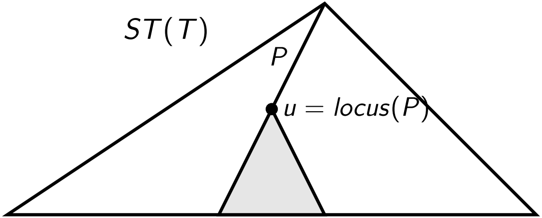
Cross-Document Pattern Matching

Gregory Kucherov[1] Yakov Nekrich[2] Tatiana Starikovskaya[3,1]

[1]Université Paris-Est & CNRS, [2]University of Chile, [3]Lomonosov Moscow State University.

Pattern Matching Problem

Given a text T and a pattern P, count all occurrences of P in T.



$O(|P|)$ **time,** $O(|T|)$ **space**

Cross-Document Pattern Matching Problem

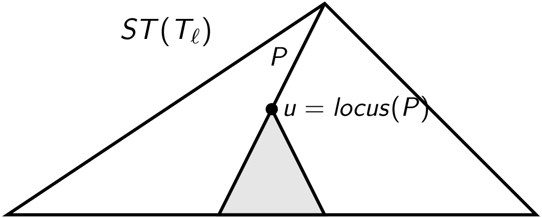
Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, count all occurrences of P in T_ℓ.

Example

documents: genomic sequences
pattern: a fragment of one of the sequences

Cross-Document Pattern Matching Problem

Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, count all occurrences of P in T_ℓ.



Standard solution: $O(|P|)$ **time,** $O(|T_\ell|)$ **space**

Faster solution? **Yes.**

Variants

- Counting
- Reporting
- Document counting and reporting
- Dynamic counting and reporting

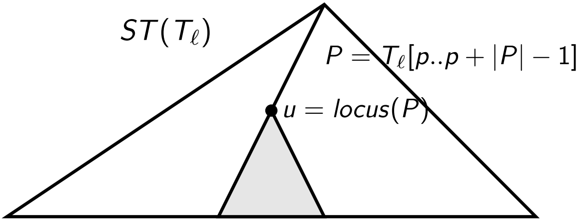
Variants

- Counting
- Reporting
- Document counting and reporting
- Dynamic counting and reporting

Counting

Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, count all occurrences of P in T_ℓ.

1) identify a position p of some occurrence of P in T_ℓ
2) find the locus of $T_\ell[p..p+|P|-1]$ in $ST(T_\ell)$, and retrieve the number of leaves in its subtree



Counting: step 1

1) identify a position p of some occurrence of P in T_ℓ

		r_1		r		r_2	
GSA		$T_\ell[p_1..]$		$T_k[i..]$		$T_\ell[p_2..]$	
D		ℓ		k		ℓ	

p_1, p_2: starting positions of the closest to $T_k[i..]$ suffixes of T_ℓ

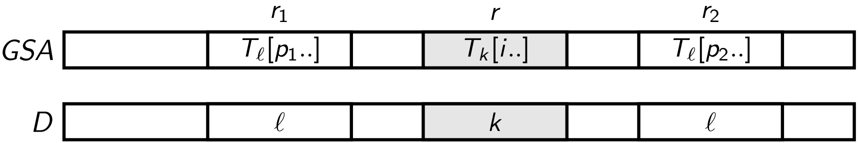
$r_1 = select(\ell, rank(D[1..r-1], \ell))$

$r_2 = select(\ell, rank(D[1..r-1], \ell) + 1)$

[Golynski et al. 2006] Rank and select queries on D can be supported in $O(1)$ and $O(\log\log m)$ time respectively.

Counting: step 1

1) identify a position p of some occurrence of P in T_ℓ



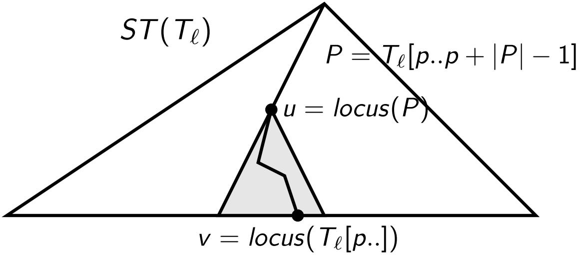
$\Rightarrow$ Positions p_1 and p_2 can be computed in $O(\log\log m)$ time.

P occurs at $p_1 \Leftrightarrow lcp(T_\ell[p_1..], T_k[i..]) \geq |P|$.

Step 1 takes $O(\log\log m)$ **time**.

