Transformation de Tseitin

Leçons : 916 ; 928

Références : BEN ARI, *Mathematical Logic for computer Science* (p.91) [?] et LESESVRE–MONTAGNON–LE BARBENCHON–PIERRON, *131 développements pour l'oral* (p. 797) [?]

Introduction :

On peut transformer tout formule en une formule en CNF équivalente, mais la transformation est exponentielle. Pour montrer que CnfSat est NP-dur, on veut une transformation polynomiale. La transformation de Tseitin que l'on présente ci-dessous permet de trouver une formule en CNF qui soit équisatisfiable à la formule de départ. On a de l'équisatisfiabilité car les formules ne sont pas sur les mêmes variables propositionnelles donc on ne peut pas espérer avoir équivalence. Ainsi cette transformation qui est polynomiale en la taille de la formule de départ, permet de prouver que CnfSat est NP-dur. Comme il est aussi dans NP, il est alors NP-complet.

Théorème 1. Pour toute formule φ du calcul propositionnel, il existe une formule $T(\varphi)$ en forme normale conjonctive (CNF) de taille linéaire en celle de φ et telle que φ et $T(\varphi)$ sont équisatisfiables.

Démonstration.

Étape 1 : *Notation et énonciation de la formule $T(\varphi)$.*

On appelle *taille d'une formule* φ, notée $|\varphi|$, le nombre de connecteurs logiques de la formule φ.

On suppose que φ ne possède que les connecteurs suivants $\wedge, \vee, \neg$. [1]

On note $SF(\varphi)$ l'ensemble des sous-formules de φ (y compris φ). On note VAR l'ensemble des variables de φ.

Pour toute sous-formule $\psi \in SF(\varphi)$, on définit une nouvelle variable a_ψ (a_ψ veut dire "ψ est vraie").

On pose

$$\boxed{T(\varphi) = a_\varphi \wedge \bigwedge_{\psi \in SF(\varphi) \setminus VAR} t(\psi)}$$ [2]

où l'on définit $t(.)$ comme ci-dessous. La transformation $t(\psi)$ doit donner le sens que l'on voudrait [3] à chaque a_ψ tout en étant en CNF (puisque l'on veut que $T(\varphi)$ soit en CNF) donc on procède comme suit

$t(\neg\psi)$:

$$\begin{aligned} & a_{\neg\psi} \longleftrightarrow \neg a_\psi \\ \sim\ & (a_{\neg\psi} \to \neg a_\psi) \wedge (\neg a_\psi \to a_{\neg\psi}) \end{aligned}$$

1. puisque l'on peut remplacer $\psi_1 \to \psi_2$ par $\neg\psi_1 \vee \psi_2$ et $\psi_1 \leftrightarrow \psi_2$ par $(\neg\psi_1 \vee \psi_2) \wedge (\neg\psi_2 \vee \psi_1)$ ce qui n'augmente la formule d'au plus 5 fois sa taille donc cela restera linéaire en $|\varphi|$
2. si φ est juste une variable propositionnelle, elle est déjà sous forme normale conjonctive
3. a_ψ veut dire "ψ est vraie"

$$\sim \ (\neg a_{\neg\psi} \vee \neg a_\psi) \wedge (a_\psi \vee a_{\neg\psi})$$

On pose donc $t(\neg\psi) = (\neg a_{\neg\psi} \vee \neg a_\psi) \wedge (a_\psi \vee a_{\neg\psi})$ de taille 5.

$t(\psi_1 \wedge \psi_2)$:

$$\begin{aligned} & a_{\psi_1\wedge\psi_2} \longleftrightarrow a_{\psi_1} \wedge a_{\psi_2} \\ \sim \ & (a_{\psi_1\wedge\psi_2} \to a_{\psi_1} \wedge a_{\psi_2}) \wedge (a_{\psi_1} \wedge a_{\psi_2} \to a_{\psi_1\wedge\psi_2}) \\ \sim \ & (\neg a_{\psi_1\wedge\psi_2} \vee a_{\psi_1}) \wedge (\neg a_{\psi_1\wedge\psi_2} \vee a_{\psi_2}) \wedge (\neg a_{\psi_1} \vee \neg a_{\psi_2} \vee a_{\psi_1\wedge\psi_2}) \end{aligned}$$

On pose donc $t(\psi_1 \wedge \psi_2) = (\neg a_{\psi_1\wedge\psi_2} \vee a_{\psi_1}) \wedge (\neg a_{\psi_1\wedge\psi_2} \vee a_{\psi_2}) \wedge (\neg a_{\psi_1} \vee \neg a_{\psi_2} \vee a_{\psi_1\wedge\psi_2})$ de taille 10.

