Quelques éléments sur le problème SAT

christian.laforest@isima.fr

March 17, 2020

Les problèmes de type SAT sont centraux en informatique (dans le domaine de la complexité et celui de l'optimisation). Nous allons nous restreindre ici à une sous-classe plus simple à décrire (mais qui est difficile à résoudre).

Notations On se donne $V = \{x_1, \ldots, x_n\}$ un ensemble de *variables booléennes* : chaque variable x_i peut se voir attribuer la valeur *vraie* (notée 1 pour simplifier) ou la valeur *faux* (notée 0).

Un *littéral* construit partir de la variable x_i est soit x_i, soit $\overline{x_i}$. Le littéral x_i a pour valeur 1 ssi x_i reçoit la valeur 1. Le littéral $\overline{x_i}$ a pour valeur 1 ssi x_i reçoit la valeur 0 ($\overline{x_i}$ peut être interprété comme "non x_i").

Une *clause* est une *disjonction* (le "ou" logique) de littéraux deux-à-deux distincts. Ici, nous nous restreindrons aux clauses composées de exactement *trois* littéraux. Ici une clause est donc : $l(x_i) \bigvee l(x_j) \bigvee l(x_k)$ avec trois littéraux, notés $l(x_i)$, $l(x_j)$, $l(x_k)$, construits à partir de trois variables x_i, x_j, x_k (pas nécessairement deux-à-deux distinctes).

Par exemple, $\overline{x_3} \bigvee x_1 \bigvee x_3$ ou $x_4 \bigvee x_3 \bigvee x_5$, etc. sont des clauses.

La valeur (booléenne) de la clause $l(x_i) \bigvee l(x_j) \bigvee l(x_k)$ est vrai (1) ssi *au moins* un littéral parmi les 3 est vrai.

Par exemple, dans la clause $x_4 \bigvee x_3 \bigvee \overline{x_5}$, il suffit que x_4 soit vrai pour que la clause soit vraie. Il suffit aussi que x_5 soit faux pour avoir la même chose. Par contre, si $x_4 = 0, x_3 = 0, x_5 = 1$ alors la clause a pour valeur 0.

Si la valeur d'une clause est vraie en attribuant une valeur à chacune des variables qui compose ses littéraux, on dit que la clause est *satisfaite* par cette attribution de valeurs.

Un ensemble S de clauses est dit *satisfiable* ssi il existe une attribution de valeur booléenne pour chaque variable rendant *toutes* les clauses vraies.

Etant donné un ensemble de clauses (même à 3 littéraux), le problème de *décider* s'il est satisfiable est un problème difficile. Actuellement personne ne connait d'algorithme efficace pour le résoudre.

Nous allons traiter ici un problème d'optimisation associé. Etant donné un ensemble de clauses, nous allons essayer de construire une attribution de valeur booléenne qui *maximise* de nombre de clauses satisfaites.

Le problème de maximisation est au moins aussi difficile que le problème de décision dans la mesure où si l'on savait résoudre "facilement" le problème de maximisation alors on saurait résoudre "facilement" le problème de décision.

Comme on ne sait rien faire de tout cela de manière déterministe en un temps "acceptable", on va donner, de manière aléatoire, équiprobable et indépendante, une valeur booléenne (0 ou 1) à chaque variable x_i. Cet algorithme aléatoire est nommé Attribution-Aléatoire.