Counting: step 2

2) find the locus of $T_\ell[p..p+|P|-1]$ in $ST(T_\ell)$, and retrieve the number of leaves in its subtree



$weight(w)$: string depth of a node w

$w = wla(v, q)$: the ancestor of v of minimal depth s.t. $weight(w) \geq q$

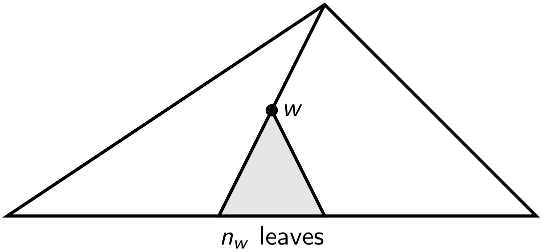
$u = wla(v, |P|)$

Weighted Level Ancestor Problem

[Farach et al. 1996, Amir et al. 2007] $w = wla(v, q)$ can be found in $O(\log\log W)$ time and linear space, where W is the maximal weight of a node in the tree.

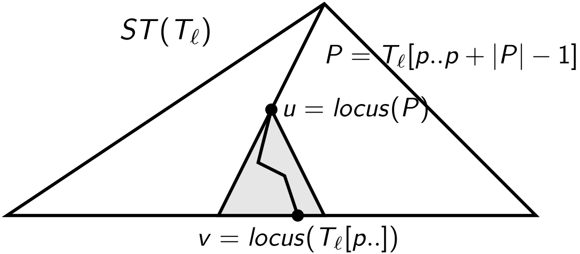
Theorem

$w = wla(v, q)$ can be found in $O(\min\{\sqrt{\log n_w / \log\log n_w}, \log\log q\})$ time and linear space.



Counting: step 2

2) find the locus of $T_\ell[p..p+|P|-1]$ in $ST(T_\ell)$, and retrieve the number of leaves in its subtree

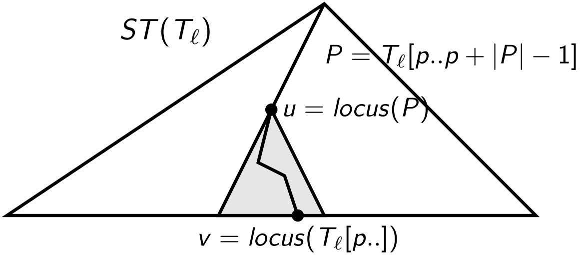


$u = wla(v, |P|), n_u = occ$

$\Rightarrow$ u **can be found in** $\min\{\sqrt{\log occ / \log\log occ}, \log\log|P|\}$ **time**.

Counting: step 2

2) find the locus of $T_\ell[p..p+|P|-1]$ in $ST(T_\ell)$, and retrieve the number of leaves in its subtree



Theorem

Counting takes $O(t + \log\log m)$ time and $O(n)$ space, where $t = \min\{\sqrt{\log occ / \log\log occ}, \log\log|P|\}$.

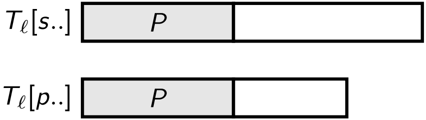
Variants

- Counting
- Reporting
- Document counting and reporting
- Dynamic counting and reporting

Reporting

Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, report all occurrences of P in T_ℓ.

1) identify a position p of T_ℓ at which P occurs
Step 1 of Counting, takes $O(\log \log m)$ **time**
2) report all s: $lcp(T_\ell[p..], T_\ell[s..]) \geq |P|$

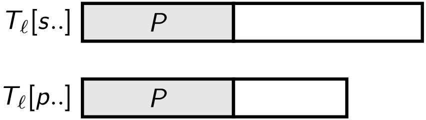


$\Leftrightarrow lcp(T_\ell[p..], T_\ell[s..]) \geq |P|$

Reporting

Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, report all occurrences of P in T_ℓ.

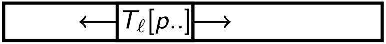
1) identify a position p of T_ℓ at which P occurs
   Step 1 of Counting, takes $O(\log\log m)$ **time**
2) report all s: $lcp(T_\ell[p..], T_\ell[s..]) \geq |P|$



$\Leftrightarrow lcp(T_\ell[p..], T_\ell[s..]) \geq |P|$

Reporting: step 2

2) report all s: $lcp(T_\ell[p..], T_\ell[s..]) \geq |P|$

$SA(T_\ell)$ 

while $lcp(T_\ell[s..], T_\ell[p..]) \geq |P|$, report s

Theorem

Reporting takes $O(\log\log m + occ)$ time and $O(n)$ space.

