmes se fera soit en langage JULIA, soit en langage C.
- Les algorithmes seront exécutés sous unix/linux.
- Le livrable comportera le compte rendu en pdf du DM2 (commenter le DM1 et DM3).
- Ce travail contribue à plusieurs résultats d'apprentissage attendus à l'issue de ce cours. Ne le voyez pas comme un simple travail applicatif des volets du cours mais comme complément à part entière. J'attire donc votre attention sur le fait d'être professionnel et scientifiquement rigoureux dans vos productions, analyses et rendus.