

Réduction de l'espace mémoire de l'arbre des suffixes

Thierry Lecroq
Université de Rouen

Arbre des suffixes

$$y = y[0..n-1]\$ = y[0..n]$$

arbre de suffixes de y : $S(y)$

Notations

- " On confond les nSuds de l'arbre avec la concaténation des étiquettes de la racine au nSud.
- " Tous les nSuds internes sont des fourches (au moins 2 successeurs).
- " $prof(w) = |w|$
- " lien suffixe : $suf(aw) = w$.
- " $tête(0) = \varepsilon$
- " $tête(j) =$ plus long préfixe de $y[j..n]$ préfixe de $y[i..n]$ avec $i < j$.

Observation 1

Soit w une fourche de $S(y)$.

Alors il existe une position i sur y telle que
 $w = \text{tête}(i)$.

Observation 2

Soit i une position sur y .

Alors $tête(i)$ est une fourche de $S(y)$.

Notations

Soit w une fourche de $S(y)$ on note $positionTête(w)$ le plus petit entier i tel que w tel que $tête(i) = w$.

Autrement dit, i est la position de la deuxième occurrence de w dans y .

Observation 3

Soit (w, u, wu) une branche de $S(y)$ avec wu une fourche.

Alors $u = y[i..i+\ell-1]$ avec

$$i = positionTête(wu) + prof(w)$$

et

$$\ell = prof(wu) - prof(w).$$

Observation 4

Soit (w, u, wu) une branche de $S(y)$ avec $wu = y[j..n]$ une feuille.

Alors $u = y[i..n]$ avec

$$i = j + \text{prof}(w).$$

Notation

Chaque feuille $y[j..n]$ est identifiée par la position j .

Soit q le nombre de fourches de $S(y)$.

Soit $b_0, b_1, \dots, b_{q-1}$ la liste des fourches dans l'ordre croissant des positions de tête :

$$positionTête(b_i) < positionTête(b_{i+1}).$$

$$num(b_i) = i \text{ (} num(b_0) = \varepsilon \text{)}$$

Observation 5

Si u et w sont deux fourches distinctes alors
 $\text{positionTête}(u) \neq \text{positionTête}(w)$.

Pour toute fourche aw de $S(y)$, w est aussi une fourche et

$$\text{positionTête}(aw) + 1 \geq \text{positionTête}(w).$$

Une fourche aw est un **petit nS ud** ssi
 $positionTête(aw) + 1 = positionTête(w)$.

Une fourche aw est un **grand nS ud** ssi
 $positionTête(aw) + 1 > positionTête(w)$.

La racine est soit petite soit grande.

Observation 6

Soit aw une fourche de $S(y)$.

Si aw est un petit nS ud alors

$$num(aw) + 1 = num(w).$$

Si $num(aw) = q-1$ et $q > 1$ alors

aw est un grand nS ud.

On peut partitionner les fourches $b_1, b_2, \& b_{q-1}$ en chaînes de zéro ou plusieurs petits nS uds suivis par un seul grand nS ud.

Une **chaîne** une suite $b_g, \& , b_d$ de fourches telles que :

- " b_{g-1} n est pas un petit nS ud ;
- " $b_g, \& , b_{d-1}$ sont des petits nS uds ;
- " b_d est un grand nS ud.

Observation 7

Soit $b_g, \dots, b_d$ une chaîne.

Alors pour $g \leq i \leq d-1$ on a

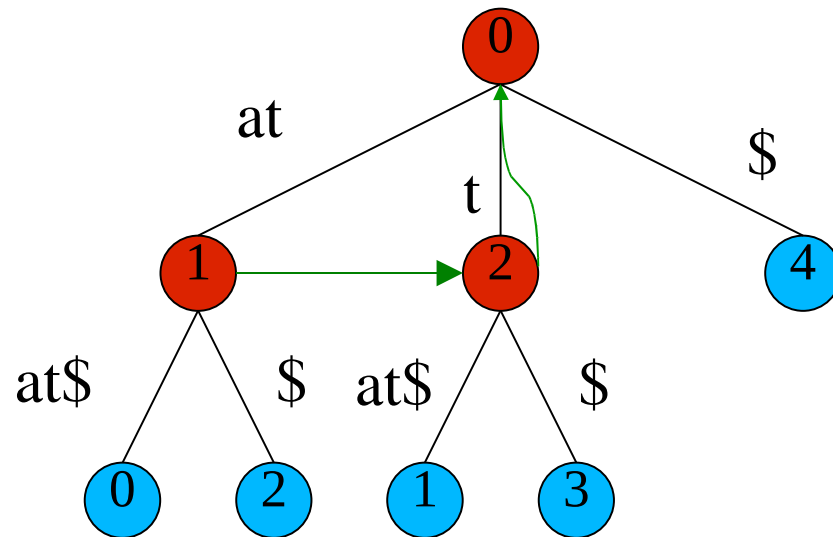
" $\text{prof}(b_i) = \text{prof}(b_d) + (d - i)$

