Validation de tableaux de bords

T. Lecroq

Université de Rouen

J-.P. Duval et A. Lefebvre

Plan

Rappels

- Bords, tableaux de bords, MP
- Automate □*.*w*

Définitions

- Valide
- Squelette

Questions

- Tableau de bords valide ?
- Mots associés ?
- Squelette associé ?

Notations

□ : alphabet

f : $f[1..n]$

étendu à $f[0]=-1$

tableau de bords d'un mot $w[1..n]$

R : relation associée à f

$i \, R \, j$ ssi $(i-1) \, f \, (j-1)$

□' : automate déterministe reconnaissant

le langage □$^*.w$

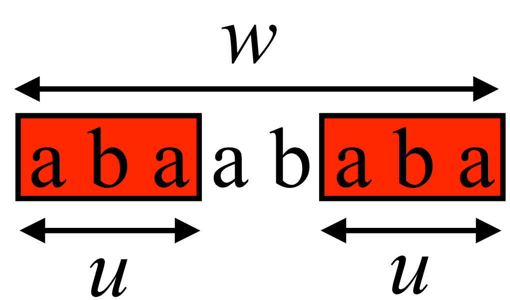
□ : squelette de □'

(flèches sans étiquettes)

Tableau de bords

Le plus long bord d'un mot *w* est le plus long préfixe et suffixe de *w* différent de *w* (par défaut le mot vide).

Exemple :



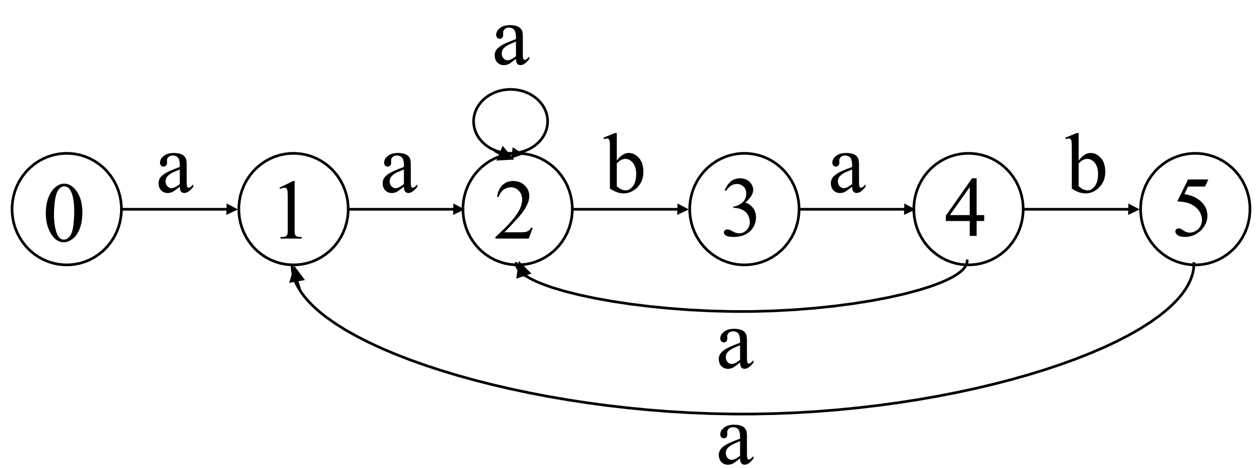
i	0	1	2	3	4	5	6	7	8
w		a	b	a	a	b	a	b	a
f	-1	0	0	1	1	2	3	2	3

Par définition, *f* est le *tableau de bords* de *w*.
(fonction de faille de Morris et Pratt [AHU, 1974])

Rappels

i	0	1	2	3	4	5
$w[i]$		a	a	b	a	b
$f[i]$	-1	0	1	0	1	0

□’

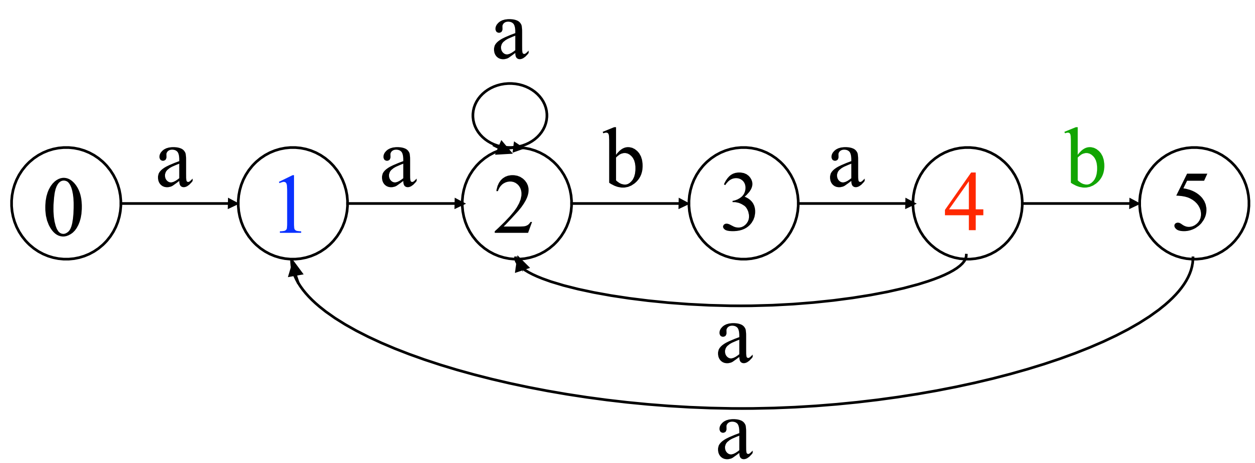


Automate reconnaissant le langage □*.w

Les flèches d’extrémité 0 sont implicites.

