

# Assembly of long, error-prone reads using repeat graphs

---

Mikhail Kolmogorov, Jeffrey Yuan, Yu Lin and Pavel A. Pevzner  
July 5, 2021

Presented by: Johannes Hausmann, Luis Kress

# Background

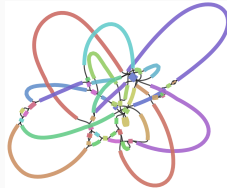
- Assembly: reconstruct target sequence from the reads

# Background

- Assembly: reconstruct target sequence from the reads
- Different assemblers, different graph structures

# Background

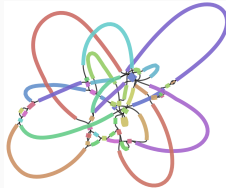
- Assembly: reconstruct target sequence from the reads
- Different assemblers, different graph structures
- Repeats → assembly fragmentation



Tangled assembly graph of *E.coli* [2]

# Background

- Assembly: reconstruct target sequence from the reads
- Different assemblers, different graph structures
- Repeats → assembly fragmentation



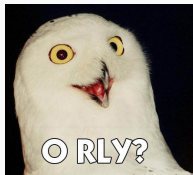
Tangled assembly graph of *E.coli* [2]

- Small differences between repeat copies → hard to resolve with error-prone reads

- Most assemblers spend much time on correct contig assembly

- Most assemblers spend much time on correct contig assembly
- Flye uses a different approach:

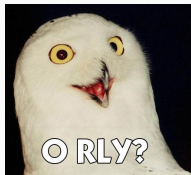
- Most assemblers spend much time on correct contig assembly
- Flye uses a different approach:
  - we don't care about the correct contig assembly





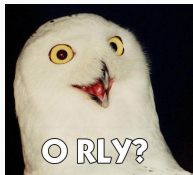
# Disjointigs

- Most assemblers spend much time on correct contig assembly
- Flye uses a different approach:
  - we don't care about the correct contig assembly
    - at least at the initial stage



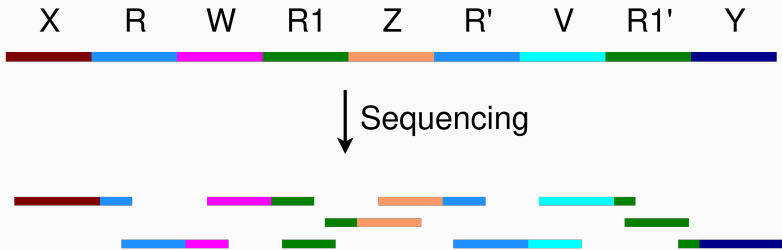
# Disjointigs

- Most assemblers spend much time on correct contig assembly
- Flye uses a different approach:
  - we don't care about the correct contig assembly  
→ at least at the initial stage

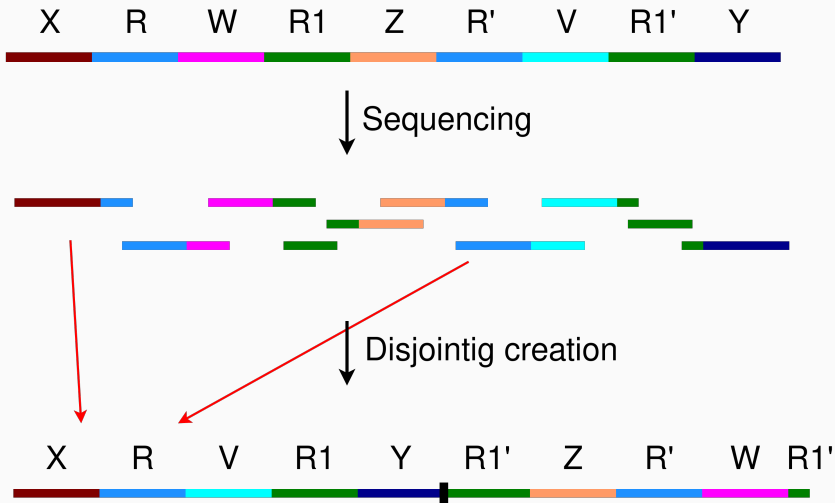


- Generate arbitrary paths from overlapping reads → Disjointigs

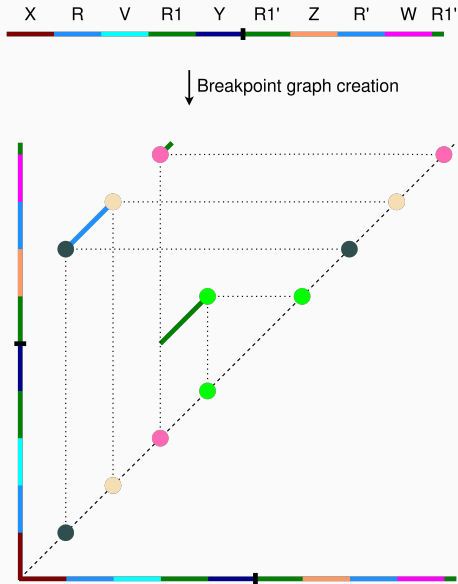
# Repeat Graph Creation



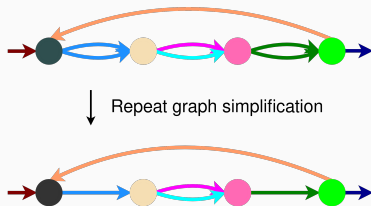
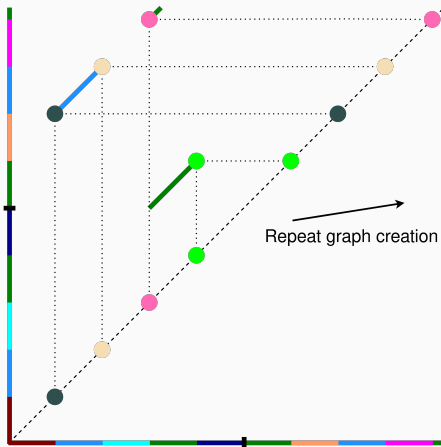
# Repeat Graph Creation