" $\text{positionTête}(b_i) = \text{positionTête}(b_d) + (d - i)$

" $\text{suf}(b_i) = b_i + 1$

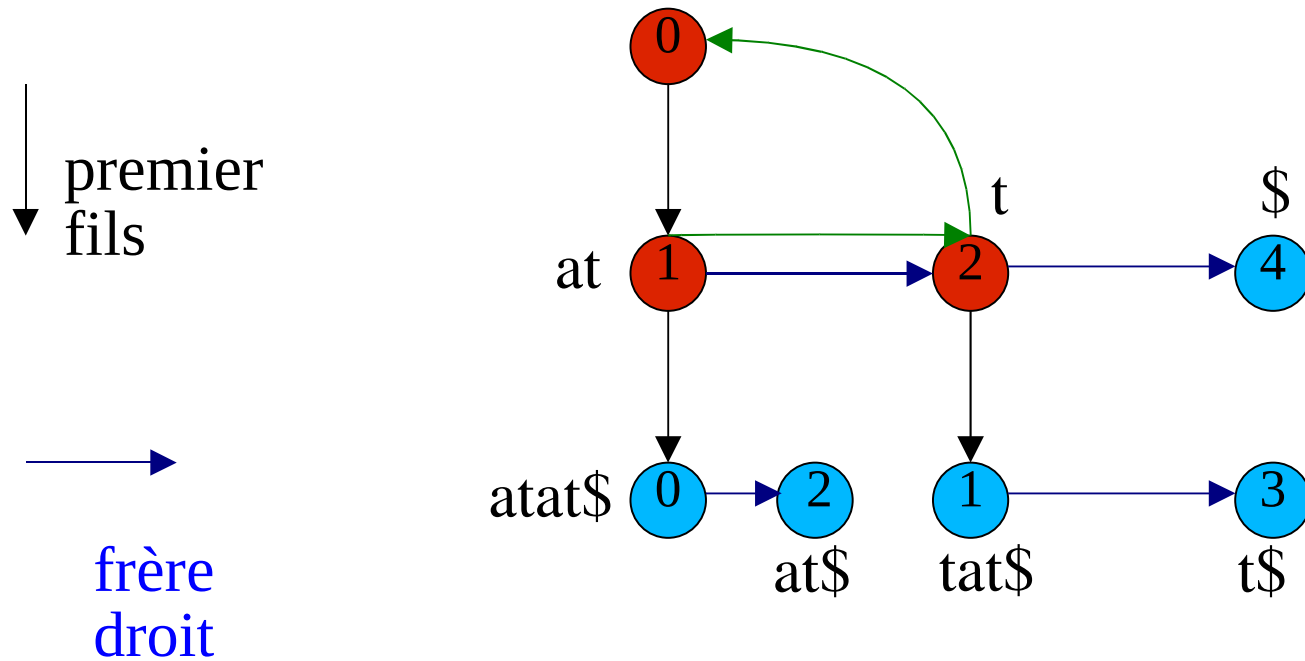
Arbre des suffixes

$y = \text{atat}\$$



Arbre des suffixes

$y = \text{atat}\$$

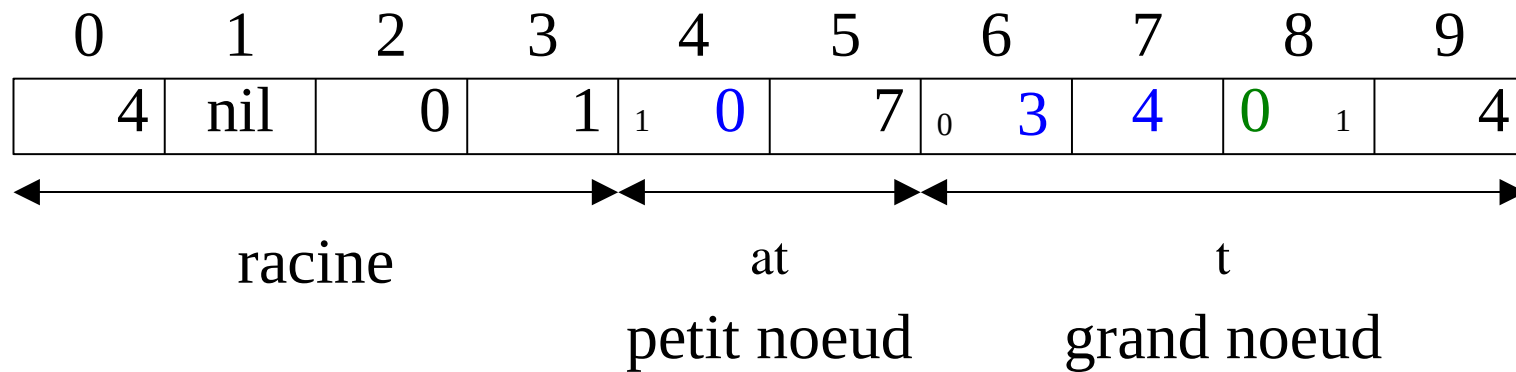


	atat\$	tat\$	at\$	t\$	\$
<i>j</i>	0	1	2	3	4
<i>F[j]</i>	2	3	nil	nil	nil

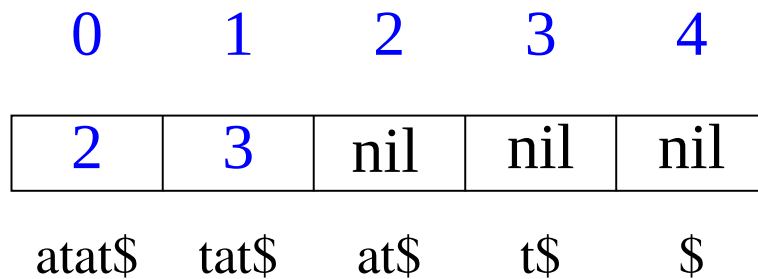
fourche	ϵ	at	t
<i>num</i>	0	1	2
premier fils	1	0	1
frère droit	nil	2	4
<i>prof</i>	0	2	1
<i>position</i> <i>Tête</i>	0	2	3
<i>suf</i>		2	1

