TD 7 : Algorithme DPLL

Stéphane Devismes

23 octobre 2026

Exercices à faire absolument : 1, 2 et 3.

Exercice 1 (Réduction, X1603 Andrews)**.** *Soit l'ensemble de clauses :*

$$\{p+q, \overline{p}+r+\overline{q}+p, p+\overline{r}, q+\overline{p}+\overline{q}, q+\overline{r}+p, r+q+\overline{p}+\overline{r}, \overline{r}+q\}.$$

1. *Réduire cet ensemble.*
2. *Indiquer si l'ensemble réduit est ou non satisfaisable.*

□

Exercice 2 (DPLL)**.** *Considérons l'ensemble de clauses suivant :*

$$\{\overline{a}+\overline{b}+\overline{f}, a+b+f, e+\overline{a}, \overline{a}+\overline{b}, \overline{a}+c, d+a+\overline{d}, a+b, \overline{a}+\overline{c}+\overline{d}, d\}.$$

— *Appliquer l'algorithme Algo_DPLL sur cet ensemble de clauses.*
— *Conclure si cet ensemble est satisfaisable ou non.*
— *Donner un modèle ou un contre-modèle obtenu à partir de la trace de l'algorithme.*

Dans l'arbre d'appel vous étiquetterez les étapes comme suit :

— *Suppression des clauses valides, en abrégé VAL.*
— *Réduction, en abrégé RE.*
— *Suppression des clauses ayant des littéraux isolés, en abrégé ELI.*
— *Résolution unitaire, abrégé en RU.*

De plus, noter les affectations effectuées à chaque étape de l'algorithme afin de retrouver facilement le modèle construit.

□

Exercice 3 (DPLL)**.** *Utiliser l'algorithme Algo_DPLL pour déterminer si les ensembles suivants de clauses sont satisfaisables ou insatisfaisables :*

— $\{a+b+c+d+e+f, \overline{a}+b, \overline{b}+a, \overline{c}+d, \overline{d}+c, \overline{b}+\overline{c}, \overline{b}+c, b+\overline{c}, \overline{e}, \overline{f}\}$.
— $\{a+b+c+d+f, \overline{a}+b, \overline{b}+a, \overline{c}+d, \overline{d}+c\}$.
— $\{b+j+\overline{a}, a+j+\overline{b}, b+a+j, a+j, j+b, \overline{b}+\overline{j}, \overline{j}+b, j+s, \overline{s}+\overline{b}\}$.
— $\{a+\overline{c}+d, \overline{b}+c+f, b+\overline{e}+f, \overline{c}+e+\overline{f}, e+f, c+d, \overline{a}, \overline{e}+\overline{f}\}$.

Donner une trace de l'algorithme.

□

Exercice 4 (Preuve)**.** *Prouver le lemme du cours : supprimer des clauses avec un littéral isolé préserve la satisfaisabilité.* □

Exercice 5 (Preuve)**.** *Prouver le lemme du cours : Soit L l'ensemble des littéraux des clauses unitaires de Γ. Soit Θ l'ensemble de clauses ainsi obtenu à partir de Γ*

— *si L comporte deux littéraux complémentaires, alors $\Theta = \{\bot\}$.*
— *sinon Θ est obtenue ainsi*
  • *enlever les clauses qui comportent un élément de L*
  • *à l'intérieur des clauses restantes enlever les littéraux complémentaires des éléments de L*

Γ a un modèle si et seulement si Θ en a un. □

Exercice 6 (DPLL)**.** *Soit Γ l'ensemble de clauses suivant :*

$$\{p+q+\bar{r},\ \bar{p}+\bar{q},\ p+\bar{q},\ r+s,\ p+\bar{r},\ \bar{s}+q\}$$

En utilisant l'algorithme Algo_DPLL, trouvez un modèle de Γ. □

Exercice 7 (DPLL). *Soit* Γ *l'ensemble de clauses suivant :*

$$\{a + b + \bar{d}, d + c, \bar{a} + b, \bar{a} + \bar{c}, \bar{b} + c, \bar{c} + a\}$$

