Validation et génération de tableaux de Knuth-Morris-Pratt

Jean-Pierre Duval, Thierry Lecroq et Arnaud Lefebvre
{Jean-Pierre.Duval,Thierry.Lecroq,Arnaud.Lefebvre}@univ-rouen.fr

Laboratoire d'Informatique, Traitement de l'Information, Systèmes
Université de Rouen

Algorithmique, combinatoire du texte et applications en bio-informatique
26, 27 et 28 septembre 2007 – Marne-la-Vallée

litis

UR
UNIVERSITÉ DE ROUEN

Plan

Plan

Recherche exacte de mot

Problème

Trouver une ou plus généralement toutes les occurrences d'un mot x de longueur m dans un texte y de longueur n.

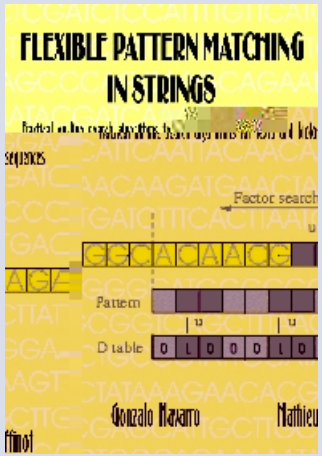
x and y sont construits sur un alphabet A de cardinal c.

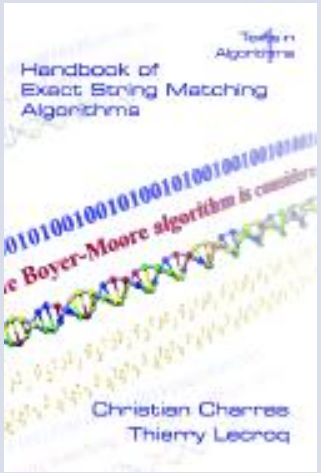
Recherche exacte de mot

Solutions

Beaucoup ! !

Voir `http://monge.univ-mlv.fr/~lecroq/string`





Les plus connues : Knuth–Morris–Pratt et Boyer–Moore, 1977

Recherche exacte de mot

Exemple

y	a	b	a	c	a	b	a	d	a	b	a	c	a	b	a	b	$\cdots$
x	a	b	a	c	a	b	a	d	a	b	a	c	a	b	a	d	
MP									a	b	a	c	a	b	a	d	
KMP													a	b	a	c	
AFD															a	b	

Bord

Définition

Un mot u est un bord d'un mot w si u est à la fois un préfixe et un suffixe de w tel que $u \neq w$.

Définition

Le bord d'un mot w est le plus long de ses bords. Il est noté $Bord(w)$.

Remarque

Le bord d'un bord d'un mot w est un bord de w.

Tableau de bords

Définition

Soit $w[1\,..\,n]$ un mot de longueur n, le tableau f_w défini par

$$f_w[i] = |Bord(w[1\,..\,i])|$$

pour $1 \leq i \leq n$ est appelé le tableau de bords de w.

fonction de suppléance (ou « failure function ») de l'algorithme de MP

$O(n)$ en temps et en espace
Au plus $2n - 3$ comparaisons de lettres

Tableau de KMP

Soit $w[1\,..\,n]$ un mot de longueur n, le tableau f défini par $g_w[1] = 0$ et

$g_w[j] = \max\{i \mid w[1\,..\,i-1]$ suffixe de $w[1\,..\,j-1]$ et $w[i] \neq w[j]\}$

ou de manière équivalente

$g_w[j] = 1+\max\{i \mid w[1\,..\,i]$ bord de $w[1\,..\,j-1]$ and $w[i+1] \neq w[j]\}$

pour $2 \leq j \leq n$ est appelé le tableau de KMP de w.

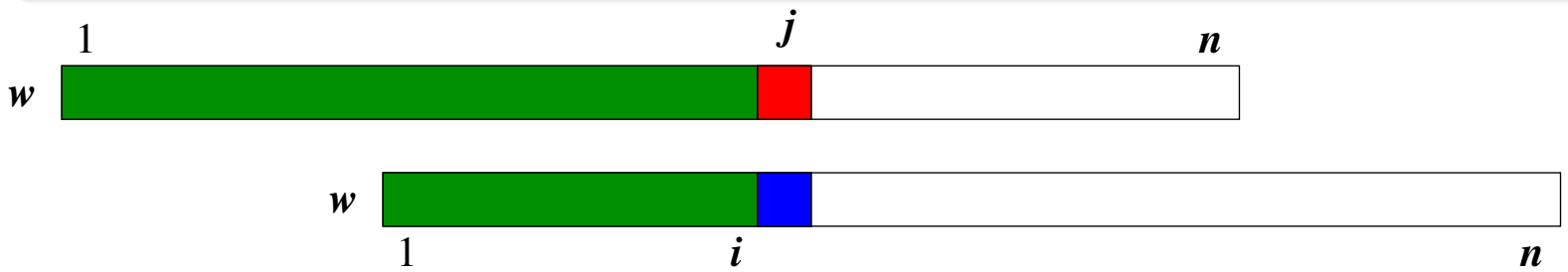


Tableau de KMP

fonction de suppléance (ou « failure function ») de l'algorithme de KMP

$O(n)$ en temps et en espace
Au plus $2n - 3$ comparaisons de lettres

Tableaux de bords et de KMP

Exemple

i	1	2	3	4	5	6	7	8	9	10	11	12	12	14	15
$w[i]$	a	b	a	b	a	c	a	a	b	c	a	b	a	b	a
$f_w[i]$	0	0	1	2	3	0	1	1	2	0	1	2	3	4	5
$g_w[i]$	0	1	0	1	0	4	0	2	1	3	0	1	0	1	0