La construction de □' est classique

□'



$□'(j, w[j+1]) = j+1$

$□'(j, c) = □'(k, c) \qquad (c \neq w[j+1])$

$j = j+1$

$k = □'(k, w[j])$

$□'(j, w[j+1]) = j+1$

$□'(j, c) = □'(f[j], c)$

$f[j+1] = □'(f[j], w[j+1])$

$j = j+1$

Remarque : la connaissance explicite de f n'est pas indispensable à ce stade.

Le problème étudié

Définition : un tableau $f[1..n]$ est un tableau de bords *valide* si et seulement si il est le tableau de bords d'au moins un mot $w[1..n]$.

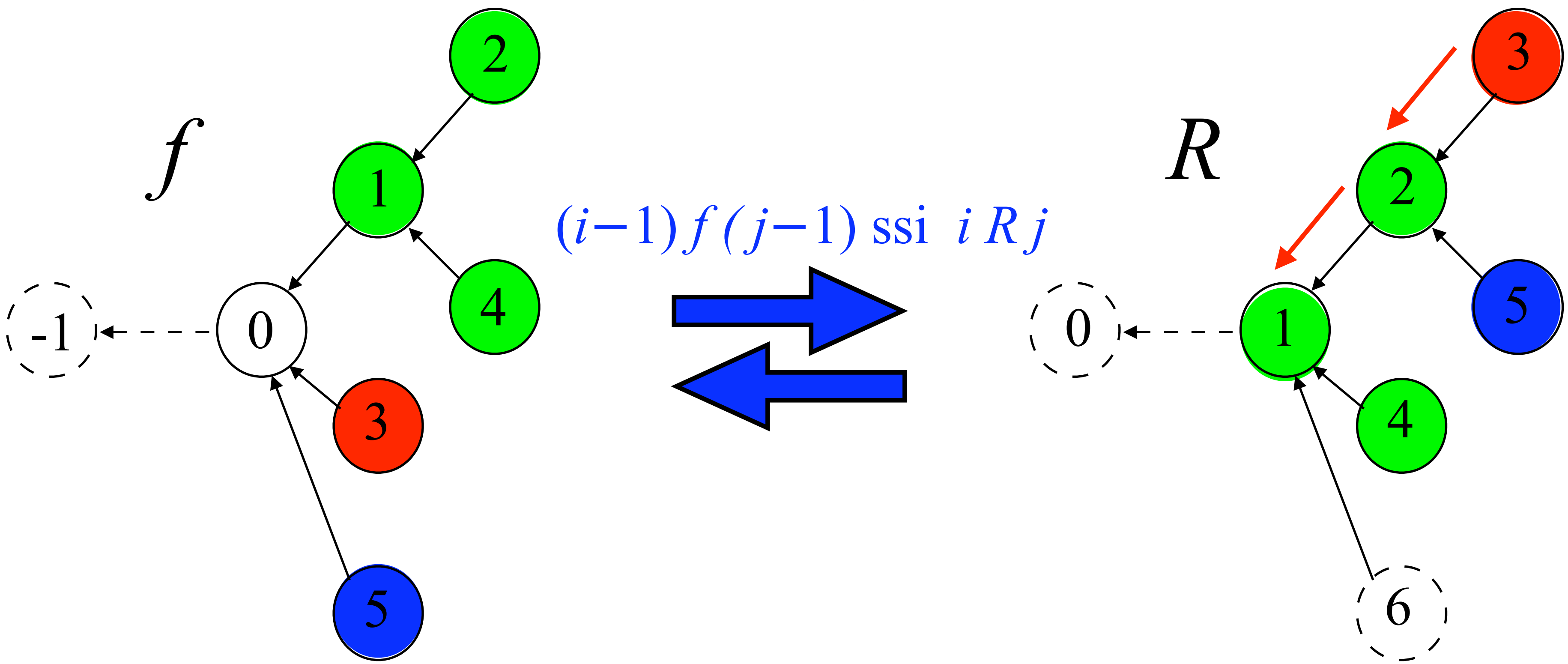
Question : un tableau f donné est-il valide ?

Introduite et en partie résolue par [Franěk *et al, 1999*], pour laquelle nous avons apporté une réponse plus simple à mettre en œuvre [Duval *et al, 2005*].

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

C'est ce problème que nous revisitons ici de manière originale en l'étendant à □' non étiqueté que nous appelons *squelette* □ *associé à* □'.

