

BWA

Burrows-Wheeler Aligner

Marcel Bargull

Fakultät für Informatik, Technische Universität Dortmund

7. Mai 2014

- 1 BWA-backtrack
- 2 BWT / SA-Intervall / FM-Index
- 3 BWA Algorithmus
- 4 BWA Implementierung

- BWT / FM-Index auf Referenzgenom
- Ausgelegt auf ≤ 100 bp Illumina Reads
- Unterstützt Mismatches und Gaps
- Backward Search
- Bounded Traversal
 - Untere Schranke für Abweichungen im Readprefix
- Nachträgliches Paired-end Mapping mit Smith-Waterman

- Toleriert bis zu k Abweichungen in Reads
 - $< 4\%$ der Reads mehr als k Unterschiede bei 2% Basenfehlerrate (gleichverteilt)

15 – 37 bp : $k = 2$

38 – 63 bp : $k = 3$

64 – 92 bp : $k = 4$

93 – 123 bp : $k = 5$

124 – 156 bp : $k = 6$

- Optional: Abweichung in Saat (= erste n Basen) beschränken
Bsp.: 70 bp Reads, 32 bp Saat, max. 2 Abweichungen



Pos

$X = \text{googol\$}$

i $S(i)$ $B[i]$
 \downarrow \downarrow
 $(6, 3, 0, 5, 2, 4, 1)$ $lo\$oogg$

- SA-Intervall: $\left[\underline{R}(W), \overline{R}(W)\right]$
- $\underline{R}(W) = \min \left\{ k : W \text{ is the prefix of } X_{S(k)} \right\}$
- $\overline{R}(W) = \max \left\{ k : W \text{ is the prefix of } X_{S(k)} \right\}$
- $C(a)$: Anzahl Symbole in X , die kleiner als a sind
- $O(a, i)$: Anzahl Vorkommen von a in $B[0, i]$
- $\underline{R}(aW) = C(a) + O(a, \underline{R}(W) - 1) + 1$
- $\overline{R}(aW) = C(a) + O(a, \overline{R}(W))$

■ INEXACTSEARCH

■ Eingaben:

- W : Read
- z : Maximale Editierdistanz

■ Ausgabe:

- I : Menge von SA-Intervallen

■ CALCULATED

■ Eingaben:

- W : Read

■ Ausgabe:

- D : Untere Schranken für Abweichungen der Präfixe $W[0, i]$ von Referenz

precalculate

BWT B for reference string X
Array $C(\cdot)$ and $O(\cdot, \cdot)$ from B
BWT B' for the reverse reference
Array $O'(\cdot, \cdot)$ from B'

procedure CALCULATED(W)

```
 $k \leftarrow 1$   
 $l \leftarrow |X| - 1$   
 $z \leftarrow 0$   
for  $i = 0$  to  $|W| - 1$  do  
   $k \leftarrow C(W[i]) + O'(W[i], k - 1) + 1$   
   $k \leftarrow C(W[i]) + O'(W[i], l)$   
  if  $k > l$  then  
     $k \leftarrow 1$   
     $l \leftarrow |X| - 1$   
     $z \leftarrow z + 1$   
 $D(i) \leftarrow z$ 
```

procedure INEXACTSEARCH(W, z)

```
CALCULATED( $W$ )  
return INEXRECUR( $W, |W| - 1, z, 1, |X| - 1$ )
```

procedure CALCULATED(W)

```
 $z \leftarrow 0$   
 $j \leftarrow 0$   
for  $i = 0$  to  $|W| - 1$  do  
  if  $W[i, j]$  not a substring of  $X$  then  
     $z \leftarrow z + 1$   
     $j \leftarrow i + 1$   
 $D(i) \leftarrow z$ 
```



```
procedure INEXRECUR( $W, i, z, k, l$ )  
  if  $z < D(i)$  then  
    return  $\emptyset$   
  if  $i < 0$  then  
    return  $\{[k, l]\}$   
   $I \leftarrow \emptyset$   
   $I \leftarrow I \cup \text{INEXRECUR}(W, i - 1, z - 1, k, l)$            // Insertion  
  for each  $b \in \{A, C, G, T\}$  do  
     $k \leftarrow C(b) + O(b, k - 1) + 1$   
     $l \leftarrow C(b) + O(b, l)$   
    if  $k \leq l$  then  
       $I \leftarrow I \cup \text{INEXRECUR}(W, i, z - 1, k, l)$            // Deletion  
      if  $b = W[i]$  then  
         $I \leftarrow I \cup \text{INEXRECUR}(W, i - 1, z, k, l)$   
      else  
         $I \leftarrow I \cup \text{INEXRECUR}(W, i - 1, z - 1, k, l)$   
  return  $I$ 
```

- BFS mit heapähnlicher Datenstruktur statt DFS
 - Priorisiert nach Alignment Score der partiellen Zuordnungen
- Unterschiedliche Bewertung für
 - Mismatch
 - Gap Open
 - Gap Extension
- Reverses Komplement wird zeitgleich abgearbeitet

- $O(\cdot, k)$ nur für jede 128te (k-te) Stelle gespeichert
- Stellen dazwischen werden “one the fly” berechnet
- $S(k)$ wird für jede 32. Stelle gespeichert
- Gesamtpeicherbedarf bei 3 Gb Genom:
 - 2,3 GB für Single End
 - ca. 3 GB bei Paired End