Automates, jetons & logiques

(introduction aux automates cheminants)

Thomas Colcombet

Séminaire 68NQRT
Rennes, le 3 octobre 2006

CONTEXTE

Automates d'arbres? automates à branchement.

Automates cheminants: modèle séquentiel d'automates d'arbres.

Automates à jetons: extension des automates cheminants.
Permet d'exprimer la sémantique de XPath.
Blocs de base des transformations XML.
(Engelfriet&Hogeboom99,Milo&Suciu&Vianu03).
$\Rightarrow$ étude dans le cas non borné.

Relation avec la logique: FO+TC.

Objectif: identifier des classes d'automates avec des langages logiques et séparer les logiques en séparant l'expressivité des familles d'automates.

PLAN

- **Automates** cheminants
- Automates à **jetons**
- **Logiques** FO+TC, FO+posTC, FO+DTC

Automates cheminants

ARBRES FINIS/TERMES

DEF: Un arbre (binaire fini) $t : dom(t) \rightarrow \Sigma$, avec:

- $dom(t)$ l'ensemble des nœuds de l'arbre,
- Σ un ensemble fini d'étiquettes,
- $fg, fd : dom(t) \rightarrow dom(t)$ applications partielles «fils gauche»/«fils droit»,
- convention: pour tout nœud x, $fg(x)$ est défini ssi $fd(x)$ l'est,
- un nœud x est une feuille si ni $fg(x)$ ni $fd(x)$ n'est défini,
- la racine $racine(t)$...

AUTOMATES À BRANCHEMENTS

DEF: $\mathcal{A} = (Q, \Sigma, F, \delta)$ avec:

- Ensemble fini d'états Q,
- Ensemble d'états finaux $F \subseteq Q$,
- Relation de transition $\delta \subseteq (Q \times Q \times \Sigma \times Q) \uplus (\Sigma \times Q)$.

AUTOMATES À BRANCHEMENTS

DEF: $\mathcal{A} = (Q, \Sigma, F, \delta)$ avec:

- Ensemble fini d'états Q,
- Ensemble d'états finaux $F \subseteq Q$,
- Relation de transition $\delta \subseteq (Q \times Q \times \Sigma \times Q) \uplus (\Sigma \times Q)$.

DEF: Une exécution $\rho : dom(t) \to Q$ avec pour tout $x \in dom(t)$:

- si x est une feuille, $(t(x), \rho(x)) \in \delta$,
- sinon $(\rho(fg(x)), \rho(fd(x)), t(x), \rho(x)) \in \delta$.

AUTOMATES À BRANCHEMENTS

DEF: $\mathcal{A} = (Q, \Sigma, F, \delta)$ avec:

- Ensemble fini d'états Q,
- Ensemble d'états finaux $F \subseteq Q$,
- Relation de transition $\delta \subseteq (Q \times Q \times \Sigma \times Q) \uplus (\Sigma \times Q)$.

DEF: Une exécution $\rho : dom(t) \to Q$ avec pour tout $x \in dom(t)$:

- si x est une feuille, $(t(x), \rho(x)) \in \delta$,
- sinon $(\rho(fg(x)), \rho(fd(x)), t(x), \rho(x)) \in \delta$.

DEF: Un arbre t est accepté par $\mathcal{A}$ s'il existe une exécution ρ de $\mathcal{A}$ sur t telle que $\rho(racine(t)) \in F$. REG est l'ensemble des langages acceptés par un automate à branchements.

AUTOMATES À BRANCHEMENTS

DEF: $\mathcal{A} = (Q, \Sigma, F, \delta)$ avec:

- Ensemble fini d'états Q,
- Ensemble d'états finaux $F \subseteq Q$,
- Relation de transition $\delta \subseteq (Q \times Q \times \Sigma \times Q) \uplus (\Sigma \times Q)$.

DEF: Une exécution $\rho : dom(t) \rightarrow Q$ avec pour tout $x \in dom(t)$:

- si x est une feuille, $(t(x), \rho(x)) \in \delta$,
- sinon $(\rho(fg(x)), \rho(fd(x)), t(x), \rho(x)) \in \delta$.

DEF: Un arbre t est accepté par $\mathcal{A}$ s'il existe une exécution ρ de $\mathcal{A}$ sur t telle que $\rho(racine(t)) \in F$. REG est l'ensemble des langages acceptés par un automate à branchements.

PROP: REG est fermé par intersection, union, complémentation, et projection. Le vide est décidable.

AUTOMATES CHEMINANTS

DEF(Aho,Ullman71): $\mathcal{A} = (Q, \Sigma, I, F, \delta)$ avec:

- états Q, états initiaux I, états finaux $F \subseteq Q$,
- relation de transition $\delta \subseteq Q \times \mathrm{Types} \times \Sigma \times \mathrm{Dirs} \times Q$,
- $\mathrm{Types} = \{racine, fils\_gauche, fils\_droit\} \times \{feuille, interieur\}$,
- $\mathrm{Dirs} = \{haut, bas\_gauche, bas\_droite, sur\_place\}$.