Tableaux de bords et de KMP

2 mots x et y peuvent avoir le même tableau de KMP et des tableaux de bords différents

Exemple

$x =$ abaab et $y =$ abacb.

i	1	2	3	4	5
$x[i]$	a	b	a	a	b
$f_x[i]$	0	0	1	1	2
$g_x[i]$	0	1	0	2	1

i	1	2	3	4	5
$y[i]$	a	b	a	c	b
$f_y[i]$	0	0	1	0	0
$g_y[i]$	0	1	0	2	1

$\mathcal{D}(w)$

Définition

L'AFD $\mathcal{D}(w)$ qui reconnaît le langage A^*w est défini par $\mathcal{D}(w[1\,..\,n]) = (Q, A, q_0, T, F)$ où

- $Q = \{0, 1, \ldots, n\}$ est l'ensemble des états ;
- A est l'alphabet ;
- $q_0 = 0$ est l'état initial ;
- $T = \{n\}$ est l'ensemble des états terminaux ;
- $F = \{(i, w[i+1], i+1) \mid 1 \leq i \leq n\} \cup \{(i, a, |Bord(w[0\,..\,i]a)|) \mid 0 \leq i < n \text{ and } a \in A \setminus \{w[i+1]\}\}$ est l'ensemble des transitions.

Le graphe sous-jacent est appelé le *squelette* de l'automate.

$\delta(i)$ et $\delta'(i)$

Définition

For $0 \leq i \leq n$:

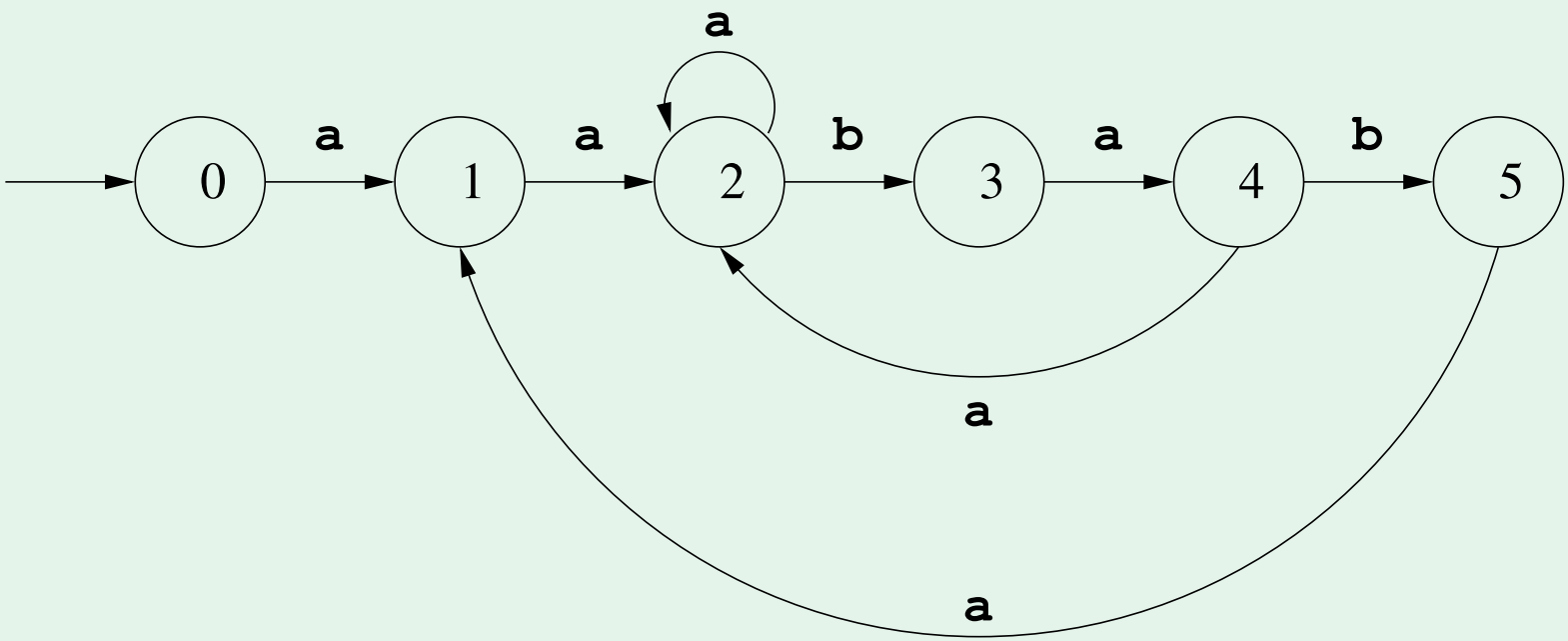
- $\delta(i) = (j \mid (i, a, j) \in F$ avec $a \in A$ et $j \neq 0)$;
- $\delta'(i) = (j \mid (i, a, j) \in F$ avec $a \in A$ et $j \notin \{0, i+1\})$.

