

TD2 : Problèmes de Satisfaction de Contraintes

Exercice 1. Backtrack

On peut représenter une exécution de l'algorithme de Backtrack vu en cours en développant l'arbre de recherche dont les nœuds représentent une assignation (de plus en plus complète) ; la racine de l'arborescence représentant l'assignation vide. Pour faciliter la représentation, on se limitera à indiquer sur chaque nœud le couple (variable, valeur) ajouté à l'assignation courante. Lorsqu'une assignation viole une contrainte, on indique par un **BT** que le nœud engendre un « backtrack » en précisant la (ou les) contrainte(s) violée(s).

- 1) Exécuter l'algorithme Backtrack sur le problème de coloration de carte vu en cours en considérant à chaque choix de variables à assigner l'ordre WA, Q, T, SA, NSW, V, NT et à chaque choix de valeurs l'ordre R, G, B .
- 2) Même question en considérant les variables dans l'ordre WA, NT, NSW, Q, V, SA, T et les valeurs dans l'ordre R, G, B .
- 3) Quel impact l'ordre d'affectation des variables a-t-il sur l'arbre de recherche ? Et celui des valeurs ?
- 4) Discuter de l'importance du choix de ces ordres pour la recherche d'une solution et pour celle de toutes les solutions.
- 5) Calculer un ordre d'application des variables en appliquant l'heuristique min-width (et en cas d'égalité suivre l'ordre lexicographique) puis tracer l'exécution de Backtrack avec cet ordre.
- 6) Calculer un ordre d'application des variables en appliquant l'heuristique max-card (et en cas d'égalité max-degree puis suivre l'ordre lexicographique si encore égalité) puis tracer l'exécution de Backtrack avec cet ordre.
- 7) Le test consistant dans BT n'a besoin que de « vérifier » un sous-ensemble des contraintes du problème. Lesquelles et pourquoi ? Proposer un algorithme pour le test de consistance.

Exercice 2. Cryptogrammes

Les puzzles arithmétiques (ou cryptogrammes) sont des opérations où une lettre correspond à un chiffre. On donne l'opération en lettres, il faut trouver celle en chiffres.

$$\begin{array}{r} T \ W \ O \\ + T \ W \ O \\ \hline = F \ O \ U \ R \end{array}$$

La résolution de ces puzzles peut être modélisée comme un CSP. Modélisez le cryptogramme précédent par un CSP. Vous représenterez graphiquement le réseau de contraintes. Vous donnerez d'abord les contraintes sous la forme d'équation puis quand cela est raisonnable donnerez la version « en extension » des contraintes. Enfin vous calculerez selon l'heuristique vue au dernier TD l'ordre d'assignation des variables.