$f, R, \overline{f} \ldots$



$\overline{f}$: fermeture réflexive, symétrique et transitive de f sur $[1..n]$

$\overline{f}$-classe de 1, 2, 4

$\overline{f}$-classe de 3

$\overline{f}$-classe de 5

R-chemin de 3

f est-il valide ?

Étant donné un tableau $f\,[1..n]$ de n entiers positifs, existe-t-il un mot w tel que f soit son tableau de bords ?

Proposition 1

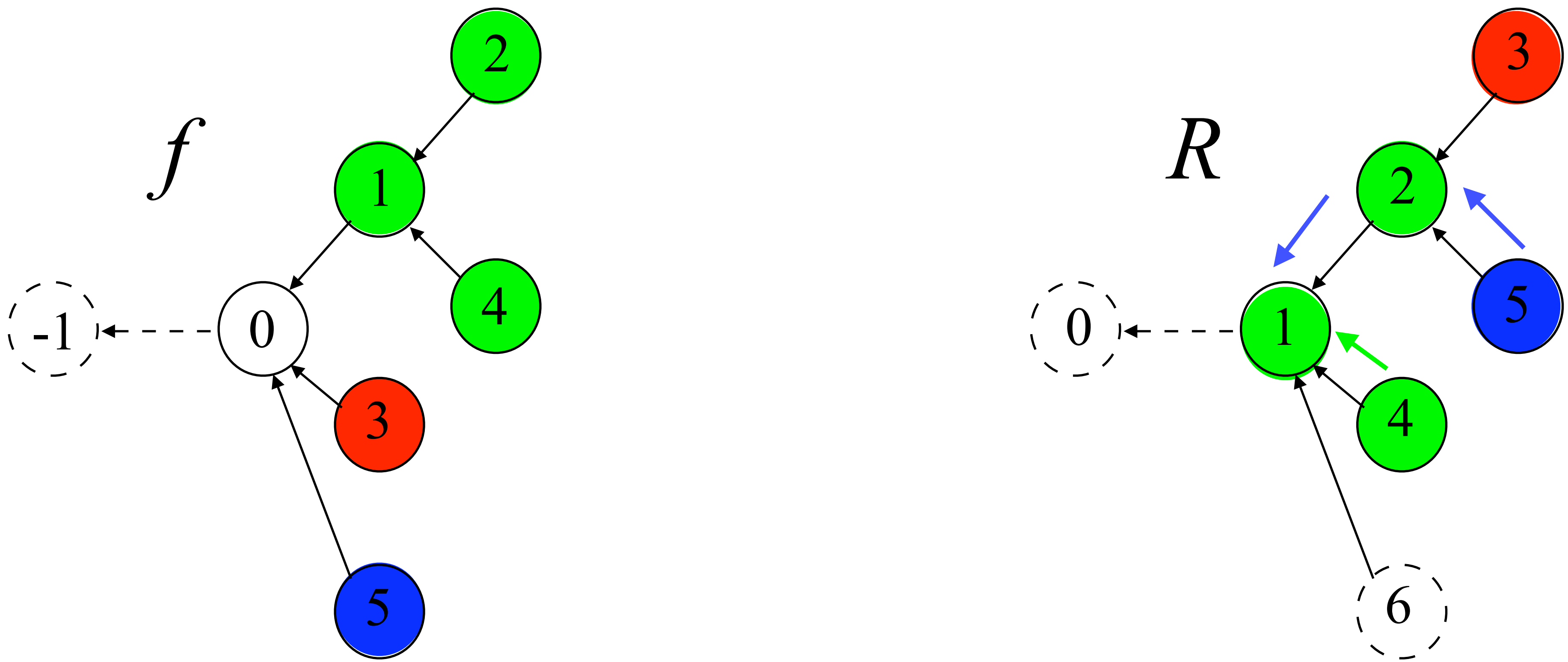
$f\,[1] = 0$ est le seul valide.

Supposons $f\,[1..j]$ valide.

$f\,[1..j{+}1]$ est valide ssi :