$t(\psi_1 \vee \psi_2)$:

$$\begin{aligned} & a_{\psi_1\vee\psi_2} \longleftrightarrow a_{\psi_1} \vee a_{\psi_2} \\ \sim \ & (a_{\psi_1\vee\psi_2} \to a_{\psi_1} \vee a_{\psi_2}) \wedge (a_{\psi_1} \vee a_{\psi_2} \to a_{\psi_1\vee\psi_2}) \\ \sim \ & (\neg a_{\psi_1\vee\psi_2} \vee a_{\psi_1} \vee a_{\psi_2}) \wedge (\neg a_{\psi_1} \vee a_{\psi_1\vee\psi_2}) \wedge (\neg a_{\psi_2} \vee a_{\psi_1\vee\psi_2}) \end{aligned}$$

On pose donc $t(\psi_1 \vee \psi_2) = (\neg a_{\psi_1\vee\psi_2} \vee a_{\psi_1} \vee a_{\psi_2}) \wedge (\neg a_{\psi_1} \vee a_{\psi_1\vee\psi_2}) \wedge (\neg a_{\psi_2} \vee a_{\psi_1\vee\psi_2})$ de taille 9.

Étape 2 : *Montrons que $|T(\varphi)|$ est linéaire en $|\varphi|$.*

Remarquons que

$$\#SF(\varphi) \leqslant 2|\varphi|$$

car chaque connecteur donne naissance à au plus 2 sous-formules de φ.

De plus, pour tout $\psi \in SF(\varphi)$, on a

$$|t(\psi)| \leqslant 10$$

En conclusion, on a

$$\begin{aligned} |T(\varphi)| &\leqslant |a_\varphi \wedge \bigwedge_{\psi \in SF(\varphi)\setminus VAR} t(\psi)| \\ &\leqslant 1 + |\bigwedge_{\psi \in SF(\varphi)\setminus VAR} t(\psi)| \\ &\leqslant 1 + 11\#SF(\varphi)\setminus VAR\text{[4]} \\ &\leqslant 1 + 11\#SF(\varphi) \\ &\leqslant 1 + 22|\varphi| \end{aligned}$$

Donc $|T(\varphi)|$ est linéaire en taille de φ.

Étape 3 : *Montrons que $T(\varphi)$ et φ sont équisatisfiables.*

$\boxed{\Longrightarrow}$ Supposons que φ est satisfiable, il existe donc une valuation ν telle que $\nu(\varphi) = 1$. On pose la valuation $\tilde{\nu}$ telle que pour tout $\psi \in SF(\varphi)$, $\tilde{\nu}(a_\psi) = \nu(\psi)$. On regarde maintenant $\tilde{\nu}(T(\varphi))$.

$$T(\varphi) = \underbrace{a_\varphi}_{\tilde{\nu}(a_\varphi)=1} \wedge \bigwedge_{\psi_1\wedge\psi_2 \in SF(\varphi)} t(\psi_1 \wedge \psi_2) \wedge \bigwedge_{\psi_1\vee\psi_2 \in SF(\varphi)} t(\psi_1 \vee \psi_2) \wedge \bigwedge_{\neg\psi \in SF(\varphi)} t(\neg\psi)$$

4. on a mis 11 car on compte 1 pour chaque $\wedge$ entre chaque clause

On va regarder juste pour $\psi_1 \wedge \psi_2 \in SF(\varphi)$, on ferait pareil pour les autres.

