

## Méthode des tableaux : Optimisation pour des formules de la forme alternée

Forme proposition pour accélérer méthode tableau mémoisation des propositions (SQL, dictionnaire, serialisation ?)

### Contents

1	Logique Propositionnelle	1
1.1	Définition	1
1.2	Principe	2
1.3	Optimisation pour les formules de forme alternée	2
2	Logique du 1er ordre (FAIT PLUS TARD)	3

## 1 Logique Propositionnelle

### 1.1 Définition

**Définition 1:** La *logique propositionnelle* est un type de logique où les formules sont obtenus par des variables propositionnelles reliées par des connecteurs.

**Définition 2 (Modèle):** Un *modèle* d'une formule  $\phi$  est une valuation qui rend vraie cette formule. On note l'ensemble des modèles de  $\phi$  par:

$$Mod(\phi) := \{v \in Val | v \models \phi\}$$

$Val$  étant l'ensemble des valuations de  $\phi$  et  $v \models \phi$  signifiant que la valuation  $v$  satisfait  $\phi$

**Définition 3 (Conséquence Logique):** Une formule  $\phi$  est *conséquence logique* d'une formule, notée  $\psi$  si  $Mod(\psi) \subseteq Mod(\phi)$ . On note cela  $\psi \models \phi$

On note  $\mathcal{V}$  un vocabulaire qu'on définit par  $\mathcal{V} := \{p, q, \dots, \neg, \wedge, \vee, \implies, \Leftrightarrow, (, )\}$  et  $\mathcal{V}^*$  l'ensemble de toutes les expressions que l'on peut former avec le vocabulaire  $\mathcal{V}$

**Définition 4 (Système formel):** Un système formel est défini par un tuple  $(\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{R})$ , tel que:

- $\mathcal{V}$  est un ensemble de symboles
- $\mathcal{E}$  est un ensemble d'expressions bien formées dans  $\mathcal{V}^*$
- $\mathcal{A}$  est un ensemble d'axiomes ( $\mathcal{A} \subset \mathcal{E}$ )
- $\mathcal{R}$  est un ensemble de règles de déduction de la forme:  $r_i : f_1, f_2, \dots, f_n \vdash g$  avec  $f_i, g \in \mathcal{E}$

**Définition 5 (Dédution ou preuve):** Soit un système SF, une preuve de  $c_p$  à partir de  $h_1, \dots, h_n$  dans SF qu'on note  $h_1, \dots, h_n \vdash c_p$  toute suite  $c_1, \dots, c_p$  telle que  $\forall i \in \mathcal{N} | 1 \leq i \leq p, c_i$  est un des suivants:

- Un axiome
- Une hypothèse
- Obtenu par application d'une règle  $r_i$  telle que  $c_{i_1}, \dots, c_{i_k} \vdash c_i$  où  $i_1, \dots, i_k < i$

On note  $n \in \mathbb{N}^*$

**Définition 6:** La *méthode des tableaux* consistent à prouver une assertion  $B$  ayant pour hypothèse  $(A_n)$  en montrant que  $\{A_1, \dots, A_n, \neg B\}$  est insatisfaisable (Cela revient à montrer qu'une implication est vraie car sa négation ne peut être vraie) i.e  $((A_1, \dots, A_n) \implies B) \Leftrightarrow Mod(\neg((A_1, \dots, A_n) \implies B)) = \emptyset \Leftrightarrow Mod((A_1, \dots, A_n) \wedge \neg B) = \emptyset$ .

Le procédé consiste donc à séparer les formules logiques complexes en plus petite formule jusqu'à que des paires complémentaires de littéraux ( $a$  et  $\neg a$ ) soit extrait ou qu'on ne peut plus simplifier la formule. Pour cela, on doit définir quelques concepts sur les arbres.

**Definition 7 (Arbre de déduction):** Arbre dont les sommets sont composés de formule, qui sont soit une hypothèse à la racine de l'arbre, soit une formule obtenue par l'application d'une règle sur une formule présente dans la même branche plus proche de la racine.

**Definition 8 (Branche fermée):** Une branche est fermée si elle contient  $\phi$  et  $\neg\phi$

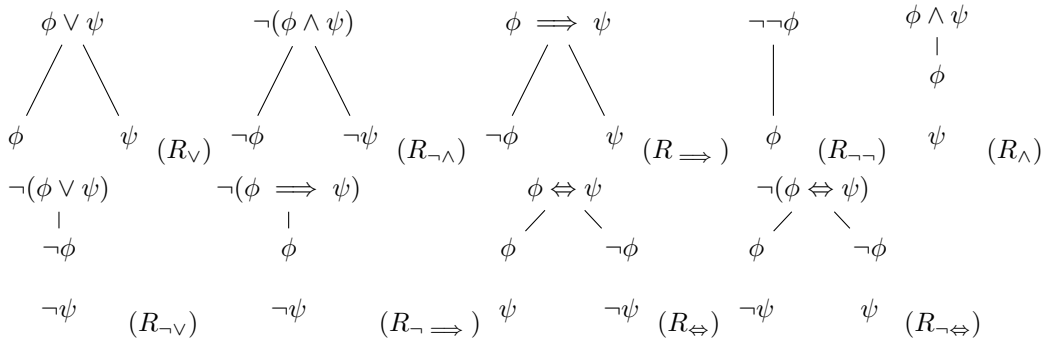
**Definition 9 (Arbre fermé):** Un arbre de déduction est fermé si toutes les branches le sont.