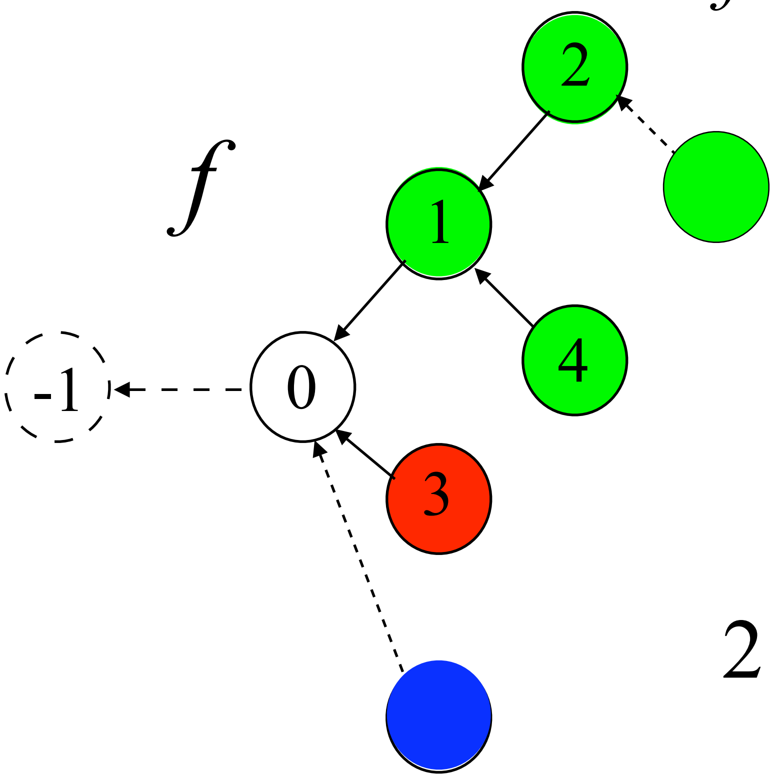
$f\,[j{+}1]$ est le plus grand représentant de la $\bar{f}$-classe de $j{+}1$, sur le R-chemin de $j{+}1$ ($j{+}1$ exclus), par défaut 0.

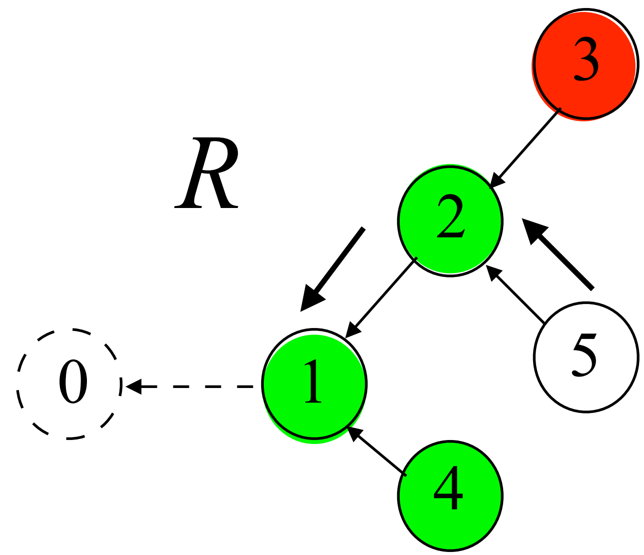
Un exemple



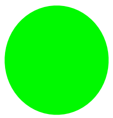
Construction d'un *f* valide

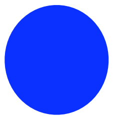
f [1..4] valide





2 solutions :

