2Sat est décidable en temps linaire

Références :

– *The Art of Computer Programming*, Volume 4A *Combinatorial Algorithms* Part 1. Pages 60 à 62.

Contexte : Une instance de 2Sat est la donnée d'une formule F sous forme normale conjonctive (FNC) dite de Krom (i.e. conjonction de clauses de la forme $a \vee b$ où a et b sont des littéraux). On se demande alors si il existe une valuation ν qui satisfasse F.

Il est remarquable que le problème un tout petit peu plus général qu'est 3Sat est en fait infiniment plus général vu qu'il est NP-Complet.

Développement : On se propose de démontrer le résultat suivant.

Théorème 1

2Sat est décidable en temps linéaire.

Construction du graphe. On se donne une formule $F = F(x_1, \cdots x_n)$ écrite comme conjonction de m clauses de Krom : $F = C_1 \wedge C_2 \wedge \cdots \wedge C_m$.

Construisons alors le graphe orienté $G = (V, E)$ avec $V = \{x_1, \overline{x_1}, x_2, \overline{x_2}, \cdots x_n, \overline{x_n}\}$. Et pour toute clause $C = (u \vee v)$; C est syntaxiquement équivalente à $(\overline{u} \Rightarrow v)$ et à $(\overline{v} \Rightarrow u)$, on crée donc deux arcs : $\overline{u} \rightarrow v$ et $\overline{v} \rightarrow u$.

Caractérisation de la satisfiabilité. On peut alors montrer le résultat suivant :

Théorème 2 (*M.* Krom)

F est satisfiable **ssi** *aucune composante fortement connexe de G ne contient à la fois une variable x et son complément $\overline{x}$.*

Le sens direct est trivial par contraposée : en effet si une composante fortement connexe contient x et $\overline{x}$, alors la formule $x \Rightarrow \overline{x} \Rightarrow x$ se déduit de F et donc F n'est pas satisfiable.

En particulier on comprend que la construction d'une valuation ν satisfaisant F nécessite que ν soit constante sur les composantes fortement connexes.

Réciproquement, supposons qu'aucun composante connexe ne contient une variable et son complémentaire. On va montrer le résultat par induction sur le nombre de composantes fortement connexes du graphe G.

Lemme 1

Le graphe G admet une composante "source" C telle qu'aucune flèche n'arrive dans C.

Preuve du lemme : On considère le graphe des composantes fortement connexes. C'est un graphe acyclique (si on avait un cycle alors toute les CFC de ce cycle n'en formeraient qu'une seule). Il suffit alors de parcourir le graphe en remontant les arêtes, ce parcours termine nécessairement car il y a un nombre fini d'état et aucun cycle. □

Suite : Soit S la composante source de G. Comme G est symétrique, l'application de G dans G, $u \mapsto \overline{u}$ envoie S sur S' une composante fortement connexe de G distincte de S par hypothèse et terminale au sens où il n'y a aucun arc sortant de S'.

On peut alors associer à chaque $x \in S$ la valeur $\nu(x) = 0$ et à $y \in S'$ la valeur $\nu(y) = 1$.

Dès lors on se restreint au graphe G_1 obtenu après le retrait des sommets de S et S'. Et on réitère l'opération jusqu'à ce que tous les littéraux se soient vus assigné une valeur via ν.

Les valeurs obtenues satisfont $\nu(u) \leq \nu(v)$ quand $u \to v$ dans le graphe, c'est-à-dire quand $\overline{u} \vee v$ (ou $\overline{v} \vee u$) est une clause de F.

Conclusion. La donnée des composantes fortement connexes du graphe nous permet immédiatement de conclure sur la satisfiabilité de F.

Reste à déterminer les composantes fortement connexes du graphe G. Or l'algorithme de Tarjan (ou de Kosaraju) nous donne le résultat en un temps linéaire vis à vis de la somme $|V| + |E|$ c'est-à-dire $2n + 2m$.

Remarque : l'algorithme de Tarjan détecte d'abord les CFC terminales, il suffit alors de modifier l'algorithme pour qu'il affecte à chaque sommet la valeur 1 à ce moment pour simplifier la mise en place.