cf. Baader.

925 Réécriture et formes normales. Exemples.

Motivations
- Vérifier algorithmiquement des identités dans des structures algébriques définies par des théories équationnelles.
- Étudier des fonctions définies récursivement.

I Systèmes de réécriture : Définition et objectifs

1 Théorie équationnelle

Déf 1 Une signature Σ est un langage égalitaire du premier ordre sans symboles de relation. C'est donc une suite $(\Sigma^{(n)})_{n\geq 0}$ où $\Sigma^{(n)}$ est un ensemble de symboles de fonctions d'arité n.
$\Sigma^{(0)}$ est aussi appelé "ensemble des constantes".

Déf 2 Soient Σ une signature et X un ensemble de variables tel que $\Sigma \cap X = \emptyset$.
On définit inductivement l'ensemble des termes $T(\Sigma, V)$ (ou T, si cela n'entraîne pas de confusion) par :
- $X \subseteq T$
- Si $t_1, \dots, t_n \in T$ et $f \in \Sigma^{(n)}$, alors $f(t_1, \dots, t_n) \in T$.

Déf 3
- Soient Σ une signature et X un ensemble dénombrable de variables tq $\Sigma \cap X = \emptyset$.
Une théorie équationnelle sur Σ est un ensemble d'égalités $E \subseteq T \times T$.
- La variété équationnelle associée à E est la classe des modèles égalitaires de E.

Ex 4 (Théorie des groupes)
Signature : $\Sigma^{(0)}_{GR} = \{e\}$, $\Sigma^{(1)}_{GR} = \{i\}$, $\Sigma^{(2)}_{GR} = \{\circ\}$.
Si par exemple $x \in X$, $\circ(x, i(\circ(x, e)))$.
On emploie la notation infixe pour $\circ$ et $^{-1}$ pour i : $x \circ (x \circ e)^{-1}$.
théorie : $E_{GR} = \{(e \circ x, x), (\circ(i(x), x), e), (\circ(\circ(x,y),z), \circ(x, \circ(y,z)))\}$
correspond à $E_{GR} = \{e \circ x \simeq x, x^{-1} \circ x \simeq e, (x \circ y) \circ z \simeq x \circ (y \circ z)\}$

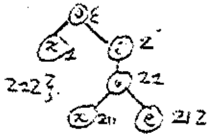
Ex 5 La classe des groupes est la variété équationnelle associée à E_{GR}.
On peut prouver que la classe des corps n'est pas une variété équationnelle.

bof !

Déf 6 On définit inductivement l'ensemble des positions d'un terme s $Pos(s)$, ensemble de suites de nombres positifs :
- si $s \in X$: $Pos(s) = \{\varepsilon\}$
- si $s = f(s_1, \dots, s_n)$, alors $Pos(s) = \{\varepsilon\} \cup \bigcup_{i=1}^{n} \{ip \mid p \in Pos(s_i)\}$

On définit aussi le sous-terme de s en position p par induction :
$s|_\varepsilon := s$ et ~~$s|_{iq}$~~ si $s = f(s_1, \dots, s_n)$, $s|_{iq} := s_i|_q$.

Ex 7 Pour $s = x \circ (x \circ e)^{-1}$:



$Pos(s) = \{\varepsilon, 1, 2, 21, 211, 212\}$.
$s|_{21} = x \circ e$.

Déf 8 Une substitution sur $T(\Sigma, X)$ est une application $\sigma : X \to T$ telle que $\sigma(x) \neq x$ pour un nombre fini de x.
On peut étendre σ par induction à $T(\Sigma, X)$ entier.

2 Systèmes de ~~[illegible]~~ réécriture

Déf 9 Un système de réécriture R est une théorie équationnelle tq si $(s,t) \in R$, alors $s \notin X$ et $Var(t) \subseteq Var(s)$.
On note souvent $s \to t$ pour $(s,t) \in R$.

bof. c déf.

Déf 10 Si R est un système de réécriture, on note $\to_R \subseteq T \times T$ la relation de réduction associée, définie par :
$s \to_R t$ ssi il existe $l \to r \in R$, $p \in Pos(s)$ et une substitution σ tq $s|_p = \sigma(l)$ et $t = s[\sigma(r)]_p$.
(on remplace une instance de l par une instance de r dans le terme s)

Ex 11 E_{GR} est un système de réécriture.
On a $x \circ (x \circ e)^{-1} \longrightarrow_{E_{GR}} x \circ x^{-1} \longrightarrow_{E_{GR}} e$.
Ce que l'on note $x \circ (x \circ e)^{-1} \xrightarrow{+}_{E_{GR}} e$.