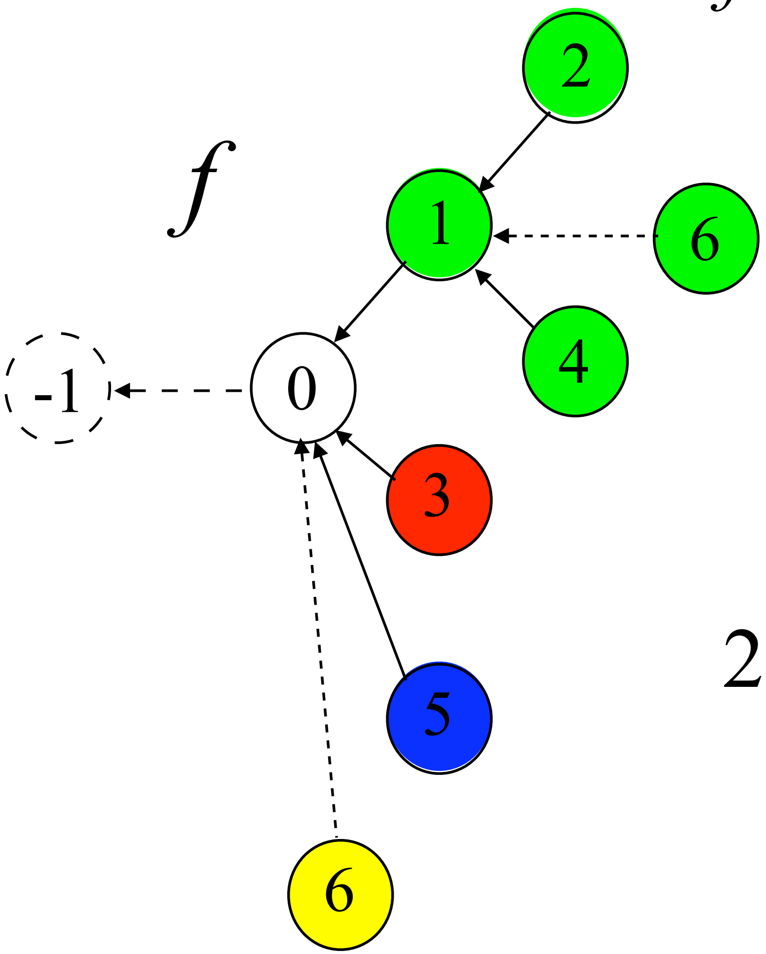
f [5]=2 et 5 est dans 

f [5]=0 et 5 est dans 

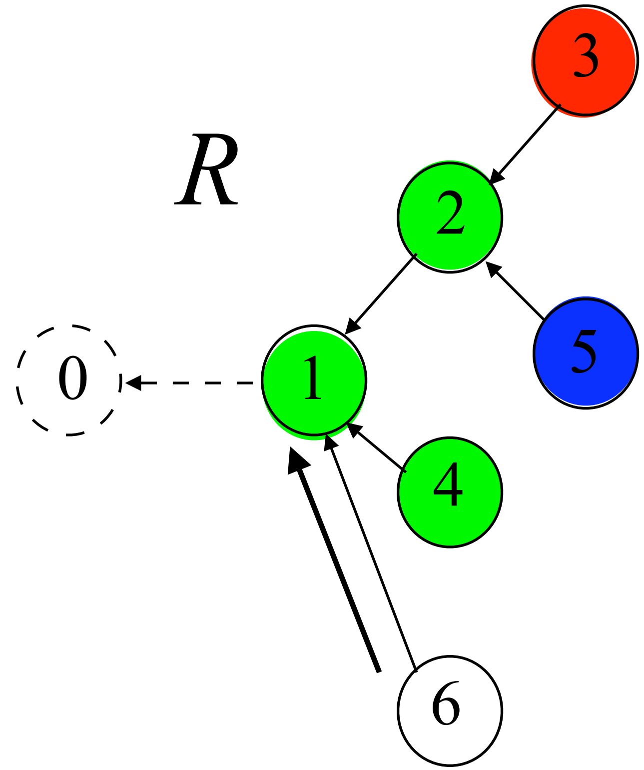
Construction d'un f valide (suite)

f[1..5] valide

f



R



2 solutions :

f[6]=1 et 6 est dans

f[6]=0 et 6 est dans

Caractérisation des mots f-équivalents

Définition : deux mots de même longueur n sont f-équivalents si et seulement si ils ont la même fonction f.

Question : étant donné f valide, quels sont les mots f-équivalents associés à f ?

