Workshop Trickle groups.
25/06/25

Systèmes de réécriture et solution au problème du mot.

I. Systèmes de réécriture : une introduction
II. L'algorithme de Trickle.
III. L'algorithme "à la Tits"

I. Systèmes de réécriture.

1) Exemple introductif.

Soient a, b, c des générateurs d'un groupe abélien libre $G=\langle$
Comment savoir si les mots

$$x = aabacbba\,bcbaba$$
$$y = baacbababcbabc.$$

Sont égaux ? Pour le savoir, on réalise mentalement l'opération

$$x \to a^6b^6c^2 \qquad y \to a^5b^6c^3.$$

Autrement dit :
- on range les générateurs dans l'ordre alphabétique
- on se convainc que les formes normales ainsi obtenues n'étant pas égales, les éléments x et y ne sont pas égaux.

2). Définitions.

On fixe A un ensemble (lettres) ; A^* le monoïde libre (mots)
ε le mot vide et $A^* = A^* \setminus \varepsilon$.

Def : Un système de réécriture R est un sous ensemble de $A^+ \times A^*$
(ses éléments sont les "règles" $u \to v$ (pour $(u,v) \in R$))

Exemple : Dans notre groupe abélien libre du départ.

$ba \to ab \quad ca \to ac \quad cb \to bc$.

Plus généralement, on déduit un monoïde de R via

$$M = \langle A \mid u = v \text{ pour } u \to v \in R \rangle.$$

Réciproquement, à partir d'un monoïde présenté on peut fabriquer un système de réécriture en "orientant les relations".

On définit maintenant une liste de relations associées à R.

- $w \xrightarrow{R} w'$ s'il existe w_1, w_2 tels que $w = w_1 u w_2$, $w' = w_1 v w_2$ et $(u,v) \in R$. (on notera $\to \; = \; \xrightarrow{R}$ s'il n'y a pas d'ambiguïté).
- $w \xrightarrow{*} w'$ la clôture réflexive transitive de $\to$.

  $w \xrightarrow{*} w' \iff \exists\; w = w_0 \to w_1 \to \dots \to w_n = w'$.

Déf : Un mot w est irréductible si $\nexists\, w'$ tel que $w \to w'$.

Déf : R est terminant/Noethérien s'il n'existe pas de suite infinie $(w_k)_{k \geq 0}$ telle que $w_k \to w_{k+1}$ $\forall k \geq 0$.

Déf : R est confluent si

$$\begin{tikzcd}
 & v_1 \arrow[dr, dashed, "\exists\, *"] & \\
u \arrow[ur, "*"] \arrow[dr, "*"'] & & w \\
 & v_2 \arrow[ur, dashed, "\exists\, *"'] &
\end{tikzcd}$$

Déf : R est complet si noethérien et confluent

Exemples :

① Groupe abélien libre $\langle a_1, \dots, a_n \rangle$. Le système $\{a_j a_i \to a_i a_j \mid i < j\}$ est terminant et confluent.

② Pour $\mathcal{B}_3$ $ab \leftarrow bc \leftarrow ca$ est $\overset{\text{homogène}}{\text{terminant}}$ mais pas confluent

$$bcab \to bbcb \to babb \qquad bcab \to abab.$$

③ Pour $\mathcal{B}_3$ toujours $aba \leftarrow bab$ est terminant car homogène mais pas confluent.

$$babab \to baaba \qquad babab \to abaab$$

④. Pour un groupe libre $\langle a_1 \dots a_n \rangle$. $R = \{a_i a_i^{-1} \to \varepsilon \;\; a_i^{-1} a_i \to \varepsilon \mid i \in [\![1,n]\!]\}$ est terminant et confluent : on retrouve la notion classique de mot réduit.

[Newman 4]

3) Conséquences

Prop : Si R est un système de réécriture terminal et confluent sur A, alors

(a) $\forall w \in A^*$, $\exists !\, f(w) \in A^*$ irred et tel que $w \xrightarrow{*} f(w)$.

(b) f induit une injection $\overline{f} : \Pi \hookrightarrow A^*$ (la forme normale).

dém : (a). On construit $f(w)$ par récurrence (terminal). (Si l'existence est fausse dans (a), alors on construit facilement une suite infinie qui contredit la noethérianité).

Ensuite, si $w \xrightarrow{*} w'$ avec R irred, alors on a

$$\begin{array}{ccc} & w' & \\ w & & w'' \\ & f(w) & \end{array} \qquad w \xrightarrow{*} w',\; w \xrightarrow{*} f(w),\; w' \xrightarrow{*} w'',\; f(w) \xrightarrow{*} w''$$

par confluence

mais par irréductibilité, $w' \xrightarrow{*} w'' \xleftarrow{*} f(w)$ n'est possible que si $f(w) = w'' = w'$. d'où (a).