On est ici dans le cas où les équations de E_{GR} sont orientées arbitrairement.
Ce qui nous intéresse est alors le problème du mot :
Savoir, étant donnés deux termes s et t si $s \overset{*}{\longleftrightarrow}_{E_{GR}} t$.

On peut aussi symboliser des fonctions récursives par des systèmes de réécriture, et l'orientation des équations n'est plus arbitraire.

…'est terminant ←

…t confluent

Idée de preuve?

Ex 12 (Fonction dérivée)

Signature $\Sigma^{(0)} = \{0, 1, x, y\}$, $\Sigma^{(1)} = \{D_x\}$ et $\Sigma^{(2)} = \{+, \times\}$.

Règles de réécriture
- $D_x(x) \to 1$
- $D_x(y) \to 0$.
- $D_x(u+v) \to D_x(u) + D_x(v)$
- $D_x(u \times v) \to u \times D_x(v) + v \times D_x(u)$.

Def 13 Si E est une théorie équationnelle, on note $\simeq_E = \{(s,t) \in T \text{ tq } E \models s \simeq t\}$ l'ensemble des identités qui sont conséquence sémantique de E.

Thm 14 (Birkhoff) $\xleftrightarrow{*}_E$ la clôture réflexive, transitive et symétrique de $\to_E$ coïncide avec $\simeq_E$.

③ Problèmes algorithmiques

Def 15 Soit R un système de réécriture. On dit que :
- $x \in T$ est réductible s'il existe $y \in T$ tq $x \to_R y$.
- $x \in T$ est sous forme normale s'il n'est pas réductible.
- $x, y \in T$ sont joignables si il existe $z \in T$ tq $x \xrightarrow{*} z \xleftarrow{*} y$. On note aussi $x \downarrow y$.

Ainsi, dans E_{GR}, $x^{-1} \circ x$ et e sont sous forme normale.

Def 16 On dit aussi que $\to_R$ est :
- confluente si $y_1 \xleftarrow{*}_R x \xrightarrow{*}_R y_2$ implique $y_1 \downarrow y_2$.
- terminante si il n'y a pas de chaîne infinie $a_0 \to_R a_1 \to_R \cdots$
- convergente si confluente et terminante.

Prop 17 Si $\to_R$ est
- confluente, chaque élément a au plus une forme normale.
- terminante, chaque élément a au moins une forme normale.

On voit ainsi l'importance de la notion de convergence dans l'étude des variétés équationnelles : pour vérifier si $s \xleftrightarrow{*}_R t$ il suffit de calculer leur unique forme normale, et de vérifier qu'elles sont identiques.

Elle est aussi utile dans l'étude des fonctions récursives :
- $\to_R$ termine si la fonction associée termine. C'est le cas de D_x dans l'exemple 12.
- la confluence correspond au déterminisme : le résultat sera le même, quelle que soit la façon d'effectuer les calculs.

III Terminaison

Thm 18 Le problème TERM est indécidable.
- entrée un système de réécriture
- Sortie oui s'il est terminant.

$\boxed{\text{DVPT 4}}$

La théorie des ordres nous permet d'étudier le problème dans de nombreux cas.

Def 19
- Un ordre de réécriture est un ordre strict $>$ sur $T(\Sigma, V)$ tq
  - compatible avec les Σ-opérations : si $s_1 > s_2$ et $f \in \Sigma^{(n)}$, alors $f(t_1, \dots, t_{i-1}, s_1, t_{i+1}, \dots, t_n) > f(t_1, \dots, t_{i-1}, s_2, \dots, t_n)$
  - fermé par substitution : si $s_1 > s_2$ et σ est une substitution, alors $\sigma(s_1) > \sigma(s_2)$.
- Un ordre de réduction est un ordre de réécriture bien fondé.

Thm 20 Un système de réécriture R termine ssi il existe un ordre de réduction tq $l > r$ pour tout $(l, r) \in R$.