Autrement dit :

- $\delta(i)$ est la liste des cibles des transitions significatives sortant de l'état i ;
- $\delta'(i)$ est la liste des cibles des transitions arrières sortant de l'état i.

AFD

Exemple



$\mathcal{D}(\mathtt{aabab})$: les transitions arrivant à l'état 0 sont omises.

$$\delta(4) = (5, 2) \text{ et } \delta'(4) = (2).$$

AFD

Théorème (Simon 1993)

Il y a au plus n transitions arrière significatives dans $\mathcal{D}(w[1\,..\,n])$.

Valide

Définition

Un tableau d'entiers $f[1\,..\,n]$ est un tableau de bords valide si et seulement si il est le tableau de bords d'au moins un mot $w[1\,..\,n]$.

Définition

Un tableau d'entiers $g[1\,..\,n]$ est un tableau de KMP valide si et seulement si il est le tableau de KMP d'au moins un mot $w[1\,..\,n]$.

Les problèmes principaux

Validation

Étant donné un tableau d'entiers, est-il un tableau de bords valide ? Est-il un tableau de KMP valide ? Sur un alphabet de quel cardinal ?

Construction d'un mot

Étant donné un tableau de bords (resp. de KMP) valide, donner un mot pour lequel ce tableau est le tableau de bords (resp. de KMP)

Construction de tableaux valides

Construire tous les tableaux de bords (de KMP) distincts pour une longueur donnée

Motivations

Générer des ensembles de tests minimaux pour divers algorithmes de recherche exacte de mot

Travaux antérieurs


D. Moore, W. F. Smyth et D. Miller.
Counting distinct strings.
Algorithmica, 23(1) :1–13, 1999.


F. Franěk, S. Gao, W. Lu, P. J. Ryan, W. F. Smyth, Y. Sun et L. Yang.
Verifying a border array in linear time.
Journal on Combinatorial Mathematics and Combinatorial Computing, 42 :223–236, 2002.


J.-P. Duval, T. Lecroq et A. Lefebvre.
Border array on bounded alphabet.
Journal of Automata, Languages and Combinatorics, 10(1) :51–60, 2005.


J.-P. Duval, T. Lecroq et A. Lefebvre.
Efficient validation and construction of border arrays
Actes des Journées Montoises, Rennes, France, 2006, 179-189.

Sur la toile

`http://al.jalix.org/Baba/Applet/baba.php`

Plan

Validation

Exemple

i	1	2	3	4	5	6	7	8	9	10	11	12	12	14	15	16
$f[i]$	0	0	1	2	3	0	1	1	2	0	1	2	3	4	5	?

Les candidats pour $f[16]$ sont

- $f[15]+1 = 5+1 = 6$
- $f[f[15]]+1 = 3+1 = 4$
- $f[f[f[15]]]+1 = 1+1 = 2$
- $f[f[f[f[15]]]] = 0+1 = 1$
- 0

Parmi ces valeurs 2 n'est pas valide car $f[4] = 2$.

Les candidats

Définition

Pour $1 \leq i \leq n$, on définit

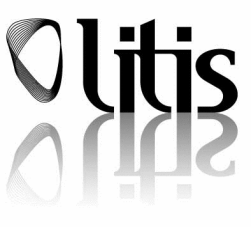
- $f^1[i] = f[i]$; et
- $f^\ell[i] = f[f^{\ell-1}[i]]$ pour $f[i] > 0$;
- $C(f,i) = (1 + f[i-1], 1 + f^2[i-1], \ldots, 1 + f^m[i-1])$ où $f^m[i-1] = 0$.

Validation

Proposition (DLL 2005)

Il y a deux conditions nécessaires et suffisantes pour qu'un tableau d'entiers f soit un tableau de bords valide :

1. $f[1] = 0$ *et pour* $2 \leq i \leq n$, *on doit avoir* $f[i] \in (0) \uplus C(f, i)$;
2. *pour* $i \geq 2$ *et pour chaque* $j' + 1 \in C(f, i)$ *tel que* $j' + 1 > f[i]$, *on doit avoir* $f[j' + 1] \neq f[i]$.