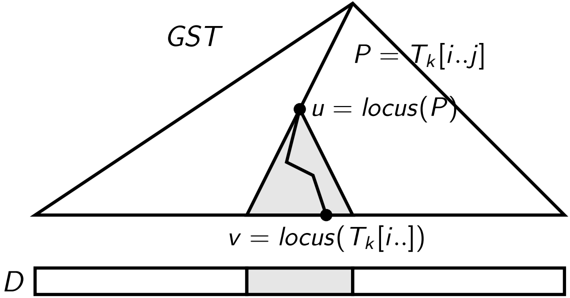
Variants

- Counting
- Reporting
- Document counting and reporting
- Dynamic counting and reporting

Document counting and reporting

Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, count or report all documents in which P occurs.

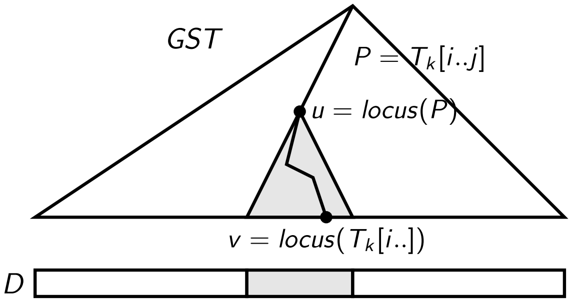
1) find $u = locus(P)$ in the generalized suffix tree
   Reduction to the WLA Problem
   $O(\min\{\sqrt{\log docc / \log\log docc}, \log\log |P|\})$ **time**
2) report or count distinct documents in the subtree of u



Document counting and reporting

Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, count or report all documents in which P occurs.

1) find $u = locus(P)$ in the generalized suffix tree
   Reduction to the WLA Problem
   $O(\min\{\sqrt{\log docc / \log\log docc}, \log\log|P|\})$ **time**
2) report or count distinct documents in the subtree of u



Document counting and reporting: step 2

2) report or count distinct documents in the subtree of $u \Leftrightarrow$ report or count distinct documents in the corresponding segment of the document array D

[Muthukrishnan 2002] Reporting of distinct documents in a segment of D takes $O(ndocs)$ time and $O(n)$ space.

Theorem

Document reporting takes $O(t + ndocs)$ time and $O(n)$ space, where $t = \min\{\sqrt{\log docc / \log\log docc}, \log\log|P|\}$.

[Bozanis et al. 1995] Counting of distinct documents in a segment of D takes $O(\log n)$ time and $O(n)$ space.

Theorem

Document counting takes $O(\log n)$ time and $O(n)$ space.

Variants

- Counting
- Reporting
- Document counting and reporting
- Dynamic counting and reporting

Dynamic counting and reporting

Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, count all occurrences of P in T_ℓ. Dynamic operation: adding a document.

1) find a position p of some occurrence of P in T_ℓ
2) find the locus of $T_\ell[p..p+|P|-1]$ in $ST(T_\ell)$, and retrieve the number of leaves in its subtree

Dynamic counting and reporting

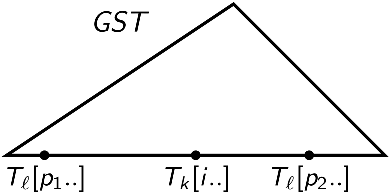
Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, count all occurrences of P in T_ℓ. Dynamic operation: adding a document.

1) find a position p of some occurrence of P in T_ℓ — $O(\log n)$ **time**

Dynamic counting and reporting

Given a set of documents $T_1, T_2, \ldots, T_m$ and a pattern $P = T_k[i..j]$, count all occurrences of P in T_ℓ. Dynamic operation: adding a document.

1) find a position p of some occurrence of P in T_ℓ — $O(\log n)$ **time**



[Dietz et al. 1987] to compare ranks of any two leaves in $O(1)$ time
Suffix array of T_ℓ

Summary of the results

m, n: the number of the documents and their total length resp.

- Counting: $O(\log\log m + \min\{\sqrt{\log occ / \log\log occ}, \log\log |P|\})$ time
- Reporting: $O(\log\log m + occ)$ time
- Document counting: $O(\log n)$ time
- Document reporting: $O(\min\{\sqrt{\log docc / \log\log docc}, \log\log |P|\} + ndocs)$ time
- Dynamic counting: $O(\log n)$ time
- Dynamic reporting: $O(\log n + occ)$ time

  (update: $O(\log n)$ time per letter)

Summary of the results

m, n: the number of the documents and their total length resp.

- Counting: $O(\log\log m + \min\{\sqrt{\log occ / \log\log occ}, \log\log |P|\})$ time
- Reporting: $O(\log\log m + occ)$ time

- Document counting: $O(\log n)$ time
- Document reporting: $O(\min\{\sqrt{\log docc / \log\log docc}, \log\log |P|\} + ndocs)$ time

- Dynamic counting: $O(\log n)$ time
- Dynamic reporting: $O(\log n + occ)$ time

  (update: $O(\log n)$ time per letter)

- Succinct data structures for counting, reporting and document reporting