Voyons un exemple particulièrement utile d'ordre de réduction.

Def 21 Soient Σ une signature finie, et $>$ un ordre strict sur les fonctions de Σ. On définit l'ordre de chemin lexicographique $>_{lpo}$ sur $T(\Sigma, V)$ par :

$s >_{lpo} t$ ssi (LPO1) $t \in Var(s)$ et $s \neq t$

(LPO2) $s = f(s_1, \dots, s_m)$, $t = g(t_1, \dots, t_n)$ et
- (2a) $\exists i \in [\![1, m]\!]$ tq $s_i \geq_{lpo} t$.
- (2b) $f > g$ et $s >_{lpo} t_j$ pour tout $j \in [\![1, n]\!]$.
- (2c) $f = g$, $s >_{lpo} t_j$ pour tout j, et il existe $i \in [\![1, n]\!]$ tq $s_1 = t_1, \dots, s_{i-1} = t_{i-1}$, $s_i >_{lpo} t_i$.

Thm 22 $>_{lpo}$ est un ordre de réduction sur $T(\Sigma, V)$.

Ex 23 Avec l'ordre $i > \circ > e$, on a dans $T(\Sigma_{GR}, V)$

1/ $\circ(e, x) >_{lpo} x$, 2/ $\circ(i(x), x) >_{lpo} e$

3/ $\circ(\circ(x, y), z) >_{lpo} \circ(x, \circ(y, z))$

Donc E_{GR} est terminant.

Ex 24 (Fonction d'Ackermann) $\Sigma^{(0)} = \{0\}$, $\Sigma^{(1)} = \{S\}$, $\Sigma^{(2)} = \{a\}$

On considère $R_{Ack} = \{ a(0, y) \to S(y),\ a(S(x), 0) \to a(x, S(0)),\ a(S(x), S(y)) \to a(x, a(S(x), y)) \}$

L'ordre $a > S > 0$ sur Σ permet de montrer que R_{Ack} termine.

Thm 25 Soient Σ une signature finie, et R un système de réécriture fini sur Σ. On peut décider si la terminaison de R peut être prouvée par un lpo.

IV Confluence

Th 26 | Le problème CONFL | Entrée : Un système de réécriture
| Sortie : Oui ssi il est confluent

est indécidable

Def 27 | Un système de réécriture est dit <u>localement confluent</u> ssi
$\forall t, t_1, t_2$ tq $t_1 \leftarrow t \rightarrow t_2$, $t_1 \downarrow t_2$

⚠ La confluence locale n'implique pas la confluence : Fig 1

Lemme 28 de Newman : Si un système de réécriture est terminant, alors la confluence locale implique la confluence.

⟿ Pour étudier la confluence locale, on étudie les paires critiques.

Def 29 Le couple $\langle \theta r_1, \theta l_1[\theta r_2]_p \rangle$ est une <u>paire critique</u> ssi
$(l_1 \rightarrow r_1) \in R$ et $(l_2 \rightarrow r_2) \in R$ et θ m.g.u de $l_1|_p$ et l_2 (variables séparées)

On note $CP(R)$ l'ensemble des paires critiques

<u>Ex</u> : En théorie des groupes, $x.(x^{-1}.y) \leftarrow (x.x^{-1}).y \rightarrow e.y$
et $\langle x.(x^{-1}.y), e.y \rangle$ est une paire critique

Lemme 30 des paires critiques : Si $t_1 \leftarrow t \rightarrow t_2$ et $t_1 \not\downarrow t_2$,
alors $\exists \langle \theta r_1, \theta l_1[\theta r_2]_p \rangle$ paire critique, $\exists i = 0$ ou 1 tq
$t \xrightarrow{(l_1 \rightarrow r_1)} t_i$, $t \xrightarrow{(l_2 \rightarrow r_2)} t_{1-i}$ et $t_1 \downarrow t_2$ ssi $\theta r_1 \downarrow \theta l_1[\theta r_2]_p$

en dvpt plutôt que thm 31 ?

