Cours de Pascal Koiran

Adrien Friggeri Julien Provillard

1 Graphes et SAT

1.1 Graphes de largeur arborescente bornée

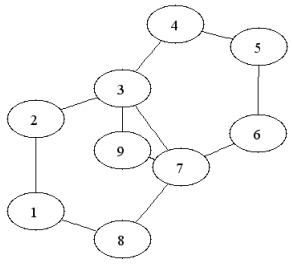
1.1.1 Définitions

Définition 1.1. Une **décomposition arborescente** d'un graphe $G = (V, E)$ est un arbre T dont chaque sommet t est étiqueté par un ensemble B_t de sommets de G tels que :

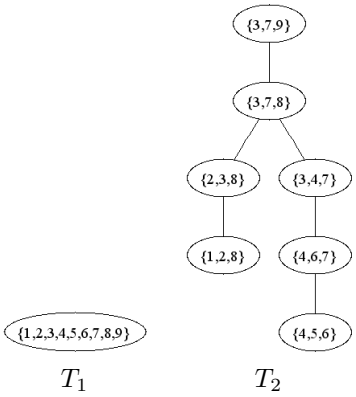
1. $\forall v \in V$, $B^{-1}(v) \stackrel{\text{def}}{=} \{t \in T : v \in B_t\}$ est non vide et connexe (*ie* c'est un sous-arbre de T).
2. $\forall (u, v) \in E$, $\exists t \in T$, $\{u, v\} \subseteq B_t$.

Définition 1.2. La **largeur de décomposition** d'une décomposition arborescente T est $max_{t \in T}|B_t|-1$.

Exemple 1.3. On considère le graphe G suivant :



Deux décompositions arborescentes possibles sont :



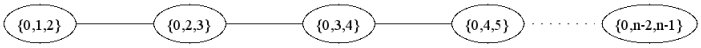
Définition 1.4. La **largeur arborescente** d'un graphe G, notée $tw(G)$, est le minimum des largeurs de décomposition des décompositions arborescentes de G.

Exemple 1.5. On donne ici quelques largeurs arborescentes classiques.

1. La largeur arborescente d'un arbre (non réduit à une racine) est 1.
2. La largeur arborescente d'un cycle est 2.
3. La largeur arborescente d'une grille $n \times n$ est n.

Démonstration. Preuve des deux premiers points,

1. Soit G un arbre non réduit à une racine, la décomposition T tel que $T = G$ et l'étiquette du noeud $i \in T$ est $\begin{cases} \{i\} \text{ si } i \text{ est la racine de } T \\ \{i, \text{père}(i)\} \text{ sinon} \end{cases}$ est de largeur de décomposition 1.
2. – Soit G un cycle $(0, 1, 2, \ldots, n-1)$, $n \geq 3$, la décomposition



est de largeur de décomposition 2.

– Soit G un graphe contenant un cycle $(0, 1, 2, \ldots, n-1)$, $n \geq 3$, supposons que G soit de largeur arborescente 1, on se donne T une décomposition de largeur 1.
Comme T est de largeur 1, pour toute arête (i, j) de G, il existe un noeud de T ayant comme étiquette $\{i, j\}$. Il existe donc des noeuds $(v_i)_{i \in \{0,\ldots,n-1\}}$ tel que v_i est étiqueté par $\{i, i+1\}$ (on considère désormais les indices modulo n).
Soit $i \in \{0, \ldots, n-1\}$, la composante connexe $B^{-1}(i)$ contient les noeuds v_{i-1} et v_i ; il existe donc un chemin entre ces noeuds dans cette composante. Donc pour tout $i \in \{0, \ldots, n-1\}$, il existe un chemin dans T entre v_{i-1} et v_i et qui est constitué de noeuds dont l'étiquette contient i.
Dans T, il existe un unique chemin entre deux sommets, donc il existe un unique chemin entre v_0 et v_{n-1}. Or ces deux noeuds sont dans $B^{-1}(0)$, donc ce chemin passe par des noeuds dont l'étiquette contient 0. Mais on sait construire un chemin de v_0 à v_{n-1} qui passe par $v_1, v_2, \ldots, v_{n-2}$, en particulier l'étiquette de v_1 est $\{1, 2\}$ et ne contient pas 0.
On obtient une contradiction qui prouve que tout graphe contenant un cycle est de largeur arborescente au moins 2.

□

Remarque. La preuve précédente montre que les seuls graphes connexes de largeur arborescente 1 sont les arbres.

