Automates d'arbres

http://list.mpri.master.univ-paris7.fr/wws/info/cours-1-18

cours-1-18@mpri.master.univ-paris7.fr

Cours: Florent Jacquemard
TD: Ștefan Ciobâcă

INRIA Saclay & LSV (CNRS-ENS Cachan)

florent.jacquemard@inria.fr
ciobaca@lsv.ens-cachan.fr

15 janvier 2010

Arbres finis orientés **non**-ordonnés étiquettés
par des symboles de fonction sans arité

Unranked unordered labeled trees

Presburger Automata (PA)

- previous lecture : unranked ordered trees
  - XML documents
  - hedge automata (HA), $\mathrm{MSO}(\rightarrow, \downarrow)$.
  - cf. TATA chap. 8.
- this lecture : unranked unordered trees
  - web data
  - Presburger automata (PA), PMSO.
  - cf. article "Numerical Document Queries",
    Helmut Seidl, Thomas Schwentick, Anca Muscholl, 2003.

Unranked Unordered Trees

- Σ is a finite alphabet.

$$\begin{array}{rcl} \text{tree} & := & a(\text{multiset}) \quad (a \in \Sigma) \\ \text{multiset} & := & \{\text{tree}, \ldots, \text{tree}\} \end{array}$$

- $a\big(\{t_1, \ldots, t_n\}\big)$ is denoted $a(t_1, \ldots, t_n)$.
- rem : the multiset can be empty. $a(\emptyset)$ is denoted a.
- The set of unranked unordered trees over Σ is denoted $\mathcal{U}_2(\Sigma)$.

Exemples de langages de $\mathcal{U}_2(\Sigma)$

- $\Sigma = \{a, b\}$.
- termes de $\mathcal{U}_2(\Sigma)$
  - de hauteur 1,
  - avec un b à la racine,
  - des feuilles qui sont a ou b :

  $i.$ avec un nombre pair de a (HA),

  $ii.$ avec le même nombre pair de a que de b ($\neg$ HA).

Arithmétique de Presburger

Formules de Presburger :

$$
\begin{array}{lcll}
\text{term} & ::= & & x \quad \text{(variable du premier ordre)} \\
 & & | & n \quad \text{(entier naturel)} \\
 & & | & \text{term} + \text{term} \\
\text{form} & ::= & & \text{term} = \text{term} \\
 & & | & \neg\text{form} \mid \text{form} \vee \text{form} \mid \text{form} \wedge \text{form} \\
 & & | & \forall x\, \text{form} \mid \exists x\, \text{form}
\end{array}
$$

Interprétation dans le domaine des entiers naturels.
$(n_1, \ldots, n_p) \models \phi(x_1, \ldots, x_p)$ ($x_1, \ldots, x_p$ variables libres)
ssi $\phi(n_1, \ldots, n_p)$ s'évalue à *vrai*.

Arithmétique de Presburger

- notation : termes $n\,x$ pour $\underbrace{x + \ldots + x}_{n}$,
- les entiers peuvent être limités à 0 et 1,
- les atomes peuvent être limités à $x = n$ et $x = y + z$.

Exemples :

- $x \leq y \equiv \exists x'\, y = x + x'$.
- $odd(x) \equiv \exists y\, x = y + y + 1$.

Arithmétique de Presburger

Theorem : Arithmétique de Presburger

L'arithmétique de Presburger est décidable en espace doublement exponentiel.

- Borne inférieure 2-EXPTIME, sup. 3-EXPTIME.
- NP-complet pour le fragment existentiel.

Arithmétique de Presburger

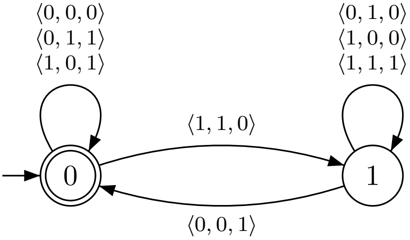
Décidabilité de l'arithmétique de Presburger.