$f \longrightarrow \delta$

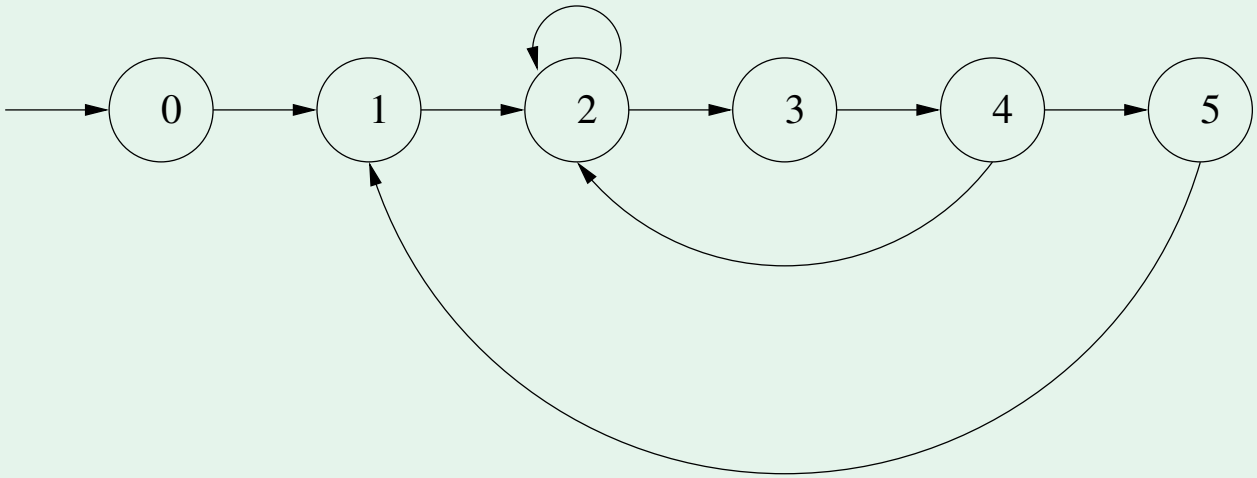
Proposition (DLL 2006)

$\delta(0) = (1)$ *et*

$\delta(j) = (j+1) \uplus \delta(f[j]) \cup (f[j+1])$ *pour* $1 \leq j < n$ *et*

$\delta(n) = \delta(f[n])$.

Exemple



$j+1$	$\uplus$	$\delta(f[j])$	$\uplus$	$f[j+1]$	$=$	$\delta(j)$	j
(1)	$\uplus$		$\uplus$		$=$	(1)	0
(2)	$\uplus$	(1)	$\uplus$	(1)	$=$	(2)	1
(3)	$\uplus$	(2)	$\uplus$		$=$	(3,2)	2
(4)	$\uplus$	(1)	$\uplus$	(1)	$=$	(4)	3
(5)	$\uplus$	(2)	$\uplus$		$=$	(5,2)	4
	$\uplus$	(1)	$\uplus$		$=$	(1)	5

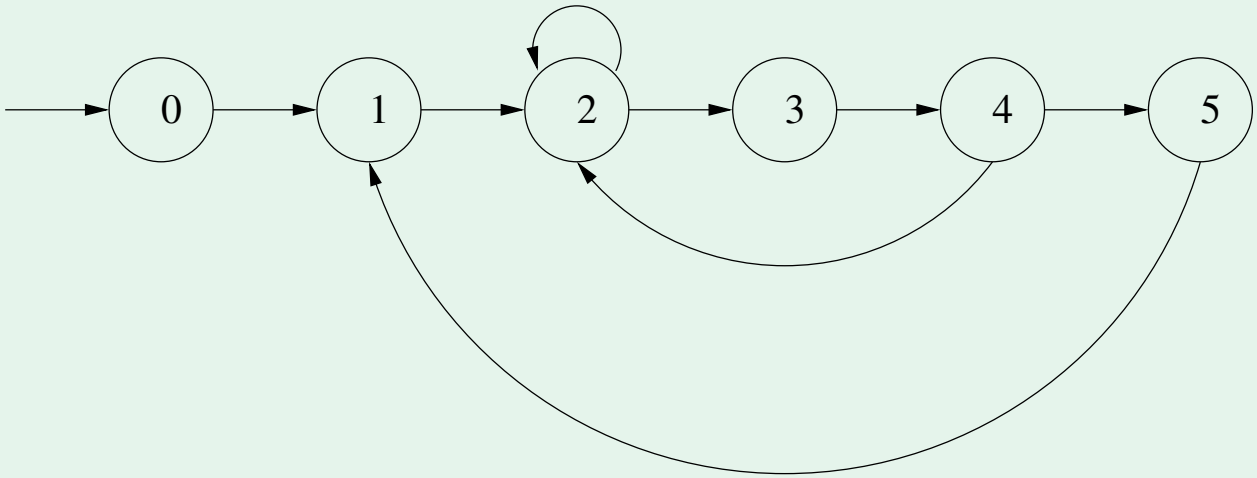
$\delta \longrightarrow f$

Corollaire (DLL 2006)

Pour $j > 0$ *:*

$$f[j+1] = \begin{cases} \delta(f[j]) \uplus \delta(j) & \text{si } \delta(f[j]) \uplus \delta(j) \text{ n'est pas vide,} \\ 0 & \text{sinon.} \end{cases}$$

Exemple



$\delta(f[j])$	$\sqcup$	$\delta(j)$	$=$	$f[j+1]$	j
	$\sqcup$	(1)	$=$	0	0
(1)	$\sqcup$	(2)	$=$	1	1
(2)	$\sqcup$	(3,2)	$=$	0	2
(1)	$\sqcup$	(4)	$=$	1	3
(2)	$\sqcup$	(5,2)	$=$	0	4

Indépendance par rapport à l'alphabet

Important

Ces calculs sont complètement indépendant des mots sous-jacents.

Algorithme

Supposons que $f[1\,..\,i]$ est un tableau de bords valide, tous les candidats pour $f[i+1]$ sont dans $\delta'(i) \uplus (0)$ et ils n'ont pas besoin d'être vérifiés.