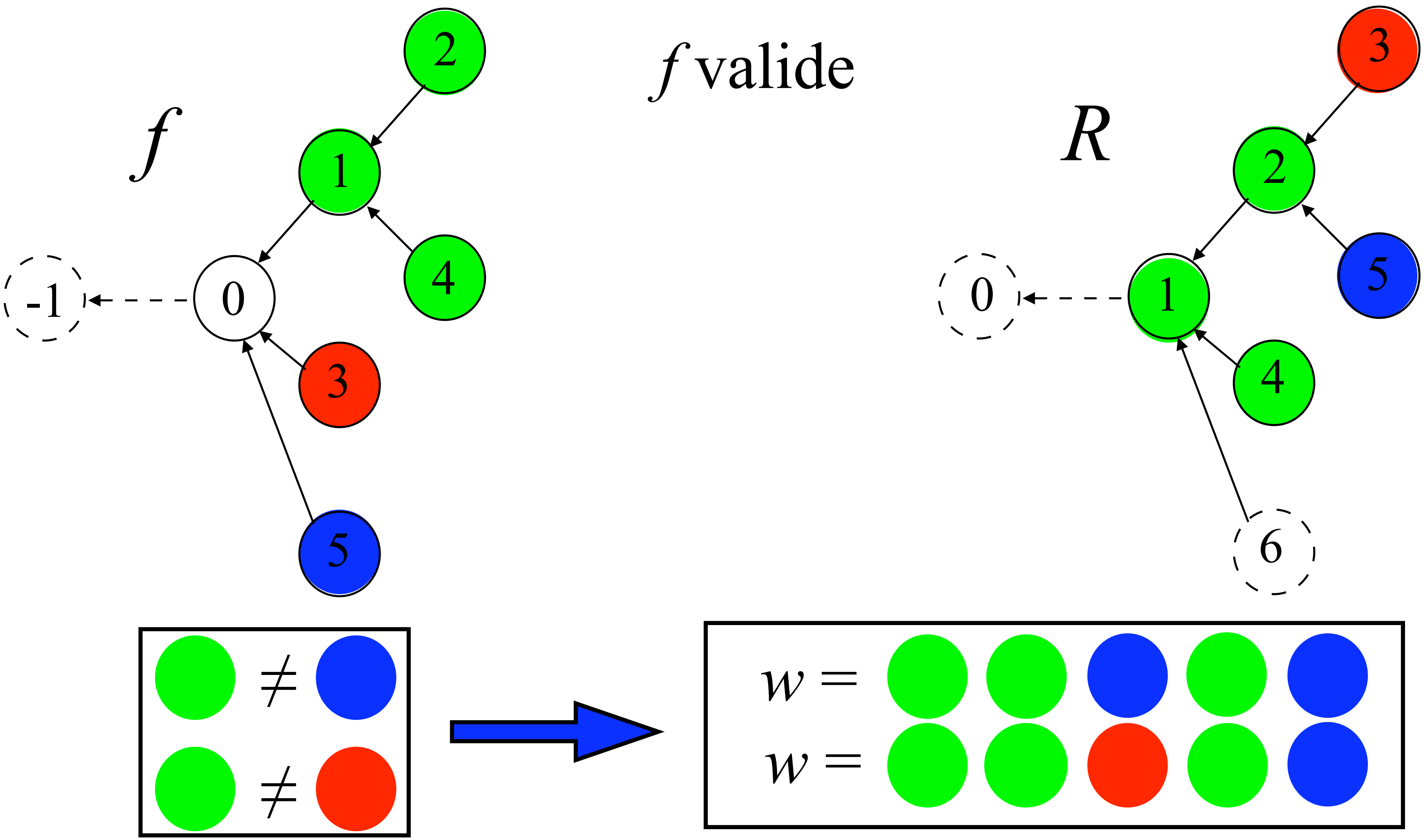
Proposition 2

Étant donné f valide, un mot w a pour tableau de bords f si et seulement si les conditions suivantes sont satisfaites :

(i) les lettres dont les indices sont dans une même $\overline{f}$-classe sont identiques ;

(ii) à deux classes différentes sur un même R-chemin doivent correspondre deux lettres différentes.

Exemple de construction des mots *f*-équivalents

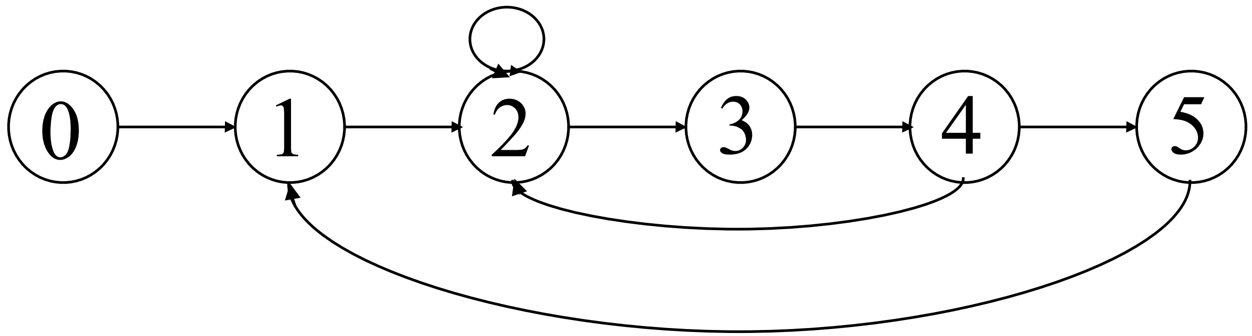


Proposition 3

Soit f un tableau d'entiers et $1 \leq j \leq n$.
Si $f\,[1..n]$ est le tableau de bords d'un mot w
et $f\,[1..j]$ est le tableau de bords d'un mot u alors il
existe v tel que $u.v$ soit f-équivalent à w.

□ associé à f

□



Proposition 4 ($f \rightarrow \square$)

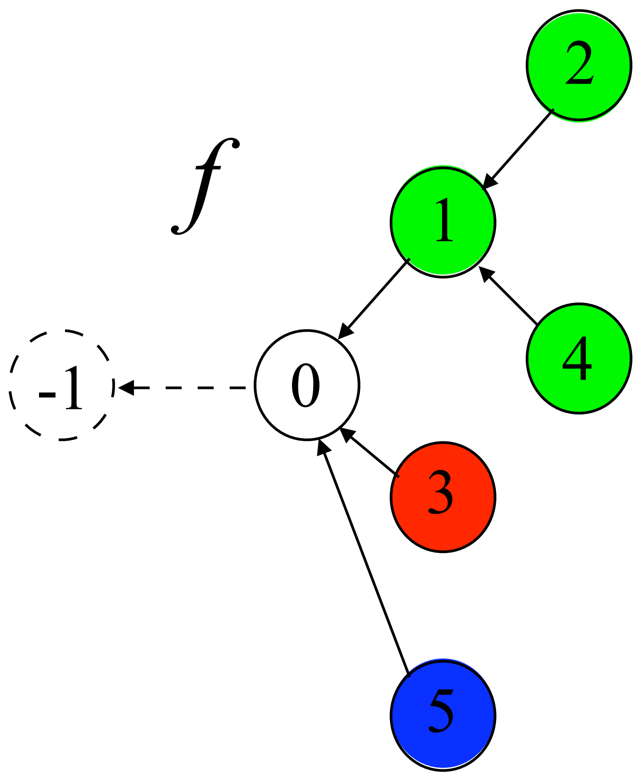
$$\square(j) = (j+1) + (\square(f[j]) \;\square\; f[j+1])$$

Proposition 5 ($\square \rightarrow f$)

$$f[j+1] = \begin{cases} \square(f[j]) \;\square\; \square(j) & \text{si non vide} \\ 0 & \text{sinon} \end{cases}$$

