Assembly of long, error-pront reads using repeat graphs

Mikhail Kolmogorov, Jeffrey Yuan, Yu Lin, and Pavel A. Pevzner

05.07.2021

Genome assembly in general

- reconstruct target sequence from the reads

Genome assembly in general

- reconstruct target sequence from the reads
- different graph structures (De-Bruijn, Overlap-layout, String)

Genome assembly in general

- reconstruct target sequence from the reads
- different graph structures (De-Bruijn, Overlap-layout, String)
- repeats $\rightarrow$ assembly fragmentation

Genome assembly in general

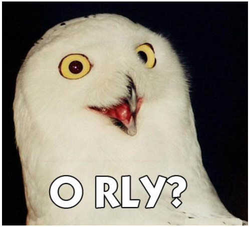
- reconstruct target sequence from the reads
- different graph structures (De-Bruijn, Overlap-layout, String)
- repeats → assembly fragmentation
- error rate long read <-> short read

Genome assembly in general

- reconstruct target sequence from the reads
- different graph structures (De-Bruijn, Overlap-layout, String)
- repeats $\rightarrow$ assembly fragmentation
- error rate long read <-> short read
- Flye should resolve these repeats correctly

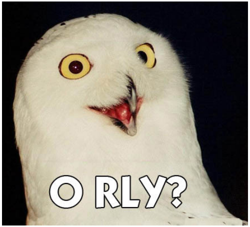
Disjointigs

- most assemblers spent much time on correct contig assembly



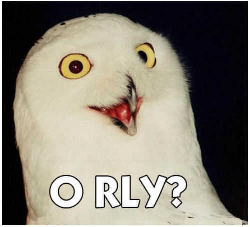
Disjointigs

- most assemblers spent much time on correct contig assembly
- Flye uses a different approach:



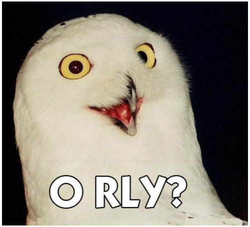
Disjointigs

- most assemblers spent much time on correct contig assembly
- Flye uses a different approach:
- we don’t care (at least at the initial stage)



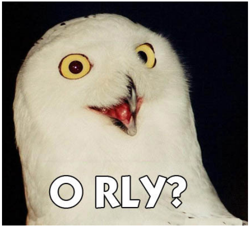
Disjointigs

- most assemblers spent much time on correct contig assembly
- Flye uses a different approach:
- we don’t care (at least at the initial stage)
- correct assembly graph



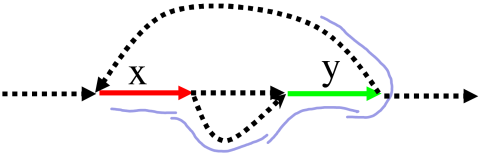
Disjointigs

- most assemblers spent much time on correct contig assembly
- Flye uses a different approach:
- we don't care (at least at the initial stage)
- correct assembly graph
- generate paths from overlapping reads without checking for correct assembly -> disjointigs



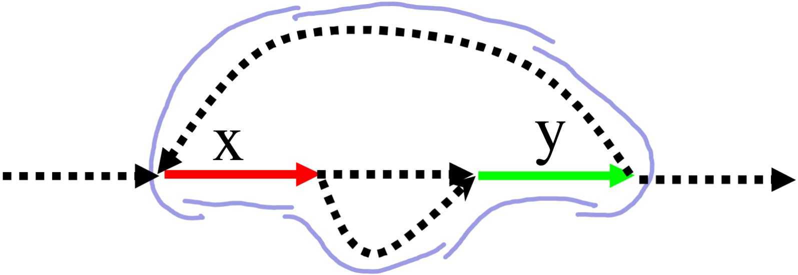
Disjointigs

- current assemblers use much time on correct contig assembly
- Flye uses a different approach:
- generate paths from overlapping reads without checking for correct assembly -> disjointigs

