Lecture 19: Genome Rearrangement

BCB 5300 Algorithms in Computational Biology

Fall 2019

Tae-Hyuk (Ted) Ahn

Department of Computer Science
Program of Bioinformatics and Computational Biology
Saint Louis University

Outline

- Learn about Genome Rearrangement

Greedy Algorithm

<u>SimpleReversalSort(π)</u>

```
for i ← 1 to n – 1
   j ← position of element i in π (i.e., π_j = i)
   if j ≠ i
      π ← π ρ(i, j)
      output π
   if π is the identity permutation
     return
```

Analyzing SimpleReversalSort

- SimpleReversalSort does not guarantee the smallest number of reversals and takes five steps on $\pi = \underline{6\ 1}\ 2\ 3\ 4\ 5$:

  - Step 1: 1 $\underline{6\ 2}$ 3 4 5
  - Step 2: 1 2 $\underline{6\ 3}$ 4 5
  - Step 3: 1 2 3 $\underline{6\ 4}$ 5
  - Step 4: 1 2 3 4 $\underline{6\ 5}$
  - Step 5: 1 2 3 4 5 6

Analyzing SimpleReversalSort

- But it can be sorted in two steps:

  π = 6 1 2 3 4 5

  - Step 1: 5 4 3 2 1 6
  - Step 2: 1 2 3 4 5 6

- So, SimpleReversalSort(π) is not optimal

- Optimal algorithms are unknown for many problems; approximation algorithms are used

Adjacencies

$\pi = \pi_1\pi_2\pi_3\ldots\pi_{n-1}\pi_n$

- A pair of elements π_i and π_{i+1} are adjacent if

$$\pi_{i+1} = \pi_i \pm 1$$

- For example:

$$\pi = 1\ 9\ \underline{3\ 4}\ \underline{7\ 8}\ 2\ \underline{6\ 5}$$

- (3, 4) or (7, 8) and (6,5) are adjacent pairs

Breakpoints

There is a breakpoint between any adjacent elements that are non-consecutive:

$$\pi = 1 \,|\, 9 \,|\, 3 \;\; 4 \,|\, 7 \;\; 8 \,|\, 2 \,|\, 6 \;\; 5$$

- Pairs (1,9), (9,3), (4,7), (8,2) and (2,5) form breakpoints of permutation π
- $b(\pi)$ - # breakpoints in permutation π

Adjacencies & Breakpoints

- An adjacency - a pair of adjacent elements that are consecutive
- A breakpoint - a pair of adjacent elements that are not consecutive

$\pi = 5\ \ 6\ \ 2\ \ 1\ \ 3\ \ 4 \longrightarrow$ Extend π with $\pi_0 = 0$ and $\pi_7 = 7$

adjacencies

0 5 6 2 1 3 4 7

breakpoints

Extending Permutations

- We put two elements $\pi_0 = 0$ and $\pi_{n+1} = n+1$ at the ends of π

$$\pi = 1 \mid 9 \mid 3 \;\; 4 \mid 7 \;\; 8 \mid 2 \mid 6 \;\; 5$$

Extending with *0* and *10*

$$\pi = 0 \;\; 1 \mid 9 \mid 3 \;\; 4 \mid 7 \;\; 8 \mid 2 \mid 6 \;\; 5 \mid 10$$

A new breakpoint was created after extending

A permutation of n may have up to $(n+1)$ breakpoints

Reversal Distance and Breakpoints

- Breakpoints are the *bottlenecks* for sorting by reversals.
- Each reversal eliminates at most 2 breakpoints.