AUTOMATES CHEMINANTS

DEF(Aho,Ullman71): $\mathcal{A} = (Q, \Sigma, I, F, \delta)$ avec:

- états Q, états initiaux I, états finaux $F \subseteq Q$,
- relation de transition $\delta \subseteq Q \times \mathrm{Types} \times \Sigma \times \mathrm{Dirs} \times Q$,
- $\mathrm{Types} = \{racine, \mathit{fils\_gauche}, \mathit{fils\_droit}\} \times \{feuille, interieur\}$,
- $\mathrm{Dirs} = \{haut, bas\_gauche, bas\_droite, sur\_place\}$.

DEF: Une configuration $(q, x) \in Q \times dom(t)$.
Une exécution $(q_0, x_0) \ldots (q_n, x_n) \in (Q \times dom(t))^+$ avec:

- $q_0 \in I$, $x_0 = racine(t)$,
- pour tout $k < n$, $(q_k, type(x_k), t(x_k), dir(x_k, x_{k+1}), q_{k+1}) \in \delta$.

AUTOMATES CHEMINANTS

DEF(Aho,Ullman71): $\mathcal{A} = (Q, \Sigma, I, F, \delta)$ avec:

- états Q, états initiaux I, états finaux $F \subseteq Q$,
- relation de transition $\delta \subseteq Q \times \mathrm{Types} \times \Sigma \times \mathrm{Dirs} \times Q$,
- $\mathrm{Types} = \{racine, fils\_gauche, fils\_droit\} \times \{feuille, interieur\}$,
- $\mathrm{Dirs} = \{haut, bas\_gauche, bas\_droite, sur\_place\}$.

DEF: Une configuration $(q, x) \in Q \times dom(t)$.
Une exécution $(q_0, x_0) \ldots (q_n, x_n) \in (Q \times dom(t))^+$ avec:

- $q_0 \in I$, $x_0 = racine(t)$,
- pour tout $k < n$, $(q_k, type(x_k), t(x_k), dir(x_k, x_{k+1}), q_{k+1}) \in \delta$.

DEF: Un arbre t est accepté par $\mathcal{A}$ s'il existe une exécution de $\mathcal{A}$ sur t telle que le dernier état appartienne à F. TWA est l'ensemble des languages acceptés par un automate cheminant.

AUTOMATES CHEMINANTS

DEF(Aho,Ullman71): $\mathcal{A} = (Q, \Sigma, I, F, \delta)$ avec:

- états Q, états initiaux I, états finaux $F \subseteq Q$,
- relation de transition $\delta \subseteq Q \times \mathrm{Types} \times \Sigma \times \mathrm{Dirs} \times Q$,
- $\mathrm{Types} = \{racine, fils\_gauche, fils\_droit\} \times \{feuille, interieur\}$,
- $\mathrm{Dirs} = \{haut, bas\_gauche, bas\_droite, sur\_place\}$.

DEF: Une configuration $(q, x) \in Q \times dom(t)$.
Une exécution $(q_0, x_0) \dots (q_n, x_n) \in (Q \times dom(t))^+$ avec:

- $q_0 \in I$, $x_0 = racine(t)$,
- pour tout $k < n$, $(q_k, type(x_k), t(x_k), dir(x_k, x_{k+1}), q_{k+1}) \in \delta$.

DEF: Un arbre t est accepté par $\mathcal{A}$ s'il existe une exécution de $\mathcal{A}$ sur t telle que le dernier état appartienne à F. TWA est l'ensemble des languages acceptés par un automate cheminant.

DEF: $\mathcal{A}$ est déterministe si $I = \{q_0\}$ et $\delta : Q \times \mathrm{Types} \times \Sigma \rightarrow \mathrm{Dirs} \times Q$.
DTWA est l'ensemble des languages acceptés par un automate cheminant déterministe.

EXEMPLES DTWA: PARCOURS EN PROFONDEUR

PROP: Les langages suivants appartiennent à DTWA:

- les arbres contenant un nœud d'étiquette a,
- les arbres ne contenant pas une étiquette a,
- les arbres dont le mot de feuille (yield) appartient à L régulier.

Exemples DTWA: parcours en profondeur

PROP: Les langages suivants appartiennent à DTWA:

- les arbres contenant un nœud d'étiquette a,
- les arbres ne contenant pas une étiquette a,
- les arbres dont le mot de feuille (yield) appartient à L régulier.