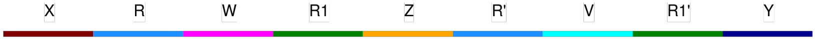


Disjointigs

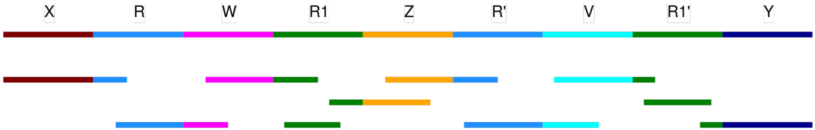
- current assemblers use much time on correct contig assembly
- Flye uses a different approach:
- generate paths from overlapping reads without checking for correct assembly -> disjointigs



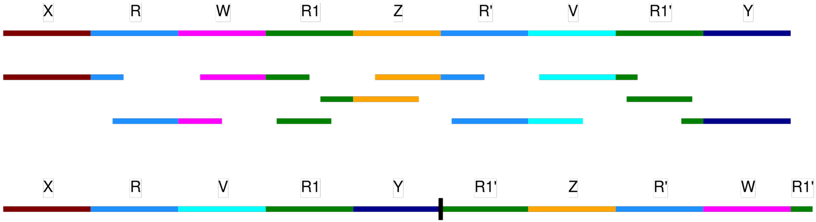
Repeat graph creation



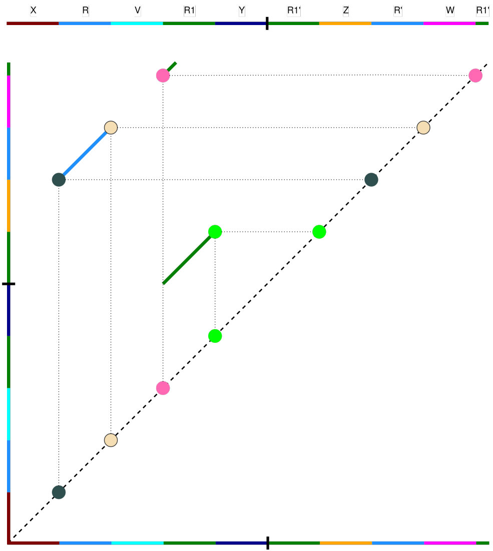
Repeat graph creation



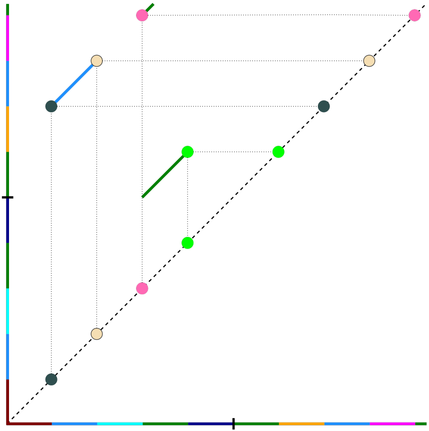
Repeat graph creation

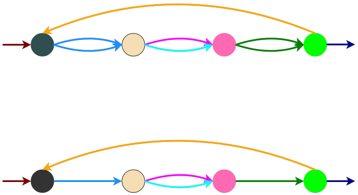


Repeat graph creation

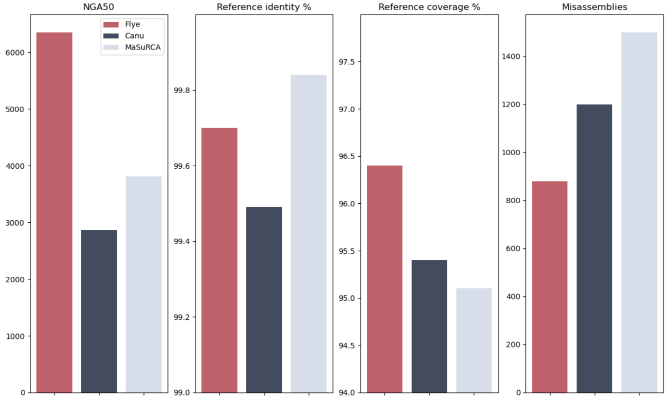


Repeat graph creation





Results

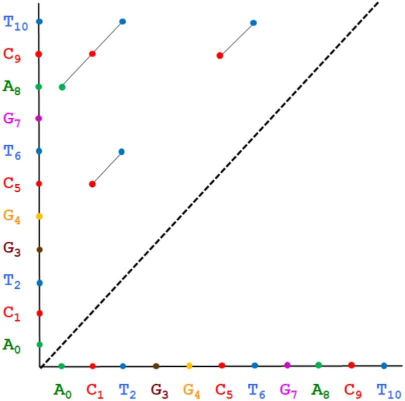

NGA50
Flye
Canu
MaSuRCA
0
1000
2000
3000
4000
5000
6000
Reference identity %
99.0
99.2
99.4
99.6
99.8
Reference coverage %
94.0
94.5
95.0
95.5
96.0
96.5
97.0
97.5
Misassemblies
0
200
400
600
800
1000
1200
1400

References

Appendix

Dot plot creation

-	9	10	-	-	9	10	-	-	5	6
8	5	6	-	-	1	2	-	0	1	2
0	1	2	3	4	5	6	7	8	9	10
A	C	T	G	G	C	T	G	A	C	T


T_{10}
C_9
A_8
G_7
T_6
C_5
G_4
G_3
T_2
C_1
A_0
A_0 C_1 T_2 G_3 G_4 C_5 T_6 G_7 A_8 C_9 T_{10}

Repeat graphs

- generalization of de bruijn graphs

```
>- from disjointigs = random walk of reads on the repeat gr
>- means the repeat graph hasn't to be known
```

Repeat graphs

- generalization of de bruijn graphs
- structure

```
>- from disjointigs = random walk of reads on the repeat gr
>- means the repeat graph hasn't to be known
```

Repeat graphs

- generalization of de bruijn graphs