## 1.2 Principe

Pour ainsi en déduire si une formule  $\phi$  est vrai ou non, on adopte des règles qu'on applique sur un arbre de déduction.

- On place  $\neg\phi$  dans la racine de l'arbre.
- On applique les règles ( $R_x$ ) suivantes à chaque branche non fermé de l'arbre
- Si l'arbre est in fine fermé, alors  $\phi$  est vrai

Les règles ( $R_x$ ) dites Smullyan-style sont:



Dans les cas où on trouve dans une même branche  $a$  et  $\neg a$ , on ferme la branche, si l'arbre devient fermé, alors  $\neg\phi$  est forcément faux, donc  $\phi$  est vrai. a1 et (a2 ou a3)

**Proposition 1:** Soit  $\Psi$  un ensemble d'hypothèse et  $\phi$  une formule, on note  $\Psi \vdash \phi$  l'existence d'un arbre fermé pour  $\Psi \cup \{\neg\phi\}$ .

$$\Psi \vdash \phi \Leftrightarrow \Psi \models \phi$$

Autrement dit, appliqué la méthode des tableaux est équivalent à prouver cette formule

La preuve de cette proposition permet de prouver la correction de l'implémentation de base.

## 1.3 Optimisation pour les formules de forme alternée

**Definition 10:** Soit  $n \in \mathbb{N}^*$  ( $\alpha_k$ ) $_{k \in [1, n]}$ , une formule  $\phi$  est de forme alternée ssi

$$\phi := \alpha_1 \wedge (\alpha_2 \vee (\alpha_3 \wedge (\dots (\alpha_n))))$$

On gardera les parenthèses dans la suite pour garder le côté intuitif de cet écriture

On remarque une CNS pour que ce genre de formule soit vrai:

**Proposition 2:**  $\phi$  de forme alternée est vrai ssi:

$$(\exists k \in \mathbb{N}^*, \alpha_{2k-2} \text{ OU } n \text{ impair} \implies \alpha_n \text{ avec } k = \frac{n-1}{2}) \text{ ET } \forall i \in [1, k], \alpha_{2i-1}$$

**Preuve 1:** (  $\Leftarrow$  ) Immédiat

(  $\Rightarrow$  ) Supposons  $\phi$  de forme alternée vrai:  $\alpha_1$  est forcément vrai, deux possibilités : soit  $\alpha_2$  est vrai, soit  $\alpha_3 \wedge (\dots)$  est vrai. Dans le deuxième cas, on répète le raisonnement sur  $\alpha_3 \wedge (\dots)$  qui est bien de forme alternée.

Deux cas de figure:

- On arrête donc le processus dès que un littéral indexé par un nombre pair est vrai. Dans ce cas là, tout les indexés impair précédents sont aussi vrai.
- Si aucun littéral pair est vrai, alors  $n$  est impair sans quoi  $\phi$  n'est pas vrai car  $\alpha_n$  doit être vrai. On en déduit que tout les littéraux impairs sont vraies, donc la formule est vrai.

Mais on cherche une CNS de satisfaisabilité.

## 2 Logique du 1er ordre (FAIT PLUS TARD)

La logique propositionnelle étant mathématiquement limité, on se propose l'utilisation de la logique du 1er ordre. Cela nous permettra ainsi d'étudier Zenon, un prouveur d'automatique de theoreme.

**Definition 11:** La *logique du 1er ordre* est un type de logique qui en plus des éléments de la logique propositionnelle permet l'utilisation de quantificateurs et de *termes*.

**Definition 12:** Soit un ensemble infini de variables  $X = \{x, y, x_1, x_2, \dots\}$  et un ensemble  $\mathcal{F} = \{c, f, g, \dots\}$  de symboles de fonction (autrement appelé signature). On rappelle que l'arité d'un symbole est son nombre d'argument. Les termes sont définis par induction:

- $\forall x \in X, x$  est un terme
- Tout symbole d'arité 0 (les constantes) est un terme
- $f(t_1, \dots, t_n)$  est un terme ssi  $f$  est un symbole d'arité  $n$  et  $t_1, \dots, t_n$  sont des termes

On note  $\mathcal{T}(\mathcal{F}, X)$  l'ensemble des termes sur  $\mathcal{F}$  et  $X$ .