CŒUR: (parcours en profondeur d'abord, de la gauche vers la droite)

$$Q = \{q, \bar{q}, r\}$$

$$I = \{q\}$$

$$\begin{array}{llllll}
\delta = \{ & (q, & ?, & interieur, & ?, & bas_gauche, & q), \\
& (q, & ?, & feuille, & ?, & sur_place, & \bar{q}), \\
& (\bar{q}, & fils_gauche, & ?, & ?, & haut, & r), \\
& (r, & ?, & ?, & ?, & bas_droite, & q), \\
& (\bar{q}, & fils_droit, & ?, & ?, & haut, & \bar{q}) \quad \}
\end{array}$$

EXEMPLES DTWA: PARCOURS EN PROFONDEUR

PROP: Les langages suivants appartiennent à DTWA:

- les arbres contenant un nœud d'étiquette a,
- les arbres ne contenant pas une étiquette a,
- les arbres dont le mot de feuille (yield) appartient à L régulier.

CŒUR: (parcours en profondeur d'abord, de la gauche vers la droite)

$$Q = \{q, \bar{q}, r\}$$

$$I = \{q\}$$

$$\begin{array}{llllll} \delta = \{ & (q, & ?, & interieur, & ?, & bas\_gauche, & q), \\ & (q, & ?, & feuille, & ?, & sur\_place, & \bar{q}), \\ & (\bar{q}, & fils\_gauche, & ?, & ?, & haut, & r), \\ & (r, & ?, & ?, & ?, & bas\_droite, & q), \\ & (\bar{q}, & fils\_droit, & ?, & ?, & haut, & \bar{q}) \quad \} \end{array}$$

PREUVE: Par adaptation de la construction ci-dessus.

EXEMPLE DTWA AVANCÉ

PROP: Soit $\Sigma = \{\vee, \wedge\}$. L'ensemble des arbres s'évaluant à vrai dans l'algèbre de Boole à deux éléments est dans DTWA.
(Sur une feuille, par convention, $\vee$ vaut «vrai», et $\wedge$ vaut «faux»)

EXEMPLE DTWA AVANCÉ

PROP: Soit $\Sigma = \{\vee, \wedge\}$. L'ensemble des arbres s'évaluant à vrai dans l'algèbre de Boole à deux éléments est dans DTWA.
(Sur une feuille, par convention, $\vee$ vaut «vrai», et $\wedge$ vaut «faux»)

IDÉE: $u \vee v$ équivaut à «if u then vrai else v»

$u \wedge v$ équivaut à «if u then v else faux»

$\Rightarrow$ Il n'est pas nécessaire de tout évaluer.

Exemple DTWA avancé

PROP: Soit $\Sigma = \{\vee, \wedge\}$. L'ensemble des arbres s'évaluant à vrai dans l'algèbre de Boole à deux éléments est dans DTWA.
(Sur une feuille, par convention, $\vee$ vaut «vrai», et $\wedge$ vaut «faux»)

$$Q = \{0, 1, 0', 1', q\}, \qquad I = \{q\}$$

$$
\begin{array}{llllllll}
\delta = \{ & (q, & ?, & \mathit{interieur}, & ?, & \mathit{bas\_gauche}, & q), & \\
 & (q, & ?, & \mathit{feuille}, & \vee, & \mathit{sur\_place}, & 1), & \\
 & (q, & ?, & \mathit{feuille}, & \wedge, & \mathit{sur\_place}, & 0), & \\
 & (v, & \mathit{fils\_gauche}, & ?, & ?, & \mathit{haut}, & v') : v = 0, 1, & \\
 & (v, & \mathit{fils\_droite}, & ?, & ?, & \mathit{haut}, & v) : v = 0, 1, & \\
 & (0', & ?, & ?, & \vee, & \mathit{bas\_droite}, & q), & \\
 & (1', & ?, & ?, & \vee, & \mathit{sur\_place}, & 1), & \\
 & (0', & ?, & ?, & \wedge, & \mathit{sur\_place}, & 0), & \\
 & (1', & ?, & ?, & \wedge, & \mathit{bas\_droite}, & 1) & \}
\end{array}
$$

PROPRIÉTÉS DE TWA

PROP: $\mathrm{TWA} \subseteq \mathrm{REG}$.

PROP: TWA est fermé par intersection et union.

Propriétés de DTWA

PROP: DTWA $\subseteq$ TWA.

Propriétés de DTWA

PROP: $\mathrm{DTWA} \subseteq \mathrm{TWA}$.

PROP: DTWA est fermé par intersection.

PROPRIÉTÉS DE DTWA

PROP: $\mathrm{DTWA} \subseteq \mathrm{TWA}$.

PROP: DTWA est fermé par intersection.

RQ: Union, complémentation? Problème de terminaison.

Propriétés de DTWA

PROP: $\mathrm{DTWA} \subseteq \mathrm{TWA}$.

PROP: DTWA est fermé par intersection.

RQ: Union, complémentation? Problème de terminaison.