On peut associer à une formule $\phi(x_1, \dots, x_p)$ un automate fini sur l'alphabet $\{0,1\}^p$ reconnaissant l'ensemble des $\langle b_{1,1}, \dots, b_{p,1} \rangle \dots \langle b_{1,k}, \dots, b_{p,k} \rangle$ tels que $b_{1,1} \dots b_{1,k}$, ..., $b_{p,1} \dots b_{p,k}$ sont les représentations en binaire d'entiers $n_1, \dots, n_p$ satisfaisant ϕ.

Donc on peut décider si il existe $n_1, \dots, n_p$ tels que $(n_1, \dots, n_p) \models \phi(x_1, \dots, x_p)$.

Arithmétique de Presburger et automates

automate fini pour $x_1 + x_2 = x_3$



Ensemble semi-linéaires

Definition :

- Un ensemble *linéaire* est un sous ensemble de $\mathbb{N}^p$ de la forme $\left\{\overline{v_0} + \overline{v_1} + \ldots + \overline{v_m} \;\middle|\; m \geq 0, \overline{v_1}, \ldots, \overline{v_m} \in B\right\}$, pour $\overline{v_0} \in \mathbb{N}^p$ et $B \subset \mathbb{N}^p$ fini donnés.
- Un ensemble *semi-linéaire* est une union finie d'ensembles linéaires.

modèles de formules Presburger $\equiv$ semi-linéaires.

Projection et théorème de Parikh

$\Sigma = \{a_1, \ldots, a_p\}$.

Definition : Projection de Parikh

La projection de Parikh d'un mot $w \in \Sigma^*$ est le tuple $\#(w) := (m_1, \ldots, m_p)$ où m_i $_{(i \leq p)}$ est le nombre d'occurrences de a_i dans w.

Pour un ensemble $L \subseteq \Sigma^*$, on note $\#(L) := \{\#(w) \mid w \in L\}$.

Theorem :

Pour tout langage $L \subseteq \Sigma^*$ context-free, il existe une formule de Presburger $\phi(x_1, \ldots, x_p)$ telle que $\#(L) := \{(n_1, \ldots, n_p) \models \phi(x_1, \ldots, x_p)\}$.

Quand L est régulier, la formule de Presburger est calculée en temps linéaire (dans la taille du NFA définissant L).

Inversement, étant donnée une formule de Presburger $\phi(x_1, \ldots, x_p)$, on peut construire un NFA $\mathcal{A}$ tel que $\#\big(L(\mathcal{A})\big) = \{(n_1, \ldots, n_k) \models \phi(x_1, \ldots, x_k)\}$.

Automates de Presburger (PA)

Definition : Automates de Presburger

Un *automate de Presburger* (PA) sur un alphabet Σ est un tuple $\mathcal{A} = (\Sigma, Q, Q^{\mathsf{f}}, \Delta)$ où $Q = \{q_1, \ldots, q_p\}$ est un ensemble fini d'*états*, $Q^{\mathsf{f}} \subseteq Q$ est le sous-ensemble des états finaux et Δ est un ensemble de règles de transition de la forme : $a(\phi) \to q$ avec $a \in \Sigma$, $q \in Q$, et $\phi = \phi(x_1, \ldots, x_p)$ est une formule de Presburger avec une variable libre pour chaque état.

Le langage du $\mathcal{A}$ dans l'état $q \in Q$, noté $L(\mathcal{A}, q)$, est le plus petit sous-ensemble de $\mathcal{U}_2(\Sigma)$ de termes $a(t_1, \ldots, t_n) \in \mathcal{U}_2(\Sigma)$ tels que

- il existe $i_1, \ldots, i_n \leq p$ tels que pour tout $j \leq n$, $t_j \in L(\mathcal{A}, q_{i_j})$,
- il existe une transition $a(\phi) \to q \in \Delta$ telle que $\#(q_{i_1}, \ldots, q_{i_n}) \models \phi(x_1, \ldots, x_p)$.

Le langage de $\mathcal{A}$ est $L(\mathcal{A}) = \bigcup_{q \in Q_{\mathsf{f}}} L(\mathcal{A}, q)$.

