



Alger le 03 décembre 2019

Projet

Partie 1.

Le projet du cours de « Bio-inspired Computing » consiste à implémenter des solutions pour le problème de satisfiabilité (SAT).

Il est demandé d'implémenter des algorithmes exacts à savoir les méthodes aveugles ensuite des méthodes heuristiques, le but étant de constater l'explosion combinatoire des méthodes exhaustives et de s'initier aux méthodes heuristiques et métaheuristiques qui présentent des approches incontournables pour la résolution des problèmes complexes. Il est clair qu'il faut utiliser des structures de données très efficaces pour aller plus loin dans la résolution du problème.

Détails

Implémenter en Java une solution pour le problème de satisfiabilité en utilisant :

- La recherche en largeur d'abord. Noter la complexité empirique temporelle et spatiale.
- La recherche en profondeur d'abord. Tester avec plusieurs seuils de profondeur et noter la complexité empirique temporelle et spatiale.
- L'algorithme A* avec plusieurs heuristiques et comparer les performances de ces heuristiques en présentant la complexité empirique temporelle et spatiale.

Commenter vos résultats.

Partie 2.

Implémenter en Java une solution pour le problème de satisfiabilité en utilisant les Algorithmes Génétiques.

Effectuer des tests de comparaison avec les approches précédentes et conclure.

Partie 3.

Implémenter en Java une solution pour le problème de satisfiabilité en utilisant ACS. Utiliser A* développé dans la partie 1 pour la construction d'une solution par une fourmi.

Effectuer des tests de comparaison avec les approches précédentes et conclure.

Données.

Les benchmarks uf75-325 / uuf75-325 serviront pour les expérimentations des algorithmes développés pour les trois parties. Une description de ces données est fournie à l'adresse :

<http://www.cs.ubc.ca/~hoos/SATLIB/benchm.html>

test-set	#instances	clause-length	#variables	#clauses
uf75-325 / uuf75-325	100	3	75	325