TH(Muscholl,Samuelides,Ségoufin05): Tout automate cheminant déterministe peut être transformé en un automate équivalent s'arrêtant sur toutes ses entrées. **(Sipser78 sur les mots)**

Propriétés de DTWA

PROP: $\mathrm{DTWA} \subseteq \mathrm{TWA}$.

PROP: DTWA est fermé par intersection.

RQ: Union, complémentation? Problème de terminaison.

TH(Muscholl,Samuelides,Ségoufin05): Tout automate cheminant déterministe peut être transformé en un automate équivalent s'arrêtant sur toutes ses entrées. **(Sipser78 sur les mots)**

COR: DTWA est fermé par intersection et complémentation.

SÉPARATIONS

TH(Bojańczyk,C.04): $\mathrm{DTWA} \subsetneq \mathrm{TWA}$.

SÉPARATIONS

TH(Bojańczyk,C.04): $\mathrm{DTWA} \subsetneq \mathrm{TWA}$.

TH(Bojańczyk,C.05): $\mathrm{TWA} \subsetneq \mathrm{REG}$.

SÉPARATIONS

TH(Bojańczyk,C.04): $\mathrm{DTWA} \subsetneq \mathrm{TWA}$.

TH(Bojańczyk,C.05): $\mathrm{TWA} \subsetneq \mathrm{REG}$.

COR: TWA et DTWA ne sont pas fermés par projection.

PREUVE: Tous les languages réguliers sont projection d'un langage de DTWA.

Questions ouvertes

Q? Election à la majorité est DTWA? TWA?

Q? Évaluation booléenne sur $\Sigma = \{\vee, \wedge, xor\}$ est DTWA? TWA?

Q? Le problème pour $L \in \mathrm{REG}$, «$L \in \mathrm{DTWA}$?» est décidable?

Automates à jetons

AUTOMATES À JETONS

Principe: Un automate à jetons est un automate cheminant avec:

- Un nombre fixé de jetons numérotés.
- L'automate peut voir les jetons situés en dessous de lui.
- L'automate peut poser et enlever des jetons.
- Discipline de pile: seul le dernier jeton peut-être modifié.

Automates à jetons

DEF(Globerman,Harel96,Engelfriet,Hogeboom99):
Un automate cheminant à k-jetons $\mathcal{A} = (Q, \Sigma, k, I, F, \delta)$ avec: $\delta \subseteq$
$$Q \times \mathrm{Types} \times \Sigma \times \underbrace{\{0, \ldots, k\}}_{\text{nombre de jetons posés}} \times \underbrace{\{0,1\}^k}_{\text{jetons à la position courante}} \times (\mathrm{Dirs} \uplus \{pose, enleve\}) \times Q$$

Une configuration $(q, x, \underbrace{x_1 \ldots x_n}_{\text{jetons posés}}) \in Q \times dom(t) \times dom(t)^*$ avec $q \leq n \leq k$

Automates à jetons

DEF(Globerman,Harel96,Engelfriet,Hogeboom99):
Un automate cheminant à k-jetons $\mathcal{A} = (Q, \Sigma, k, I, F, \delta)$ avec: $\delta \subseteq$
$$Q \times \text{Types} \times \Sigma \times \underbrace{\{0, \ldots, k\}}_{\text{nombre de jetons posés}} \times \underbrace{\{0, 1\}^k}_{\text{jetons à la position courante}} \times (\text{Dirs} \uplus \{pose, enleve\}) \times Q$$

Une configuration $(q, x, \underbrace{x_1 \ldots x_n}_{\text{jetons posés}}) \in Q \times dom(t) \times dom(t)^*$ avec $q \leq n \leq k$

- Les actions de Dirs sont identiques (et laissent les jetons inchangés),

AUTOMATES À JETONS

DEF(Globerman,Harel96,Engelfriet,Hogeboom99):
Un automate cheminant à k-jetons $\mathcal{A} = (Q, \Sigma, k, I, F, \delta)$ avec: $\delta \subseteq Q \times \mathrm{Types} \times \Sigma \times \underbrace{\{0, \ldots, k\}}_{\text{nombre de jetons posés}} \times \underbrace{\{0,1\}^k}_{\text{jetons à la position courante}} \times (\mathrm{Dirs} \uplus \{pose, enleve\}) \times Q$

Une configuration $(q, x, \underbrace{x_1 \ldots x_n}_{\text{jetons posés}}) \in Q \times dom(t) \times dom(t)^*$ avec $q \leq n \leq k$

- Les actions de Dirs sont identiques (et laissent les jetons inchangés),
- $pose$ pose un jeton à la position courante:
  $(q, x, x_0 \ldots x_n) \Rightarrow (q', x, x_0 \ldots x_n x)$

