Introduction à l'algorithmique de texte

Chaîne de caractères (Mot) de longueur n sur un alphabet Σ.

Trouver dans cette chaîne les occurrences { d'un motif, d'un mot, d'un ensemble de mots }

Applications
- → Extraction d'information
- → Recherche dans un éditeur
- → Génomique, bioinformatique
- → Correction orthographique
- → ...

Déf. Un **préfixe** (strict) d'un mot $u \in \Sigma^*$ est un mot $v \in \Sigma^*$ (différent de u) t.q. il existe un mot $w \in \Sigma^*$ avec $u = vw$.

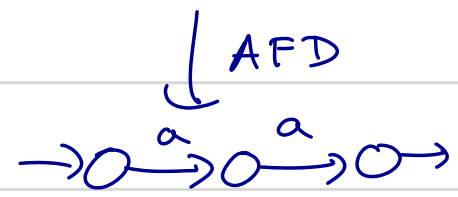
Un **suffixe** d'un mot $u \in \Sigma^*$ est un mot $v \in \Sigma^*$ t.q. il existe un mot $w \in \Sigma^*$ avec $u = wv$.

Mot / Chaîne } de taille n dans laquelle on veut rechercher des motifs.

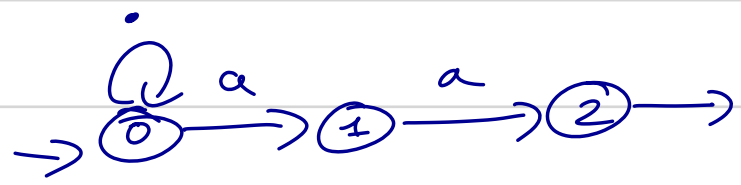
Motif	Recherche	Technique	Complexité de compilation	Complexité de recherche
e.r.e de taille k	1 occurrence ou Ttes les occurrences	AFD pour $.*e$ (une occurrence à chaque passage dans un état final)	$O(\beta^k)$ pour $\beta > 2$	$O(n)$
chaîne de taille k	1 occ. ou Ttes les occ.	naïf	0	$O(n \times k)$
chaîne de taille k	1 occ. ou Ttes les occ.	Knuth-Morris-Pratt (KMP)	$O(k)$	$O(n)$
ensemble de m chaînes de taille k	correspondance exacte	Trie (ou Arbre PATRICIA)	$O(m \times k)$	$O(n)$
ensemble de m chaînes de taille k	1 occ. ou Ttes les occ.	$m \times$ KMP	$O(m \times k)$	$O(m \times n)$
		Aho-Corasick (Trie + KMP, automates à repli arborescents)	$O(m \times k)$	$O(n$ + taille des résultats$)$

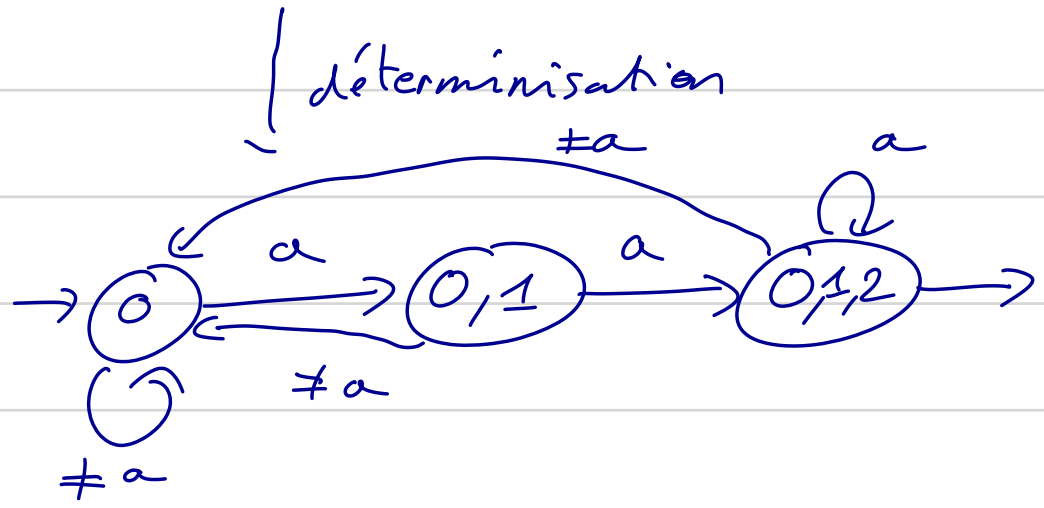
Ex. pour .*e

Rechercher tt.es les occurrences de "aa".