1.1.2 Propriétés des décompositions arborescentes

Définition 1.6. On dit que T est une **k-décomposition** d'un graphe G si T est une décompostion arborescente de G de largeur de décomposition au plus $k-1$.

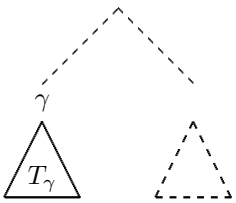
Propriété 1.7. *Soit $G = (V, E)$ un graphe de largeur arborescente au plus $k-1$. Il existe une k-décomposition T de G telle que T a au plus $|V|$ sommets.*

Propriété 1.8. *A partir d'une k-décomposition T d'un graphe G, on peut construire une autre décomposition arborescente T' de G de taille au plus $2|T|$, dans laquelle chaque sommet est de degré au plus 3.*

Corollaire 1.9. *Si G est un graphe tel que $tw(G) \leq k-1$, il existe une k-décomposition de G qui est un arbre binaire enraciné contenant au plus $2|V|$ sommets.*

1.1.3 Application à SAT

On note $I(\Sigma)$ le graphe d'incidence d'un ensemble de clauses Σ et on se donne une k-décomposition T de $I(\Sigma)$ qui soit un arbre binaire enraciné. On note T_γ le sous-arbre de T enraciné en γ.

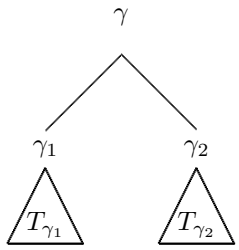


T

Remarque. On observe que

1. Si une clause C apparaît dans T_γ mais pas dans γ, alors toutes les variables de C apparaissent dans T_γ. En effet, toutes les variables de C apparaîssent dans $B^{-1}(C)$ (car elles sont reliées à C dans $I(\Sigma)$) qui est inclus dans T_γ.

2. Si γ a deux fils γ_1 et γ_2 et si v apparaît dans T_{γ_1} et dans T_{γ_2} alors v apparaît dans γ. En effet, $B^{-1}(v)$ doit être connexe.



Définition 1.10. Un **assignement partiel** est une fonction $w : W \to \{0, 1\}$ où $W \subseteq V$ ensembles des variables.

On dit que w satisfait un littéral l si
- $l = x$, avec $x \in V$ et $w(x) = 1$
- $l = \bar{x}$, avec $x \in V$ et $w(x) = 0$

1.1.4 Algorithme

Le principe de l'algorithme est de calculer des assignements partiels en remontant l'arbre depuis les feuilles. L'invariant est le suivant :

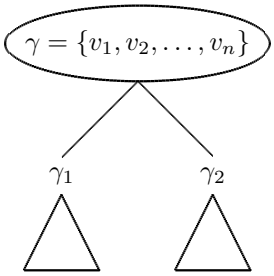
En chaque noeud γ, on calcule tous les couples (w, S) tels que
- $w : V_\gamma \to \{0, 1\}$ est un assignement partiel (il n'est défini que sur les variables qui apparaissant en γ).
- $S \subseteq C_\gamma$ (clauses apparaissant en γ).
- Il existe un autre assignement partiel $w' : V(T_\gamma) \to \{0, 1\}$ qui étend w ($w'_{|V_\gamma} = w$), satisfait toutes les clauses de $C(T_\gamma) \setminus C_\gamma$ et parmi les clauses de C_γ satisfait uniquement celles de S.

Remarque. Le nombre de couples à calculer en γ peut être exponentiel en k.

Aux feuilles Un couple (w, S) est accepté si w satisfait toutes les clauses de S, et uniqement celles-ci.

Aux noeuds internes

Cas particulier où γ ne comporte que des variables



Le couple (w, S) est accepté si on produit en γ_1 un couple (w_1, S_1) et en γ_2 un couple (w_2, S_2) tels que :
- $S_i = C(\gamma_i)$;
- w et w_i coïncident sur les variables communes à γ et γ_i.

Cas général
- w et w_i coïncident sur les variables communes à γ et γ_i ;
- S est égal à l'union de $S_1 \cap C_\gamma$, de $S_2 \cap C_\gamma$ et de l'ensemble des clauses de C_γ satisfaites par w.
- les clauses de $C(\gamma_i)$ qui ne sont pas dans $C(\gamma)$ sont dans S_i.

A la racine On déclare Σ satisfaisable si on produit un couple de la forme $(w, C(\gamma))$.

Remarque. On a supposé ici qu'on connaît la décomposition arborescente. Pour avoir un algorithme complet de résolution de Sat, il faut savoir construire cette décomposition.