AUTOMATES À JETONS

DEF(Globerman,Harel96,Engelfriet,Hogeboom99):
Un automate cheminant à k-jetons $\mathcal{A} = (Q, \Sigma, k, I, F, \delta)$ avec: $\delta \subseteq Q \times \mathrm{Types} \times \Sigma \times \underbrace{\{0, \ldots, k\}}_{\text{nombre de jetons posés}} \times \underbrace{\{0, 1\}^k}_{\text{jetons à la position courante}} \times (\mathrm{Dirs} \uplus \{pose, enleve\}) \times Q$

Une configuration $(q, x, \underbrace{x_1 \ldots x_n}_{\text{jetons posés}}) \in Q \times dom(t) \times dom(t)^*$ avec $q \leq n \leq k$

- Les actions de Dirs sont identiques (et laissent les jetons inchangés),
- $pose$ pose un jeton à la position courante:
  $(q, x, x_0 \ldots x_n) \Rightarrow (q', x, x_0 \ldots x_n x)$
- $enleve$ version forte enlève le dernier jeton:
  $(q, x, x_0 \ldots x_n) \Rightarrow (q', x, x_0 \ldots x_{n-1})$

AUTOMATES À JETONS

DEF(Globerman,Harel96,Engelfriet,Hogeboom99):
Un automate cheminant à k-jetons $\mathcal{A} = (Q, \Sigma, k, I, F, \delta)$ avec: $\delta \subseteq$

$$Q \times \text{Types} \times \Sigma \times \underbrace{\{0, \dots, k\}}_{\text{nombre de jetons posés}} \times \underbrace{\{0,1\}^k}_{\text{jetons à la position courante}} \times (\text{Dirs} \uplus \{pose, enleve\}) \times Q$$

Une configuration $(q, x, \underbrace{x_1 \dots x_n}_{\text{jetons posés}}) \in Q \times dom(t) \times dom(t)^*$ avec $q \leq n \leq k$

- Les actions de Dirs sont identiques (et laissent les jetons inchangés),
- $pose$ pose un jeton à la position courante:
  $(q, x, x_0 \dots x_n) \Rightarrow (q', x, x_0 \dots x_n x)$
- $enleve$ version forte enlève le dernier jeton:
  $(q, x, x_0 \dots x_n) \Rightarrow (q', x, x_0 \dots x_{n-1})$
- $enleve$ version faible enlève le dernier jeton à la position courante:
  $(q, x, x_0 \dots x_{n-1} x) \Rightarrow (q', x, x_0 \dots x_{n-1})$

Automates à jetons

DEF:

sPA_k: languages acceptés par un automate à k-jetons forts.

sDPA_k: languages acceptés par un automate déterministe à k-jetons forts.

wPA_k: languages acceptés par un automate à k-jetons faibles.

wDPA_k: languages acceptés par un automate déterministe à k-jetons faible

$$\mathrm{sPA} = \bigcup_k \mathrm{sPA}_k, \quad \mathrm{sDPA} = \bigcup_k \mathrm{sDPA}_k, \quad \mathrm{wPA} = \bigcup_k \mathrm{wPA}_k, \quad \mathrm{wDPA} = \bigcup_k \mathrm{wDPA}_k.$$

AUTOMATES À JETONS

DEF:

sPA_k: languages acceptés par un automate à k-jetons forts.

sDPA_k: languages acceptés par un automate déterministe à k-jetons forts.

wPA_k: languages acceptés par un automate à k-jetons faibles.

wDPA_k: languages acceptés par un automate déterministe à k-jetons faible

$$\mathrm{sPA} = \bigcup_k \mathrm{sPA}_k, \quad \mathrm{sDPA} = \bigcup_k \mathrm{sDPA}_k, \quad \mathrm{wPA} = \bigcup_k \mathrm{wPA}_k, \quad \mathrm{wDPA} = \bigcup_k \mathrm{wDPA}_k.$$

TH(Bojańczyk,Samuelides,Schwentik,Ségoufin06): $\mathrm{sDPA}_k = \mathrm{wDPA}_k$, $\mathrm{sPA}_k = \mathrm{wPA}_k$.

Consequence: On ne parle plus que de PA et DPA.

AUTOMATES À JETONS

DEF:

sPA_k: languages acceptés par un automate à k-jetons forts.

sDPA_k: languages acceptés par un automate déterministe à k-jetons forts.

wPA_k: languages acceptés par un automate à k-jetons faibles.

wDPA_k: languages acceptés par un automate déterministe à k-jetons faible

$$\mathrm{sPA} = \bigcup_k \mathrm{sPA}_k, \quad \mathrm{sDPA} = \bigcup_k \mathrm{sDPA}_k, \quad \mathrm{wPA} = \bigcup_k \mathrm{wPA}_k, \quad \mathrm{wDPA} = \bigcup_k \mathrm{wDPA}_k.$$

TH(Bojańczyk,Samuelides,Schwentik,Ségoufin06): $\mathrm{sDPA}_k = \mathrm{wDPA}_k$,
$\mathrm{sPA}_k = \mathrm{wPA}_k$.

