FM-index of an alignment: A compressed index for highly similar strings

Joong Chae Na [1] Hyunjoon Kim [2] Heejin Park [3] Thierry Lecroq [4] Martine Léonard [4] Laurent Mouchard [4] Kunsoo Park [2]

[1] Sejong University, Seoul, Korea
[2] Seoul National University, Korea
[3] Hanyang University, Seoul, Korea
[4] Normandie Université, Université de Rouen, LITIS EA 4108, France

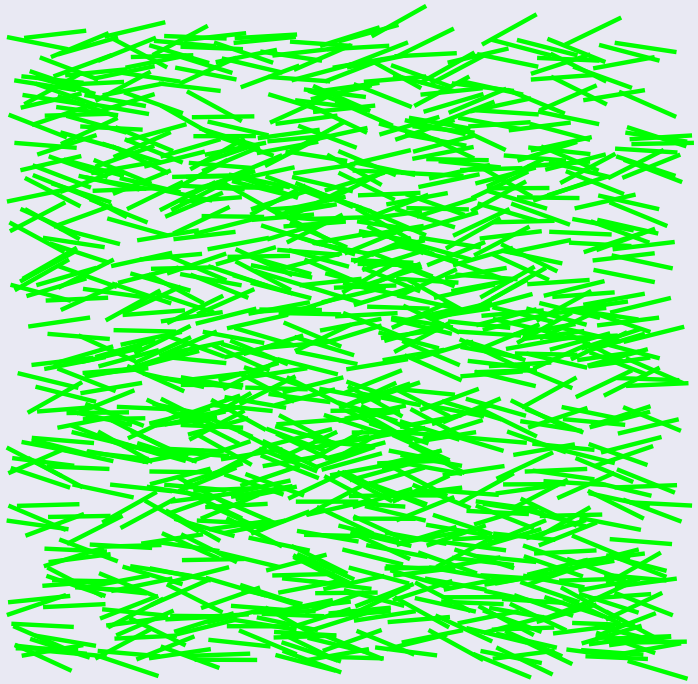
SeqBio 2015
November 26th-27th 2015 – Orsay, France

Next Generation Sequencing (NGS)

- millions of sort (length ~ 150) fragments called *reads*
- 2 types of projects:
  - re-sequencing: (*mapping*) reads on a reference gnome
  - *de novo* (*assembling*) reads