- structure
- creation

```
>- from disjointigs = random walk of reads on the repeat gr
>- means the repeat graph hasn't to be known
```

Difference repeat graph de Bruijn graph

- A-Bruijn graph (alignments) generalizes the de Bruijn graph

Difference repeat graph de Bruijn graph

- A-Bruijn graph (alignments) generalizes the de Bruijn graph
- We thus argue that the time has come to explain that the breakpoint graphs and the de Bruijn graphs are two identical data structures (if one ignores a cosmetic difference between them) as they both represent specific instances of a general notion of the A-Bruijn graph introduced in [13]. The A-Bruijn graphs are based on representing genomes as sets of labeled paths and further gluing identically labeled edges (breakpoint graphs) or vertices (de Bruijn graphs) in the resulting paths.

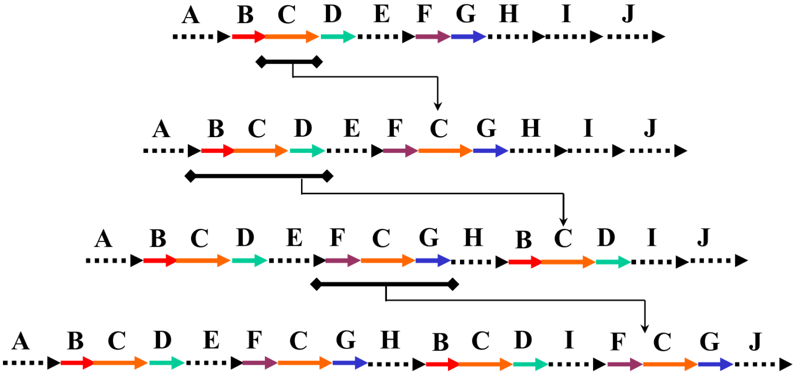
Difference repeat graph de Bruijn graph

- A-Bruijn graph (alignments) generalizes the de Bruijn graph
- We thus argue that the time has come to explain that the breakpoint graphs and the de Bruijn graphs are two identical data structures (if one ignores a cosmetic difference between them) as they both represent specific instances of a general notion of the A-Bruijn graph introduced in [13]. The A-Bruijn graphs are based on representing genomes as sets of labeled paths and further gluing identically labeled edges (breakpoint graphs) or vertices (de Bruijn graphs) in the resulting paths.
- de Bruijn graphs need correct bases

