Équivalence entre Logique Monadique du Second Ordre et Automates

Marc Bagnol

1 Logique du Second Ordre (SO)

En logique du premier ordre (FO), les quantifications ne peuvent porter que sur les éléments de l'univers M. La logique du second ordre étend FO en introduisant la possibilité de quantifier sur les parties de M^k, $k \geq 1$.

Définition 1. *SO*
On suppose fixé un vocabulaire σ.
On dispose, en plus des symboles habituels, de symboles de variables du second ordre X_1^k,...,X_n^k,... où k désigne l'arité de la variable.

Formules et variables libres : on donne ici les constructions de formules qui diffèrent de FO et les règles de liberté des variables de ces formules :
• Aux formules atomiques habituelles on ajoute celles de la forme : $X^k(t_1, ..., t_k)$ où les t_i sont des termes de σ et X^k une variable k-aire du second ordre. Les variables libres de cette formule sont X^k et celles des t_i.
• Les connecteurs logiques ($\vee$, $\wedge$, $\neg$) permettent les mêmes constructions que pour FO et ne changent pas la liberté des variables.
• La quantification sur les variables du premier ordre est inchangée, on a de plus : *($\overrightarrow{x}$ représente un uplet de variables)*
Si $F(\overrightarrow{x}, Y, \overrightarrow{X})$ est une formule, alors $\exists Y F(\overrightarrow{x}, Y, \overrightarrow{X})$ et $\forall Y F(\overrightarrow{x}, Y, \overrightarrow{X})$ sont des formules dans lesquelles Y est liée et où la liberté des autres variables est inchangée.

Exemple . $\forall X^2(\exists y X^2(x, y)) \Rightarrow Y^1(x)$ est une formule de SO dans laquelle X^2 et y sont liées, Y^1 et x sont libres.

Sémantique : définissons alors l'interprétation de ces nouvelles formules. (les interprétations habituelles de FO sont inchangées)
Étant donné une σ-structure $\mathfrak{M}$ on a :
• Si $\phi(X^k, \overrightarrow{x})$ est $X^k(t_1, ..., t_k)$ les t_i étant des termes de σ avec $\overrightarrow{x}$ pour variables libres, alors $\mathfrak{M} \models \phi(B, \overrightarrow{b})$ si et seulement si $(t_1(\overrightarrow{b}), ..., t_k(\overrightarrow{b})) \in B$.
• Si $\phi(\overrightarrow{x}, \overrightarrow{X})$ est $\forall Y \varphi(\overrightarrow{x}, Y, \overrightarrow{X})$ (respectivement $\forall Y \varphi(\overrightarrow{x}, Y, \overrightarrow{X})$) alors

$\mathfrak{M} \models \phi(B, \overrightarrow{b})$ si et seulement si il existe (respectivement pour tout) Y on a $\mathfrak{M} \models \varphi(B, Y, \overrightarrow{b})$.
Exemple . $F(X^1, Y^1) = \forall Z^2 \, \exists x \exists y \, (X^1(x) \wedge Y^1(y) \wedge Z^2(x, y))$ *est satisfaite si pour toute relation binaire* Z^2 *il existe* x *et* y *tels que* $x \in X^1$ *et* $y \in Y^1$ *et* $(x, y) \in Z^2$.

Définition 2. *MSO*
La logique monadique du second ordre (MSO) est une restriction de SO dans laquelle on se limite aux variables unaires du second ordre.
Autrement dit, on se limite à la quantification sur des sous-ensembles de l'univers, ce qui empêche notamment de quantifier sur des fonctions (vues comme des cas particuliers de relations binaires).

Définition 3. $\exists SO$
La logique existentielle du second ordre ∃SO est une restriction de SO aux formules de la forme : $\exists X_1 ... \exists X_n \varphi$ où φ est sans quantification du second ordre.
On définit de même ∀SO, ∃MSO et ∀MSO.

2 Équivalence MSO-Automates

On supposera, pour alléger les notations, que l'on travaille sur l'alphabet $\Sigma = \{0, 1\}$*, l'extension à n'importe quel alphabet ne posant pas de problème.*

On va s'interesser à la définition de langages par des formules logiques. Nous allons voir dans cette partie que les langages MSO-définissables sont exactement les langages rationnels. Plus précisément :

Définition 4. *Langage défini par un énoncé.*
On utilise le vocabulaire $\sigma = \langle \leq, P_0, P_1 \rangle$.
À un mot $s = s_1 ... s_n$ sur Σ on peut associer la σ-structure :
$M_s = \langle \{1, ..., n\}, \leq, P_0, P_1 \rangle$ où $\leq$ est la relation d'ordre usuelle sur les entiers, P_0 l'ensemble des i tels que $s_i = 0$ et P_1 l'ensemble des i tels que $s_i = 1$.
Le langage défini par l'énoncé ϕ, noté $L(\phi)$, est donné par :

$$s \in L(\phi) \underset{déf.}{\iff} M_s \models \phi$$

Théorème 1. ***(Büchi)***
Un langage est MSO-définissable si et seulement si il est rationnel.