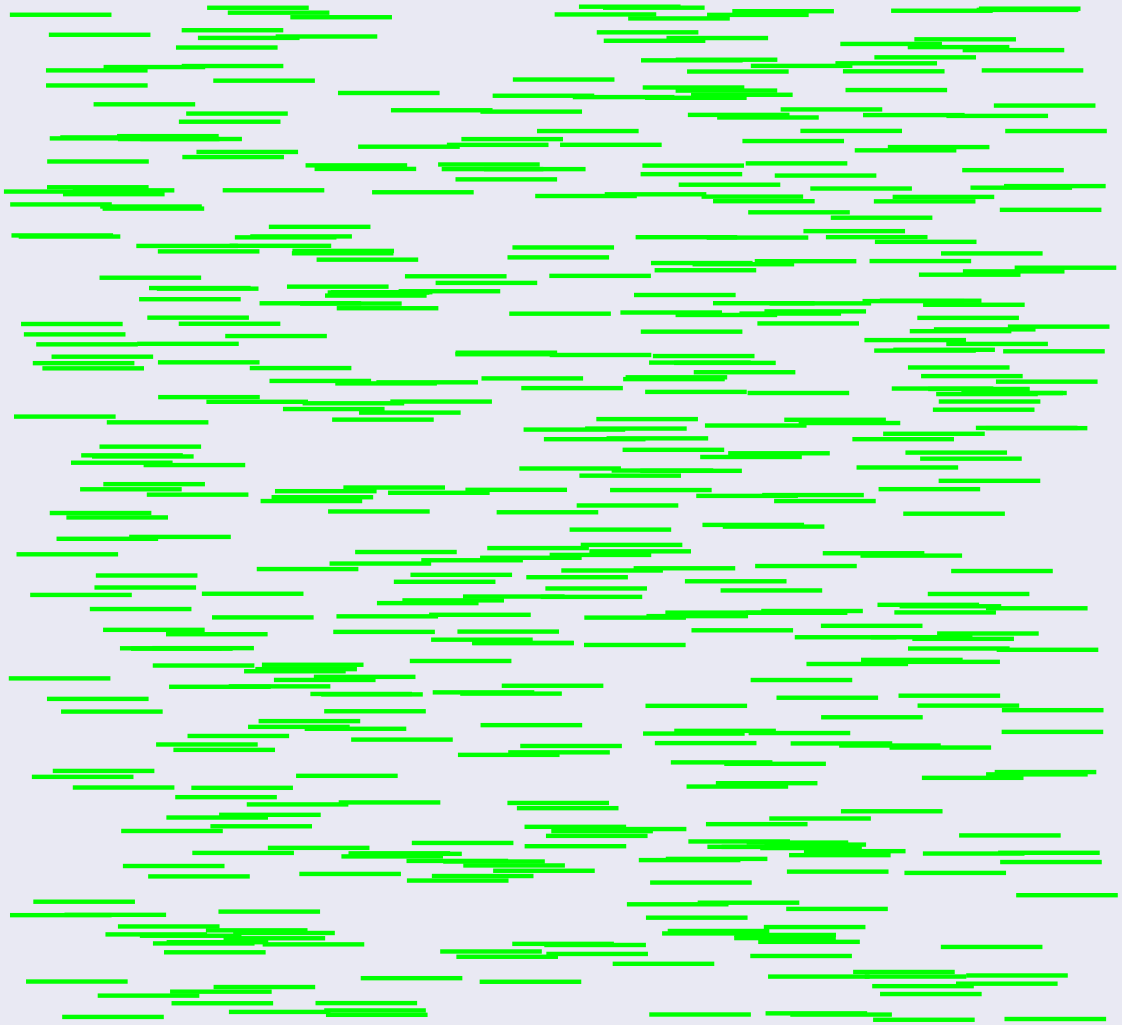
Sequencer

Reads

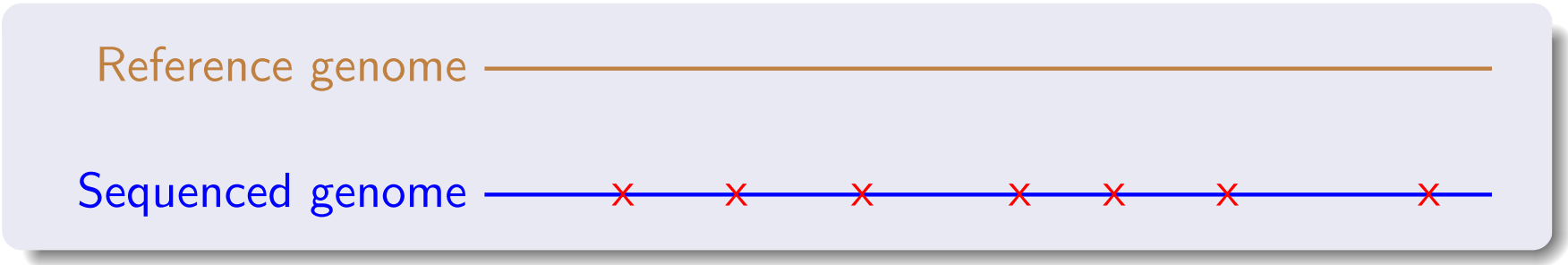


Re-sequencing

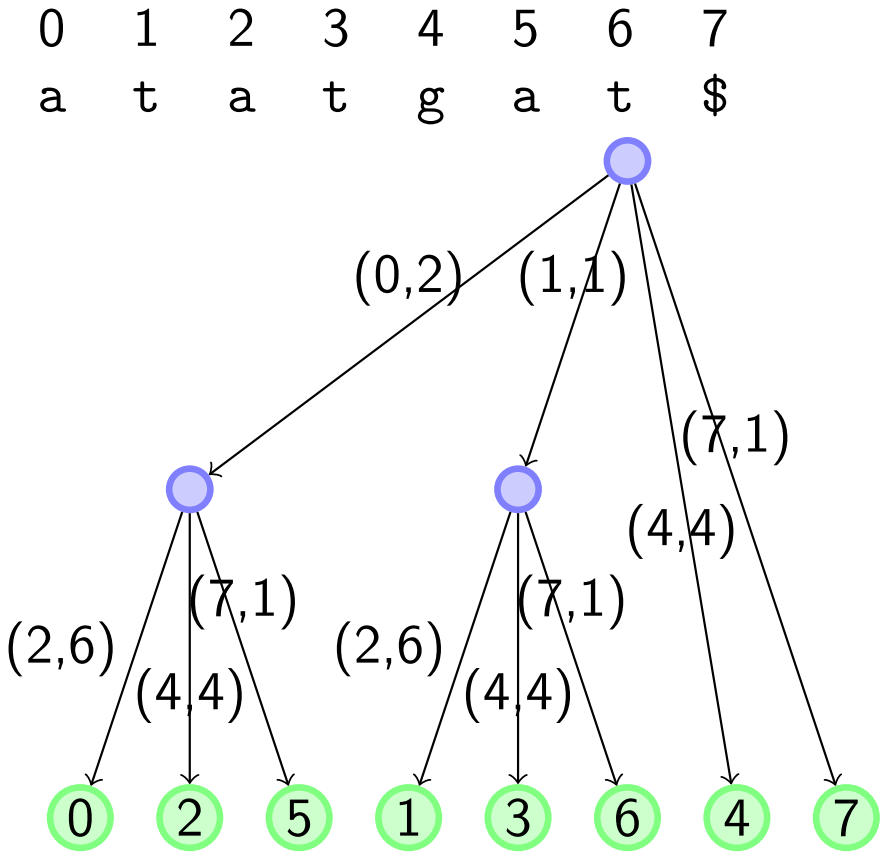
Reference genome



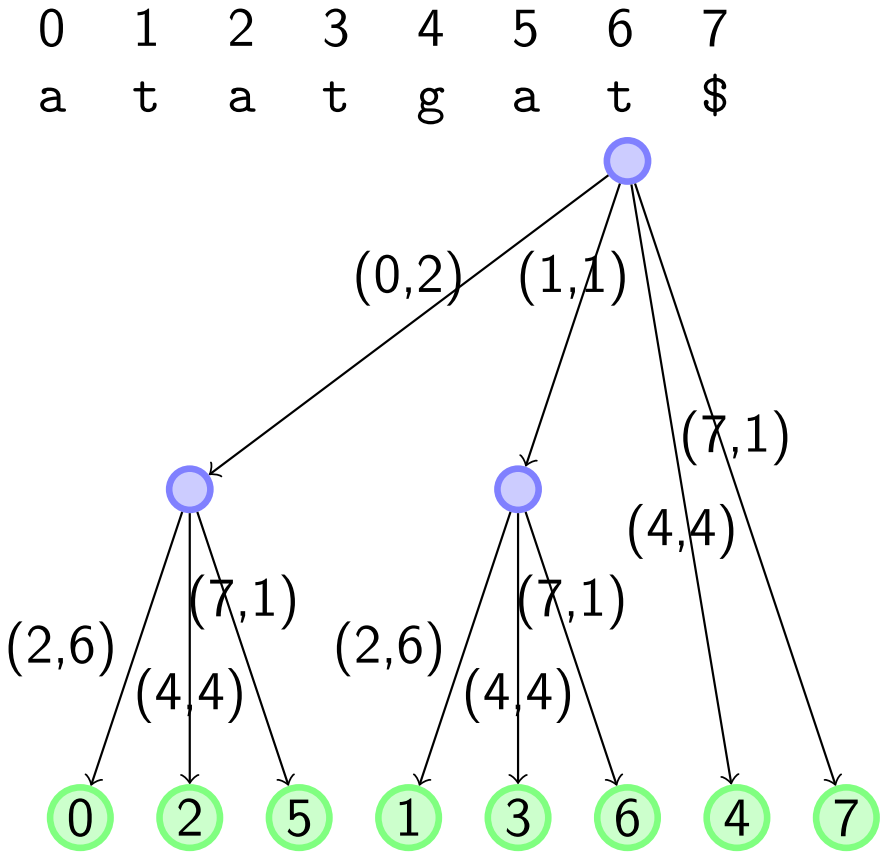
Variations



Suffix Tree



Suffix Tree



Human genome 3.3 Gbp

3.3 Gbp $\times$ 12.5 bytes $\rightarrow$ 41.25 GB

Suffix Array

		0	1	2	3	4	5	6	7
y	=	a	t	a	t	g	a	t	$

SA	sort of the suffixes
7	$
5	at$
0	atatgat$
2	atgat$
4	gat$
6	t$
1	tatgat$
3	tgat$

Suffix Array

		0	1	2	3	4	5	6	7
y	$=$	a	t	a	t	g	a	t	$

SA	sort of the suffixes
7	$
5	at$
0	atatgat$
2	atgat$
4	gat$
6	t$
1	tatgat$
3	tgat$

$n \log n$ bits

Burrows-Wheeler Transform (BWT)

		0	1	2	3	4	5	6	7
y	=	a	t	a	t	g	a	t	$

sort of the conjugates

F		L
$	atatga	t
a	t$atat	g
a	tatgat	$
a	tgat$a	t
g	at$ata	t
t	$atatg	a
t	atgat$	a
t	gat$at	a

BWT = tg$ttaaa

Burrows-Wheeler Transform (BWT)

		0	1	2	3	4	5	6	7
y	$=$	a	t	a	t	g	a	t	$

sort of the conjugates

F		L
$	atatga	t
a	t$atat	g
a	tatgat	$
a	tgat$a	t
g	at$ata	t
t	$atatg	a
t	atgat$	a
t	gat$at	a

BWT $=$ tg$ttaaa

$n \log \sigma$ bits

Burrows-Wheeler Transform (BWT)

	0	1	2	3	4	5	6	7
y =	a	t	a	t	g	a	t	$

	a	c	g	t
C	1	1	4	5

a	c	g	t
0	0	0	1
0	0	1	0
0	0	0	0
0	0	0	1
0	0	0	1
1	0	0	0
1	0	0	0
1	0	0	0

$\sigma \log n + \sigma n$ bits

FM-index

Self-index

sampled suffix array + BWT

FM-index

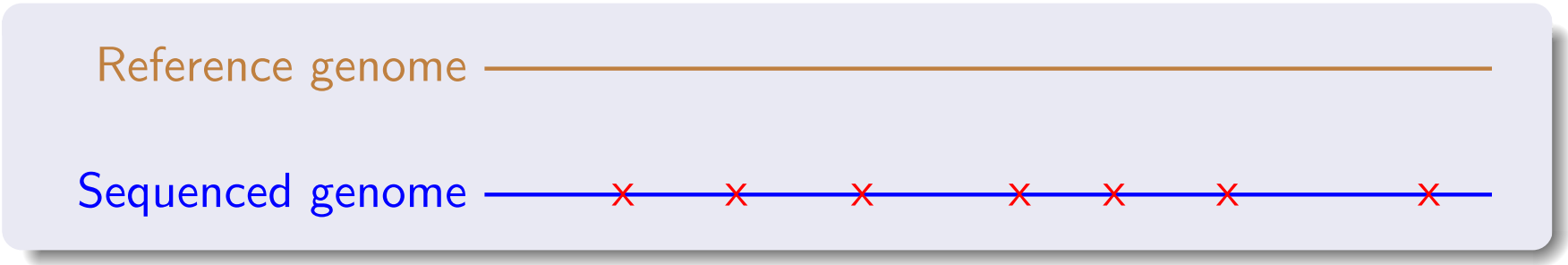
Self-index

sampled suffix array + BWT

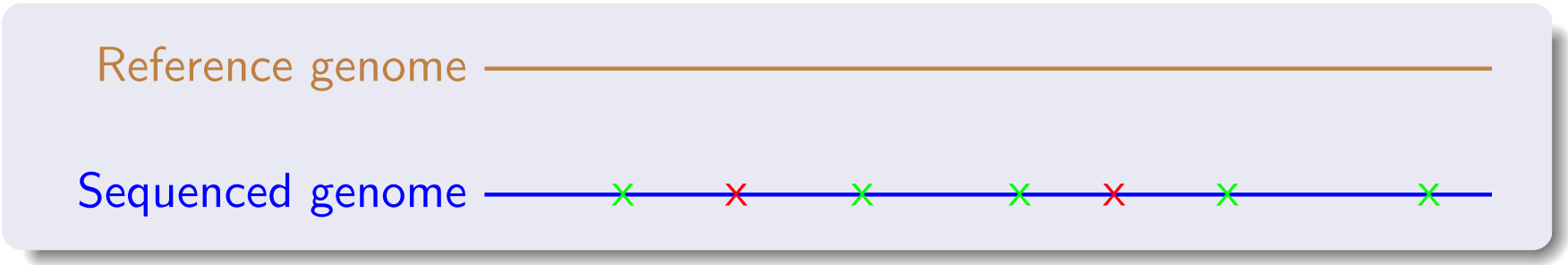
Human genome

less than 2GB

After sequencing



After sequencing



Reference genomes

Similar sequences

Example with 4 sequences

$S^1 = \texttt{atccaactccc\$}$
$S^2 = \texttt{attcaactcac\$}$
$S^3 = \texttt{atcttactcac\$}$
$S^4 = \texttt{atacaactccc\$}$