Th 31 Un système de réécriture est localement confluent
ssi toutes les paires critiques sont joignables } DVPT 2

Th 32 La confluence de système de réécriture fini et terminant
est ~~indécidable~~ $\boxed{\text{décidable}}$? (cf Th 26)

V Complétion

<u>But</u> : Construire une procédure effective pour décider l'égalité
(càd un TRS confluent et terminant)

⚠ c'est pas un algo. c'est une $\boxed{\text{procédure}}$ ici

<u>Knuth-Bendix</u> <u>Entrée</u> : • un ensemble fini d'égalités E
• un ordre de réduction $>$

<u>Sortie</u> : Si la procédure termine sans échec :
<u>Un système</u> de réécriture confluent, terminant, compatible avec E

$R := \{ \text{Ordonner}(s,t) \quad (s,t) \in E \}$

<u>Tant qu'il existe</u> $\langle s,t \rangle \in CP(R)$, $s \not\downarrow t$
  $R := R \cup \{ \text{Ordonner}(\hat{s}, \hat{t}) \quad , \hat{s} = s\downarrow, \hat{t} = t\downarrow, \langle s,t \rangle \in CP(R) \}$

Retourner R

<u>Ordonner</u> (s,t)
<u>Si</u> $s \not\equiv t$ et $s \not> t$ et $t \not> s$: <u>Echec</u>
<u>Sinon</u> | si $s > t$ alors Retourner $(s \rightarrow t)$
| si $t > s$ alors Retourner $(t \rightarrow s)$

unification : Hors sujet ?

$$\begin{tikzcd} c & a \arrow[l] \arrow[r, bend right] & b \arrow[l, bend right] \arrow[r] & d \end{tikzcd}$$

Fig 1

Pb du mot : Entrée : système de E,
$s, t \in T$. ?
Sortie : oui ssi $s \overset{*}{\underset{E}{\longleftrightarrow}} t$

⚠ À la déf de $\overset{*}{\longleftrightarrow}$
ça veut dire $t \searrow^{*} t_1 \swarrow^{*} t_2 \searrow^{*} \cdots \searrow^{*} s$.
(= clôture symétrique transitive).

. Pas bcp d'exs ?
↳ Exs des grammaires ?, (regarder dans le carton ?).

• Formes normales ??

Thm des paires critiques)

[Thm] Un système de réécriture R est localement confluent ssi toutes ses paires critiques sont joignables.

On note $T(\Sigma, V)$ l'ens. des termes de variables V et signature Σ.

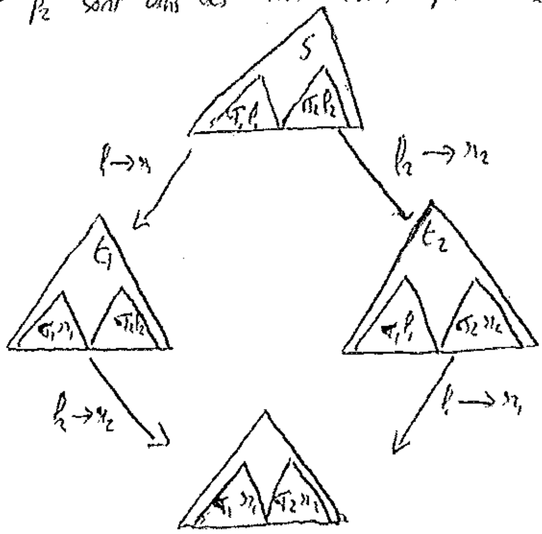
Preuve

Soient $s \in T(\Sigma, V)$, des
- règles $l_i \to r_i \in R$ $(i = 1, 2)$
- positions $p_i \in Pos(s)$
- substitutions σ_i avec $s/_{p_i} = \sigma_i l_i$

, et notons $t_i = s[\sigma_i r_i]_{p_i}$ $(i = 1, 2)$.

On a donc $t_1 \xleftarrow{l_1 \to r_1} s \xrightarrow{l_2 \to r_2} t_2$. On veut savoir si t_1 et t_2 sont joignables.

Cas 1 p_1 et p_2 sont dans des sous-arbres séparés : la confluence locale a bien lieu.

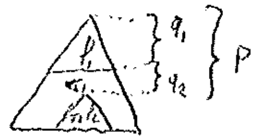


Cas 2 p_1 est un préfixe de p_2 : donc il existe un mot p tq $p_2 = p_1 p$.