Arbre des suffixes

fourches



feuilles

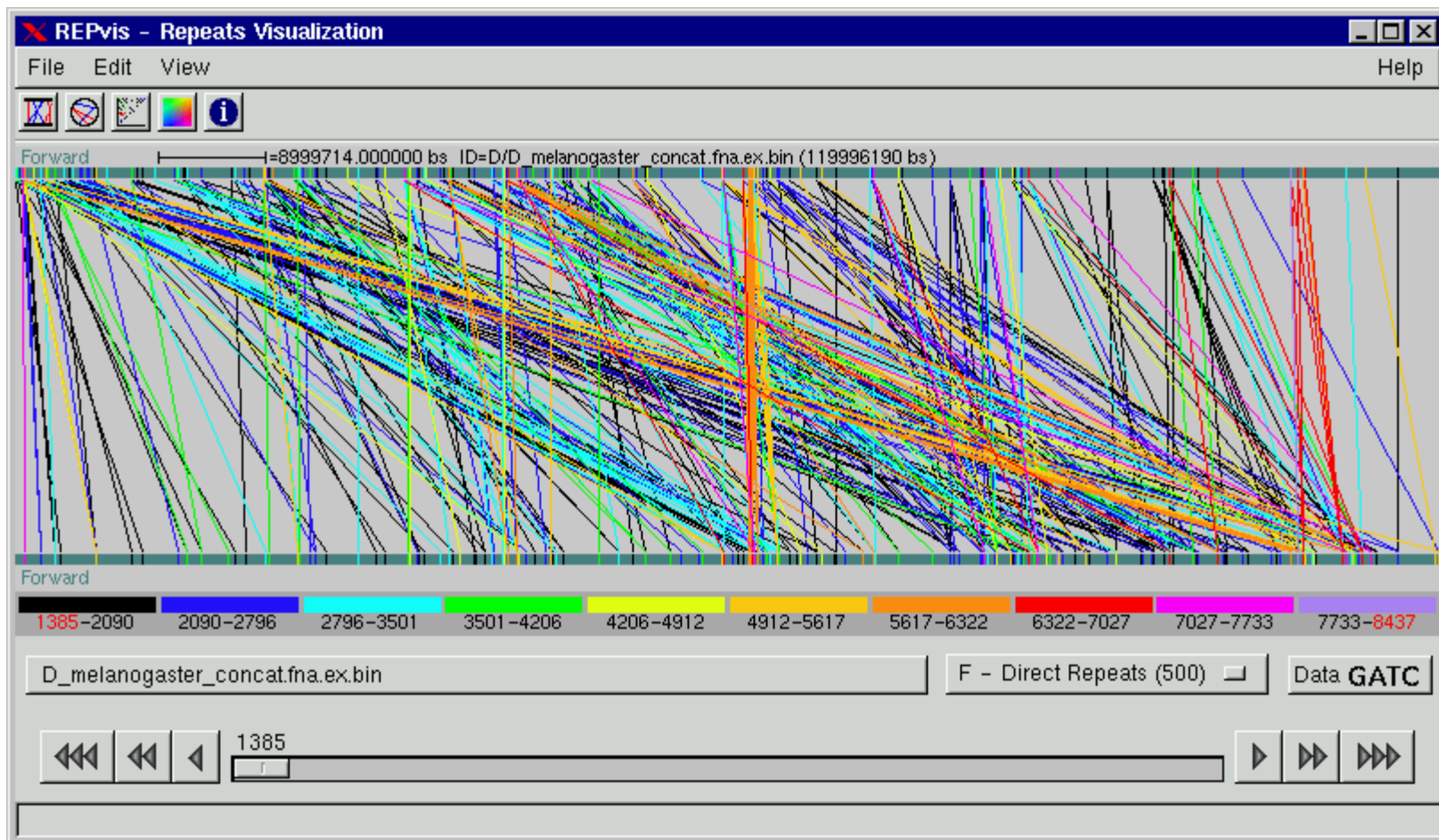


10+5 = 15 entiers

Taille des différentes implantations

• Mc Creight	$2n + 5(g+p)$
• Listes chaînées	$n + 5(g+p)$
• Table de hachage	$4n + 3(g+p)$
• Listes chaînées améliorées	$n + 2g + 4p$
• Table de hachage améliorée	$3(n+p)$

- n : longueur de la séquence