PA : exemple 1

$\Sigma = \{a, b, f\}$.
Ensemble des arbres de $\mathcal{U}_2(\Sigma)$ où tous les a et b étiquettent des feuilles :

$$\mathcal{A} = \big(\{q\}, \{q\}, \{a(x_q = 0) \rightarrow q, b(x_q = 0) \rightarrow q, f(true) \rightarrow q\}\big)$$

PA : exemple 2

$\Sigma = \{a, b, c\}$.
Ensemble des arbres de $\mathcal{U}_2(\Sigma)$ avec le même nombre de a et de b sous chaque noeud.

$$\mathcal{A} = \big(\{q_a, q_b, q\}, \{q_a, q_b, q\}, \{a(\phi) \to q_a, b(\phi) \to q_b, c(\phi) \to q\}\big)$$

avec $\phi \equiv x_{q_a} = x_{q_b}$.

PA : exemple 3

$\Sigma = \{a, b\}$.
Ensemble des arbres de $\mathcal{U}_2(\Sigma)$ dont tout noeud interne est étiqueté par a et tout noeud étiqueté par a a au moins autant de fils sans b que de fils contenant b.

$Q = Q_{\mathsf{f}} = \{q_a, q_b\}$.
L'état q_b accepte les arbres contenant un b et q_a les autres.

$$\Delta = \left\{ \begin{array}{lcl} a(x_{q_a} \geq x_{q_b} = 0) & \rightarrow & q_a \end{array} \quad \begin{array}{lcl} a(x_{q_a} \geq x_{q_b} > 0) & \rightarrow & q_b \\ b(x_{q_a} = x_{q_b} = 0) & \rightarrow & q_b \end{array} \right\}$$

Automates de Presburger normalisés

Definition :

Un PA $\mathcal{A} = (\Sigma, Q, Q^{\mathsf{f}}, \Delta)$ sur Σ est dit *normalisé* si pour tous $a \in \Sigma$ et $q \in Q$, il y a au plus une transition de la forme $a(\phi) \to q$ dans Δ.

Quand $\mathcal{A}$ est normalisé, on note $a(\phi_{a,q}) \to q$ l'unique transition avec a et q.

Proposition :

Pour tout PA $\mathcal{A}$, il existe un PA normalisé $\mathcal{A}_n$ reconnaissant le même langage.

La taille de $\mathcal{A}_n$ est linéaire dans la taille de $\mathcal{A}$.

Automates de Presburger complets et déterministes

Definition : PA déterministes

Un PA $\mathcal{A} = (\Sigma, Q, Q^{\mathsf{f}}, \Delta)$ sur Σ est *déterministe* si pour tout $a \in \Sigma$ et tous $q_{i_1}, \ldots, q_{i_n} \in Q^*$, si $a(\phi) \to q \in \Delta$ et $a(\phi') \to q' \in \Delta$ sont telles que $\#(q_{i_1}, \ldots, q_{i_n}) \models \phi$ et $\#(q_{i_1}, \ldots, q_{i_n}) \models \phi'$, alors $q = q'$.

Lemma :

Le déterminisme de PA est décidable.

Definition : PA complets

Un PA $\mathcal{A} = (\Sigma, Q, Q^{\mathsf{f}}, \Delta)$ sur Σ est *complet* si pour tout $a \in \Sigma$ et tous $q_{i_1}, \ldots, q_{i_n} \in Q^*$ il existe au moins une transition $a(\phi) \to q \in \Delta$ telle que $\#(q_{i_1}, \ldots, q_{i_n}) \models \phi$.

Complétion de PA

Proposition : Complétion

Pour tout PA $\mathcal{A}$, il existe un PA complet $\mathcal{A}_c$ reconnaissant le même langage.

La taille de $\mathcal{A}_c$ est linéaire dans la taille de $\mathcal{A}$.

pr.: Soit $\mathcal{A} = (Q, Q_{\mathsf{f}}, \Delta)$ un PA supposé normalisé.
On ajoute l'état $q_\perp$: $\mathcal{A}_c = \big(Q \cup \{q_\perp\}, Q_{\mathsf{f}}, \Delta_c\big)$ avec

$$\begin{array}{rrcl} \Delta_c := & a\big(\phi_{a,q} \wedge x_{q_\perp} = 0\big) & \rightarrow & q \text{ t.q. } a(\phi_{a,q}) \rightarrow q \in \Delta \\ \cup & a\big(\bigwedge_{q \in Q} \neg\phi_{a,q} \vee x_{q_\perp} > 0\big) & \rightarrow & q_\perp \end{array}$$

Déterminisation de PA

Proposition : Déterminisation

Pour tout PA $\mathcal{A}$, il existe un PA déterministe $\mathcal{A}_d$ reconnaissant le même langage.

