

Suffix sorting algorithms in parallel computation

Nurgissa Umatay

Background

Suffix Array Construction

- DC3 (Skew) algorithm
- Parallel DC3 (pDC3) algorithm

DC3 (Skew) - step 1

T = yabbadabbado\$
5 1 2 2 1 3 1 2 2 2 1 3 0

a=1, b=2, d=3,
o=4, y=5, \$=0

$$B_k = \{i \in [0, n] \mid i \bmod 3 = k\}.$$

B0 B1 B2

Index:	0	1	2	3	4	5	6	7	8	9	10	11	12
Value:	5	1	2	2	1	3	1	2	2	1	3	4	0

DC3 (Skew) cont. - step 1

Radix Sorting B12

Rank	Letters	Index
1	1 2 2	1
2	1 3 1	4
3	2 1 3	8
4	2 2 1	2
4	2 2 1	7
5	3 1 2	5
6	3 4 0	10
7	4 0 0	11

Evaluating of Radix Sorting B12

Indices: 0 1 2 3 4 5 6 7 8 9 10

B12 Index: 1, 4, 7, 10, 2, 5, 8, 11

Rank: 1, 2, 4, 6, 4, 5, 3, 7

New Array: 1 2 4 6 4 5 3 7 0 0 0

DC3 (Skew) cont. - step 1

First Level of Recursion

Indices: 0 1 2 3 4 5 6 7 8 9 10
StringToSort: 1 2 4 6 4 5 3 7 0 0 0

$B_{12} = \{1, 4, 7, 2, 5, 8\}$

Rank	Letters	Index
1	0 0 0	8
2	2 4 6	1
3	4 5 3	4
4	4 6 4	2
5	5 3 7	5
6	7 0 0	7

DC3 (Skew) cont. - step 2

Sorting Non-Sample triplets

Indices: 0 1 2 3 4 5 6 7 8 9 10
StringToSort: ① 2 4 ⑥ 4 5 ③ 7 0 0 0
Ranks: ? ② 4 ? ③ 5 ? ⑥ 1 0 0

$B0 = \{0, 3, 6\}$

Pairs to sort = { (1, 2), (6, 3), (3, 6) }

Pairs	Index
(1, 2)	0
(3, 6)	6
(6, 3)	3

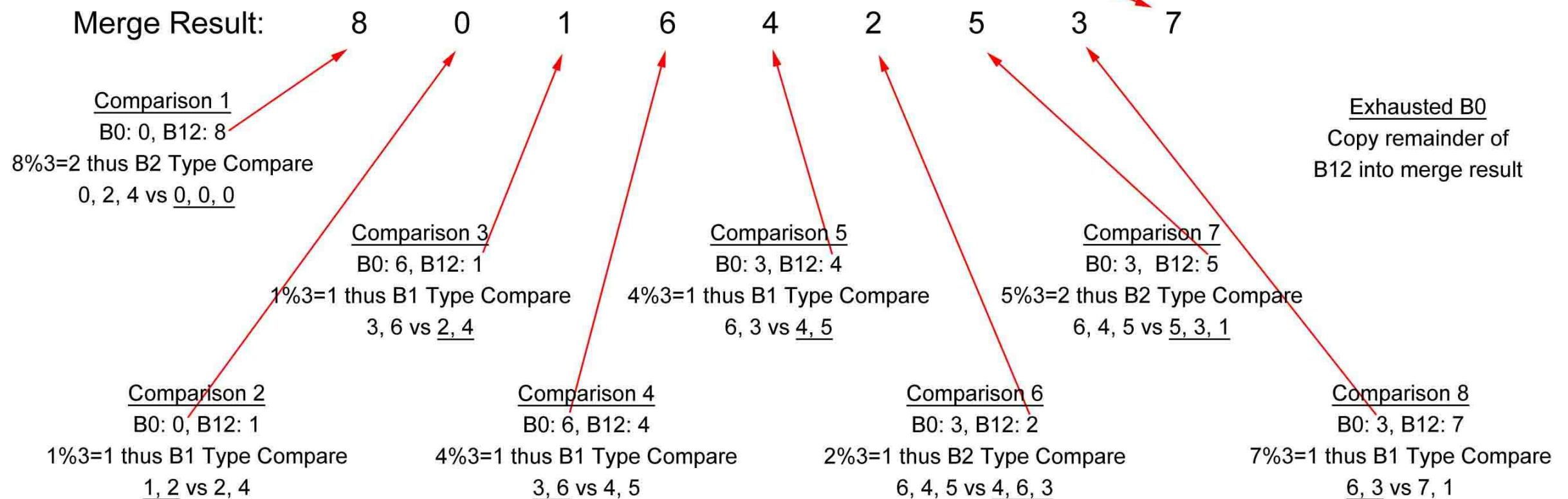
DC3 (Skew) cont. - step 3

Merging sample & non-sample triplets

Indices: 0 1 2 3 4 5 6 7 8 9 10
StringToSort: 1 2 4 6 4 5 3 7 0 0 0
Ranks: ? 2 4 ? 3 5 ? 6 1 0 0