- g : nombre de grands noeuds
- p : nombre de petits noeuds

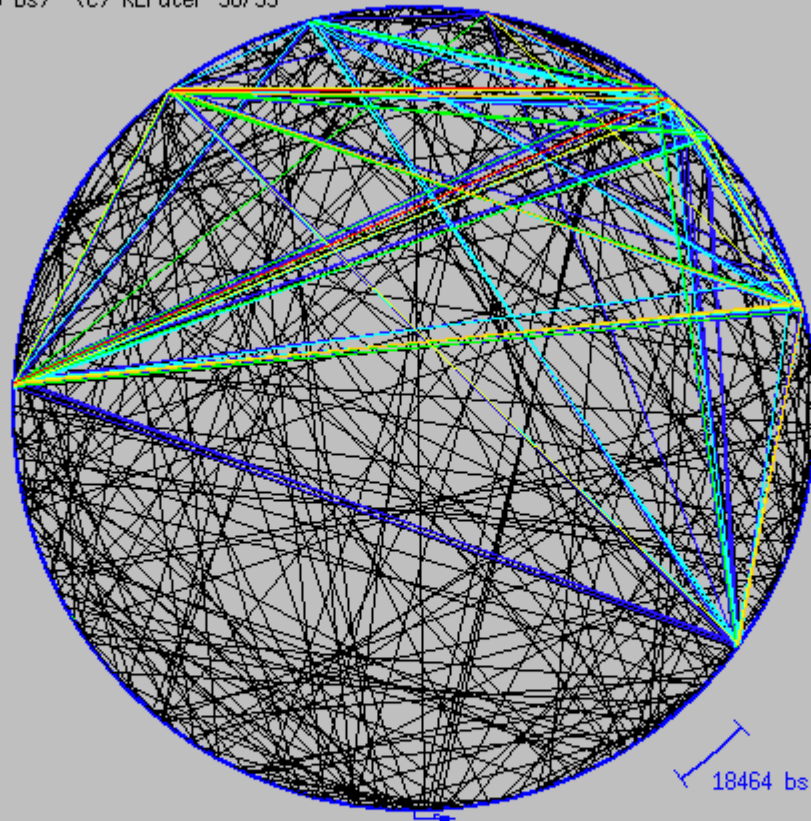


Repeats Visualizer V1.0

About

ID=/vol/pi/share/xreputer/mgen.bin (580073 bs) (C) REPuter 98/99

Forward vs. Forward



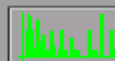
15-38 38-61 61-83 83-106 106-129 129-152 152-175 175-197 197-220 220-243 Min=15 Max=243