La taille de $\mathcal{A}_d$ est exponentielle dans la taille de $\mathcal{A}$ (borne inférieure).

pr.: Soit $\mathcal{A} = (Q, Q_{\mathsf{f}}, \Delta)$, normalisé avec $Q = \{q_1, \ldots, q_b\}$.

$$\mathcal{A}_d = \left(2^Q, \{S \subseteq Q \mid S \cap Q_{\mathsf{f}} \neq \emptyset\}, \Delta_d\right)$$

Formules de Presburger dans Δ_d : une variable libre x_S pour chaque $S \subseteq Q$.

ε-transitions et élimination

Remark :

PA avec transitions $q \to q' \equiv$ PA.

PA : opérations Booléennes

Proposition :

La classe des langages de PA est fermée par union, intersection et complément.

- $\cup$ union disjointe (linéaire) ou produit (quadratique, préserve déterminisme).
- $\cap$ loi de De Morgan ou produit (quadratique).
- $\neg$ complète, déterminise, inverse états finaux (exponentiel).

PA : décision de l'appartenance

Proposition : $\in$

Le problème de l'appartenance est décidable pour les PA.

- en temps polynomial pour les DPA,
- NP-complet pour les NPA.

PA : décision du vide

Proposition : $\emptyset$

Le problème du vide est décidable en temps polynomial pour les PA.

pr.: marquage d'états, construction de $M_i \subseteq Q$.

- initialement, $M_0 = \emptyset$.
- à chaque étape, si pour $q \in Q \setminus M_i$, $\displaystyle\bigwedge_{p \in Q \setminus M_i} x_p = 0 \wedge \bigvee_{a \in \Sigma} \phi_{a,q}$ est satisfiable, alors $M_{i+1} := M_i \cup \{q\}$.

PA : décision de l'inclusion, équivalence

Proposition : $\subseteq$, $\equiv$

Les problèmes de l'inclusion et équivalence sont décidables pour les PA.

pr.: $L(\mathcal{A}_1) \subseteq L(\mathcal{A}_2)$ ssi $L(\mathcal{A}_1) \cap (\mathcal{U}_2(\Sigma) \setminus L(\mathcal{A}_2)) = \emptyset$.

Correspondance entre PA, HA et AC-TA

Il y a correspondance entre :

- les langages de PA
- les langages de CF-HA dont les langages (CF) de transitions sont clos par permutation
- la classe des clôtures de langages d'arbres réguliers modulo AC

Rappel : CF-HA

Definition : CF-HA

CF-HA : tuple $(\Sigma, Q, Q_\mathsf{f}, \Delta)$ comme un HA dont les transitions sont de la forme $a(L) \rightarrow q$ avec $a \in \Sigma$, $q \in Q$, et $L \subseteq Q^*$ est hors contexte (*context-free*).

Langages réguliers modulo AC (AC-TA)

Signature $\Sigma = \Sigma_\emptyset \uplus \{a\}$.
Le symbole a est binaires et suit les axiomes d'associativité et commutativité :

$$a(x_1, a(x_2, x_3)) = a(a(x_1, x_2), x_3) \tag{A}$$

$$a(x_1, x_2)) = a(x_2, x_1) \tag{C}$$

Pour un TA $\mathcal{B}$ sur Σ, on note
$\mathsf{AC}\big(L(\mathcal{B})\big) := \big\{t \in \mathcal{T}(\Sigma) \;\big|\; t =_{\mathsf{AC}} s \in L(\mathcal{B})\big\}$ (langage d'AC-TA)

Proposition :

- la classe des langages d'arbres réguliers est strictement incluse dans la classe des langages d'AC-TA.
- La classe des langages d'AC-TA est close par opérations Booléennes.