$\pi = 2\ \ 3\ \ 1\ \ 4\ \ 6\ \ 5$

$0 \mid \underline{2\ \ 3 \mid 1} \mid 4 \mid 6\ \ 5 \mid 7$ $\qquad b(\pi) = 5$

$0\ \ 1 \mid \underline{3\ \ 2} \mid 4 \mid 6\ \ 5 \mid 7$ $\qquad b(\pi) = 4$

$0\ \ 1\ \ 2\ \ 3\ \ 4 \mid \underline{6\ \ 5} \mid 7$ $\qquad b(\pi) = 2$

$0\ \ 1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6\ \ 7$ $\qquad b(\pi) = 0$

$$d(\pi) \geq \frac{b(\pi)}{2}$$

Sorting By Reversals: A Better Greedy Algorithm

SimpleReversalSort(π)

for $i \leftarrow 1$ to $n-1$
    $j \leftarrow$ position of element i in π (i.e., $\pi_j = i$)
    if $j \neq i$
        $\pi \leftarrow \pi \cdot \rho(i, j)$
        output π
    if π is the identity permutation
        return

BreakPointReversalSort(π)

while $b(\pi) > 0$
    Among all possible reversals, choose reversal ρ minimizing $b(\pi \cdot \rho)$
    $\pi \leftarrow \pi \cdot \rho(i, j)$
    output π
return

Does it always terminate?

How can we be sure that removing some breakpoints does not introduce others?

Strips

- Strip: an interval between two consecutive breakpoints in a permutation
  - Decreasing strip: *strip* of elements in decreasing order (e.g. 6 5 and 3 2).
  - Increasing strip: *strip* of elements in increasing order (e.g. 7 8)

0 1 9 4 3 7 8 2 5 6 *10*

  - A single-element strip can be declared either increasing or decreasing. We will choose to declare them as decreasing with exception of the strips with *0* and *n+1*

Reducing the Number of Breakpoints

Consider $\pi = 1\ 4\ 6\ 5\ 7\ 8\ 3\ 2$

$0\ \ 1 \mid 4 \mid 6\ \ 5 \mid 7\ \ 8 \mid 3\ \ 2 \mid 9$ $\qquad b(\pi) = 5$

If permutation π contains at least one decreasing strip, then there exists a reversal ρ which decreases the number of breakpoints (i.e. $b(\pi \bullet \rho) < b(\pi)$).

Things to Consider

Consider π = 1 4 6 5 7 8 3 2

0 1 | 4 | 6 5 | 7 8 | 3 2 | 9 $\quad b(\pi) = 5$

reduced by 1!

0 1 2 3 | 8 7 | 5 6 | 4 | 9 $\quad b(\pi) = 4$

- Choose the decreasing strip with the smallest element *k* in π
- Find *k – 1* in the permutation
- Reverse the segment between *k* and *k-1*
- Repeat until there is no decreasing strip

Things to Consider

Consider π = 1 4 6 5 7 8 3 2

0 1 2 3 | 8 7 | 5 6 | 4 | 9 $\qquad b(\pi) = 4$

0 1 2 3 4 | 6 5 | 7 8 9 $\qquad b(\pi) = 2$

- Choose the decreasing strip with the smallest element *k* in π
- Find *k – 1* in the permutation
- Reverse the segment between *k* and *k-1*
- Repeat until there is no decreasing strip

Things to Consider

Consider π = 1 4 6 5 7 8 3 2

0 1 2 3 4 5 6 7 8 9 $\qquad b(\pi) = 0$

- Choose the decreasing strip with the smallest element k in π
- Find $k - 1$ in the permutation
- Reverse the segment between k and k-1
- Repeat until there is no decreasing strip

Things to Consider

Consider p = 1 4 6 5 7 8 3 2

$0\ 1 \mid 4 \mid 6\ 5 \mid 7\ 8 \mid 3\ 2 \mid 9$ $\quad b(\pi) = 5$

$0\ 1\ 2\ 3 \mid 8\ 7 \mid 5\ 6 \mid 4 \mid 9$ $\quad b(\pi) = 4$

$0\ 1\ 2\ 3\ 4 \mid 6\ 5 \mid 7\ 8\ 9$ $\quad b(\pi) = 2$

$0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9$ $\quad b(\pi) = 0$

$d(\pi) = 3$

Q: Does it work for arbitrary permutation?