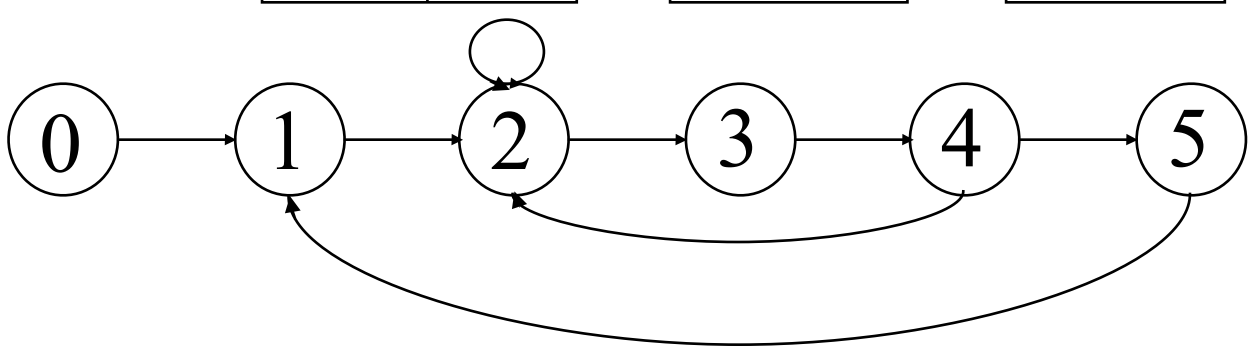
Exemple f→□

f

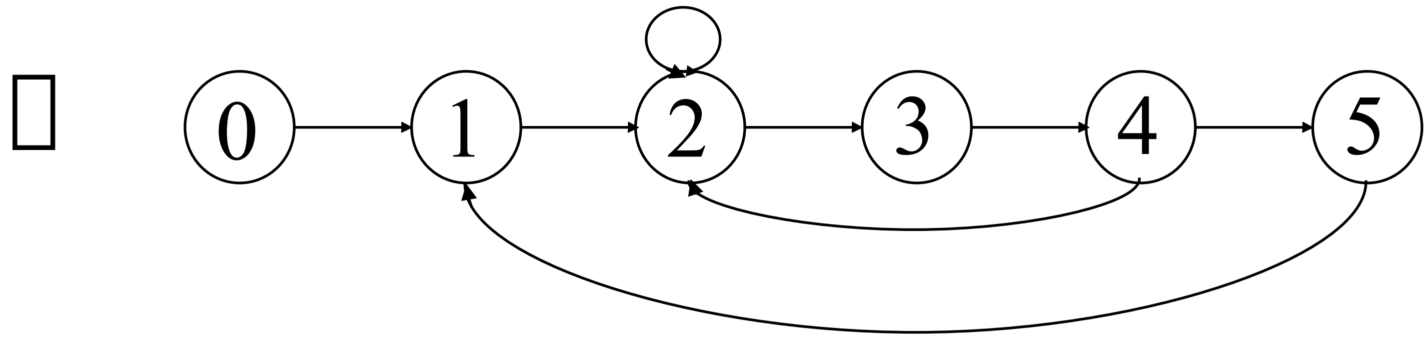


	$j+1$	+	$\delta(f[j])$	-	$f[j+1]$	=	$\delta(j)$
$j=0$	(1)	+		-		=	(1)
$j=1$	(2)	+	(1)	-	(1)	=	(2)
$j=2$	(3)	+	(2)	-		=	(3,2)
$j=3$	(4)	+	(1)	-	(1)	=	(4)
$j=4$	(5)	+	(2)	-		=	(5,2)
$j=5$		+	(1)	-		=	(1)