Correspondences $\mathcal{T}(\Sigma) \leftrightarrow \mathcal{U}_2(\Sigma)$

On suppose $\Sigma_{\mathsf{AC}} = \{a\}$.

$$
\begin{array}{rrcl}
\mathit{flat}: & \mathcal{T}(\Sigma) & \to & \mathcal{U}_2(\Sigma) \\
\mathit{hflat}: & \mathcal{T}(\Sigma)^* & \to & \mathcal{H}(\Sigma) \\
\mathit{flat}^{-1}: & \mathcal{U}_2(\Sigma) & \to & \mathcal{T}(\Sigma)
\end{array}
$$

Définitions ($g \in \Sigma_n \setminus \{a\}$) :

$$
\begin{array}{rcl}
\mathit{flat}\big(g(t_1,\ldots,t_n)\big) & = & g(\mathit{flat}(t_1)\ldots\mathit{flat}(t_n)) \\
\mathit{flat}\big(a(t_1,t_2)\big) & = & a(\mathit{hflat}(t_1\,t_2)) \\[1ex]
\mathit{hflat}\big(g(s_1,\ldots,s_n)\,t_2\ldots t_m\big) & = & \mathit{flat}\big(g(s_1,\ldots,s_n)\big)\,\mathit{hflat}(t_2\ldots t_m) \\
\mathit{hflat}\big(a(s_1,s_2)\,t_2\ldots t_m\big) & = & \mathit{hflat}(s_1 s_2\,t_2\ldots t_m) \\[1ex]
\mathit{flat}^{-1}\big(g(t_1\ldots t_n)\big) & = & g(\mathit{flat}^{-1}(t_1),\ldots,\mathit{flat}^{-1}(t_n)) \\
\mathit{flat}^{-1}\big(a(t_1\ldots t_m)\big) & = & a(t_1, a(t_2,\ldots,a(t_{m-1},t_m))) \\
 & & (m \geq 2)
\end{array}
$$

Logique monadique faible du second ordre

PMSO

Logique monadique du second ordre de Presburger

Syntaxe des formules de PMSO.

- variables du premier ordre $x\dots$
- variables du second ordre $X\dots$
- $$\begin{array}{rcl}
\text{form} & ::= & x = y \;\big|\; x \downarrow y \;\big|\; a(x) \;\big|\; x \in X \;\big|\; x/\text{pres} \quad (a \in \Sigma) \\
& \big| & \text{form} \wedge \text{form} \;\big|\; \text{form} \vee \text{form} \;\big|\; \neg\text{form} \\
& \big| & \exists x\, \text{form} \;\big|\; \exists X\, \text{form} \;\big|\; \forall x\, \text{form} \;\big|\; \forall X\, \text{form} \\
\text{pres} & ::= & \text{term} = \text{term} \\
& \big| & \neg\text{pres} \;\big|\; \text{pres} \vee \text{pres} \;\big|\; \text{pres} \wedge \text{pres} \\
& \big| & \forall z\, \text{pres} \;\big|\; \exists z\, \text{pres} \\
\text{term} & ::= & z \quad \text{(variable entière du premier ordre)} \\
& \big| & n \quad \text{(entier naturel)} \\
& \big| & [X] \quad (X \text{ var. du second ordre}) \\
& \big| & \text{term} + \text{term}
\end{array}$$

formules pres telles que les variables z sont liées.

rem. pas d'atomes $x \to y$ comme pour les arbres ordonnés.

Logique monadique du second ordre de Presburger

Sémantique de PMSO.

- domaine : ensemble $\|t\|$ des sommets d'un arbre $t \in \mathcal{U}_2(\Sigma)$,
- σ : variables du premier ordre $\to \|t\|$,
- ρ : variables du second ordre $\to 2^{\|t\|}$,
- $t, \sigma, \rho \models x = y$ ssi $\sigma(x)$ identique à $\sigma(y)$,
- $t, \sigma, \rho \models x \downarrow y$ ssi $\sigma(x)$ est le père de $\sigma(y)$ dans t,
- $t, \sigma, \rho \models a(x)$ ssi $\sigma(x)$ étiqueté par a dans t,
- $t, \sigma, \rho \models x \in X$ ssi $\sigma(x) \in \rho(X)$,
- $t, \sigma, \rho \models x/\phi$ ssi $(n_1, \ldots, n_p) \models \phi$ où n_i est le nombre de fils (dans t) de $\sigma(x)$ dans $\rho(X_i)$ (avec $dom(\rho) = \{X_1, \ldots, X_p\}$),
- $t, \sigma, \rho \models \psi_1 \vee \psi_2$ ssi $t, \sigma, \rho \models \psi_1$ ou $t, \sigma, \rho \models \psi_2$,
- $t, \sigma, \rho \models \psi_1 \wedge \psi_2$ ssi $t, \sigma, \rho \models \psi_1$ et $t, \sigma, \rho \models \psi_2$,
- $t, \sigma, \rho \models \neg\psi$ ssi $u, \sigma \not\models \psi$,
- $t, \sigma, \rho \models \exists x\, \psi$ ssi il existe $p \in \|t\|$ t.q. $t, \sigma \cup \{p \to x\}, \rho \models \psi$,
- $t, \sigma, \rho \models \exists X \psi$ ssi il existe $P \subseteq \|t\|$ t.q. $t, \sigma, \rho \cup \{P \to X\} \models \psi$,