Démonstration. Montrons d'abord que tout langage rationnel est MSO-définissable :

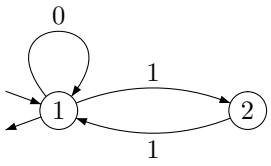
Soit L rationnel, reconnu par un automate $\mathcal{A} = (Q, I, F, \Delta)$. On supposera que $Q = \{1, ..., m\}$.

On va construire une formule ϕ :

$$\exists X_1 ... \exists X_m \, \varphi_{part} \wedge \varphi_{init} \wedge \varphi_{trans} \wedge \varphi_{fin} \text{ qui définit } L.$$

L'idée est de dire qu'un mot $s = s_1...s_n$ est reconnu par $\mathcal{A}$ si et seulement si il existe une lecture de s sur $\mathcal{A}$, c'est à dire si à toute position i dans s on peut associer l'état k dans lequel est $\mathcal{A}$ avant de lire s_i.
Ainsi, les X_k vont représenter l'ensemble des i auxquels on associe l'état k.

Exemple . L'automate



reconnait 0110 *avec* $X_1 = \{1, 2, 4\}$ *et* $X_2 = \{3\}$.

• φ_{part} assure que $X_1, ..., X_m$ forme une partition de l'univers de M_s :

$$\forall x \, (\bigvee_{i=1}^{m} (X_i(x) \bigwedge_{j \neq i} \neg X_j(x)) \,)$$

• φ_{init} vérifie que l'on est bien dans l'état initial avant de lire la première lettre :

$$\forall x \, (\forall y \; x \leq y) \Rightarrow \bigvee_{i \in I} X_i(x)$$

• φ_{trans} contrôle la validité des transitions lors de la lecture :

$$\forall x \forall y (x \prec y) \Rightarrow (\bigvee_{(i,a,j) \in \Delta} (X_i(x) \wedge P_a(x) \wedge X_j(y)) \,)$$

(où $x \prec y$ est une abréviation de "y est le successeur de x")

• φ_{fin} vérifie que la lecture termine sur un état final :

$$\forall x \, (\forall y \; y \leq x) \Rightarrow \bigvee_{\substack{(i,a,f) \in \Delta \\ f \in F}} X_i(x)$$

La satisfaction de ϕ dans M_s équivaut bien à l'existence d'une lecture de s sur $\mathcal{A}$ et donc $L(\phi) = L(\mathcal{A}) = L$.

Pour la réciproque, on commence par étendre la notion de définition de langage aux formules :

Étant donnés un mot $s = s_1..s_n$, des entiers $x_1, ..., x_k \in \{1, ...n\}$ et des ensembles $X_1, ..., X_r \in \mathcal{P}(\{1, ..., n\})$, on peut associer au k+r+1-uplet $(s, x_1, ..., x_k, X_1, ..., X_r)$ le mot $(s, \overline{x_1}, ..., \overline{x_k}, \overline{X_1}, ..., \overline{X_r})$ sur $(\Sigma^{k+r+1})^*$, où la $j^{ème}$ lettre de $\overline{x_i}$ est 1 si et seulement si $i = j$ et la $j^{ème}$ lettre de $\overline{X_i}$ est 1 si et seulement si $j \in X_i$.
Exemple . *à* $(0101, 3, \{1, 2\})$ *on associe* $(0101, 0010, 1100)$.

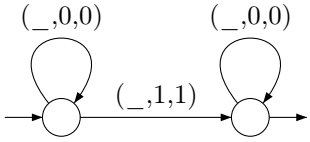
On peut alors définir pour $\phi(\overrightarrow{x}, \overrightarrow{X})$ le langage $L(\phi)$:

$$(s, \overrightarrow{\overline{b}}, \overrightarrow{\overline{B}}) \in L(\phi) \underset{déf.}{\iff} M_s \models \phi(\overrightarrow{b}, \overrightarrow{B}).$$

On va maintenant prouver par induction que le langage défini par une formule (et donc en particulier par un énoncé) est toujours rationnel.

