Recherche de motif

Référence : 131 Développements pour l'oral, D.Lesesvre, P. Montagnon.

Introduction. Le problème de recherche d'un motif M dans un texte T consiste à déterminer si M apparaît comme facteur (sous-mot) de T. On pose $T = t_1...t_n$ et $M = m_1...m_k$, et pour $1 \leqslant i \leqslant k$, on pose $M_i = m_1...m_i$ le préfixe de M de longueur i. On pose aussi par convention $M_0 = \epsilon$.

Si $u, v \in \Sigma^*$, on note $u \sqsubset v$ qi u est suffixe de v. On note aussi

$$\sigma(u) = \max\{i : M_i \sqsubset u\}$$

c'est-à-dire la taille du plus grand préfixe de M qui est également suffixe de u.

But : On veut construire un algorithme résolvant le problème en $\mathcal{O}(n)$ avec un prétraitement polynomial en k.

Construction d'un automate. Soit $A = (Q, \Sigma, I, F, \delta)$ l'automate déterministe complet défini par :

- $Q = \{0, ..., k\}$;
- $I = \{0\}$;
- $F = \{k\}$;
- pour tout $q \in Q$ et $a \in \Sigma$, $\delta(q, a) = \sigma(M_q a)$.

Lemme 1. *Pour tout mot $u \in \Sigma^*$, on a $\delta^*(0, u) = \sigma(u)$. Ainsi, $L(A) = \Sigma^* M$.*

Un premier lemme.

Lemme 2. *Soient $u, v \in \Sigma^*$ et $a \in \Sigma$.*

1. *Si $u \sqsubset v$, alors $\sigma(u) \leqslant \sigma(v)$.*
2. *$\sigma(ua) \leqslant \sigma(u) + 1$.*
3. *$\sigma(ua) = \sigma(M_{\sigma(u)} a)$*

Remarque 1. *Si $M_i \sqsubset u$ ($0 \leqslant i \leqslant k$), alors $i \leqslant \sigma(u)$. De plus, on a $\sigma(M_i) = i$.*

Démonstration.

1. Si $u \sqsubset v$, alors tout suffixe de u est suffixe de v. Ainsi, on a $\{i : M_i \sqsubset u\} \subset \{i : M_i \sqsubset v\}$, et donc $\sigma(u) \leqslant \sigma(v)$.
2. Si $\sigma(ua) = 0$, alors alors puisque $\sigma(u) \geqslant 0$, l'inégalité est vérifiée. Sinon, on a $\sigma(ua) - 1 \geqslant 0$. On remarque alors que $M_{\sigma(ua)-1} \sqsubset u$, et ainsi,

$$\sigma(ua) - 1 \leqslant \sigma(u)$$

3. On montre le résultat par double inégalité.
   - Par définition $M_{\sigma(u)} \sqsubset u$ et donc $M_{\sigma(u)} a \sqsubset ua$ et donc d'après le point 1, $\sigma(M_{\sigma(u)a}) \leqslant \sigma(ua)$.

— Pour l'autre inégalité, on observe que $M_{\sigma(u)}a$ et $M_{\sigma(ua)}$ sont deux suffixes de ua. Ainsi, le plus court des deux mots est suffixe de l'autre. D'après le point 2, on a

$$|M_{\sigma(ua)}| = \sigma(ua) \leqslant \sigma(u) + 1 = |M_{\sigma(u)}a|$$

Ainsi, $M_{\sigma(ua)} \sqsubset M_{\sigma(u)}a$, d'où

$$\sigma(ua) = \sigma(M_{\sigma(ua)}) \leqslant \sigma(M_{\sigma(u)}a)$$

d'après le point 1.

□

Démonstration du lemme 1. On procède par récurrence sur la longueur l de u.

- Si $l = 0$, alors $u = \epsilon$ et $\delta^*(0, \epsilon) = 0 = \sigma(\epsilon)$.
- Supposons que $l > 0$ et que pour tout mot $v \in \Sigma^{l-1}$ vérifie $\delta^*(0, v) = \sigma(v)$. Soit $u \in \Sigma^l$, qu'on écrit $v = ua$ avec $|u| = l - 1$ et $a \in \Sigma$. En appliquant l'hypothèse de récurrence :

$$\delta^*(0, u) = \delta(\delta^*(0, v), a) = \delta(\sigma(v), a) = \sigma\left(M_{\sigma(v)}a\right) = \sigma(va) = \sigma(u)$$

Cela conclut la récurrence.

Enfin, par définition de σ, $\sigma(u) = k$ si et seulement si $M \sqsubset u$. Ainsi,

$$u \in L(A) \Leftrightarrow \delta^*(0, u) = k \Leftrightarrow \sigma(u) = k \Leftrightarrow M \sqsubset u \Leftrightarrow u \in \Sigma^* M$$

et donc $L(A) = \Sigma^* M$. □

Lemme 3. *Pour $i, j \in Q$ tels que $i > j$, le mot $m_{i+1}...m_k$ est accepté par A à partir de i mais pas de j. Ainsi, A est minimal.*

Démonstration. On note $N_i = m_{i+1}...m_k$ le suffixe de M de taille $k - i$. Par construction, on a $\delta^*(0, M_i) = i$, d'où

$$\delta^*(i, N_i) = \delta^*(0, M_i N_i) = \delta^*(0, M) = k$$

Ainsi, N_i est accepté à partir de i dans A.

Pour $j < i$, alors $|M_j N_i| = j + k - i < k = |M|$. Ainsi, M n'est pas un suffixe de $M_j N_i$ et donc

$$\delta^*(j, N_i) = \delta^*(0, M_j N_i) = \sigma(M_j N_i) < k$$

donc N_i n'est pas accepté à partir de j dans A.

On en déduit de même que si $i > j$, les états i et j de A ne sont pas équivalents pour l'équivalence de Nérode. Comme A est déterministe et complet, ceci assure que A est minimal.

□

Algorithme. La construction de A (et plus particulièrement de sa fonction de transition) δ n'utilise que le motif m et non le texte T. On peut représenter cette fonction via un tableau bidimensionnelle de taille $(k + 1) \times |\Sigma|$. On peut alors remplir cette table avec les différentes valeurs de σ, ce qui se fait en temps polynomial en k.

Enfin, on lit le mot T lettre par lettre. Si on atteint l'état k, on a trouvé une occurrence de du motif M. Si on atteint la fin de Tsans jamais atteindre k, alors M n'apparaît pas dans T.