PMSO : exemples

- racine :

  $x = \varepsilon \equiv \neg \exists y \, y \downarrow x$
- feuille :

  $\mathsf{leaf}(x) \equiv \neg \exists y \, x \downarrow y$
- $x \downarrow y \equiv \exists Y \, Y = \{y\} \wedge x/_{[Y]=1}$
- ordre préfixe = clôture transitive de $\downarrow$:

  $x \downarrow^* y \equiv \forall X \left(x \in X \wedge \forall z \forall z' \left(z \in X \wedge z \downarrow z' \Rightarrow z' \in X\right)\right) \Rightarrow y \in X$

langages de PMSO

Definition : langage

Le langage défini formule close ψ de PMSO sur Σ est l'ensemble des termes $t \in \mathcal{U}_2(\Sigma)$ t.q. $t \models \psi$.

langage de PMSO : exemple 1

L'ensemble des arbres de la forme $f(a, \ldots, a, b, \ldots, b)$ avec le même nombre de a que de b.

$$
\begin{aligned}
\exists X_a \, \exists X_b \, f(\varepsilon) \quad & \wedge \forall y \, (y \in X_a \Leftrightarrow a(y)) \wedge (y \in X_b \Leftrightarrow b(y)) \\
& \wedge \forall y \; \varepsilon \downarrow y \Rightarrow (\mathsf{leaf}(y) \wedge (y \in X_a \vee y \in X_b)) \\
& \wedge \varepsilon /_{[X_a]=[X_b]}
\end{aligned}
$$

PMSO : PA exemple 2

$\Sigma = \{a, b\}$.
Ensemble des arbres de $\mathcal{U}_2(\Sigma)$ dont tout noeud interne est étiqueté par a et tout noeud étiqueté par a a au moins autant de fils sans b que de fils contenant b.

$$
\begin{array}{lcl}
\exists X_a \, \exists X_b \, \forall x & & (b(x) \Rightarrow \mathsf{leaf}(x)) \\
& \wedge & (a(x) \wedge x/_{[X_a] \geq [X_b] > 0} \Rightarrow x \in X_b) \\
& \wedge & (a(x) \wedge x/_{[X_a] \geq [X_b] = 0} \Rightarrow x \in X_a) \\
& \wedge & (b(x) \wedge x/_{[X_a] = [X_b] = 0} \Rightarrow x \in X_b)
\end{array}
$$

$Q = Q_{\mathsf{f}} = \{q_a, q_b\}$.
L'état q_b accepte les arbres contenant un b et q_a les autres.

$$
\Delta = \left\{ \begin{array}{lcl lcl}
a(x_{q_a} \geq x_{q_b} = 0) & \rightarrow & q_a & a(x_{q_a} \geq x_{q_b} > 0) & \rightarrow & q_b \\
& & & b(x_{q_a} = x_{q_b} = 0) & \rightarrow & q_b
\end{array} \right\}
$$

PMSO : exemples de requetes

Base de clients d'un vendeur de musique en ligne, stockée dans un arbre d'arité variable non-ordonné.

Un client est un sous-arbre :

- racine étiquettée par `client`
- informations dans les fils (achats, étiquettés en tête par le genre).