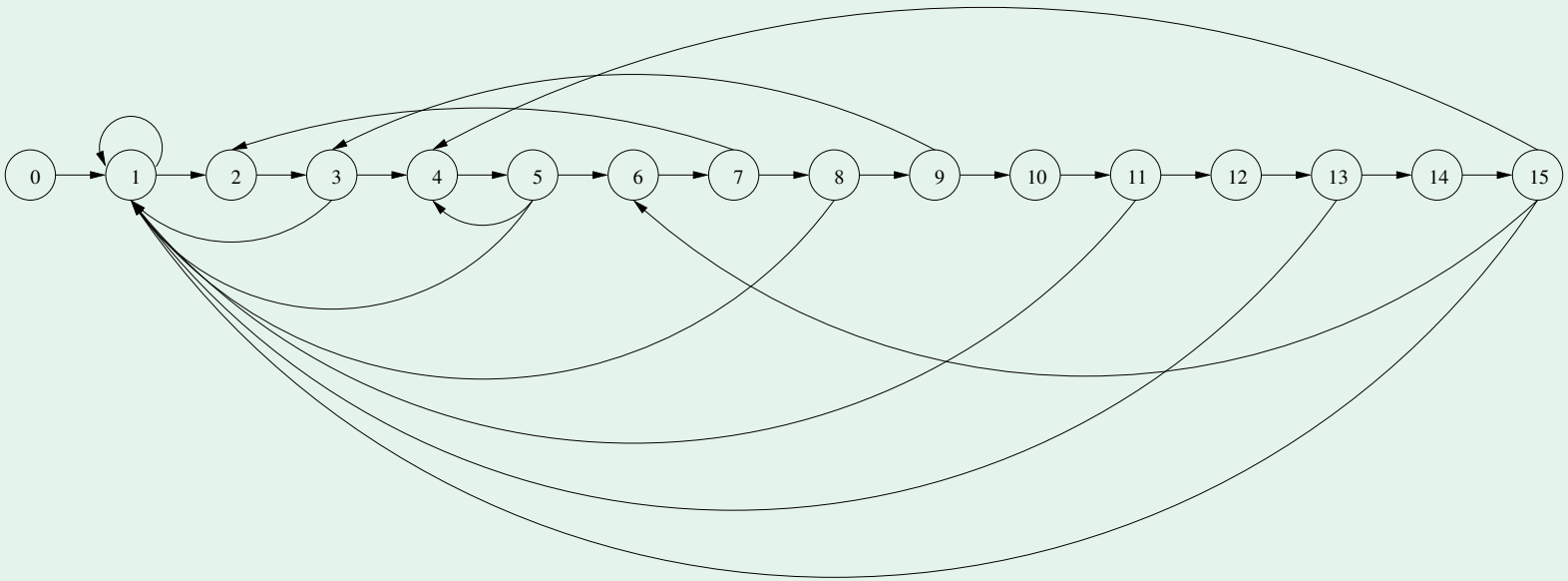
Le squelette de l'automate est construit en même temps que la vérification du tableau f.

Si $f[i+1]$ est égal à 0, il suffit de vérifier que la cardinalité de $\delta'(i)$ est strictement plus petite que la cardinalité s de l'alphabet pour s'assurer que f est valide jusqu'à la position $i+1$.

Algorithme

Exemple

i	1	2	3	4	5	6	7	8	9	10	11	12	12	14	15	16
$f[i]$	0	0	1	2	3	0	1	1	2	0	1	2	3	4	5	?



Les candidats pour $f[16]$ sont dans $\delta'(15) \uplus (0) = (6, 4, 1, 0)$.

Complexité

Théorème (DLL 2006)

La validité d'un tableau de n entiers $f[1\,..\,n]$ peut être vérifiée en temps et espace $O(n)$.
Si f est un tableau de bords valide, un mot w pour lequel f est le tableau de bords peut être calculé avec les mêmes complexités.

