# Compilation de formules CNF

Florent Capelli

IMJ-PRG, Paris Diderot

## Le problème

On a une base de connaissance sur un ensemble de variables $X$ à valeur dans $\{0,1\}$ représentée par une CNF $F$. On veut effectuer de nombreuses requêtes sur $F$ comme savoir si elle admet des solutions, combien elle en admet, énumérer ces solutions etc. Tous ces problèmes sont NP-durs. On aurait donc à payer un temps exponentiel à chaque requête. L'idée de la compilation est de perdre une seule fois du temps pour transformer $F$ en une structure de données qui supporte ces requêtes en temps polynomial:

**Sans compilation :**

```
> F est-elle satisfiable ?
Veuillez patienter longtemps, nous résolvons un problème NP-complet...
OUI
> Combien de solutions a F[x ↦ 0, y ↦ 1] ?
Veuillez patienter longtemps, nous résolvons un problème #P-complet...
237
> Quel est l'ensemble des solutions de ∃x.F ?
Veuillez patienter longtemps, nous cherchons une première solution...
01100110110
Veuillez patienter longtemps, nous cherchons une deuxième solution...
01100111111
```

**Avec compilation :**

```
Veuillez patienter longtemps, nous cherchons une représentation de F
plus intéressante.
> F est-elle satisfiable ?
OUI
> Combien de solutions a F[x ↦ 0, y ↦ 1] ?
237
> Quel est l'ensemble des solutions de ∃x.F ?
01100110110
01100111111
01100111101
```

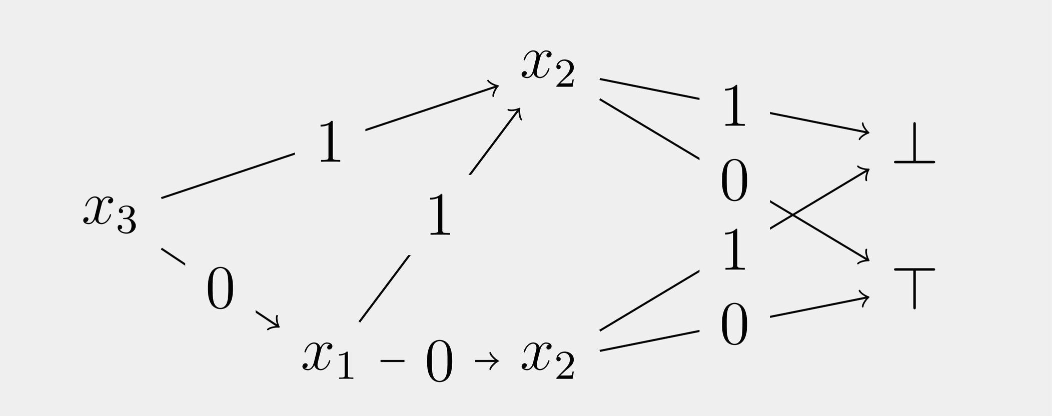
## Diagrammes de décision (FBDD)

### Définition

Un graphe acyclique avec deux feuilles $\top$ et $\bot$ et une source $s$.

- Chaque sommet teste une variable
- Tous les chemins de la source vers une feuille testent chaque variable au plus une fois.
- Chaque chemin définit donc une unique assignation partielle des variables : ce sont les valeurs de vérité de la fonction calculée.