Prouver que cet ensemble de clauses est contradictoire en utilisant l'algorithme Algo_DPLL. □

Exercice 8 (De SAT à 3-SAT,***, BONUS). *SAT est un problème de décision qui peut être énoncé comme suit : « étant donné un ensemble de clauses* Γ*, existe-t-il une assignation qui soit modèle de* Γ *? » 3-SAT est une restriction de ce problème où toutes les clauses de* Γ *ont exactement trois littéraux.*

SAT est un problème NP-complet. Dans cet exercice, nous proposons d'étudier une réduction polynomiale de SAT vers 3-SAT, prouvant ainsi que 3-SAT est aussi NP-complet[1]*.*

Une réduction polynomiale de SAT vers 3-SAT consiste à transformer en temps polynomial un ensemble de clauses Γ *en un ensemble de clauses* Γ' *vérifiant les deux propriétés suivantes :*

(a) Γ' *est uniquement constitué de clauses ayant trois littéraux distincts.*

(b) Γ *a un modèle si et seulement si* Γ' *a un modèle.*

Généralement, de telles réductions vérifient une propriété supplémentaire :

(c) Tout modèle de Γ' *est un modèle de* Γ*.*

Le but de cet exercice est de montrer que la transformation polynomiale présentée ci-dessous vérifie les trois propriétés énoncées ci-dessus. Noter que cette transformation introduit des variables supplémentaires.

Soit $\Gamma = \{c_1, \ldots, c_m\}$ *un ensemble de clauses. La réduction consiste à remplacer toute clause* $c_i = z_1 + \ldots + z_k$ *de* Γ *(ATTENTION :* $z_1, \ldots, z_k$ *sont des littéraux) par un ensemble de clauses* C'_i *construit en fonction de la valeur de* k*, comme suit :*

$k = 1$ **:** $C'_i = \{z_1 + y_1 + y_2,\ z_1 + y_1 + \overline{y_2},\ z_1 + \overline{y_1} + y_2,\ z_1 + \overline{y_1} + \overline{y_2}\}$ *où* y_1 *et* y_2 *sont des variables supplémentaires.*

$k = 2$ **:** $C'_i = \{z_1 + z_2 + y_1,\ z_1 + z_2 + \overline{y_1}\}$ *où* y_1 *est une variable supplémentaire.*

$k = 3$ **:** $C'_i = \{c_i\}$.

$k > 3$ **:** $C'_i = \{z_1 + z_2 + y_1,\ \overline{y_1} + z_3 + y_2,\ \overline{y_2} + z_4 + y_3,\ \overline{y_3} + z_5 + y_4,\ \ldots,\ \overline{y_{k-3}} + z_{k-1} + z_k\}$ *où* $y_1, \ldots, y_{k-3}$ *sont des variables supplémentaires.*

Ainsi, $\Gamma' = \bigcup_{i=1}^{m} C'_i$.

Par construction, la réduction proposée vérifie la propriété (a)*. Les réponses aux questions suivantes permettent de prouver les propriétés* (b) *et* (c) *:*

1. *Montrer (sans table de vérité) que toute assignation donne la même valeur à* c_i *et* C'_i *lorsque* c_i *est constituée d'un seul littéral.*
2. *Montrer (sans table de vérité) que toute assignation donne la même valeur à* c_i *et* C'_i *lorsque* c_i *est constituée de deux littéraux.*
3. *Soit* c_i *une clause de plus de 3 littéraux. Montrer que si* c_i *a un modèle, alors* C'_i *a aussi un modèle.*
4. *Soit* c_i *une clause de plus de 3 littéraux. Montrer que tout modèle de* C'_i *est modèle de* c_i*.*
5. *La réduction proposée maintient uniquement la satisfaisabilité. Montrer qu'elle ne maintient pas le sens.*

□

1. Il faut noter qu'en général, un problème k-SAT n'est pas nécessairement NP-complet. Par exemple, 2-SAT a été prouvé polynomial.