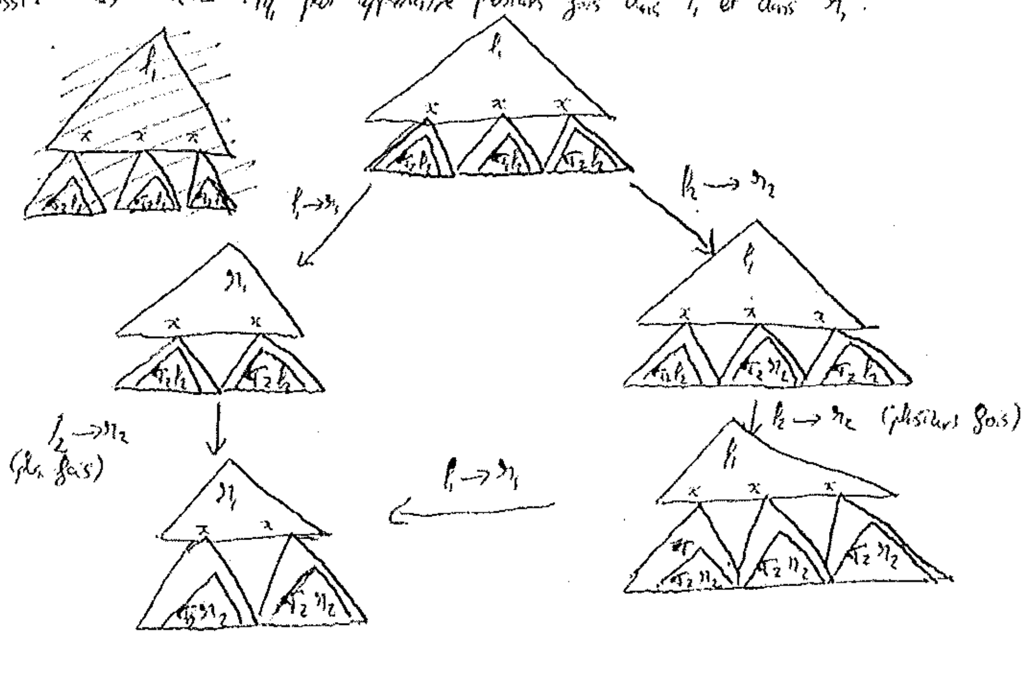
La relation $\xrightarrow{}_{p_1}$ étant compatible avec le contexte, on peut se ramener au cas $p_1 = \varepsilon$.

Cas 2.1 $\sigma_2 l_2$ n'apparaît pas dans l_1, mais dans σ_1.

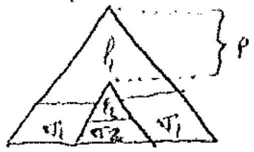
Formellement : $p = q_1 q_2$, où q_1 $l_1/_{q_1} \in V$.



La confluence locale tient aussi : mais $x := l_1/_{q_1}$ peut apparaître plusieurs fois dans l_1 et dans r_1.



<u>Cas 2.2.</u> Il existe $p \in Pos(l_1)$, $l_1|_p \notin V$, avec $\sigma_1(l_1|_p) = \sigma_2 l_2$.



Montrons que c'est une instance d'une paire critique.

<u>Déf.</u> Soient $l_i \to r_i \in R$ $(i=1,2)$ où les variables ont été renommées de sorte que $Var(l_1, r_1) \cap Var(l_2, r_2) = \emptyset$.
~~On sup~~ Soit $p \in Pos(l_1)$ tq $l_1|_p \notin V$, et on suppose qu'il existe un mgu θ de $l_1|_p =? l_2$.
On dit alors que $< \theta r_1, (\theta l_1)[\theta r_2]_p >$ est une <u>paire critique</u>.

Dans notre situation, supposons qu'on est dans ~~la situation~~ le cas où $Var(l_1, r_1) \cap Var(l_2, r_2) = \emptyset$.

On a donc $Dom(\sigma_1) \cap Dom(\sigma_2) = \emptyset$, et la substitution $\sigma = \sigma_1 \cup \sigma_2$ est bien définie.

Mais alors $\sigma(l_1|_p) = \sigma_1(l_1|_p) = \sigma_2 l_2 = \sigma l_2$, donc σ est un unificateur de $l_1|_p$ et l_2.