### Exemple




| Décision | Comptage | Conditionnement | Enumération |
|---|---|---|---|
| ✔ | ✔ | ✔ | ✔ |

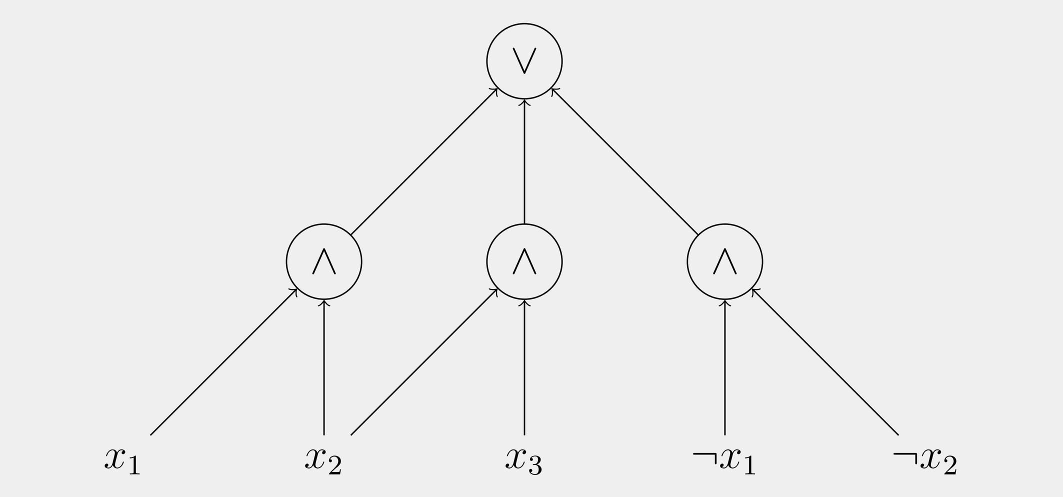
## DNNF

### Définition

Une DNNF est un circuit booléen:

- Les entrées du circuit sont étiquetées par une variable ou sa négation.
- Les autres portes sont étiquetées par des $\wedge$ ou des $\vee$.
- Pour chaque porte $\alpha$ étiquetée par $\wedge$, calculant $\alpha_1 \wedge \alpha_2$, on a $\mathsf{var}(\alpha_1) \cap \mathsf{var}(\alpha_2) = \emptyset$.

### Exemple




| Décision | Comptage | Conditionnement | Enumération |
|---|---|---|---|
| ✔ | ✖ | ✔ | ✔ |

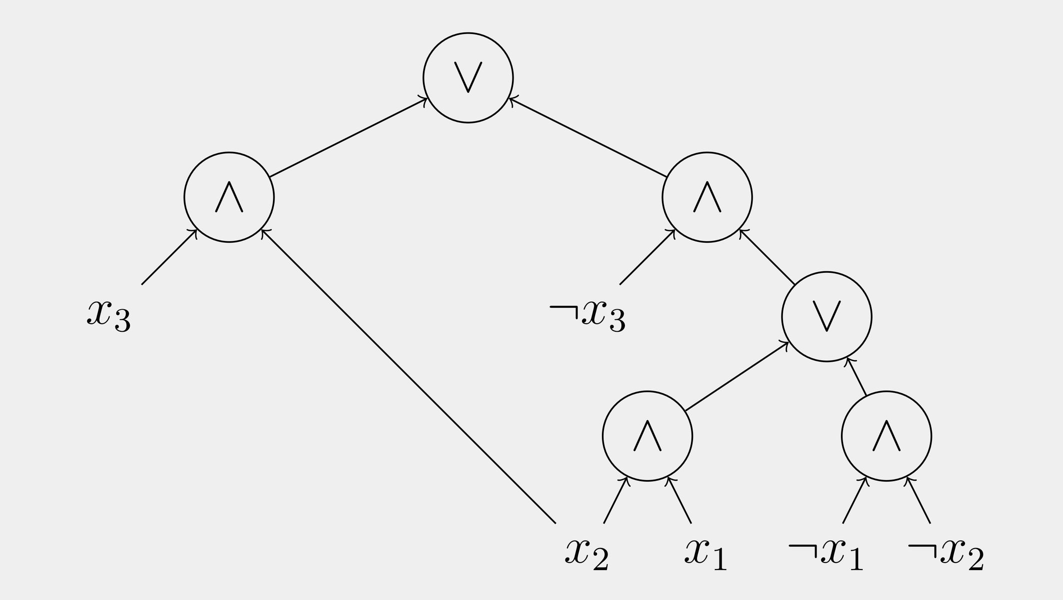
## DNNF déterministes

### Définition

Une DNNF déterministe, noté d-DNNF, est une DNNF vérifiant en plus:

- pour chaque porte $\alpha$ étiquetée par $\vee$, calculant $\alpha_1 \vee \alpha_2$, on a $\alpha_1 \wedge \alpha_2 = \bot$ ($\alpha_1$ et $\alpha_2$ ont des modèles disjoints)

### Exemple




| Décision | Comptage | Conditionnement | Enumération |
|---|---|---|---|
| ✔ | ✔ | ✔ | ✔ |

## Résultats d'impossibilité (bornes inférieures)

### Théorème

Sous des hypothèses de complexité raisonnables ($\mathsf{NP} \not\subseteq \mathsf{P/poly}$), il existe une famille $(F_n)$ de CNF tel que $F_n$ est de taille polynomial en $n$ et toute DNNF calculant $F_n$ est de taille non polynomial en $n$.

**A-t-on vraiment besoin d'une hypothèse de complexité ? NON :**

### CNF d'un graphe

On associe à un graphe $G = (V, E)$ la 2-CNF monotone $F_G$ sur les variables $V$ et $|E|$ clauses:

$$\bigwedge_{(x,y) \in E} (x \vee y)$$

### Théorème (*avec Bova, Mengel et Slivovsky*)

Il existe une famille de graphe $(G_n)_{n\in\mathbb{N}}$ et une constante $c > 0$ tel que pour tout $n$, $G_n$ a $n$ sommet et toute DNNF $D_n$ calculant $F_{G_n}$ est de taille au moins $2^{cn}$.

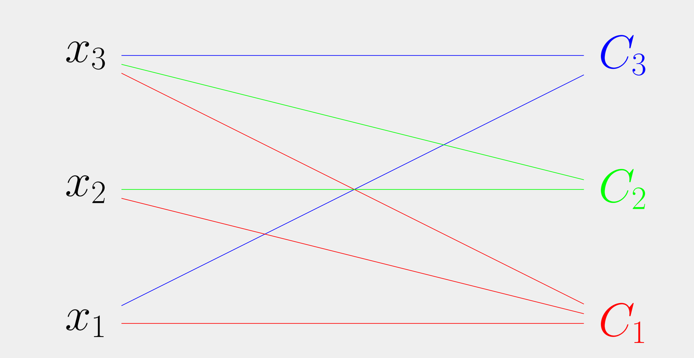
**Remarque :** le théorème précédent fonctionne pour toute famille $(G_n)_{n\in\mathbb{N}}$ de graphes expanders.

## Résultats algorithmiques (bornes supérieures)

### Graphe d'incidence d'une formule CNF

À une CNF $F$, on associe un graphe biparti, le graphe d'incidence $G = (\mathsf{var}(F) \cup \mathsf{cla}(F), E)$ avec $(x, C) \in E$ si et seulement si $x \in \mathsf{var}(C)$.

### Exemple




**Peut-on exploiter des propriétés structurelles du graphe pour compiler ?**
Tous les résultats connus à ce jour de complexité paramétrée pour #SAT fonctionnent aussi pour la compilation vers les d-DNNF:

| Le graphe d'incidence est | Taille de la d-DNNF | Remarque |
|---|---|---|
| de treewidth $k$ | $4^k\mathsf{poly}(\lvert F\rvert)$ | FPT |
| de cliquewidth $k$ | $\lvert F\rvert^{3k}\mathsf{poly}(\lvert F\rvert)$ | pas FPT sauf si $\mathsf{W}[1] = \mathsf{FPT}$ |
| $\beta$-acyclic | $\mathsf{poly}(\lvert F\rvert)$ | $\beta$-hypertreewidth reste ouvert |

LaTeX TikZposter