SortedB0 = {0, 6, 3}

SortedB12 = {8, 1, 4, 2, 5, 7}



Preliminary results

Name	Size	Time	Memory	Level of recursion
Drosophila.dna.chromosome	~24 400000 ntd	NA	~30 Gb	NA
Drosophila.dna.chromosome	~12 200 000 ntd	665 sec	~18 Gb	9
Drosophila.dna.chromosome	~6100000 ntd	286 sec	~9 Gb	8



```
[numatay1@ugradx ~/genomics_project]$ python dc3_sa_builder.py < Drosophila
File loaded...
4000000 - 66
2666667 - 259026
1777778 - 1760768
1185186 - 1178933
790124 - 788195
526750 - 526326
351167 - 351113
234112 - 234112
286.795931101 seconds
8
[numatay1@ugradx ~/genomics_project]$ python dc3_sa_builder.py < Drosophila
File loaded...
7999960 - 66
5333307 - 261808
3555538 - 3518654
2370359 - 2355181
1580240 - 1575284
1053494 - 1052242
702330 - 702110
468220 - 468219
312147 - 312147
664.243397951 seconds
9
```

A	B	A - B	Level of Recursion
7999960	66	7999894	0
5333307	261808	5071499	1
3555538	3518654	36884	2
2370359	2355181	15178	3
1580240	1575284	4956	4
1053494	1052242	1252	5
702330	702110	220	6
468220	468219	1	7
312147	312147	0	8

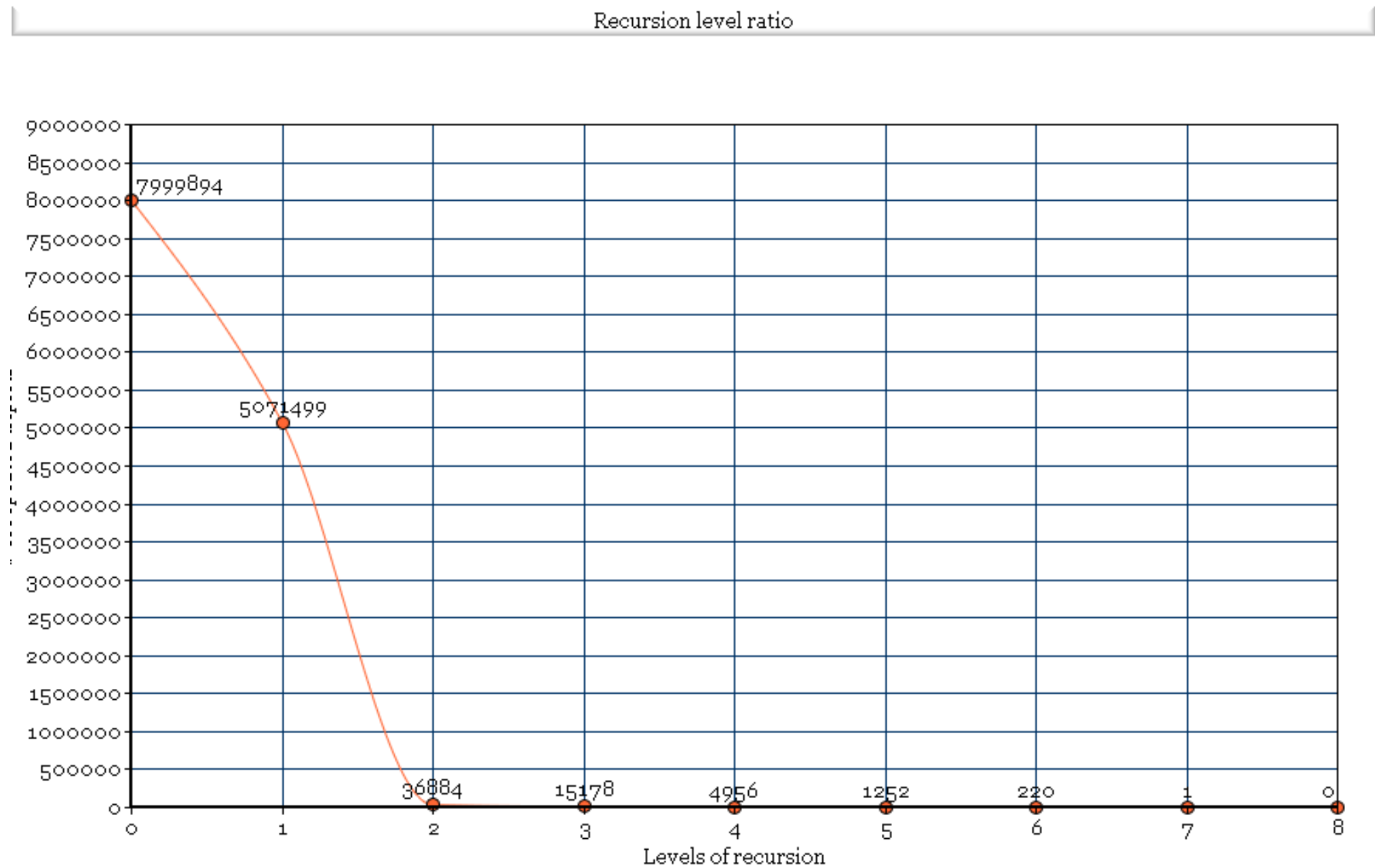
Ranking optimization

B0 B1 B2

Index:	0	1	2	3	4	5	6	7	8	9	10	11	12
Value:	5	1	2	2	1	3	1	2	2	1	3	4	0



Ranking optimization



Getting things done

- Finish implementation of pDC3 (currently works slower than DC3)
- Test ranking optimization on pDC3