On voit que $w \xrightarrow{*} w'$ dans $A \Rightarrow f(w) = f(w')$ car $f(w')$ est irréductible et $w \xrightarrow{*} w' \xrightarrow{*} f(w')$.

(b). Soient $w, w' \in A^*$ représentant le même élément de Π. On a alors

$$w = w_0 \sim w_1 \cdots \sim w_m = w'$$

avec $\sim = \leftarrow$ ou $\rightarrow$. d'où $f(w) = f(w')$. $\square$.

II. L'algorithme de Trickle.

On fixe $(\Gamma, \leq, \mu, (\varphi_x)_{x \in V(\Gamma)})$ un trickle graph, et

$$T_r(\Gamma) = \langle V(\Gamma) \mid x^{\mu(x)} = 1,\ \varphi_x(y)x = x\varphi_y(x) \text{ for } \{x,y\} \in E(\Gamma) \rangle$$

le groupe de trickle associé.

Au besoin, on fixe un ordre total $\sqsubseteq$ qui raffine $\leq$

1) Définitions

(On pose $\mathbb{Z}_k = \mathbb{Z}/k\mathbb{Z}$)

Les lettres du système que nous allons étudier sur $T_r(\Gamma)$ ne sont pas $V(\Gamma) \cup V(\Gamma)^{-1}$, mais un ensemble de <u>strates</u>.

* $S(\Gamma) = \{x^a \mid x \in V(\Gamma)$ et $a \in \mathbb{Z}_{\mu(x)} \setminus 0\}$ l'ensemble des <u>syllabes</u> (fini ssi $\mu(x) < \infty$ pour tout x).

* Une <u>strate</u> $U = \{x_1^{a_1}, \ldots, x_p^{a_p}\}$ où $\{x_i, x_j\} \in E(\Gamma)$ $i \neq j$ (en part. $x_i \neq x_j$). Pour tout sous-graphe complet $G \subseteq \Gamma$, on a $\prod_{x \in V(G)} \mu(x)$ strates ayant $V(G)$ pour <u>support</u>.
  + $p = \lg_{st}(U)$ en est la longueur
  + par convention, $\emptyset$ est une strate de support $\emptyset$ et de longueur 0.

L'ensemble $\Omega = \Omega(\Gamma)$ des strates est notre alphabet

* Ajouter y^b dans U : C'est possible si $y \in \mathrm{Supp}\,U$, ou si $\{y, x\} \in C$ pour tout $x \in \mathrm{Supp}(U)$. (i.e si adjoindre y au support de U laisse un sous graphe complet de Γ).

- Si $y \notin \mathrm{Supp}(U)$ et peut être ajouté, alors

$$R(U, y^b) = \{ \varphi_y^{-b}(x_1)^{a_1}, \dots, \varphi_y^{-b}(x_p)^{a_p}, y^b \}$$

↳ à mettre au bon endroit pour C

- Si $y = x_i \in \mathrm{Supp}(U)$ alors soit

* $b + a_i = 0 \in \mathbb{Z}_{\mu(y)}$ et

$$R(U, y^b) = \{ \varphi_y^{-b}(x_1)^{a_1} \dots, \overbrace{\varphi_y^{-b}(x_i)}^{}{}^{a_i}, \dots \varphi_y^{-b}(x_p)^{a_p} \}.$$

* $b + a_i \neq 0 \in \mathbb{Z}_{\mu(y)}$ et

$$R(U, y^b) = \{ \varphi_y^{-b}(x_1)^{a_1} \dots, \varphi_y^{-b}(x_i)^{a_i + b}, \dots \varphi_y^{-b}(x_p)^{a_p} \}$$

$$= \{ \varphi_y^{-b}(x_1)^{a_1}, \dots, y^{a_i + b}, \dots, \varphi_y^{-b}(x_p)^{a_p} \}$$

car $y = x_i = \varphi_y^{-b}(x_i)$.

En fait, pour $x_i \sqsupset y$, on a $\varphi_y^{-b}(x_i) = x_i$. Donc on peut plus généralement écrire

$$R(U, y^b) = \{ x_1^{a_1} \dots x_{i-1}^{a_{i-1}}, x_i^{a_i + b}, \varphi_y^{-b}(x_p)^{a_p} \dots \varphi_y^{-b}(x_p)^{a_p} \}$$

Là encore, le mot associé à $R(U, y^b)$ est bien $U \cdot y^b$.

(pour $y \sqsupset x_i$, on a $x_i y = \varphi_y(\varphi_y^{-1}(x_i)) y = y \, \varphi_y^{-1}(x_i)$. (on pense φ_y comme une conjugaison).

Un élément de Ω^* est donc de la forme $u = U_1 \dots U_k$ où les U_i sont des strates. On pose $\lg_{pi}(u) = k$ et on appelle u un piling (empilement).

Rq : $V(\Gamma)^{\pm 1} \subseteq \Omega(\Gamma)$, donc on obtient facilement un ensemble de générateurs de $T\pi(\Gamma)$.

2) Fonctionnement.