Reconnaissance d'un automate

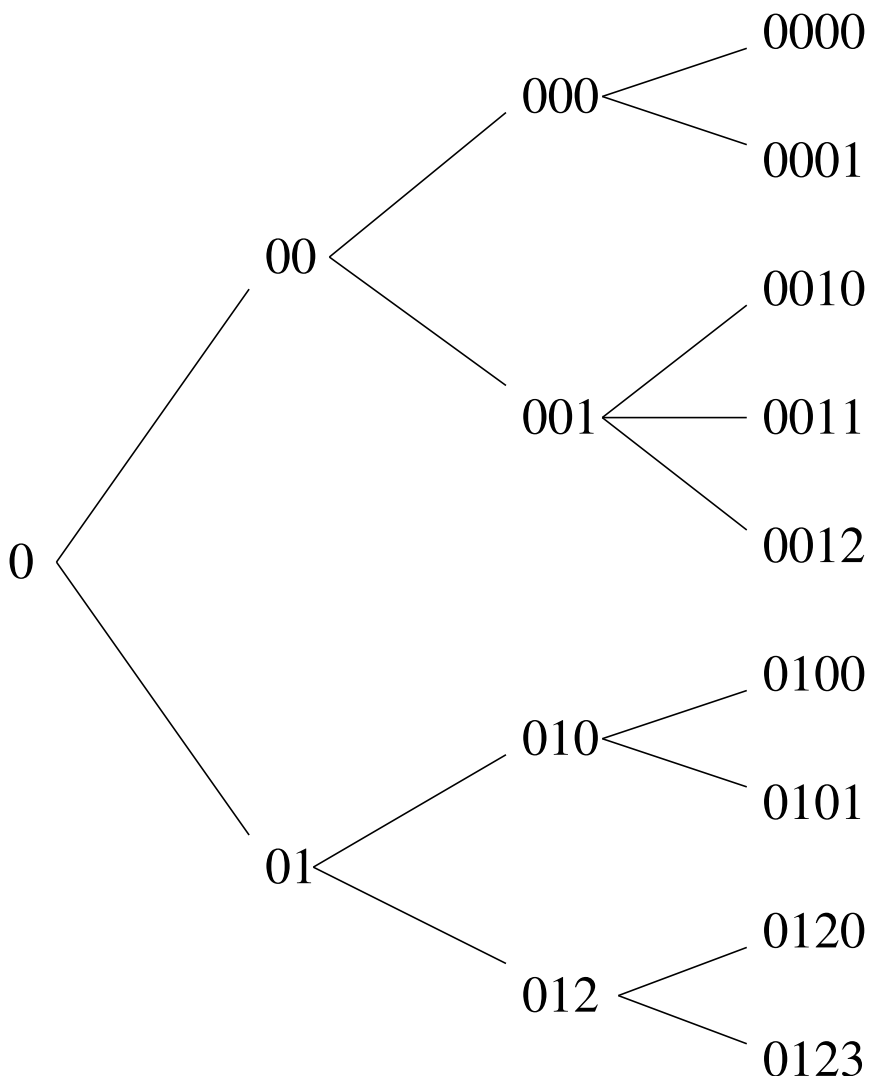
Conséquence

Il est possible de vérifier si un graphe donné peut être le graphe reconnaissant A^*w et de calculer w en temps et espace linéaire.

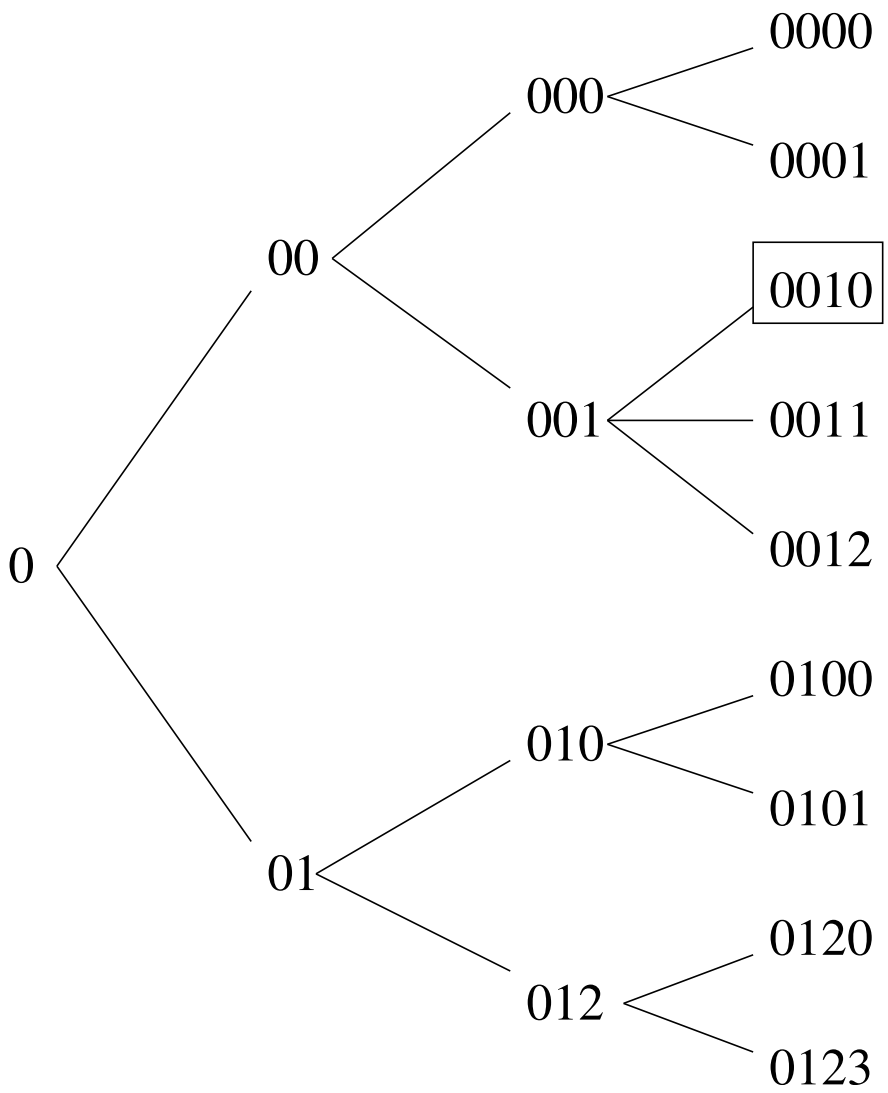
Construction des tableaux de bords

Un algorithme pour générer tous les tableaux de bords distincts devient alors évident : tous les candidats valides pour $f[i]$ sont dans $\delta'(i-1) \uplus (0)$.