What if there is no Decreasing Strip?

$0\ \ 1\ \ 2 \mid 5\ \ 6\ \ 7 \mid 3\ \ 4 \mid 8\ \ 9$ $\qquad b(\pi) = 3$

- If there is no decreasing strip, there may be no reversal *r* that reduces the number of breakpoints (i.e. $b(p \bullet r) \geq b(p)$ for any reversal r).
- By reversing an <u>increasing</u> strip (# of breakpoints remains unchanged), we will create a decreasing strip. Then the number of breakpoints will be reduced in the following step.

What if there is no Decreasing Strip?

$$0 \;\; 1 \;\; 2 \,\big|\, 7 \;\; 6 \;\; \overset{k}{\downarrow}{5} \,\big|\, 3 \;\; \overset{k-1}{\downarrow}{4} \,\big|\, 8 \;\; 9 \qquad b(\pi) = 3$$

- If there is no decreasing strip, there may be no reversal *r* that reduces the number of breakpoints (i.e. $b(p \bullet r) \geq b(p)$ for any reversal *r*).
- By reversing an <u>increasing</u> strip (# of breakpoints remains unchanged), we will create a decreasing strip. Then the number of breakpoints will be reduced in the following step.

What if there is no Decreasing Strip?

$$\begin{array}{c} \quad\quad\quad\quad\quad\quad k \;\; k\text{-}1 \\ \quad\quad\quad\quad\quad\quad \downarrow \;\;\; \downarrow \\ \underrightarrow{0\ 1\ 2}\,|\,\underleftarrow{7\ 6\ 5\ 4\ 3}\,|\,\underrightarrow{8\ 9} \end{array} \qquad b(\pi) = 2$$

- If there is no decreasing strip, there may be no reversal *r* that reduces the number of breakpoints (i.e. $b(p \bullet r) \geq b(p)$ for any reversal r).
- By reversing an <u>increasing</u> strip (# of breakpoints remains unchanged), we will create a decreasing strip. Then the number of breakpoints will be reduced in the following step.

What if there is no Decreasing Strip?

k-1 $\quad\quad\quad\quad\quad$ k

$0 \;\; 1 \;\; 2 \mid 7 \;\; 6 \;\; 5 \;\; 4 \;\; 3 \mid 8 \;\; 9$ $\qquad b(\pi) = 2$

- If there is no decreasing strip, there may be no reversal r that reduces the number of breakpoints (i.e. $b(p \bullet r) \geq b(p)$ for any reversal r).
- By reversing an <u>increasing</u> strip (# of breakpoints remains unchanged), we will create a decreasing strip. Then the number of breakpoints will be reduced in the following step.

What if there is no Decreasing Strip?

$k\text{-}1$ k

$\underrightarrow{0 \;\; 1 \;\; 2 \;\; 3 \;\; 4 \;\; 5 \;\; 6 \;\; 7 \;\; 8 \;\; 9}$ $\qquad b(\pi) = 0$

- If there is no decreasing strip, there may be no reversal r that reduces the number of breakpoints (i.e. $b(p \bullet r) \geq b(p)$ for any reversal r).
- By reversing an increasing strip (# of breakpoints remains unchanged), we will create a decreasing strip. Then the number of breakpoints will be reduced in the following step.

ImprovedBreakpointReversalSort

ImprovedBreakpointReversalSort(π)

while $b(\pi) > 0$
  if π has a decreasing strip
    Among all possible reversals, choose reversal ρ that minimizes $b(\pi \cdot \rho)$
  else
    Choose a reversal ρ that flips an increasing strip in π
  $\pi \leftarrow \pi \cdot \rho$
  output π
return

BreakPointReversalSort(π)

while $b(\pi) > 0$
  Among all possible reversals, choose reversal ρ minimizing $b(\pi \cdot \rho)$
  $\pi \leftarrow \pi \cdot \rho(i, j)$
  output π
return