Similar sequences

Example with 4 sequences

$S^1 = \texttt{atccaactccc\$}$
$S^2 = \texttt{attcaactcac\$}$
$S^3 = \texttt{atcttactcac\$}$
$S^4 = \texttt{ataca actccc\$}$

Similar sequences

Example with 4 sequences

$S^1 = \texttt{atccaactccc\$}$
$S^2 = \texttt{attcaactcac\$}$
$S^3 = \texttt{atcttactcac\$}$
$S^4 = \texttt{ataacaactccc\$}$

$\rho = \texttt{at(cca/tca/ctt/aca)actc(c/a/a/c)c\$}$

Similar sequences

Formally

A set of sequences, for $1 \leq j \leq r$:

$$S^j = \alpha_1 \Delta_1^j \cdots \alpha_k \Delta_k^j \alpha_{k+1}$$

the alignment is thus denoted by:

$$\rho = \alpha_1 (\Delta_1^1 / \Delta_1^2 / \cdots / \Delta_1^r) \cdots \alpha_k (\Delta_k^1 / \Delta_k^2 / \cdots / \Delta_k^r) \alpha_{k+1} \$$$

where

- α_is are common parts
- Δ_i^js are non-common parts

Longest repeated suffix of α_is

For each $1 \leq i \leq k$ and $1 \leq j \leq r$

$$\alpha_i = \alpha_i^{\diamond^j} \alpha_i^{*^j}$$

where $\alpha_i^{*^i}$ is the longest suffix of α_i occurring at least twice in S^j