Requete pour clients x qui ont acheté plus de jazz que de blues :

$$\begin{array}{rcl} query_1(x) \equiv & & (\exists X_{\texttt{jazz}} \, \forall y \, y \in X_{\texttt{jazz}} \Leftrightarrow \texttt{jazz}(y)) \\ & \wedge & (\exists X_{\texttt{blues}} \, \forall y \, y \in X_{\texttt{blues}} \Leftrightarrow \texttt{blues}(y)) \\ & \wedge & \texttt{client}(x) \wedge x/_{[X_{\texttt{jazz}}] > [X_{\texttt{blues}}]} \end{array}$$

Requete pour clients x ayant acheté plus de jazz que n'importe quel autre genre :

$$\begin{array}{rcl} query_1(x) \equiv & & (\exists X_{\texttt{jazz}} \, \forall y \, y \in X_{\texttt{jazz}} \Leftrightarrow \texttt{jazz}(y)) \\ & \wedge & (\exists X_{\texttt{other}} \, \forall y \, y \in X_{\texttt{other}} \Leftrightarrow \neg\texttt{jazz}(y)) \\ & \wedge & \texttt{client}(x) \wedge x/_{[X_{\texttt{jazz}}] > [X_{\texttt{other}}]} \end{array}$$

PMSO et PA

Theorem

$L \subseteq \mathcal{U}(\Sigma)$ *est définissable en* PMSO *ssi* L *est un langage de PA.*

Contraintes de Presburger et arbres finis orientés **ordonnés** étiquettés par des symboles de fonction sans arité

Presburger Hedge Automata (PHA)

Automates d'Hedges et Presburger (PHA)

Definition : Automates d'Hedges et Presburger

Un *automate d'Hedges et Presburger* (PHA) sur un alphabet Σ est un tuple $\mathcal{A} = (\Sigma, Q, Q^{\mathsf{f}}, \Delta)$ où $Q = \{q_1, \dots, q_p\}$ est un ensemble fini d'*états*, $Q^{\mathsf{f}} \subseteq Q$ est le sous-ensemble des états finaux et Δ est un ensemble de règles de transition de la forme : $a\big(\bigvee_i (L_i \wedge \phi_i)\big) \to q$ avec $a \in \Sigma$, $q \in Q$, $L_i \subseteq Q^*$ langage régulier et $\phi_i = \phi(x_1, \dots, x_p)$ est une formule de Presburger avec une variable libre p. ch. état.

Pour $w \in Q^*$ on définit $w \models L_i \wedge \phi_i$ par $w \in L_i$ et $\#(w) \models \phi_i$.

PHA : langages

Le langage $L(\mathcal{A}, q)$ de $\mathcal{A}$ dans l'état $q \in Q$, est le plus petit ensemble d'arbres d'arité variable ordonnés $a(t_1, \ldots, t_n) \in \mathcal{U}(\Sigma)$ t.q.

- il existe $i_1, \ldots, i_n \leq p$ tels que pour tout $j \leq n$, $t_j \in L(\mathcal{A}, q_{i_j})$,
- il existe une transition $a\big(\bigvee_i (L_i \wedge \phi_i)\big) \to q \in \Delta$ telle que $q_{i_1} \ldots q_{i_n} \models \bigvee_i (L_i \wedge \phi_i)$, i.e. il existe i t.q.
  - $q_{i_1} \ldots q_{i_n} \in L_i$,
  - $\#(q_{i_1} \ldots q_{i_n}) \models \phi_i(x_1, \ldots, x_p)$.

Le langage de $\mathcal{A}$ est $L(\mathcal{A}) = \bigcup_{q \in Q_\mathsf{f}} L(\mathcal{A}, q)$.

PHA : propriétés

Proposition :

- La classe des langages de PHA est fermée par union et intersection,
- La classe des langages de PHA n'est pas fermée par complémentaire.

- $\cup$, $\cap$: produit
- csq problème de l'universalité.

Proposition :

DPHA $\neq$ NPHA.

PHA : problèmes de décision

Proposition : $\in$

L'appartenance est décidable em PTIME pour PHA.

Lemma :

Étant donné Q, $p = |Q|$, $M \subseteq Q$, et une contrainte $L \wedge \phi$ où $L \subseteq Q^*$ est régulier et $\phi = \phi(x_1, \ldots, x_p)$ est une formule de Presburger, il est décidable si il existe $w \in M^* \cap L$ tel que $\#(w) \models \phi$.