Consequence: On ne parle plus que de PA et DPA.

PROP: $\mathrm{DPA}_k \subseteq \mathrm{DPA}_{k+1}$, $\mathrm{PA}_k \subseteq \mathrm{PA}_{k+1}$, $\mathrm{DPA}_k \subseteq \mathrm{PA}_k \subseteq \mathrm{REG}$.

PROPRIÉTÉS DES AUTOMATES À JETONS

PROP: PA_k est fermé par intersection et union.

PROPRIÉTÉS DES AUTOMATES À JETONS

PROP: PA_k est fermé par intersection et union.

TH(BSSS06): DPA_k est fermé par intersection, union et **complémentation**.

Propriétés des automates à jetons

PROP: PA_k est fermé par intersection et union.

TH(BSSS06): DPA_k est fermé par intersection, union et **complémentation**.

TH(BSSS06):
$$\begin{aligned}
&\mathrm{PA}_k \subsetneq \mathrm{PA}_{k+1}, \\
&\mathrm{DPA}_k \subsetneq \mathrm{DPA}_{k+1}, \\
&\mathrm{DPA}_k \subsetneq \mathrm{PA}_k, \\
&\mathrm{TWA} \not\subseteq \mathrm{DPA}_k, \\
&\mathrm{PA} \subsetneq \mathrm{REG}.
\end{aligned}$$

Propriétés des automates à jetons

PROP: PA_k est fermé par intersection et union.

TH(BSSS06): DPA_k est fermé par intersection, union et **complémentation**.

TH(BSSS06):

$$\begin{aligned}
&\mathrm{PA}_k \subsetneq \mathrm{PA}_{k+1}\,,\\
&\mathrm{DPA}_k \subsetneq \mathrm{DPA}_{k+1}\,,\\
&\mathrm{DPA}_k \subsetneq \mathrm{PA}_k\,,\\
&\mathrm{TWA} \not\subseteq \mathrm{DPA}_k\,,\\
&\mathrm{PA} \subsetneq \mathrm{REG}.
\end{aligned}$$

Q? $\begin{cases} \mathrm{PA} = \mathrm{DPA}? \\ \mathrm{DPA} \subseteq \mathrm{TWA}? \end{cases}$

Logiques

LOGIQUES

FO(premier-ordre): $\exists x,\ \forall x,\ a(x),\ fg(x),\ fd(x),\ x \sqsubseteq y,\ \vee,\ \wedge,\ \neg$.

LOGIQUES

FO(premier-ordre): $\exists x,\ \forall x,\ a(x),\ fg(x),\ fd(x),\ x \sqsubseteq y,\ \vee,\ \wedge,\ \neg$.

DEF(Immerman87): Fermeture transitive de $\phi(x, y, z_1, \ldots, z_k)$;
$t \models \phi^*(u, v, w_1, \ldots, w_k)$ ssi
il existe $u = u_1, \ldots, u_n = v$ tels que pour tout i, $t \models \phi(u_i, u_{i+1}, w_1, \ldots, w_k)$.

LOGIQUES

FO(**premier-ordre**): $\exists x,\ \forall x,\ a(x),\ fg(x),\ fd(x),\ x \sqsubseteq y,\ \vee,\ \wedge,\ \neg$.

DEF(Immerman87): Fermeture transitive de $\phi(x, y, z_1, \ldots, z_k)$;
$t \models \phi^*(u, v, w_1, \ldots, w_k)$ ssi
il existe $u = u_1, \ldots, u_n = v$ tels que pour tout i, $t \models \phi(u_i, u_{i+1}, w_1, \ldots, w_k)$.

DEF: Une formule $\phi(x, y, z_1, \ldots, z_k)$ est déterministe si
pour tous $t, u, w_1, \ldots w_k$, il existe au plus un v tel que $t \models \phi(u, v, w_1, \ldots, w_k)$.

Logiques

FO(**premier-ordre**): $\exists x,\ \forall x,\ a(x),\ fg(x),\ fd(x),\ x \sqsubseteq y,\ \vee,\ \wedge,\ \neg$.

DEF(Immerman87): Fermeture transitive de $\phi(x, y, z_1, \ldots, z_k)$;
$t \models \phi^*(u, v, w_1, \ldots, w_k)$ ssi
il existe $u = u_1, \ldots, u_n = v$ tels que pour tout i, $t \models \phi(u_i, u_{i+1}, w_1, \ldots, w_k)$.

DEF: Une formule $\phi(x, y, z_1, \ldots, z_k)$ est déterministe si
pour tous $t, u, w_1, \ldots w_k$, il existe au plus un v tel que $t \models \phi(u, v, w_1, \ldots, w_k)$.