Difference repeat graph de Bruijn graph

- A-Bruijn graph (alignments) generalizes the de Bruijn graph
- We thus argue that the time has come to explain that the breakpoint graphs and the de Bruijn graphs are two identical data structures (if one ignores a cosmetic difference between them) as they both represent specific instances of a general notion of the A-Bruijn graph introduced in [13]. The A-Bruijn graphs are based on representing genomes as sets of labeled paths and further gluing identically labeled edges (breakpoint graphs) or vertices (de Bruijn graphs) in the resulting paths.
- de Bruijn graphs need correct bases
- otherwise tangled graph

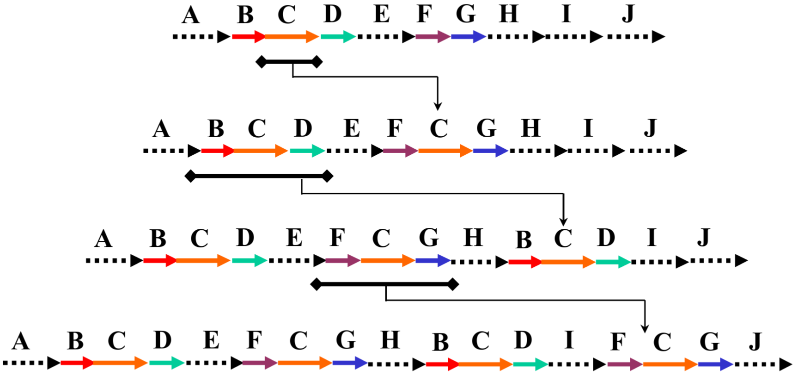
Segmental duplications



A B C D E F G H I J
A B C D E F C G H I J
A B C D E F C G H B C D I J
A B C D E F C G H B C D I F C G J

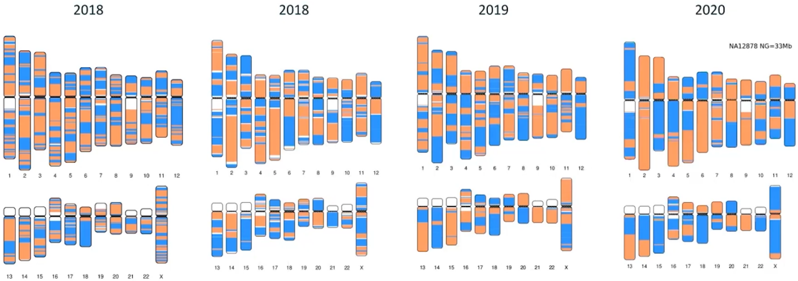
- Segmental duplications are duplicated blocks of genomic DNA typically ranging in size from 1-200 kb (IHGSC 2001)

Segmental duplications


A B C D E F G H I J
A B C D E F C G H I J
A B C D E F C G H B C D I J
A B C D E F C G H B C D I F C G J

- Segmental duplications are duplicated blocks of genomic DNA typically ranging in size from 1-200 kb (IHGSC 2001)
- They often contain sequence features such as high-copy repeats and gene sequences with intron-exon structure