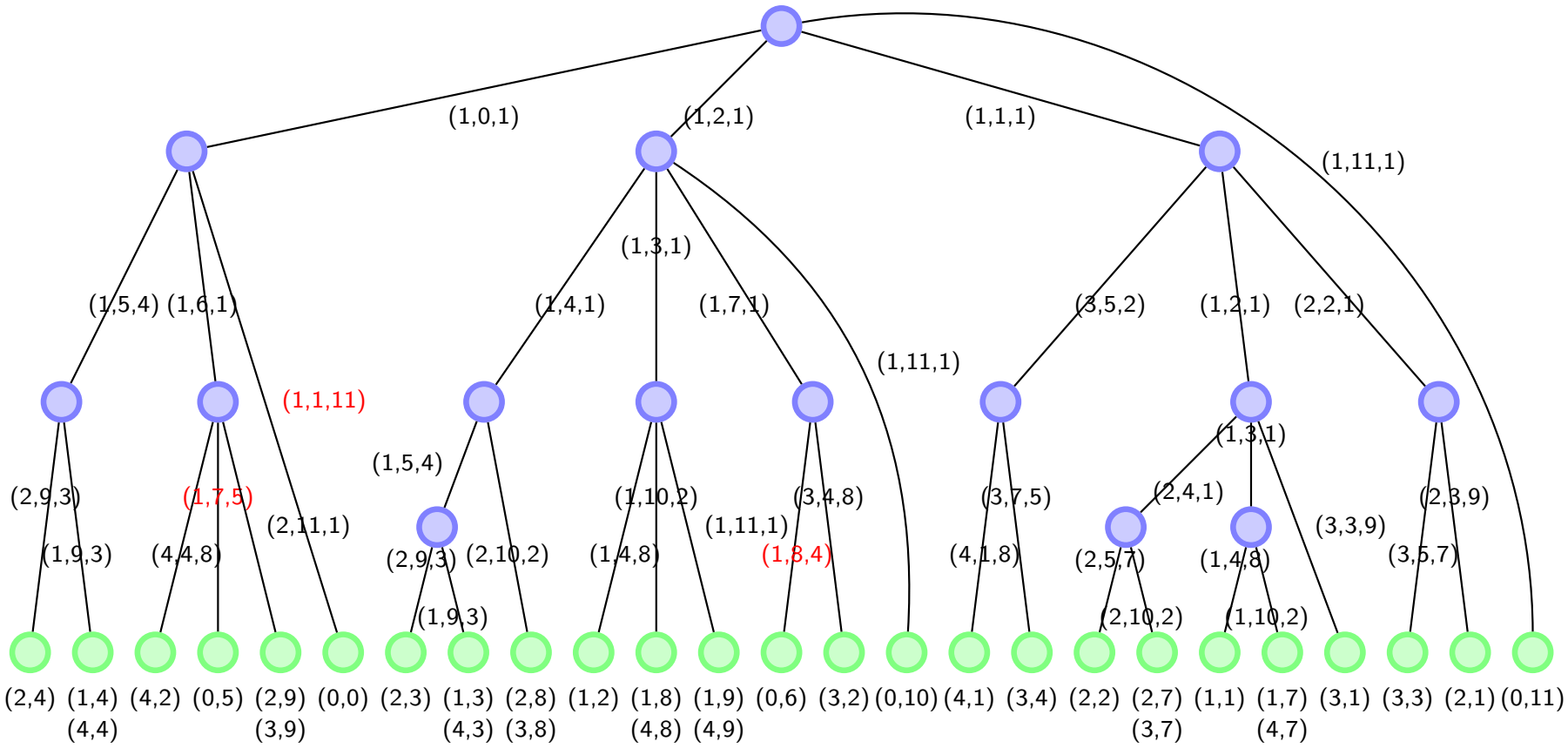
Then

$$\alpha_i = \alpha_i^{\diamond} \alpha_i^{*}$$

where α_i^{*} is the longest among the $\alpha_i^{*^j}$s

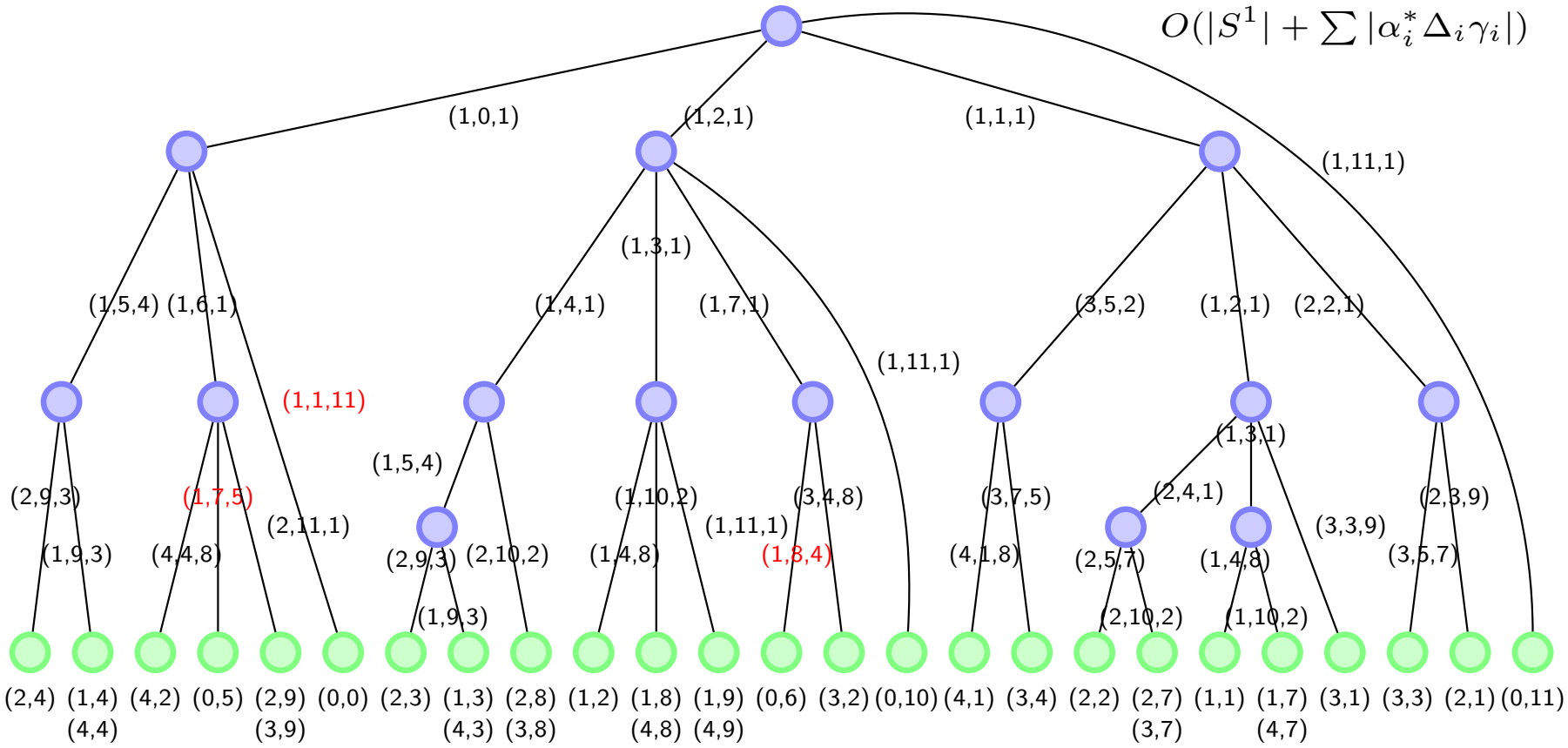
Suffix tree of an alignment

	0	1	2	3	4	5	6	7	8	9	10	11
S_1	a	t	c	c	a	a	c	t	c	c	c	$
S_2	a	t	t	c	a	a	c	t	c	a	c	$
S_3	a	t	c	t	t	a	c	t	c	a	c	$
S_4	a	t	a	c	a	a	c	t	c	c	c	$
	$\alpha_1^\diamond$	α_1^*	Δ_1			$\alpha_2^\diamond$		α_2^*		Δ_2	α_3	



Suffix tree of an alignment

	0	1	2	3	4	5	6	7	8	9	10	11
S_1	a	t	c	c	a	a	c	t	c	c	c	$
S_2	a	t	t	c	a	a	c	t	c	a	c	$
S_3	a	t	c	t	t	a	c	t	c	a	c	$
S_4	a	t	a	c	a	a	c	t	c	c	c	$
	$\alpha_1^\diamond$	α_1^*	Δ_1			$\alpha_2^\diamond$		α_2^*		Δ_2	α_3	



Suffix array of an alignment

i		SA	sort of the suffixes
0	0	11	$
1	2	4	aactcac$
2	1	4	aactccc$
3	4	4	aactccc$
4	2	9	ac$
5	4	9	ac$
6	4	2	acaactccc$
7	0	5	actc(c/a/a/c)c$
8	0	0	at(cca/tca/ctt/aca)actc(c/a/a/c)c$
9	0	10	c$
10	2	3	caaactcac$
11	1	3	caaactccc$
12	4	3	caaactccc$
13	2	8	cac$
14	3	8	cac$
15	1	9	cc$
16	4	9	cc$
17	1	2	ccaactccc$
18	1	8	ccc$
19	4	8	ccc$
20	0	6	ctc(c/a/a/c)c$
21	3	2	cttactcac$
22	4	1	tacaactccc$
23	3	4	tactcac$
24	2	2	tcaactcac$
25	2	7	tcac$
26	3	7	tcac$
27	1	1	tccaactccc$
28	1	7	tccc$
29	4	7	tccc$
30	3	1	tcttactcac$
31	3	3	ttactcac$
32	2	1	ttcaactcac$

Suffix array of an alignment

i		SA	sort of the suffixes
0	0	11	$
1	2	4	aactcac$
2	1	4	aactccc$
3	4	4	aactccc$
4	2	9	ac$
5	4	9	ac$
6	4	2	acaactccc$
7	0	5	actc(c/a/a/c)c$
8	0	0	at(cca/tca/ctt/aca)actc(c/a/a/c)c$
9	0	10	c$
10	2	3	caaactcac$
11	1	3	caaactccc$
12	4	3	caaactccc$
13	2	8	cac$
14	3	8	cac$
15	1	9	cc$
16	4	9	cc$
17	1	2	ccaactccc$
18	1	8	ccc$
19	4	8	ccc$
20	0	6	ctc(c/a/a/c)c$
21	3	2	cttactcac$
22	4	1	tacaactccc$
23	3	4	tactcac$
24	2	2	tcaactcac$
25	2	7	tcac$
26	3	7	tcac$
27	1	1	tccaactccc$
28	1	7	tccc$
29	4	7	tccc$
30	3	1	tcttactcac$
31	3	3	ttactcac$
32	2	1	ttcaactcac$

Burrows-Wheeler Transform of an alignment

i	SA		F		L
0	0	11	$	at(cca/tca/ctt/aca)actc(c/a/a/c)	c
1	1, 2, 4	4	a	actc(c/a/c)$at(c/t/a)	c
2	2, 3	9	a	c$at(tca/ctt)act	c
3	4	2	a	caactccc$a	t
4	0	5	a	ctc(c/a/a/c)c$at(cc/tc/ct/ac)	a,t
5	0	0	a	t(cca/tca/ctt/aca)actc(c/a/a/c)	$
6	0	10	c	$at(cca/tca/ctt/aca)actc	a,c
7	1, 2, 4	3	c	(ca/ca/ca)actc(c/a/c)c$at	a,c,t
8	2, 3	8	c	(a/a)c$at(tca/ctt)ac	t
9	1, 4	9	c	c$at(cca/aca)act	c
10	1	2	c	caactccc$a	t
11	1, 4	8	c	(c/c)c$at(cca/aca)ac	t
12	0	6	c	tc(c/a/a/c)c$at(cca/tca/ctt/aca)	a
13	3	2	c	ttactcac$a	t
14	4	1	t	acaactccc$	a
15	3	4	t	actcac$atc	t
16	2	2	t	caactcac$a	t
17	2, 3	7	t	c(a/a)c$at(tca/aca)ac	c
18	1	1	t	ccaactccc$	a
19	1, 4	7	t	c(c/c)c$at(cca/aca)ac	c
20	3	1	t	cttactcac$	a
21	3	3	t	tactcac$at	c
22	2	1	t	tcaactcac$	a

Burrows-Wheeler Transform of an alignment

i	SA		F		L	a		c		g	t
0	0	11	$	at(cca/tca/ctt/aca)actc(c/a/a/c)	c	0		1		0	0
1	1, 2, 4	4	a	actc(c/a/c)$at(c/t/a)	c	0		1		0	0
2	2, 3	9	a	c$at(tca/ctt)act	c	0		1		0	0
3	4	2	a	caactccc$a	t	0		0		0	1
4	0	5	a	ctc(c/a/a/c)c$at(cc/tc/ct/ac)	a,t	1		0		0	1
5	0	0	a	t(cca/tca/ctt/aca)actc(c/a/a/c)$	$	0		0		0	0
6	0	10	c	$at(cca/tca/ctt/aca)actc	a,c	1		1		0	0
7	1, 2, 4	3	c	(ca/ca/ca)actc(c/a/c)c$at	a,c,t	1		1		0	1
8	2, 3	8	c	(a/a)c$at(tca/ctt)ac	t	0		0		0	1
9	1, 4	9	c	c$at(cca/aca)act	c	0		1		0	0
10	1	2	c	caactccc$a	t	0		0		0	1
11	1, 4	8	c	(c/c)c$at(cca/aca)ac	t	0		0		0	1
12	0	6	c	tc(c/a/a/c)c$at(cca/tca/ctt/aca)	a	1		0		0	0
13	3	2	c	ttactcac$a	t	0		0		0	1
14	4	1	t	acaactccc$	a	1	1	0		0	0
15	3	4	t	actcac$atc	t	0		0		0	1
16	2	2	t	caactcac$a	t	0		0		0	1
17	2, 3	7	t	c(a/a)c$at(tca/aca)a	c	0		1	1	0	0
18	1	1	t	ccaactccc$	a	0	1	0		0	0
19	1, 4	7	t	c(c/c)c$at(cca/aca)a	c	0		0	1	0	0
20	3	1	t	cttactcac$	a	0	1	0		0	0
21	3	3	t	tactcac$at	c	0		1		0	0
22	2	1	t	tcaactcac$	a	0	1	0		0	0