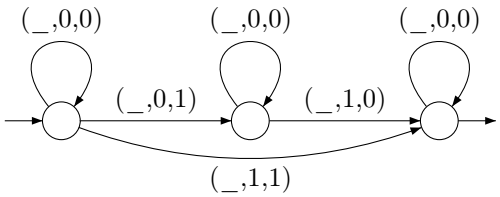
• Formules atomiques :

- Si $\phi(x, y)$ est $(x = y)$, $L(\phi)$ est reconnu par l'automate :

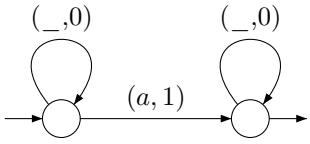


où $(\_, a)$ *signifie que l'on a toutes les transitions* (b, a), $b \in \Sigma$

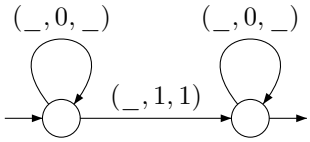
- Si $\phi(x, y)$ est $(x \leq y)$, $L(\phi)$ est reconnu par :



- Si $\phi(x)$ est $P_a(x)$ pour $a \in \Sigma$, $L(\phi)$ est reconnu par :



- Si $\phi(x, X)$ est $X(x)$, $L(\phi)$ est reconnu par :



• Connecteurs logiques : soient ϕ et ψ deux formules. On suppose qu'elles ont le même ensemble de variables libres (ce qui ne pose pas de problème quitte à modifier légèrement les automates correspondants afin que la valeur prise par les variables n'apparaissant pas effectivement dans les formules soit indifférente). Dans ce cas : $L(\phi \vee \psi) = L(\phi) \cup L(\psi)$ et $L(\phi \wedge \psi) = L(\phi) \cap L(\psi)$.
De plus, on a $L(\neg\phi) = \complement(L(\phi))$.

• Quantificateurs : si ϕ est $\exists y\, \varphi(\overrightarrow{x}, y, \overrightarrow{X})$, soit $\mathcal{A} = (Q, q_1, F, \Delta)$ tel que $L(\mathcal{A}) = L(\varphi)$. On pose $\mathcal{A}' = (Q, q_1, F, \Delta')$, avec Δ' défini par $(p, (\overrightarrow{b}, \overrightarrow{B}), q) \in \Delta'$ si et seulement si il existe $y \in \{0, 1\}$ tel que $(p, (\overrightarrow{b}, y, \overrightarrow{B}), q) \in \Delta$. On a alors $L(\mathcal{A}') = L(\phi)$.
Si ϕ est $\forall y\, \varphi(\overrightarrow{x}, y, \overrightarrow{X})$ on peut utiliser l'équivalence $\forall x F \iff \neg\exists x \neg F$ et appliquer la construction précédente pour obtenir un automate qui reconnait $L(\phi)$.
On peut faire de même pour les quantification sur les variables du second ordre.

Finalement, les langages définis par les formules atomiques sont rationnels, l'ensemble des formules définissant un langage rationnel est stable par connection logique et quantification. Ceci prouve que le langage défini par n'importe quelle formule de MSO est rationnel. □

Corollaire 1. *Décidabilité*

L'aspect algorithmique de la deuxième partie de la preuve montre que l'on peut décider si un énoncé dans $\langle \leq, P_0, P_1 \rangle$ sur les ensembles finis, totalement ordonnés par $\leq$ et partitionnés par P_0, P_1 est vrai (en déterminant si le langage qu'il définit est Σ^*).
Toutefois, la complexité de ce problème est "non-élémentaire" (*i.e.* elle ne peut être bornée par une composée d'exponentielles), ce qui compromet une mise en oeuvre efficace de l'algorithme.

Corollaire 2. *Pour la définition de langages, ∃MSO est équivalente à MSO.*

En effet, toute formule de MSO définit un langage reconnu par un automate. Ce langage peut être également défini par la formule de ∃MSO donnée dans la première partie de la preuve.

Citons en conclusion deux autres théorème reliant la logique et les langages formels :

Théorème 2. ***(Mc Naughton-Papert)***
Un langage est FO-définissable si et seulement si il est sans étoile.

Théorème 3. ***(Fagin)***
Un langage est ∃SO-définissable si et seulement si il est dans NP.

En particulier, on voit que FO est trop faible pour définir tous les langages rationnels (le langage $(00)^*$ ne peut être décrit comme un langage sans étoile), et que SO peut définir des langages non reconnus par des automates. Cela confirme que MSO est le bon niveau d'expressivité pour définir les langages rationnels.