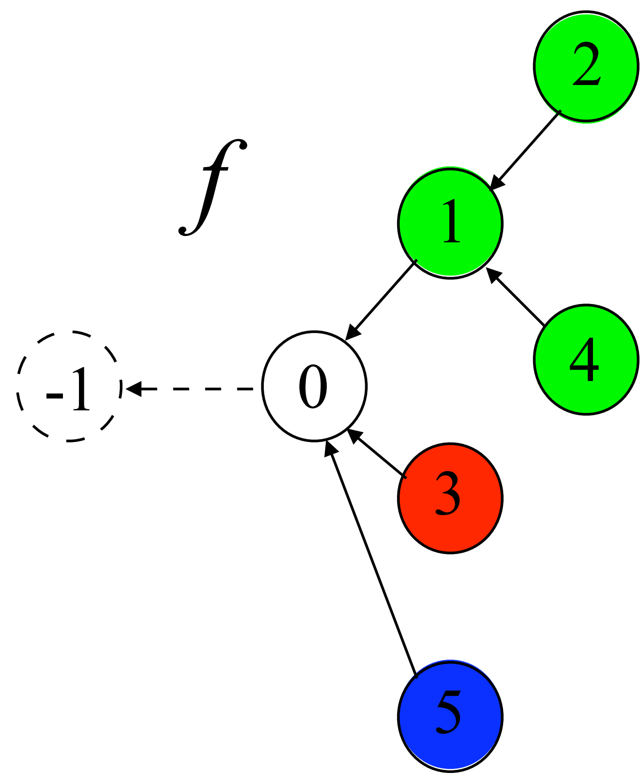
□

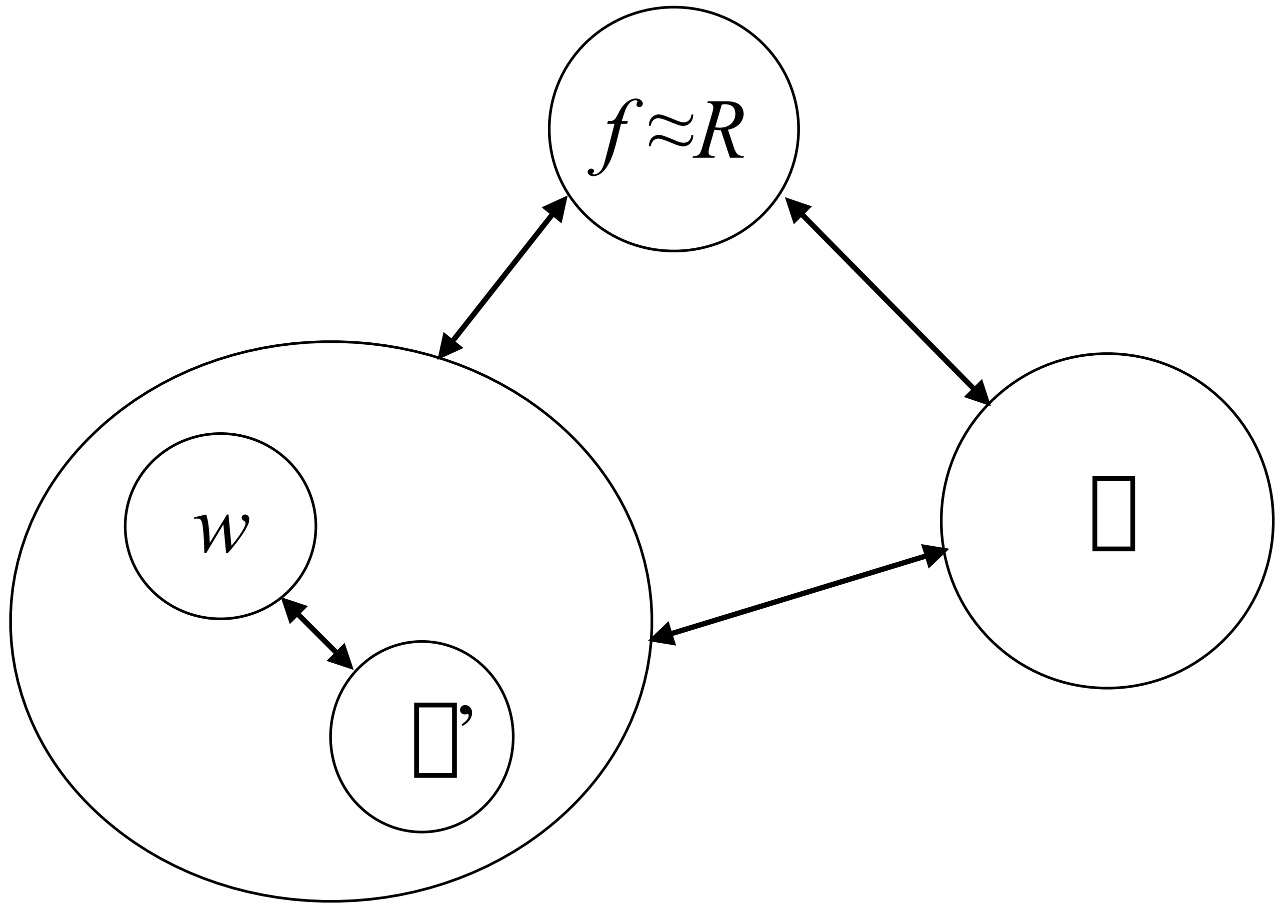


Exemple □→f



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





f-équivalence

=

□-équivalence

Perspectives

Rappelons la fonction g (*next* de [KMP, 1977]) :

$$g[j] = \max\{i \mid w[1..i-1] \text{ suffixe de } w[1..j-1] \text{ avec } w[i] \neq w[j]\}$$

On sait que

$$g[j] = \max\{\square(j\text{-}1)\square(j)\}$$
$$= \max\{\square(f\,[\mathrm{j}\square 1])\square f\,[j])\}$$

Mais on n'a pas pour g l'équivalent de la proposition 3.

Références

[AHU, 1974] AV Aho , JE Hopcroft , and JD Ullman ,
The Design and Analysis of Computer Algorithms,
Addison-Wesley, Reading, MA, 1974.

[Franěk *et al, 1999*] F. Franek, W. Lu, P.J. Ryan, W.F. Smyth, Y. Sun, L. Yang
Verifying a border array in linear time
Proc. Tenth Australasian Workshop on Combinatorial Algorithms (AWOCA), School of Computing, Curtin University of Technology, Perth, Australia, August 25-27, 1999, pp. 26-33.

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