Search for tactcc

Burrows-Wheeler Transform of an alignment

	i	SA		F		L	a	c	g	t
	0	0	11	$	at(cca/tca/ctt/aca)actc(c/a/a/c)	c	0	1	0	0
	1	1, 2, 4	4	a	actc(c/a/c)$at(c/t/a)	c	0	1	0	0
	2	2, 3	9	a	c$at(tca/ctt)act	c	0	1	0	0
	3	4	2	a	caactccc$a	t	0	0	0	1
	4	0	5	a	ctc(c/a/a/c)c$at(cc/tc/ct/ac)	a,t	1	0	0	1
	5	0	0	a	t(cca/tca/ctt/aca)actc(c/a/a/c)$	$	0	0	0	0
→	6	0	10	c	$at(cca/tca/ctt/aca)actc	a,c	1	1	0	0
	7	1, 2, 4	3	c	(ca/ca/ca)actc(c/a/c)c$at	a,c,t	1	1	0	1
	8	2, 3	8	c	(a/a)c$at(tca/ctt)ac	t	0	0	0	1
	9	1, 4	9	c	c$at(cca/aca)act	c	0	1	0	0
	10	1	2	c	caactccc$a	t	0	0	0	1
	11	1, 4	8	c	(c/c)c$at(cca/aca)ac	t	0	0	0	1
	12	0	6	c	tc(c/a/a/c)c$at(cca/tca/ctt/aca)	a	1	0	0	0
→	13	3	2	c	ttactcac$a	t	0	0	0	1
	14	4	1	t	acaactccc$	a	1	1 0	0	0
	15	3	4	t	actcac$atc	t	0	0	0	1
	16	2	2	t	caactcac$a	t	0	0	0	1
	17	2, 3	7	t	c(a/a)c$at(tca/aca)a	c	0	1	1 0	0
	18	1	1	t	ccaactccc$	a	0	1 0	0	0
	19	1, 4	7	t	c(c/c)c$at(cca/aca)a	c	0	0	1 0	0
	20	3	1	t	cttactcac$	a	0	1 0	0	0
	21	3	3	t	tactcac$at	c	0	1	0	0
	22	2	1	t	tcaactcac$	a	0	1 0	0	0

Search for tactcc

c

Burrows-Wheeler Transform of an alignment

	i	SA		F		L	a	c	g	t
	0	0	11	$	at(cca/tca/ctt/aca)actc(c/a/a/c)	c	0	1	0	0
	1	1, 2, 4	4	a	actc(c/a/c)$at(c/t/a)	c	0	1	0	0
	2	2, 3	9	a	c$at(tca/ctt)act	c	0	1	0	0
	3	4	2	a	caactccc$a	t	0	0	0	1
	4	0	5	a	ctc(c/a/a/c)c$at(cc/tc/ct/ac)	a,t	1	0	0	1
	5	0	0	a	t(cca/tca/ctt/aca)actc(c/a/a/c)$	$	0	0	0	0
	6	0	10	c	$at(cca/tca/ctt/aca)actc	a,c	1	1	0	0
	7	1, 2, 4	3	c	(ca/ca/ca)actc(c/a/c)c$at	a,c,t	1	1	0	1
	8	2, 3	8	c	(a/a)c$at(tca/ctt)ac	t	0	0	0	1
→	9	1, 4	9	c	c$at(cca/aca)act	c	0	1	0	0
	10	1	2	c	caactccc$a	t	0	0	0	1
→	11	1, 4	8	c	(c/c)c$at(cca/aca)ac	t	0	0	0	1
	12	0	6	c	tc(c/a/a/c)c$at(cca/tca/ctt/aca)	a	1	0	0	0
	13	3	2	c	ttactcac$a	t	0	0	0	1
	14	4	1	t	acaactccc$	a	1	1 0	0	0
	15	3	4	t	actcac$atc	t	0	0	0	1
	16	2	2	t	caactcac$a	t	0	0	0	1
	17	2, 3	7	t	c(a/a)c$at(tca/aca)a	c	0	1	1 0	0
	18	1	1	t	ccaactccc$	a	0	1 0	0	0
	19	1, 4	7	t	c(c/c)c$at(cca/aca)a	c	0	0	1 0	0
	20	3	1	t	cttactcac$	a	0	1 0	0	0
	21	3	3	t	tactcac$at	c	0	1	0	0
	22	2	1	t	tcaactcac$	a	0	1 0	0	0

Search for tactcc

cc

Burrows-Wheeler Transform of an alignment

	i	SA		F		L	a	c	g	t
	0	0	11	$	at(cca/tca/ctt/aca)actc(c/a/a/c)	c	0	1	0	0
	1	1, 2, 4	4	a	actc(c/a/c)$at(c/t/a)	c	0	1	0	0
	2	2, 3	9	a	c$at(tca/ctt)act	c	0	1	0	0
	3	4	2	a	caactccc$a	t	0	0	0	1
	4	0	5	a	ctc(c/a/a/c)c$at(cc/tc/ct/ac)	a,t	1	0	0	1
	5	0	0	a	t(cca/tca/ctt/aca)actc(c/a/a/c)$	$	0	0	0	0
	6	0	10	c	$at(cca/tca/ctt/aca)actc	a,c	1	1	0	0
	7	1, 2, 4	3	c	(ca/ca/ca)actc(c/a/c)c$at	a,c,t	1	1	0	1
	8	2, 3	8	c	(a/a)c$at(tca/ctt)ac	t	0	0	0	1
	9	1, 4	9	c	c$at(cca/aca)act	c	0	1	0	0
	10	1	2	c	caactccc$a	t	0	0	0	1
	11	1, 4	8	c	(c/c)c$at(cca/aca)ac	t	0	0	0	1
	12	0	6	c	tc(c/a/a/c)c$at(cca/tca/ctt/aca)	a	1	0	0	0
	13	3	2	c	ttactcac$a	t	0	0	0	1
	14	4	1	t	acaactccc$	a	1	1 0	0	0
	15	3	4	t	actcac$atc	t	0	0	0	1
	16	2	2	t	caactcac$a	t	0	0	0	1
	17	2, 3	7	t	c(a/a)c$at(tca/aca)a	c	0	1	1 0	0
→	18	1	1	t	ccaactccc$	a	0	1 0	0	0
→	19	1, 4	7	t	c(c/c)c$at(cca/aca)a	c	0	0	1 0	0
	20	3	1	t	cttactcac$	a	0	1 0	0	0
	21	3	3	t	tactcac$at	c	0	1	0	0
	22	2	1	t	tcaactcac$	a	0	1 0	0	0

Search for tactcc

tcc

Burrows-Wheeler Transform of an alignment

	i	SA		F		L	a	c	g	t
	0	0	11	$	at(cca/tca/ctt/aca)actc(c/a/a/c)	c	0	1	0	0
	1	1, 2, 4	4	a	actc(c/a/c)$at(c/t/a)	c	0	1	0	0
	2	2, 3	9	a	c$at(tca/ctt)act	c	0	1	0	0
	3	4	2	a	caactccc$a	t	0	0	0	1
	4	0	5	a	ctc(c/a/a/c)c$at(cc/tc/ct/ac)	a,t	1	0	0	1
	5	0	0	a	t(cca/tca/ctt/aca)actc(c/a/a/c)$	$	0	0	0	0
	6	0	10	c	$at(cca/tca/ctt/aca)actc	a,c	1	1	0	0
	7	1, 2, 4	3	c	(ca/ca/ca)actc(c/a/c)c$at	a,c,t	1	1	0	1
	8	2, 3	8	c	(a/a)c$at(tca/ctt)ac	t	0	0	0	1
	9	1, 4	9	c	c$at(cca/aca)act	c	0	1	0	0
	10	1	2	c	caactccc$a	t	0	0	0	1
	11	1, 4	8	c	(c/c)c$at(cca/aca)ac	t	0	0	0	1
→	12	0	6	c	tc(c/a/a/c)c$at(cca/tca/ctt/aca)	a	1	0	0	0
	13	3	2	c	ttactcac$a	t	0	0	0	1
	14	4	1	t	acaactccc$	a	1	0	0	0
	15	3	4	t	actcac$atc	t	0	0	0	1
	16	2	2	t	caactcac$a	t	0	0	0	1
	17	2, 3	7	t	c(a/a)c$at(tca/aca)a	c	0	1	0	0
	18	1	1	t	ccaactccc$	a	0	0	0	0
	19	1, 4	7	t	c(c/c)c$at(cca/aca)a	c	0	0	0	0
	20	3	1	t	cttactcac$	a	0	0	0	0
	21	3	3	t	tactcac$at	c	0	1	0	0
	22	2	1	t	tcaactcac$	a	0	0	0	0