Select Genome

mgen.bin



GATC



F

P

C

R

17

Logs

Exit

Résultats expérimentaux

- | | Mb | Mo | exact | $H < 5$ | $E < 5$ |
|-------------------------------|--------|---------|--------|---------|---------|
| • <i>B. subtilis</i> | 4,02 | 55,60 | 18,80 | 18,86 | 18,88 |
| • <i>E. coli</i> | 4,42 | 61,19 | 20,66 | 20,89 | 20,98 |
| • <i>H. sapiens</i> XXII | 32,06 | 443,04 | 185,88 | 186,71 | 187,33 |
| • <i>A. thaliana</i> II et IV | 35,47 | 490,23 | 226,43 | 227,30 | 227,64 |
| • <i>C. elegans</i> | 92,40 | 1277,27 | 762,44 | 767,31 | |
| | 769,86 | | | | |
- 400 MHz, 2 Go RAM
- espace = $13,82n$

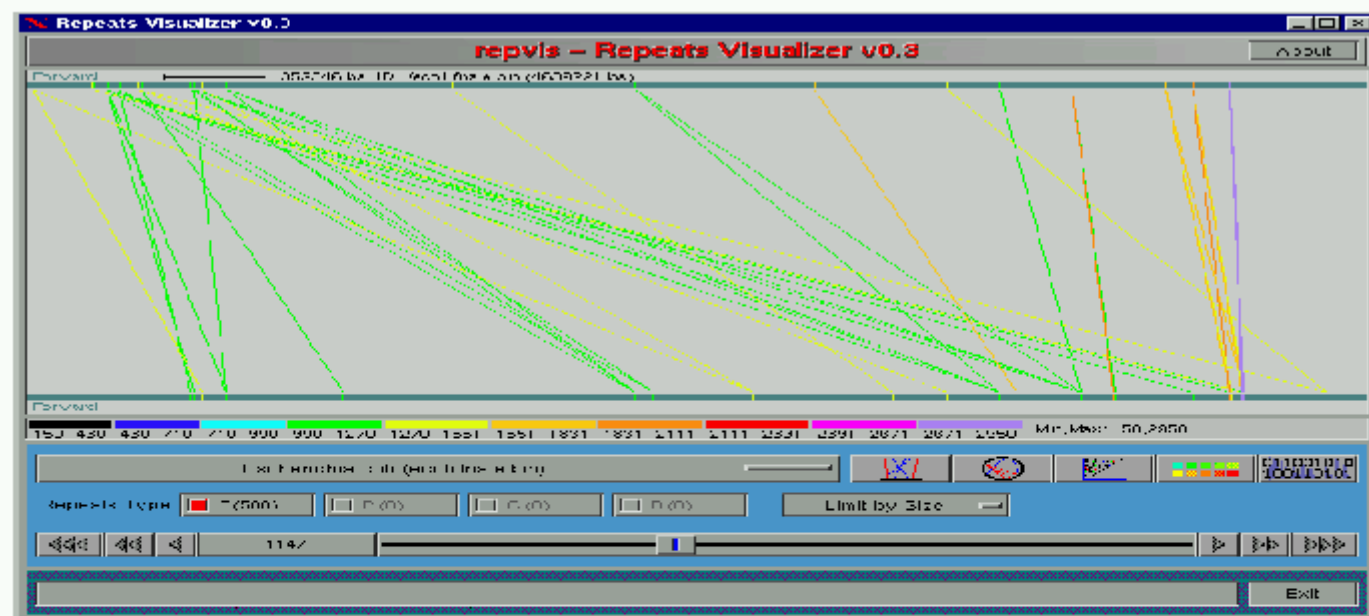


Figure 4: A typical application of *REPvis*, showing a view of the 50 most significant direct repeats in *E. coli* (4.6Mb), ranging from 1147 to 2950 bases in length. There are five repeats longer than the longest one found in *M. tuberculosis*; see Figure 5. In the main window graphics panel, two horizontal lines depict the input sequence and a copy of it. Diagonal lines stand for repeats by connecting their respective starting positions. Below the graphics panel, a choice box lists all calculated sequences in a user specified directory. Three further buttons switch the visualization mode to square graph, circle graph or dot plot. An additional button leads to the complete list of all repeats and their size distribution. Selector buttons specify which type of repeat to display. The symbols *F*, *P*, *C*, and *R* indicate direct (forward), palindromic (reverse complemented), complemented and reversed repeats; the number of repeats for each type is shown on the button.

Références

- " Reducing the space requirement of suffix trees, S. Kurtz, *Software - Practice & Experience*, 1998.
- " REPuter: fast computation of maximal repeats in complete genomes, S. Kurtz et C. Schleiermacher, *Bioinformatics*, 1999.
- " Computation and vizualisation of degenerate repeats in complete genomes, S. Kurtz, E. Ohlebusch, C. Schleiermacher, J. Stoye et R. Giegerich, *ISMB* 2000.