DEF: FO+TC = FO + fermeture transitive
FO+posTC = FO + fermeture transitive apparaissant positivement
FO+DTC = FO + fermeture transitive de formules déterministes

Logiques

FO(premier-ordre): $\exists x,\ \forall x,\ a(x),\ fg(x),\ fd(x),\ x \sqsubseteq y,\ \vee,\ \wedge,\ \neg$.

DEF(Immerman87): Fermeture transitive de $\phi(x, y, z_1, \ldots, z_k)$;
$t \models \phi^*(u, v, w_1, \ldots, w_k)$ ssi
il existe $u = u_1, \ldots, u_n = v$ tels que pour tout i, $t \models \phi(u_i, u_{i+1}, w_1, \ldots, w_k)$.

DEF: Une formule $\phi(x, y, z_1, \ldots, z_k)$ est déterministe si
pour tous $t, u, w_1, \ldots w_k$, il existe au plus un v tel que $t \models \phi(u, v, w_1, \ldots, w_k)$.

DEF: FO+TC = FO + fermeture transitive
FO+posTC = FO + fermeture transitive apparaissant positivement
FO+DTC = FO + fermeture transitive de formules déterministes

MSO: ...

LOGIQUES

FO(**premier-ordre**): $\exists x,\ \forall x,\ a(x),\ fg(x),\ fd(x),\ x \sqsubseteq y,\ \vee,\ \wedge,\ \neg$.

DEF(Immerman87): Fermeture transitive de $\phi(x, y, z_1, \ldots, z_k)$;
$t \models \phi^*(u, v, w_1, \ldots, w_k)$ ssi
il existe $u = u_1, \ldots, u_n = v$ tels que pour tout i, $t \models \phi(u_i, u_{i+1}, w_1, \ldots, w_k)$.

DEF: Une formule $\phi(x, y, z_1, \ldots, z_k)$ est déterministe si
pour tous $t, u, w_1, \ldots w_k$, il existe au plus un v tel que $t \models \phi(u, v, w_1, \ldots, w_k)$.

DEF: FO+TC = FO + fermeture transitive
FO+posTC = FO + fermeture transitive apparaissant positivement
FO+DTC = FO + fermeture transitive de formules déterministes

MSO: ...

TH(Thatcher,Wright68): MSO=REG.

LOGIQUES

FO(premier-ordre): $\exists x,\ \forall x,\ a(x),\ fg(x),\ fd(x),\ x \sqsubseteq y,\ \vee,\ \wedge,\ \neg.$

DEF(Immerman87): Fermeture transitive de $\phi(x, y, z_1, \ldots, z_k)$;
$t \models \phi^*(u, v, w_1, \ldots, w_k)$ ssi
il existe $u = u_1, \ldots, u_n = v$ tels que pour tout i, $t \models \phi(u_i, u_{i+1}, w_1, \ldots, w_k)$.

DEF: Une formule $\phi(x, y, z_1, \ldots, z_k)$ est déterministe si pour tous $t, u, w_1, \ldots w_k$, il existe au plus un v tel que $t \models \phi(u, v, w_1, \ldots, w_k)$.

DEF: FO+TC = FO + fermeture transitive
FO+posTC = FO + fermeture transitive apparaissant positivement
FO+DTC = FO + fermeture transitive de formules déterministes

MSO: ...

TH(Thatcher,Wright68): MSO=REG. $\left.\begin{array}{l}\forall \mathcal{A}.\exists \Phi. \\ \forall \Phi.\exists \mathcal{A}.\end{array}\right\}$ $\forall t.t \models \Phi$ ssi $\mathcal{A}$ accepte t.

CORRESPONDANCES

TH(Engelfriet,Hoogeboom06): FO+posTC = PA.

CORRESPONDANCES

TH(Engelfriet,Hoogeboom06): FO+posTC = PA.

PREUVE: $\Leftarrow$: récurrence sur le nombre de jetons.
$\Rightarrow$: récurrence sur la formule.

HR: $t \models \phi(u_1, \ldots, u_k)$
ssi $\mathcal{A}$ accepte t à partir de la configuration $(q_0, racine(t), u_1 \ldots u_k)$.