Search for tactcc

ctcc $Z = \{1, 4\}$

Burrows-Wheeler Transform of an alignment

	i	SA		F	L	a	c	g	t
	0	0	11	$ at(cca/tca/ctt/aca)actc(c/a/a/c)	c	0	1	0	0
	1	1, 2, 4	4	a actc(c/a/c)$at(c/t/a)	c	0	1	0	0
	2	2, 3	9	a c$at(tca/ctt)act	c	0	1	0	0
	3	4	2	a caactccc$a	t	0	0	0	1
→	4	0	5	a ctc(c/a/a/c)c$at(cc/tc/ct/ac)	a,t	1	0	0	1
	5	0	0	a t(cca/tca/ctt/aca)actc(c/a/a/c)$	$	0	0	0	0
	6	0	10	c $at(cca/tca/ctt/aca)actc	a,c	1	1	0	0
	7	1, 2, 4	3	c (ca/ca/ca)actc(c/a/c)c$at	a,c,t	1	1	0	1
	8	2, 3	8	c (a/a)c$at(tca/ctt)ac	t	0	0	0	1
	9	1, 4	9	c c$at(cca/aca)act	c	0	1	0	0
	10	1	2	c caactccc$a	t	0	0	0	1
	11	1, 4	8	c (c/c)c$at(cca/aca)ac	t	0	0	0	1
	12	0	6	c tc(c/a/a/c)c$at(cca/tca/ctt/aca)	a	1	0	0	0
	13	3	2	c ttactcac$a	t	0	0	0	1
	14	4	1	t acaactccc$	a	1	0	0	0
	15	3	4	t actcac$atc	t	0	0	0	1
	16	2	2	t caactcac$a	t	0	0	0	1
	17	2, 3	7	t c(a/a)c$at(tca/aca)a	c	0	1	0	0
	18	1	1	t ccaactccc$	a	0	0	0	0
	19	1, 4	7	t c(c/c)c$at(cca/aca)a	c	0	0	0	0
	20	3	1	t cttactcac$	a	0	0	0	0
	21	3	3	t tactcac$at	c	0	1	0	0
	22	2	1	t tcaactcac$	a	0	0	0	0

Search for tactcc

actcc $Z = \{1, 4\}$

Burrows-Wheeler Transform of an alignment

	i	SA		F	L	a	c	g	t
	0	0	11	$ at(cca/tca/ctt/aca)actc(c/a/a/c)	c	0	1	0	0
	1	1, 2, 4	4	a actc(c/a/c)$at(c/t/a)	c	0	1	0	0
	2	2, 3	9	a c$at(tca/ctt)act	c	0	1	0	0
	3	4	2	a caactccc$a	t	0	0	0	1
	4	0	5	a ctc(c/a/a/c)c$at(cc/tc/ct/ac)	a,t	1	0	0	1
	5	0	0	a t(cca/tca/ctt/aca)actc(c/a/a/c)$	$	0	0	0	0
	6	0	10	c $at(cca/tca/ctt/aca)actc	a,c	1	1	0	0
	7	1, 2, 4	3	c (ca/ca/ca)actc(c/a/c)c$at	a,c,t	1	1	0	1
	8	2, 3	8	c (a/a)c$at(tca/ctt)ac	t	0	0	0	1
	9	1, 4	9	c c$at(cca/aca)act	c	0	1	0	0
	10	1	2	c caactccc$a	t	0	0	0	1
	11	1, 4	8	c (c/c)c$at(cca/aca)ac	t	0	0	0	1
	12	0	6	c tc(c/a/a/c)c$at(cca/tca/ctt/aca)	a	1	0	0	0
	13	3	2	c ttactcac$a	t	0	0	0	1
	14	4	1	t acaactccc$	a	1	0	0	0
→	15	3	4	t actcac$atc	t	0	0	0	1
	16	2	2	t caactcac$a	t	0	0	0	1
	17	2, 3	7	t c(a/a)c$at(tca/aca)a	c	0	1	0	0
	18	1	1	t ccaactccc$	a	0	0	0	0
	19	1, 4	7	t c(c/c)c$at(cca/aca)a	c	0	0	0	0
	20	3	1	t cttactcac$	a	0	0	0	0
	21	3	3	t tactcac$at	c	0	1	0	0
	22	2	1	t tcaactcac$	a	0	0	0	0

Search for tactcc

tactcc $Z = \{1, 4\} \cap \{3\} = \emptyset$

Experimental results

Size

Human chromosomes 20 (63 Mbp) from 1000 genomes project

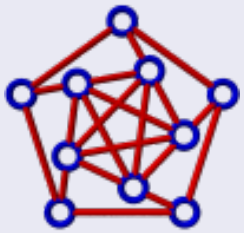
Number of sequences	10	30	60	100
FM-index	223,5	619,6	1111,2	-
RLCSA	104,7	133,0	170,0	209,0
FMA	47,6	60,2	85,7	117,6

Experimental results

Search times

# seq.	Index	Pattern length				
		10	20	30	40	50
10	FM-index	660,92	138,54	6,72	2,63	2,36
	RLCSA	57,4	12,68	0,84	0,31	0,3
	FMA	197,16	34,71	2,19	0,82	0,82
30	FM-index	1735,26	77,17	21,38	9,36	6,17
	RLCSA	151,13	9,68	2,37	1,06	0,7
	FMA	185,73	8,71	2,5	1,3	0,93
60	FM-index	3295,62	108,4	31,38	12,07	8,58
	RLCSA	441,31	13,87	4,16	1,45	0,72
	FMA	158,24	4,08	1,79	0,95	0,62
100	FM-index	-	-	-	-	-
	RLCSA	1286,89	59,54	8,04	2,01	0,98
	FMA	263,38	13,65	2,09	0,65	0,57

References



J. C. Na, H. Park, M. Crochemore, J. Holub, C. S. Iliopoulos, L. Mouchard and K. Park
Suffix Tree of an Alignment: An Efficient Index for Similar Data
Proceedings of the 24th International Workshop on Combinatorial Algorithms (IWOCA 2013), LNCS 8288, 337–348

J. C. Na, H. Park, S. Lee, M. Hong, T. Lecroq, L. Mouchard and K. Park
Suffix Array of Alignment: A Practical Index for Similar Data
Proceedings of the 20th Symposium on String Processing and Information Retrieval (SPIRE 2013), LNCS 8214, 243–254

J. C. Na, H. Kim, H. Park, T. Lecroq, M. Lonard, L. Mouchard and K. Park
FM-index of Alignment: A Compressed Index for Similar Strings
to appear in *Theoretical Computer Science*

Perspectives

- better deal with indels
- plug the method into a read aligner
- propose sets of reference genomes

Thank you for your attention

Backward search: example

i	0	1	2	3	4	5	6	7	8	9	10
y	a	c	a	a	a	c	a	t	a	t	$

	i	SA	F									L	$B^{\$}$	B^{a}	B^{c}	B^{t}
	0	10	$	a	c	a	a	a	c	a	t	a	t	0	0	0	1
	1	2	a	a	a	c	a	t	a	t	$	a	c	0	0	1	0
	2	3	a	a	c	a	t	a	t	$	a	c	a	0	1	0	0
	3	0	a	c	a	a	a	c	a	t	a	t	$	1	0	0	0
	4	4	a	c	a	t	a	t	$	a	c	a	a	0	1	0	0
	5	8	a	t	$	a	c	a	a	a	c	a	t	0	0	0	1
	6	6	a	t	a	t	$	a	c	a	a	a	c	0	0	1	0
	7	1	c	a	a	a	c	a	t	a	t	$	a	0	1	0	0
	8	5	c	a	t	a	t	$	a	c	a	a	a	0	1	0	0
	9	9	t	$	a	c	a	a	a	c	a	t	a	0	1	0	0
	10	7	t	a	t	$	a	c	a	a	a	c	a	0	1	0	0
													BWT				

c	$	a	c	t	#
C	0	1	7	9	11

$x =$ aca

Backward search: example

i	0	1	2	3	4	5	6	7	8	9	10
y	a	c	a	a	a	c	a	t	a	t	$

	i	SA	F										L	$B^{\$}$	B^{a}	B^{c}	B^{t}
	0	10	$	a	c	a	a	a	c	a	t	a	t	0	0	0	1
→	1	2	a	a	a	c	a	t	a	t	$	a	c	0	0	1	0
	2	3	a	a	c	a	t	a	t	$	a	c	a	0	1	0	0
	3	0	a	c	a	a	a	c	a	t	a	t	$	1	0	0	0
	4	4	a	c	a	t	a	t	$	a	c	a	a	0	1	0	0
	5	8	a	t	$	a	c	a	a	a	c	a	t	0	0	0	1
→	6	6	a	t	a	t	$	a	c	a	a	a	c	0	0	1	0
	7	1	c	a	a	a	c	a	t	a	t	$	a	0	1	0	0
	8	5	c	a	t	a	t	$	a	c	a	a	a	0	1	0	0
	9	9	t	$	a	c	a	a	a	c	a	t	a	0	1	0	0
	10	7	t	a	t	$	a	c	a	a	a	c	a	0	1	0	0
													BWT				

