

Enhanced Suffix Arrays

Thierry Lacroq

Thierry.Lacroq@univ-rouen.fr

Laboratoire d'Informatique, du Traitement de l'Information et des
Systèmes.

International PhD School in Formal Languages and Applications

Tarragona, November 20th and 21st, 2006



Plan

- 1 Introduction
- 2 Lcp-interval tree
- 3 Enhanced suffix array

Plan

- 1 Introduction
- 2 Lcp-interval tree
- 3 Enhanced suffix array

Suffix Array

Exemple pour $y = \text{acaaacatat}\$$

i	$p[i]$	$LCP[i]$											
0	2	0	a	a	a	c	a	t	a	t	\$		
1	3	2	a	a	c	a	t	a	t	\$			
2	0	1	a	c	a	a	a	c	a	t	a	t	\$
3	4	3	a	c	a	t	a	t	\$				
4	6	1	a	t	a	t	\$						
5	8	2	a	t	\$								
6	1	0	c	a	a	a	c	a	t	a	t	\$	
7	5	2	c	a	t	a	t	\$					
8	7	0	t	a	t	\$							
9	9	1	t	\$									
10	10	0	\$										

lcp-interval

An interval $[i, j]$, $0 \leq i < j \leq n$, is an **lcp-intervalle** of **lcp value** ℓ if

- 1 $LCP[i] < \ell$,
- 2 $LCP[k] \geq \ell$ for $i + 1 \leq k \leq j$,
- 3 $LCP[k] = \ell$ for at least one k with $i + 1 \leq k \leq j$,
- 4 $LCP[j + 1] < \ell$.

We denote **ℓ -interval** (or **ℓ - $[i, j]$**) for an lcp-interval of lcp value ℓ .

*l*indices

Each index k , $i + 1 \leq k \leq j$, with $LCP[k] = \ell$ is called an *ℓ -index*.

The set of ℓ -indices of an ℓ -interval $[i, j]$ is denoted by $\ell\text{indices}(i, j)$.

If $[i, j]$ is an ℓ -interval such that
 $w = y[p[i]..p[i] + \ell - 1]$
is the longest common prefix to suffixes
 $y[p[i]..n], y[p[i + 1]..n], \dots, y[p[j]..n]$
then $[i, j]$ is called an *w -interval*.

Example

For $y = \text{acaaacatat}\$$

$[0, 5]$ is a 1-interval since

- 1 $p[0] = 0 < 1$,
- 2 $LCP[k] \geq 1$ for $1 \leq k \leq 5$,
- 3 $LCP[2] = 1$,
- 4 $LCP[5 + 1] = 0 < 1$.

$[0, 5]$ is an a-interval and $\ell_{indices}(0, 5) = \{2, 4\}$

Successor Interval

An m -interval $[g, d]$ is **embedded** in an ℓ -interval $[i, j]$ if

- $[g, d]$ is a subinterval of $[i, j]$ ($i \leq g < d \leq j$) and
- $m > \ell$.

The ℓ -interval $[i, j]$ is called the **enclosing** interval of the interval $[g, d]$.

If $[i, j]$ encloses $[g, d]$ and there is no enclosing interval in $[i, j]$ that encloses $[g, d]$ then $[g, d]$ is a **child interval** of $[i, j]$.

Plan

- 1 Introduction
- 2 Lcp-interval tree**
- 3 Enhanced suffix array

Lcp-interval tree

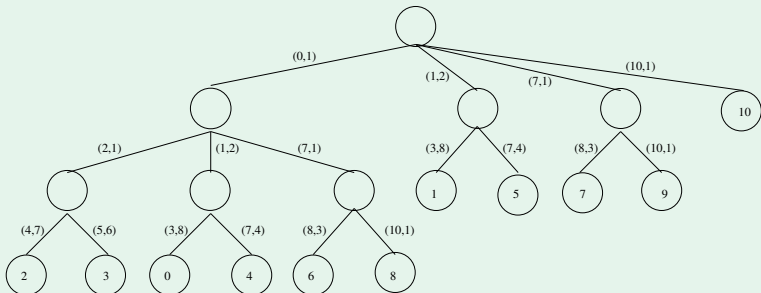
The parent-child relationship on the intervals constitutes the **lcp-interval tree** of the suffix array.

The root of the tree is the 0-interval $[0, n]$.

The lcp-interval tree is basically the suffix tree without leaves.