Soient deux pilings U, V. Le système de reecriture de trickle consiste à faire passer des syllabes de V à U.

À partir d'ici, on numérote les syllabes dans une strate par ordre $\sqsubseteq$ décroissant.

Soit $V = \{x_1^{a_1} \dots x_p^{a_p}\}$ une strate. et $i \in [\![1, p]\!]$.

* Retirer $x_i^{a_i}$ de V : on obtient $L(V, x_i^{a_i}) = \{x_1^{a_1}, \dots, \widehat{x_i^{a_i}}, \dots, x_p^{a_p}\}$.

* Si $j \sqsupset i$, alors $x_j^{a_j} x_i^{a_i} = \left(\varphi_{x_j}^{a_j}(x_i)\right)^{a_i} x_j^{a_j}$ par définition d'un trickle group. Par récurrence, on a donc.

$$x_1^{a_1} \dots x_i^{a_i} = \underbrace{\left[\varphi_{x_1}^{a_1} \circ \varphi_{x_2}^{a_2} \dots \circ \varphi_{x_{i-1}}^{a_{i-1}}(x_i)\right]^{a_i}}_{\gamma(V, x_i^{a_i})} x_1^{a_1} \dots x_{i-1}^{a_{i-1}}$$

En identifiant la strate $\{x_1^{a_1} \dots, x_p^{a_p}\}$ au mot $x_1^{a_1} \dots x_p^{a_p}$ (ce qu'on peut faire grace à notre ordre total, on a

$$V = \gamma(V, x_i^{a_i}) \cdot L(V, x_i^{a_i}).$$

Rq : en fait, $\gamma(V, x_i^{a_i})$ ne dépend pas du choix de $\sqsubseteq$. car $\{x_j^{a_j} \in V \mid x_j \leq x_i\}$ est déjà totalement ordonné par $\leq$. (cond. (b))
Par cond (c), ces éléments sont les seuls tels que φ_{x_j} engine non triviale.

Avec ces opérations, on peut (enfin) définir notre système de réécriture ! $\mathcal{R}$:

* $\emptyset \to \varepsilon \quad \in \Omega^* \times \Omega^*$

* $UV \to R(U, y^b)\, L(V, x^b)$ où $x^b \in V$ est tel que $y^b = \gamma(V, x^b)$ peut être ajoutée à U.

Si Π désigne le monoïde présenté par R, on note déjà que l'application $\Omega^* \to T\pi(\Gamma)$ envoyant un piling sur le mot que nous lui avons associé induit un morphisme $\Pi \to T\pi(\Gamma)$ car

$$UV = U\,\gamma(V\, x^b)\, L(V, x^b) = U y^b\, L(V, x^b) = R(U, y^b)\, L(V, x^b).$$

<u>Théo (2.4)</u> le morphisme $\Pi \to T\pi(\Gamma)$ est un isomorphisme de monoïde (en particulier, Π est un groupe) De plus, R est un système de réécriture complet

<u>Rq</u> : l'application $\Omega^* \to T\pi(\Gamma)$ ne dépend pas du choix de $\exists$.

<u>Cor</u> : L'application $S(\Gamma) \to T\pi(\Gamma)$ est injective

<u>Cor</u> : $T\pi(\Gamma)$ est fini ssi Γ est un graphe fini complet et $\mu(x) \neq \infty\ \forall x$.

III. L'algorithme "À la Tits".

Les strates forment un ensemble assez gros, on préférerais un système de générateurs plus petit. Par exemples les syllabes.

$S(\Gamma) \to Tr(\Gamma)$ induit $S(\Gamma)^* \to Tr(\Gamma)$.

Pour $g \in \Gamma$, on pose $\ell_S(g)$ pour la plus petite longueur d'un élément de $S(\Gamma)^*$ qui exprime g.

On considère trois systèmes de reecriture.

$$R_I : \quad x^a x^{-a} \to \varepsilon \qquad x^a x^b \to x^{a+b}$$

$$R_{II} : \quad x^a \, y^b \to \varphi_x^a(y)^b \, \varphi_y^{-b}(x)^a , \qquad x - y \in E(\Gamma).$$

$$R_M = R_I \cup R_{II}.$$

- Si $w \xrightarrow{I} w'$, alors $\ell(w') < \ell(w)$.
- Si $w \xrightarrow{II} w'$ alors $\ell(w) = \ell(w')$ et $w' \xrightarrow{II} w$.

Donc aucune chance que R_M soit terminant !

Theo 2.8 : Soit $g \in Tr(\Gamma)$, représenté par $w \in S(\Gamma)$. LASSE

(i). $\ell(w) = \ell_S(g)$ (ii) il n'y a aucun mot w' tel que $w \xrightarrow{M}{}^{*} w'$ et $\ell(w') < \ell(w)$

Si w, w' sont deux mots de longueur minimale representant g, alors $w \xrightarrow{H} w'$

(Matsumoto like)