Contigity improvement



Canu 1.7
NG50=10 Mb

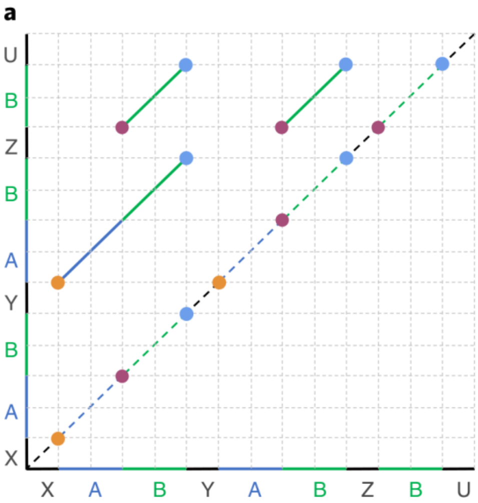
Flye 2.3.7
NG50=16 Mb

Flye 2.5
NG50=21 Mb

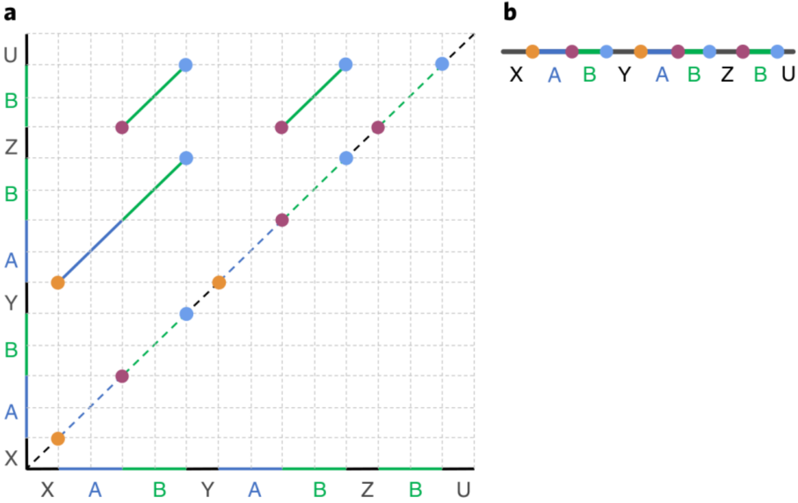
Flye 2.7
NG50=33 Mb

- colors are contigs
- color changes -> fragmented

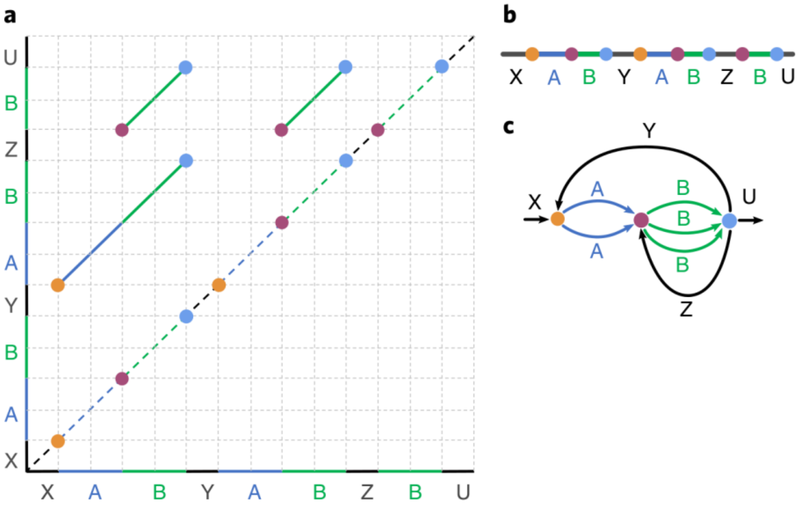
Repeat graph creation



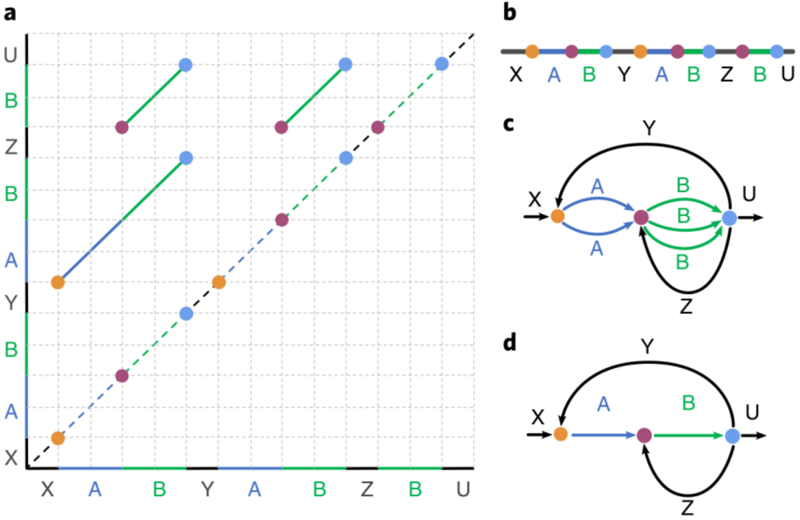
Repeat graph creation



Repeat graph creation



Repeat graph creation



Repeat resolution