Construction des tableaux de bords



Construction des tableaux de bords



abac

Construction des tableaux de bords

- temps proportionnel au nombre
- espace linéaire

Comptage

i	$B(i)$	$B(i,2)$	$B(i,3)$	$B(i,4)$
1	1	1	1	1
2	2	2	2	2
3	4	4	4	4
4	9	**8**	9	9
5	20	16	20	20
6	47	32	47	47
7	110	64	110	110
8	263	128	**262**	263
9	630	256	626	630
10	1525	512	1509	1525
11	3701	1024	3649	3701
12	9039	2048	8872	9039
13	22 140	4096	21 640	22 140
14	54 460	8192	52 993	54 460
15	134 339	16 384	130 159	134 339
16	332 439	32 768	320 696	**332 438**

Nombre de tableaux de bords distincts

Proposition

$B(n, 2) = 2^{n-1}$.

Proposition

$B(j, s) = B(j)$ *pour* $j < 2^s$.

Nombre de tableaux de bords distincts

Proposition

$B(2^s, s) = B(2^s) - 1.$

Le tableau de bords manquant à la forme suivante :
$0..2^0 - 1 \cdot 0..2^1 - 1 \cdots 0..2^{s-1} - 1.$

Il correspond au mot $w_s \cdot \sigma[s+1]$ *(de longueur* 2^s*) où*
w_s *est défini récursivement par :*
$w_1 = \sigma[1]$ *et*
$w_i = w_{i-1} \cdot \sigma[i] \cdot w_{i-1}$ *pour* $i > 1$.

Exemple

Le tableau suivant $f[1\,..\,16]$ est un tableau de bords valide sur un alphabet de cardinalité au moins 5 :

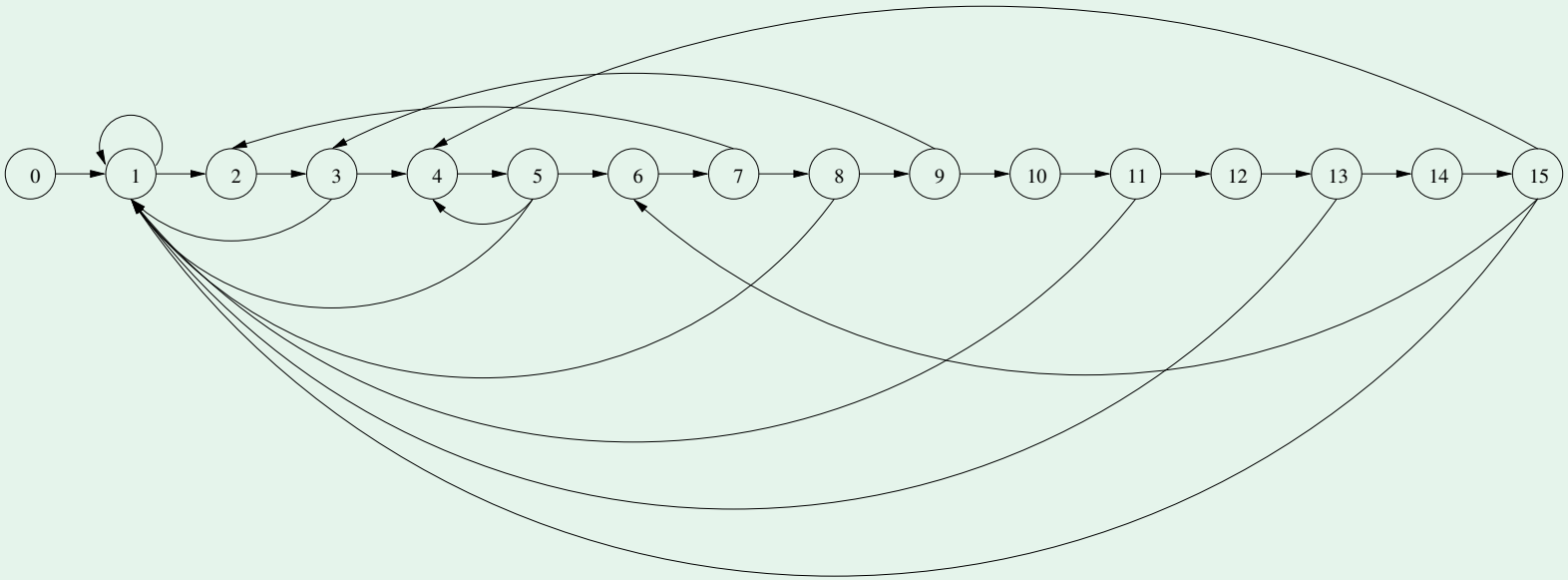
i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$w_4[i]$	a	b	a	c	a	b	a	d	a	b	a	c	a	b	a	e
$f[i]$	0	0	1	0	1	2	3	0	1	2	3	4	5	6	7	0

Plan

Tableaux de KMP

Exemple

i	1	2	3	4	5	6	7	8	9	10	11	12	12	14	15	16
$f[i]$	0	0	1	2	3	0	1	1	2	0	1	2	3	4	5	?
$g[i]$	0	1	0	1	0	4	0	2	1	3	0	1	0	1	0	?