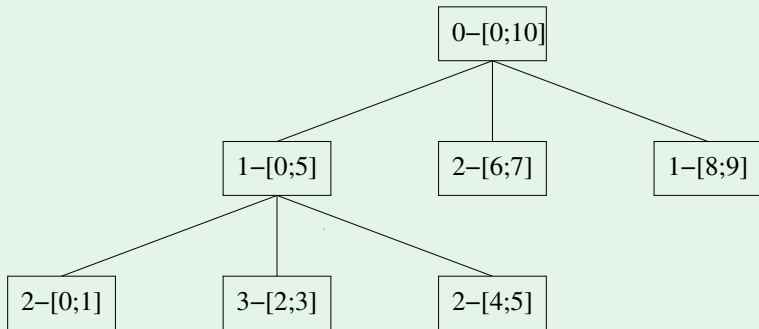
Suffix tree

Exemple



Lcp-interval tree

Exemple



Plan

- 1 Introduction
- 2 Lcp-interval tree
- 3 Enhanced suffix array**

Additional tables

$$\begin{aligned} \text{childtab}[i].\text{up} &= \min\{q \in [0, i - 1] \mid \text{LCP}[q] > \text{LCP}[i] \text{ and} \\ &\quad \forall k \in [q + 1, i - 1], \text{LCP}[k] \geq \text{LCP}[q]\} \end{aligned}$$

$$\begin{aligned} \text{childtab}[i].\text{down} &= \min\{q \in [i + 1, n] \mid \text{LCP}[q] > \text{LCP}[i] \text{ and} \\ &\quad \forall k \in [i + 1, q - 1], \text{LCP}[k] > \text{LCP}[q]\} \end{aligned}$$

$$\begin{aligned} \text{childtab}[i].\text{next} &= \min\{q \in [i + 1, n] \mid \text{LCP}[q] = \text{LCP}[i] \text{ and} \\ &\quad \forall k \in [i + 1, q - 1], \text{LCP}[k] > \text{LCP}[i]\} \end{aligned}$$

Additional tables

For a ℓ -interval $[i, j]$ with ℓ -indices $i_1 < i_2 < \dots < i_k$, $childtab[i].up$ and $childtab[i].down$ are used to compute the first ℓ -index i_1 .

The other ℓ -indices are $childtab[i_1].next$, $childtab[i_2].next$, \dots , $childtab[i_{k-1}].next$.

Additional tables

Lemme 3.1

Let $[i, j]$ be a ℓ -interval. If $i_1 < i_2 < \dots < i_k$ are the ℓ -indices of $[i, j]$ then the child intervals of $[i, j]$ are $[i, i_1 - 1], [i_1, i_2 - 1], \dots, [i_k, j]$.

Additional tables

Lemme 3.2

For each ℓ -interval $[i, j]$ we have:

- 1 $i < \text{childtab}[j + 1].\text{up} \leq j$ or $i < \text{childtab}[i].\text{down} \leq j$.
- 2 $\text{childtab}[j + 1].\text{up}$ is the first ℓ -index of $[i, j]$ if $i < \text{childtab}[j + 1].\text{up} \leq j$.
- 3 $\text{childtab}[i].\text{down}$ is the first ℓ -index of $[i, j]$ if $i < \text{childtab}[i].\text{down} \leq j$.

Enhanced Suffix Array

Example for $y = \text{acaaacatat}\$$

i	$p[i]$	$LCP[i]$	up	$down$	$next$	
0	2	0		2	6	a a a c a t a t \$
1	3	2				a a c a t a t \$
2	0	1	1	3	4	a c a a a c a t a t \$
3	4	3				a c a t a t \$
4	6	1	3	5		a t a t \$
5	8	2				a t \$
6	1	0	2	7	8	c a a a c a t a t \$
7	5	2				c a t a t \$
8	7	0	7	9	10	t a t \$
9	9	1				t \$
10	10	0	9			\$

Applications

- Exact string matching.
- Computation of supermaximal repeats and of maximal unique matches (MUM).
- Computation of maximal repeats and of maximal exact matches (MEM).
- Computation of tandem repeats.

Example

For $y = \text{acaaacatat}\$$

$[0, 5]$ is a 1-interval and $\ell\text{indices}(0, 5) = \{2, 4\}$

The first 1-index 2 is stored in $\text{childtab}[0].\text{down}$ and in $\text{childtab}[6].\text{up}$.

The second 1-index 4 is stored in $\text{childtab}[2].\text{next}$.

Thus the child intervals of $[0, 5]$ are $[0, 1]$, $[2, 3]$ and $[4, 5]$.

References



M.I. Abouelhoda, S. Kurtz, E. Ohlebusch.
Replacing suffix trees with enhanced suffix arrays.
J. Discrete Algorithms 2(1): 53-86 (2004).



M.I. Abouelhoda, E. Ohlebusch, S. Kurtz.
Optimal Exact String Matching Based on Suffix Arrays.
SPIRE 2002 31-43 (2002).



M.I. Abouelhoda, S. Kurtz, E. Ohlebusch.
Enhanced suffix arrays and applications.
In S. Aluru, editor, *Handbook on Computational Molecular Biology*, pp
7-1-7-27. Chapman and Hall/CRC, 2006.