CORRESPONDANCES

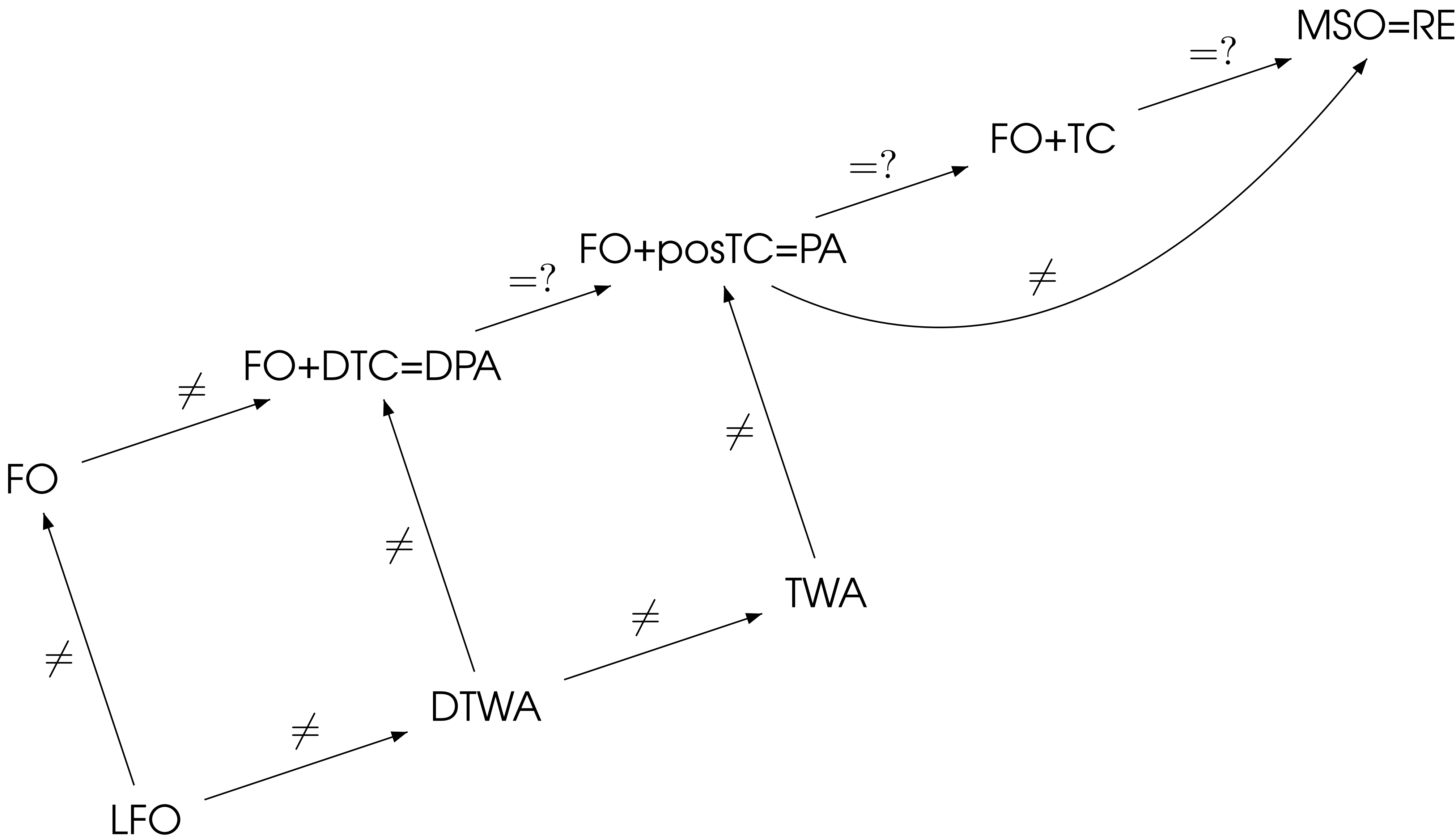
TH(Engelfriet,Hoogeboom06): FO+posTC = PA.

PREUVE: $\Leftarrow$: récurrence sur le nombre de jetons.
$\Rightarrow$: récurrence sur la formule.

HR: $t \models \phi(u_1, \ldots, u_k)$
ssi $\mathcal{A}$ accepte t à partir de la configuration $(q_0, racine(t), u_1 \ldots u_k)$.

TH(EH06): FO+DTC = DPA.

HIERARCHIE LOGIQUE



LA QUESTION À RÉSOUDRE!!!

Q? $\mathrm{PA} = \mathrm{DPA}$?

La question à résoudre!!!

Q? $\mathrm{PA} = \mathrm{DPA}$?

Si oui...

La question à résoudre!!!

Q? $\mathrm{PA} = \mathrm{DPA}$?

Si oui...

Conséquence: PA est clos par complémentaire.

LA QUESTION À RÉSOUDRE!!!

Q? $\mathrm{PA} = \mathrm{DPA}$?

Si oui...

Conséquence: PA est clos par complémentaire.

Conséquence: FO+TC=FO+posTC.

La question à résoudre!!!

Q? $\mathrm{PA} = \mathrm{DPA}$?

Si oui...

Conséquence: PA est clos par complémentaire.

Conséquence: FO+TC=FO+posTC.

Conséquence: FO+TC$\subsetneq$MSO.

Merci.