Les candidats pour $g[16]$ sont :

- 6 et alors les candidats pour $f[16]$ sont dans $\delta'(15) \uplus (0) \uplus (6) = (4, 1, 0)$;
- $g[6] = 4$ et alors $f[16] = 6$.

Tableaux de KMP

Proposition

$g[i+1]$ peut être égal soit à $f[i]+1$ soit à $g[f[i]+1]$.

Proposition

Si $g[i+1] = g[f[i]+1]$ alors $f[i+1] = f[i]+1$.

Proposition

Si $g[i+1] = f[i]+1$ alors $f[i+1]$ peut être une valeur parmi $\delta'(i) \uplus (f[i]+1) \uplus (0)$.

Tableaux de KMP

Complexité

Le tableau de bords $0 \cdot 1 \cdot 0 \cdot (2 \cdot 1 \cdot 0)^* \cdot (1|2 \cdot 0|2 \cdot 1 \cdot 1)$ nécessite les nombres d'appels à la procédure de vérification suivants :

- $3(((n/3) \times (n/3) + 1)/2)$ si $n \bmod 3 = 1$;
- $2 + 3((((n+1)/3) \times ((n+1)/3) + 1)/2) - n/3$ si $n \bmod 3 = 0$;
- $2 + 3((((n-1)/3) \times ((n-1)/3) + 1)/2) + n/3 + 1$ si $n \bmod 3 = 2$.

Tableaux de KMP

Exemple

$f = 01021021021021021 1$

1					
2					
3					
4					
5	5				
6	6				
7	7				
8	8	8			
9	9	9			
10	10	10			59 appels
11	11	11	11		
12	12	12	12		
13	13	13	13		
14	14	14	14	14	
15	15	15	15	15	
16	16	16	16	16	
17	17	17	17	17	
18	18	18	18	18	
				19	

Tableaux de KMP

Conjecture

Il est possible de vérifier si un tableau d'entiers $g[1\,..\,n]$ est un tableau de KMP valide en temps $O(n^2)$.

Comptage

Exemple

1	2	3	4	5	6	7
a	b	a	c	a	b	a

							b
0	0	1	0	1	2	3	2
0	1	0	2	0	1	0	4

							a
0	0	1	0	1	2	3	1
0	1	0	2	0	1	0	4

							d
0	0	1	0	1	2	3	0
0	1	0	2	0	1	0	4

							c
0	0	1	0	1	2	3	4
0	1	0	2	0	1	0	2

Construction des tableaux de bords

- temps proportionnel au nombre
- espace proportionnel au nombre

Comptage

i	$K(i)$	$K(i,2)$	$K(i,3)$	$K(i,4)$	i	$K(i)$	$K(i,2)$	$K(i,3)$	$K(i,4)$
1	1	1	1	1	10	1106	512	**1104**	1106
2	2	2	2	2	11	2656	1024	2644	2656
3	4	4	4	4	12	6414	2048	6365	6414
4	8	8	8	8	13	15 582	4096	15 406	15 582
5	17	**16**	17	17	14	38 011	8192	37 430	38 011
6	37	32	37	37	15	93 124	16 384	91 317	93 124
7	85	64	85	85	16	228 927	32 768	223 524	228 927
8	197	128	197	197	17	564 674	65 536	548 969	564 674
9	465	256	465	465	18	1 396 860	131 072	1 352 193	**1 396 859**

Comptage

$K(5,2) = K(5) - 1$: le tableau de KMP manquant est $0 \cdot 1 \cdot 0 \cdot 2 \cdot 0$, il correspond à abaca.

$K(10,3) = K(10) - 2$: les 2 tableaux de KMP manquants sont $0 \cdot 1 \cdot 0 \cdot 2 \cdot 0 \cdot 1 \cdot 0 \cdot 4 \cdot 0 \cdot 1$ et $0 \cdot 1 \cdot 0 \cdot 2 \cdot 0 \cdot 1 \cdot 0 \cdot 4 \cdot 1 \cdot 1$, ils correspondent à abacabadab et abacabadbb respectivement.

$K(18,4) = K(18) - 1$: le tableau de KMP manquant est $0 \cdot 1 \cdot 0 \cdot 2 \cdot 0 \cdot 1 \cdot 0 \cdot 4 \cdot 0 \cdot 1 \cdot 0 \cdot 2 \cdot 0 \cdot 1 \cdot 0 \cdot 8 \cdot 1 \cdot 1$, il correspond à abacabadabacabaebb.

Comptage

Proposition

Soit $g_1 = 0$.
Soit $g_i = g_{i-1} \cdot 2^i \cdot g_{i-1}$ *pour* $i > 1$.
Pour $i \geq 4$, $K(2^i + 2, i) = K(2^i + 2) - 1$ *: le tableau de KMP manquant est* $g_i \cdot 1 \cdot 1$, *il correspond à* $w_i \cdot \sigma[2] \cdot \sigma[2]$.

Plan

Conclusion

- $f \leftrightarrow \delta$
- validation de f et de δ
- calcul des f
- validation de g
- calcul des g

Perspectives

- complexité de la validation de g
- formules pour le nombre de f et g