c	$	a	c	t	#
C	0	1	7	9	11

$x =$ aca
a $\rightarrow C[\text{a}], C[\text{a}+1]-1 = 1, 6$

Backward search: example

i	0	1	2	3	4	5	6	7	8	9	10
y	a	c	a	a	a	c	a	t	a	t	$

	i	SA	F									L	$B^{\$}$	B^{a}	B^{c}	B^{t}
	0	10	$	a	c	a	a	a	c	a	t	a	t	0	0	0	1
	1	2	a	a	a	c	a	t	a	t	$	a	c	0	0	1	0
	2	3	a	a	c	a	t	a	t	$	a	c	a	0	1	0	0
	3	0	a	c	a	a	a	c	a	t	a	t	$	1	0	0	0
	4	4	a	c	a	t	a	t	$	a	c	a	a	0	1	0	0
	5	8	a	t	$	a	c	a	a	a	c	a	t	0	0	0	1
	6	6	a	t	a	t	$	a	c	a	a	a	c	0	0	1	0
$\rightarrow$	7	1	c	a	a	a	c	a	t	a	t	$	a	0	1	0	0
$\rightarrow$	8	5	c	a	t	a	t	$	a	c	a	a	a	0	1	0	0
	9	9	t	$	a	c	a	a	a	c	a	t	a	0	1	0	0
	10	7	t	a	t	$	a	c	a	a	a	c	a	0	1	0	0
													BWT				

c	$	a	c	t	#
C	0	1	7	9	11

$x =$ aca
a $\rightarrow C[\text{a}], C[\text{a}+1]-1 = 1, 6$
ca $\rightarrow C[\text{c}] + rank_{\text{c}}(BWT, 1-1), C[\text{c}] + rank_{\text{c}}(BWT, 6) - 1 = 7+0, 7+2-1 = 7, 8$

Backward search: example

i	0	1	2	3	4	5	6	7	8	9	10
y	a	c	a	a	a	c	a	t	a	t	$

	i	SA	F									L	$B^{\$}$	B^{a}	B^{c}	B^{t}
	0	10	$	a	c	a	a	a	c	a	t	a	t	0	0	0	1
	1	2	a	a	a	c	a	t	a	t	$	a	c	0	0	1	0
	2	3	a	a	c	a	t	a	t	$	a	c	a	0	1	0	0
→	3	0	a	c	a	a	a	c	a	t	a	t	$	1	0	0	0
→	4	4	a	c	a	t	a	t	$	a	c	a	a	0	1	0	0
	5	8	a	t	$	a	c	a	a	a	c	a	t	0	0	0	1
	6	6	a	t	a	t	$	a	c	a	a	a	c	0	0	1	0
	7	1	c	a	a	a	c	a	t	a	t	$	a	0	1	0	0
	8	5	c	a	t	a	t	$	a	c	a	a	a	0	1	0	0
	9	9	t	$	a	c	a	a	a	c	a	t	a	0	1	0	0
	10	7	t	a	t	$	a	c	a	a	a	c	a	0	1	0	0
													BWT				

c	$	a	c	t	#
C	0	1	7	9	11

$x =$ aca
a $\rightarrow C[\text{a}], C[\text{a}+1]-1 = 1, 6$
ca $\rightarrow C[\text{c}] + rank_{\text{c}}(BWT, 1-1), C[\text{c}] + rank_{\text{c}}(BWT, 6) - 1 = 7+0, 7+2-1 = 7, 8$
aca $\rightarrow C[\text{a}] + rank_{\text{a}}(BWT, 7-1), C[\text{a}] + rank_{\text{a}}(BWT, 8) - 1 = 1+2, 1+4-1 = 3, 4$