AFND pour .*aa :





a b a a b a a a a b a a b

← états finals
≡ fins de motifs

Ex. algo naïf

Motif a b c a b d

Chaîne a b a c a b c a b c a b d

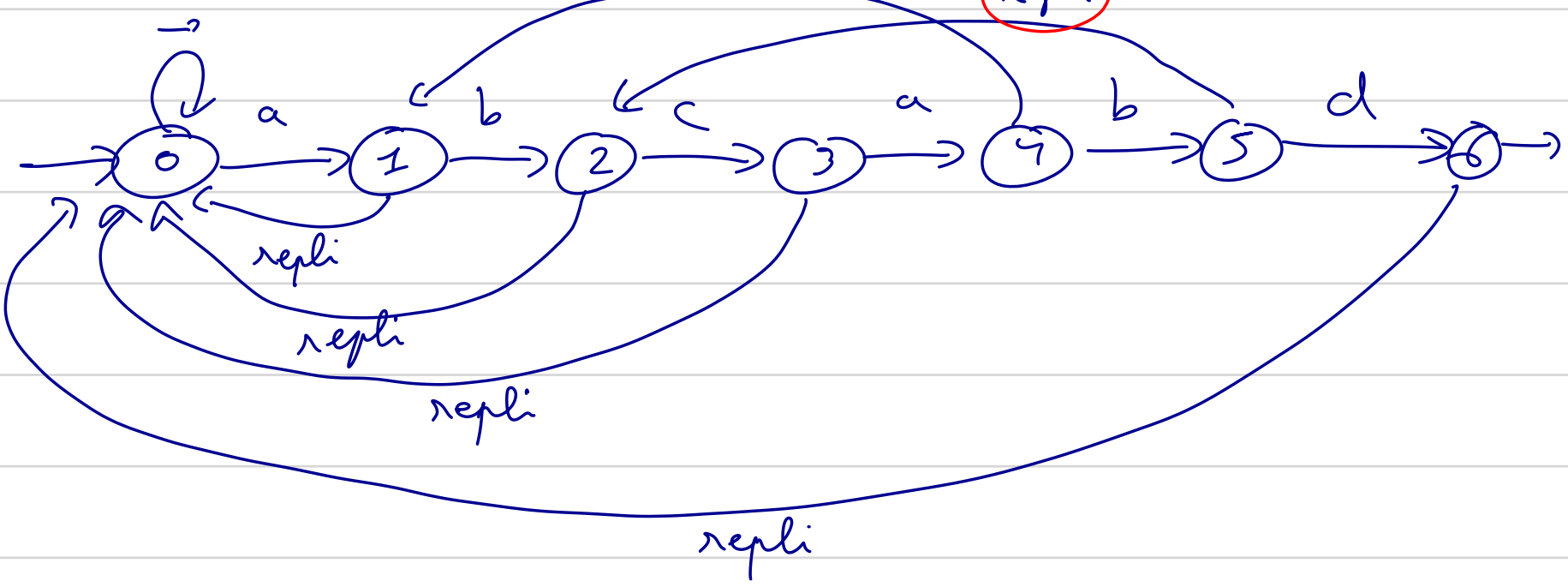
a b —
—
a —
— a b c a b —
—
— a b c a b d

KMP

Compilation : On se souvient, dans le motif, de la taille du plus long préfixe strict qui est un suffixe de chaque préfixe du motif.

a	b	c	a	b	d
(-1)	0	0	1	2	0

1	2	3	4	5	6
a	b	c	a	b	d
(-1)	0	0	1	2	0



a b a c a b c a b c a b d

1 2 → 0 → 1 → 0 → 1 2 3 4 5 →

2 3 4 5 6 : match

KMP

Complexité de reconnaissance : $O(n + \#\text{replis})$

$\#\text{replis} \leq n$: nombre de fois où on va de droite à gauche ≤ nombre de fois où on va de gauche à droite.