$$\begin{aligned}
\tilde{\nu}(t(\psi_1 \wedge \psi_2)) &= \tilde{\nu}(a_{\psi_1 \wedge \psi_2} \leftrightarrow a_{\psi_1} \wedge a_{\psi_2}) \\
&= \mathbb{1}_{\tilde{\nu}(a_{\psi_1 \wedge \psi_2}) = \tilde{\nu}(a_{\psi_1} \wedge a_{\psi_2})} \\
&= \mathbb{1}_{\tilde{\nu}(a_{\psi_1 \wedge \psi_2}) = \tilde{\nu}(a_{\psi_1})\tilde{\nu}(a_{\psi_2})} \\
&= \mathbb{1}_{\nu(\psi_1 \wedge \psi_2) = \nu(\psi_1)\nu(\psi_2)} \\
&= 1
\end{aligned}$$

Ainsi la valuation $\tilde{\nu}$ rend vrai chaque clause, donc $\tilde{\nu}(T(\varphi)) = 1$. Ainsi $T(\varphi)$ est satisfiable.

$\boxed{\Longleftarrow}$ Supposons que $T(\varphi)$ est satisfiable, il existe donc une valuation ν telle que $\nu(T(\varphi)) = 1$.

On pose la valuation $\tilde{\nu}$ telle que pour tout $p \in VAR$, $\tilde{\nu}(p) = \nu(a_p)$.

Or $\nu(T(\varphi)) = 1$, donc $\nu(a_\varphi) = 1$ ce qui implique que $\tilde{\nu}(\varphi) = 1$ car la variable associée à une sous-formule est équivalente à cette sous-formule par construction [5].

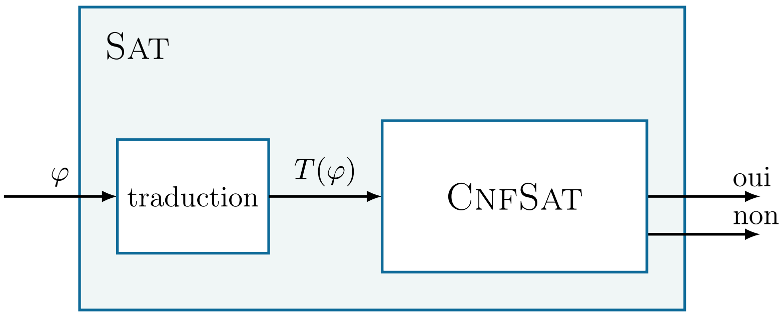
Donc φ est satisfiable. □

Astuces de l'agrégatif :

Il peut être intéressant de faire un exemple pour illustrer cette méthode, mais le temps ne le permet pas forcément. On peut couper l'explication $t(\psi_1 \vee \psi_2)$ pour gagner du temps car cela fonctionne comme $t(\psi_1 \wedge \psi_2)$.

Remarques :

On peut voir ce théorème comme réduction du problème Sat au problème CnfSat [6]



La transformation de Tseitin est donc la traduction de φ en $T(\varphi)$ qui est bien polynomiale. Ainsi CnfSat est NP-dur. De plus comme CnfSat est dans NP pour les mêmes raisons que Sat, on a alors que CnfSat est NP-complet.

Questions possibles :

- Que se passe-t-il si φ est juste une variable propositionnelle ?
- Montrer que l'on a en fait la majoration plus fine

$$|F_\varphi| \leqslant |\varphi| + |V_\varphi|.$$

- Montrer rigoureusement la réduction de CnfSat à 3Sat.
- Pourrait-on trouver une transformation similaire pour les formules sous forme normale disjonctive ? Quelles en seraient les conséquences ?

5. on peut le prouver par induction
6. $\boxed{\text{CnfSat}}$ $\begin{cases} \underline{\text{entrée :}} \text{ une formule } \varphi \text{ du calcul propositionnel sous forme normale conjonctive} \\ \underline{\text{sortie :}} \text{ oui si } \varphi \text{ est satisfiable, non sinon} \end{cases}$

- Montrer qu'une formule sous CNF équivalente à

$$(x_1 \wedge y_1) \vee (x_2 \wedge y_2) \vee \cdots \vee (x_n \wedge y_n)$$

  est de taille exponentielle en la taille de la formule de départ.
- Montrer que le système $\{\leftrightarrow, \neg\}$ n'est pas complet.
- Donner un système complet de connecteurs logiques qui ne contient qu'un seul connecteur logique.