Proposition : $\emptyset$

Le vide est décidable pour PHA.

Proposition : $\forall$

L'universalité est indécidable pour PHA.

PHA : logique

Theorem :

Un ensemble d'arbres de $\mathcal{U}(\Sigma)$ est reconnaissable par un PHA ssi il est défini par une formule de PMSO de la form $\exists X_1 \ldots \exists X_k \, \phi$ où ϕ est du premier ordre.

Corollary :

- EPMSO (fragment existentiel) est décidable dans $\mathcal{U}(\Sigma)$.
- PMSO est indécidable dans $\mathcal{U}(\Sigma)$.

Arbres finis orientés mixtes (ordonnés et non ordonnés) étiquettés par des symboles de fonction sans arité

Mixed Trees

- $\Sigma = \Sigma_A \cup \Sigma_{AC}$.

$$\begin{array}{rcl} \text{tree} & := & a(\text{hedge}) \mid c(\text{multiset}) \quad (a \in \Sigma_A, c \in \Sigma_{AC}) \\ \text{hedge} & := & \text{tree}, \ldots, \text{tree} \\ \text{multiset} & := & \{\text{tree}, \ldots, \text{tree}\} \end{array}$$

- $c\big(\{t_1, \ldots, t_n\}\big)$ is denoted $c(t_1, \ldots, t_n)$.
- The set of mixed unranked trees over Σ is denoted $\mathcal{M}(\Sigma)$.

Presburger m-Tree Automata (PMA)

Definition : Presburger m-Tree Automata

A *Presburger m-Tree Automaton* (PMA) over an alphabet $\Sigma = \Sigma_A \cup \Sigma_{AC}$ is a tuple $\mathcal{A} = (\Sigma, Q, Q^{\mathsf{f}}, \Delta)$ where $Q = \{q_1, \ldots, q_p\}$ is a finite set of *states*, $Q^{\mathsf{f}} \subseteq Q$ is the subset of final states and Δ is a set of transition rules of the form :

- $a(L) \to q$ with $a \in \Sigma_A$, $q \in Q$, $L \subseteq Q^*$ is a regular language or
- $c(\phi) \to q$ with $c \in \Sigma$, $q \in Q$, $\phi = \phi(x_1, \ldots, x_p)$ is a Presburger formula with one free variable for each state.

PMA : languages

The language $L(\mathcal{A}, q)$ of the PMA $\mathcal{A}$ in state $q \in Q$, is the smallest set of mixed trees

- $a(t_1, \ldots, t_n)$, $a \in \Sigma_A$, such that
  - there exists $i_1, \ldots, i_n \leq p$ with $t_j \in L(\mathcal{A}, q_{i_j})$ for all $j \leq n$,
  - there exists a transition $a(L) \to q \in \Delta$ such that $q_{i_1} \ldots q_{i_n} \in L$,
- or $c(t_1, \ldots, t_n)$, $c \in \Sigma_{AC}$, such that
  - there exists $i_1, \ldots, i_n \leq p$ with $t_j \in L(\mathcal{A}, q_{i_j})$ for all $j \leq n$,
  - there exists a transition $c(\phi) \to q \in \Delta$ such that $\#(q_{i_1} \ldots q_{i_n}) \models \phi(x_1, \ldots, x_p)$.

The language of $\mathcal{A}$ is $L(\mathcal{A}) = \bigcup_{q \in Q_{\mathsf{f}}} L(\mathcal{A}, q)$.

PMA : properties

Proposition :

The class of PMA languages is closed under all Boolean operations.

Proposition :

DPMA $\equiv$ NPMA.

PMA : decision problems

Proposition : $\in$

Membership is decidable for PMA.

Proposition : $\emptyset$

Emptiness is decidable for PMA.

Consequences of the same results for PHA ($\mathrm{PMA} \subseteq \mathrm{PHA}$).

PMA : logique

Theorem :

The class of languages of $\mathcal{M}(\Sigma)$ definable by formulae de PMSO is the class of PMA languages.

Corollary

PMSO over $\mathcal{M}(\Sigma)$ *is decidable.*