Il existe ainsi un mgu θ de $l_1|_p$ et l_2, et $< \sigma r_1, (\sigma l_1)[\sigma r_2]_p >$ est une instance de la paire critique $< \theta r_1, (\theta l_1)[\theta r_2]_p >$.

<u>Conclusion</u> Si $s \to_R t_i$ $(i=1,2)$, alors $t_1 \downarrow_R t_2$ ou $t_i = s[u_i]_p$ $(i=1,2)$ où $< u_1, u_2 >$ ou $< u_2, u_1 >$ est une instance d'une paire critique

<u>Preuve du théorème</u> $\boxed{\Leftarrow}$ Soient ~~$t_1 t_2$~~ $t_i = s[u_i]_p$ $(i=1,2)$ où $< u_1, u_2 >$ est une instance d'une paire critique $< v_1, v_2 >$: il existe une substitution δ tq $u_i = \delta v_i$.

On a $v_1 \downarrow_R v_2$: soit $t \in T(\Sigma, V)$ tq $v_i \xrightarrow{*}_R t$ $(i=1,2)$, alors $u_i \xrightarrow{*}_R \delta t$ et donc $t_i \xrightarrow{*}_R s[\delta t]_p$.

Donc $t_1 \downarrow_R t_2$ et R est localement confluent.

$\boxed{\Rightarrow}$ Si $< \theta r_1, (\theta l_1)[\theta r_2]_p >$ est une paire critique on a $\theta r_1 \leftarrow \theta l_1 \to (\theta l_1)[\theta r_2]_p$

et par locale confluence, $\theta r_1 \downarrow (\theta l_1)[\theta r_2]_p$.

$\boxtimes$

Terminaison d'un système de réécriture

__Th 1__ : __TERM__ | Entrée : Un S.R R | est indécidable
| Sortie : Oui ssi R est terminant

__Dem__ : Par réduction de ATC (ARRET_TOUTE_CONF(#)) :

| Entrée : Une M.T. M
| Sortie : Oui ssi $\forall K$ configuration de M, il n'existe pas de calcul infini partant de K.

$\# \in \Gamma$: symbole vide

Soit M une M.T. : $M = (\Gamma, Q, q_i, F, \Delta)$, on va construire R_M tel que

($\exists K$, config de M, M effectue un calcul infini à partir de K)

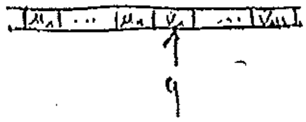
ssi ($\exists t$, un terme et une suite infinie de réécritures à partir de t)

__Termes__ : $\Sigma = \Sigma^{(1)} = \vec{\Gamma} \cup Q \cup \overleftarrow{\Gamma} \cup \{\vec{l}, \overleftarrow{r}\}$

$V = \{x_i, i \in \mathbb{N}\}$

$\longrightarrow$ On représentera un terme par un mot de Σ^* : $a_1(\ldots(a_n(x))\ldots) \rightsquigarrow a_1 \ldots a_n$

__Def__ : On appelle terme de configuration (t.c) un terme du langage $(\vec{\Gamma})^* Q (\overleftarrow{\Gamma})^*$

$t : \vec{u}_1 \ldots \vec{u}_n \, q \, \overleftarrow{v}_1 \ldots \overleftarrow{v}_m \quad \longleftrightarrow \quad K_t =$ 

__Prop 1__ . Soit t un t.c, $\exists! K$ tq $K = K_t$

. Soit K, $\exists t$ tq $K = K_t$

On construit alors les règles de la manière suivante :

- Pour tout $(q,a) \longrightarrow (q',b,r) \in \Delta$, on ajoute

$$q(\overleftarrow{a}(x)) \longrightarrow_R \overrightarrow{b}(q'(x))$$

et si $a = \#$, $\quad q(\overleftarrow{\pi}(x)) \longrightarrow_R \overrightarrow{b}(q'(\overleftarrow{\pi}(x)))$ $\qquad$ à R_M