Compilation : $O(k)$ par programmation dynamique

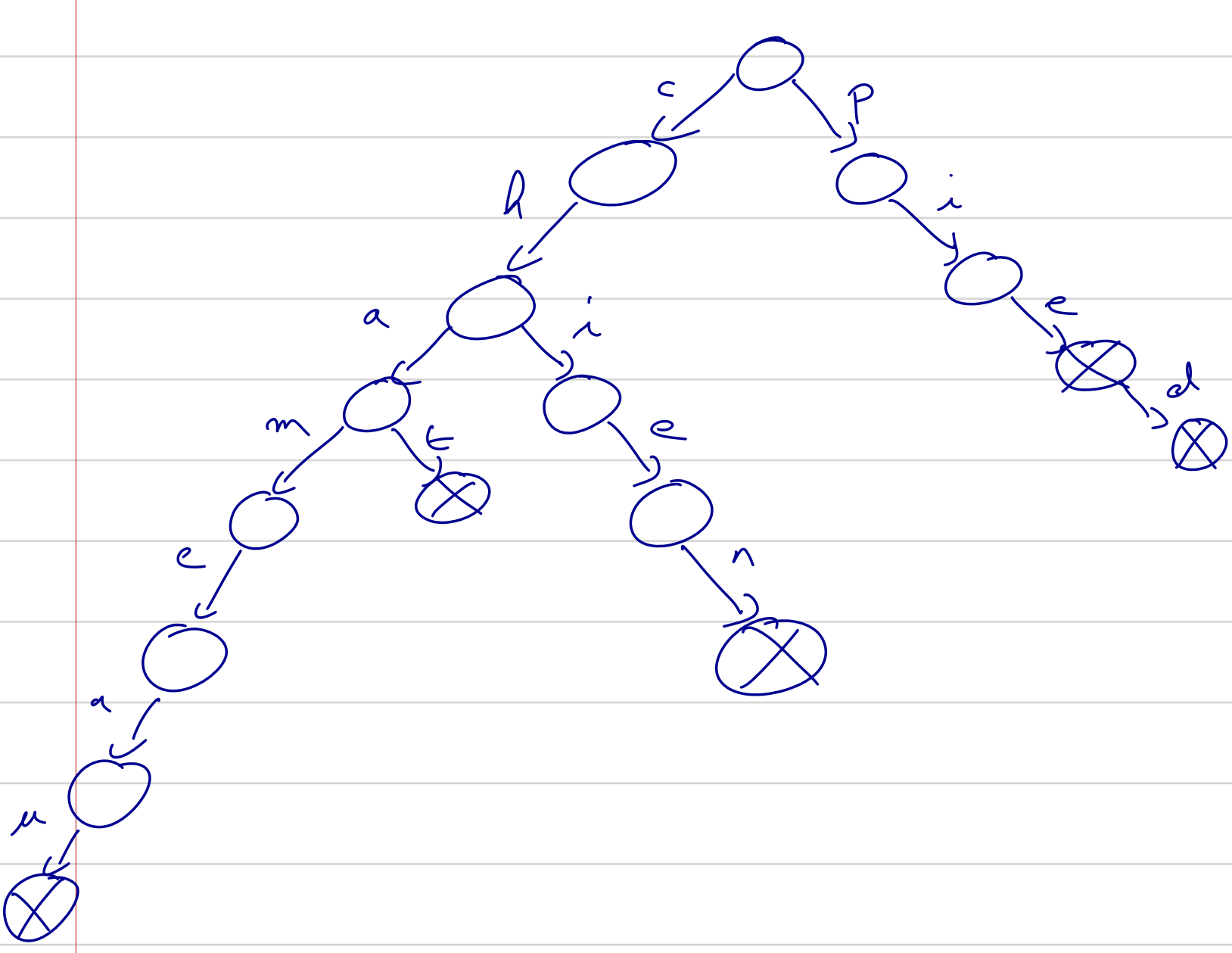
1	2	3	4	5	6
a	b	c	a	b	d
(-1)	0	~~1~~			
		0	1	2	~~3~~
					~~1~~
					0

Trie

("retrieval")

Arbre dont chaque noeud correspond aux préfixes des mots d'un ensemble, chaque lien enfant – parent correspond à un symbole, et où les noeuds des mots de l'ensemble sont marqués.

{ chien, chat, chameau, pie, pied } $n = 5$, $k = 7$



Arbres radix ou PATRICIA

Raffinement des Trie dans lequel les arêtes enfant – parent peuvent être étiquetées par des sous-chaînes