BWT

L

t

g

$

t

t

a

a

a

BWT

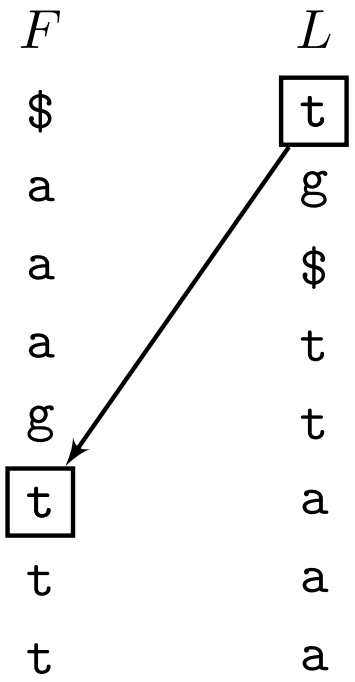
F	L
$	t
a	g
a	$
a	t
g	t
t	a
t	a
t	a

BWT

F	L
$	t
a	g
a	$
a	t
g	t
t	a
t	a
t	a

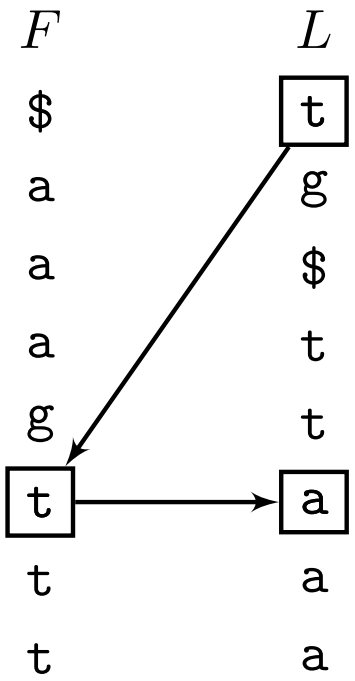
t

BWT



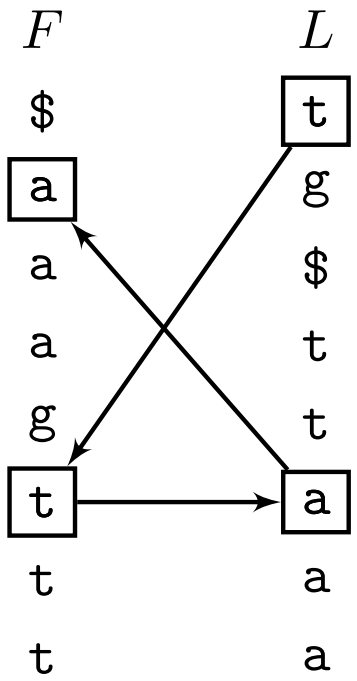
t

BWT



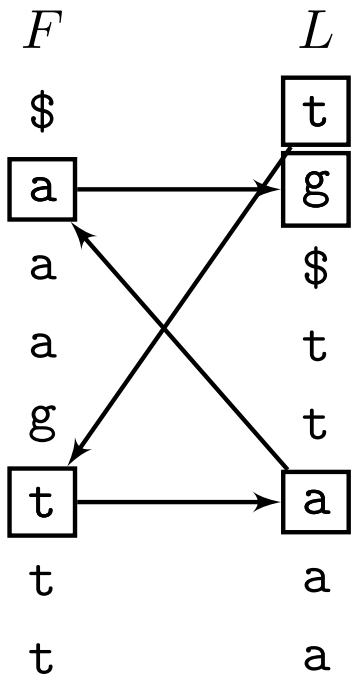
a t

BWT



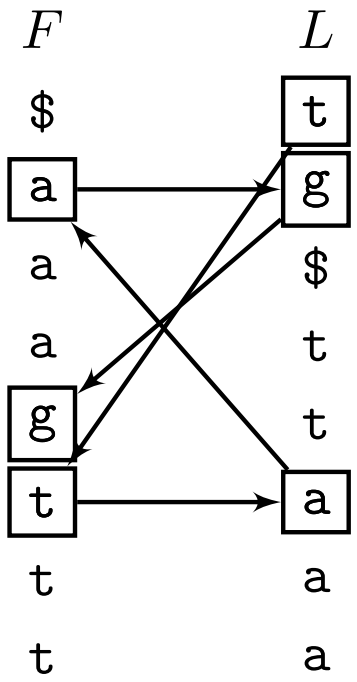
a t

BWT



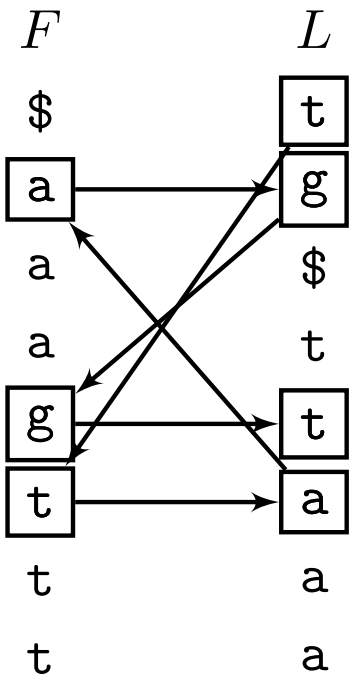
g a t

BWT



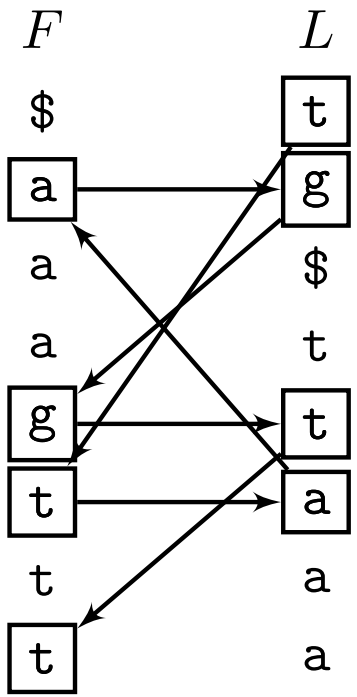
g a t

BWT



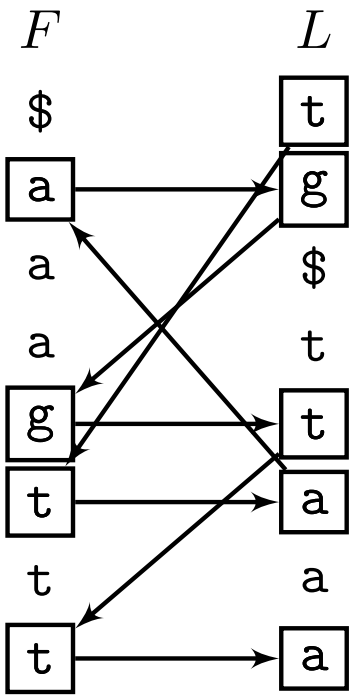
t g a t

BWT



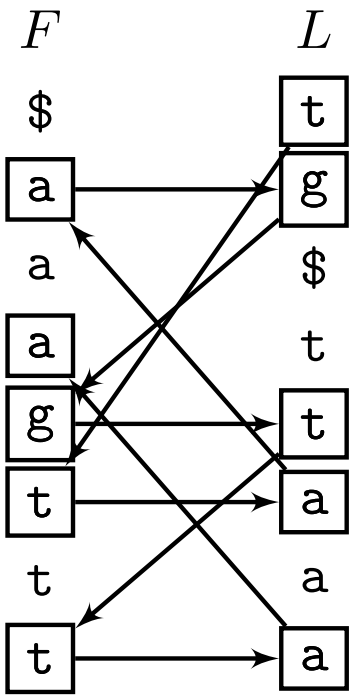
t g a t

BWT



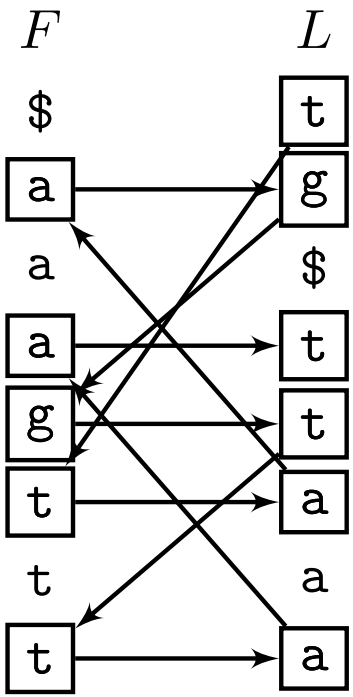
a t g a t

BWT



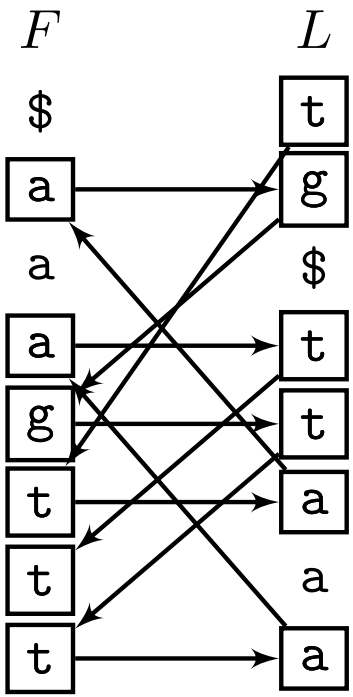
a t g a t

BWT



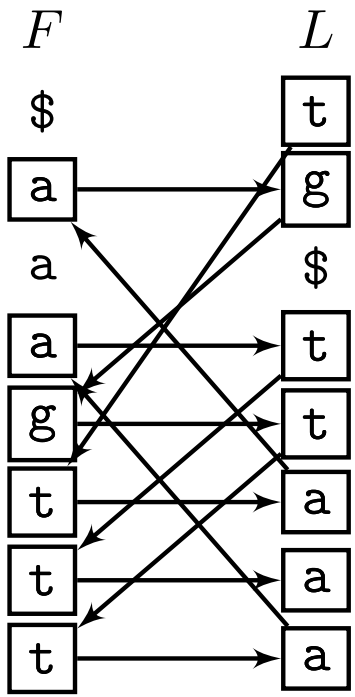
t a t g a t

BWT



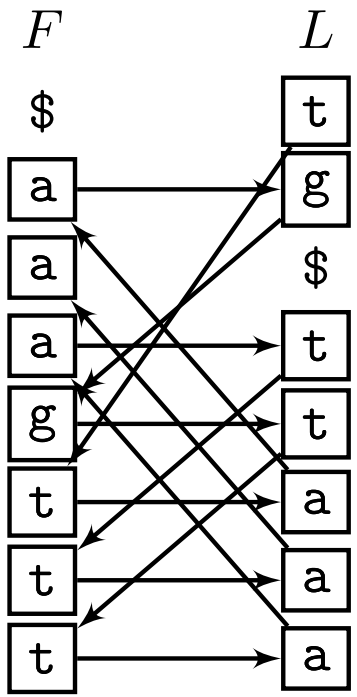
t a t g a t

BWT



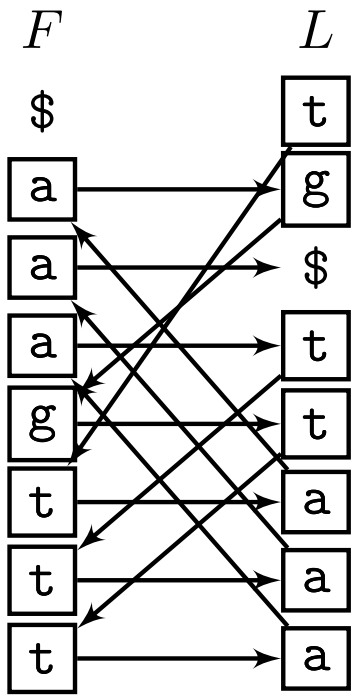
a t a t g a t

BWT



a t a t g a t

BWT



a t a t g a t

LF Function

atatgat$

	$	a	g	t
C	0	1	4	5

$$LF(i) = C[L[i]] + \mathit{rank}_{L[i]}(L, i) - 1$$

LF Function

atatgat$

	$	a	g	t
C	0	1	4	5

$$LF(i) = C[L[i]] + \mathit{rank}_{L[i]}(L, i) - 1$$

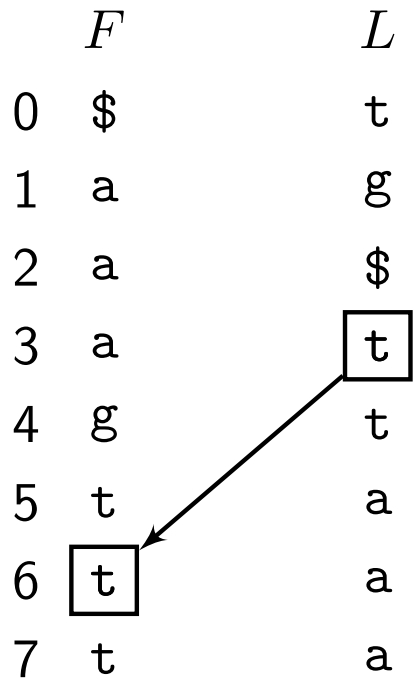
	F	L
0	$	t
1	a	g
2	a	$
3	a	t
4	g	t
5	t	a
6	t	a
7	t	a

LF Function

atatgat$

	$	a	g	t
C	0	1	4	5

$$LF(i) = C[L[i]] + \mathit{rank}_{L[i]}(L, i) - 1$$



$rank$ and $select$

$rank_c(y, i)$: number of symbols c in $y[0\,..\,i]$
and
$select_c(y, i)$: position of the i-th occurrence of symbol c in y

Representation of the BWT

For $c \in A$

$$B^c[i] = \begin{cases} 1 & \text{if } BWT[i] = c \\ 0 & \text{otherwise} \end{cases}$$

then

$rank_c(y, i) = rank_1(B^c, i)$

and

$select_c(y, i) = select_1(B^c, i)$

$rank_1$ and $select_1$ can be computed in constant time by using $n\sigma + o(n\sigma)$ bits