- Pour tout $(q,a) \longrightarrow (q',b,l) \in \Delta$, on ajoute

$$\forall c \in \Gamma \quad \overrightarrow{c}(q(\overleftarrow{a}(x))) \longrightarrow_R q'(\overleftarrow{c}(\overleftarrow{b}(x)))$$

$$\text{et } \overrightarrow{\ell}(q(\overleftarrow{a}(x))) \longrightarrow_R \overrightarrow{\ell}(q'(\overleftarrow{\#}(\overleftarrow{b}(x))))$$

et si $a = \#$, $\quad \forall c \in \Gamma, \quad \overrightarrow{c}(q(\overleftarrow{\pi}(x))) \longrightarrow_R q'(\overleftarrow{c}(\overleftarrow{b}(\overleftarrow{\pi}(x))))$

$$\text{et } \overrightarrow{\ell}(q(\overleftarrow{\pi}(x))) \longrightarrow_R \overrightarrow{\ell}(q'(\overleftarrow{\#}(\overleftarrow{b}(\overleftarrow{\pi}(x)))))$$

Par construction, on a la propriété suivante :

Prop2 :
- Soient t, t' deux termes. Si t est un t.c. et $t \longrightarrow_R t'$, alors t' est un t.c. et $K_t \longrightarrow_M K_{t'}$
- Soient K, K' tq $K \longrightarrow_M K'$, alors $\forall t$ tq $K = K_t$, $\exists t'$ tq $K' = K_{t'}$, $t \longrightarrow_R t'$

Montrons alors Th 1 : $\boxed{\Rightarrow}$ Soit $(K_i)_{i\in\mathbb{N}}$ tq $K_i \longrightarrow_M K_{i+1}$ une suite infinie de calculs. Par les propositions 1 et 2, on peut construire $(t_i)_{i\in\mathbb{N}}$ tq $t_i \longrightarrow t_{i+1}$ suite infinie de réécritures.

$\boxed{\Leftarrow}$ Soit t un terme donnant lieu à une suite infinie de réécritures. Il existe une unique décomposition de t de la forme $t = u_1 t_1 \ldots u_n t_n u_{n+1}$ où les t_i sont des termes de configuration de taille maximale.

Puisque toute réécriture se passe à l'intérieur d'un t.c. et donne lieu à un autre t.c., $\exists i$ tq t_i donne lieu à une suite infinie de réécritures

ce qui nous permet, grâce aux propriétés 2 et 3, de trouver un calcul infini de M.

Remarque, on peut, avec cette construction, prouver le théorème suivant :

<u>Th</u> : TERM' Entrée : Un S.R. R, un terme t
Sortie : Oui ssi il n'existe pas une suite infinie de réécritures partant de t

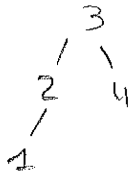
<u>Preuve</u>. Par réduction du problème de l'arrêt.

ARRET Entrée : Une MT M, un mot d'entrée ω
Sortie : Oui ssi il n'existe pas de calcul infini de M sur l'entrée ω.

On a immédiatement :

$$(M,\omega) \in \text{ARRET} \quad \text{ssi} \quad (R_M, K_0)_\sigma \in \text{TERM}' \quad \text{où } K_0 = q_1(\overleftarrow{\omega}(\triangleright))$$

Questions :

- Ens. total^T ordonné _ Est-ce qu'on en a besoin tout le temps ?
- Vous connaissez des ens. tot^T ordonnés ?
- ______ d'autres hyp. qu'on pourrait faire sur les clés ?
- Améliorer suppression avec les pointeurs ? [Suppr. en O(1)] ?
  ↳ listes doublement chainées
- AVL :  + insérer 0

- recherche par rangs
  ↳ il y a pas un algo linéaire ?
  [En parler ?]

Intersection de segments

↳ Comment on calcule si et où deux segments s'intersectent ?

- Est-ce qu'il existe une structure de données avec les avantages des AVL, avec successeur et prédécesseur en O(1) ?
  ↳ On sait pas...

Plan :
↳ Recherche de motif ds un texte // arbre courant
Pas dans le cadre de cette leçon.
↳ Arbres radix pq pas ?

Questions : - Pourquoi opérat^{ions} AVL en $O(\lg n)$
- Pourquoi la hauteur est en $O(\lg n)$?
(R donnée. Considérer l'arbre de hauteur h avec le moins de nœuds (→ arbres de Fibonacci).)

- Arbre rouge/noir → c'est quoi ?
- lien codage Huffman, ABR optimaux ?

App : - programmation dynamique - accès aux valeurs intermédiaires