# Repeat Graph Creation



# Repeat Graph Creation



# Results

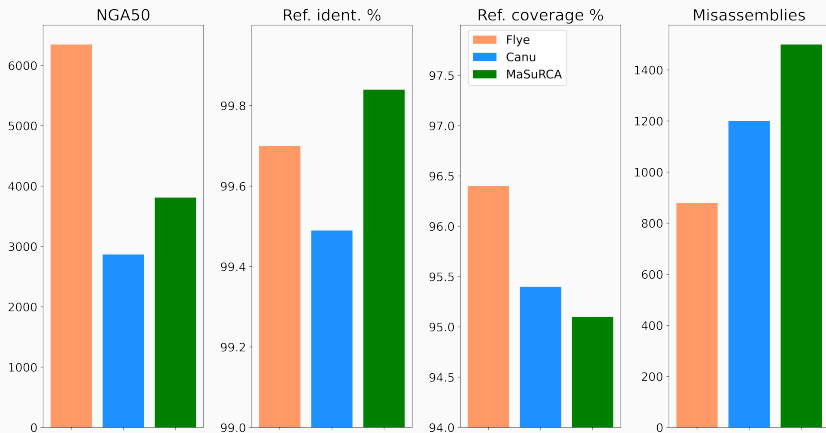


Figure 1: Results for HUMAN testset



<https://github.com/LKress/ASA>





A. Gurevich, V. Saveliev, N. Vyahhi, and G. Tesler.

**QUAST: quality assessment tool for genome assemblies.**

*Bioinformatics (Oxford, England)*, 29(8):1072–1075, Apr. 2013.



M. Kolmogorov, J. Yuan, Y. Lin, and P. A. Pevzner.

**Assembly of long, error-prone reads using repeat graphs.**

*Nature Biotechnology*, 37(5):540–546, May 2019.



Y. Lin, S. Nurk, and P. A. Pevzner.

**What is the difference between the breakpoint graph and the de Bruijn graph?**

*BMC genomics*, 15 Suppl 6:S6, 2014.



P. A. Pevzner, P. A. Pevzner, H. Tang, and G. Tesler.

**De novo repeat classification and fragment assembly.**

*Genome Research*, 14(9):1786–1796, Sept. 2004.



B. J. Walker, T. Abeel, T. Shea, M. Priest, A. Abouelliel, S. Sakthikumar, C. A. Cuomo, Q. Zeng, J. Wortman, S. K. Young, and A. M. Earl.

**Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement.**

*PloS One*, 9(11):e112963, 2014.



A. V. Zimin, G. Marçais, D. Puiu, M. Roberts, S. L. Salzberg, and J. A. Yorke.

**The MaSuRCA genome assembler.**

*Bioinformatics*, 29(21):2669–2677, Nov. 2013.

# Assembly of long, error-prone reads using repeat graphs

---

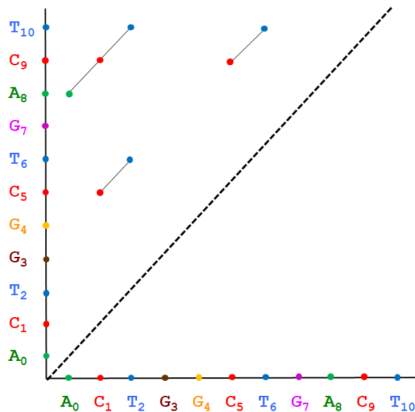
Mikhail Kolmogorov, Jeffrey Yuan, Yu Lin and Pavel A. Pevzner  
July 5, 2021

Presented by: Johannes Hausmann, Luis Kress

# 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



Dot plot creation [2]

# Repeat graphs

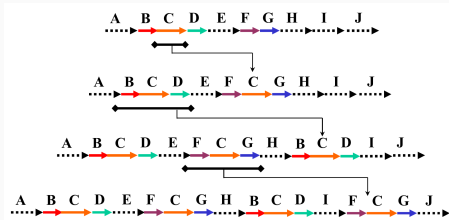
- generalization of de Bruijn graphs
- structure
- creation
  - from disjointigs = random walk of reads on the repeat graph
  - means the repeat graph hasn't to be known
  - set of disjointigs of a genome is complete if each  $k+1$ -mer from the genome is present in a disjointig from this set  $\rightarrow$  repeat graph construction of complete set of disjointigs same result as repeat graph construction of the genome

## 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 [3]. 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."  
[4]
- de Bruijn graphs need correct bases
- otherwise tangled graph



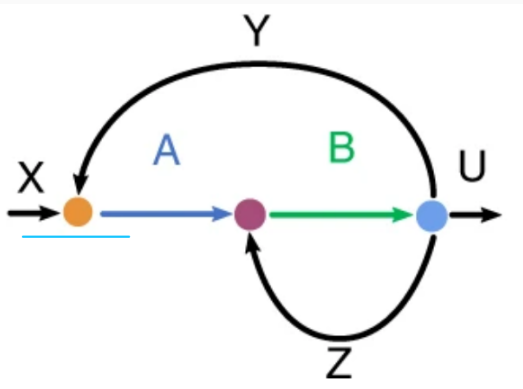
# Segmental duplications



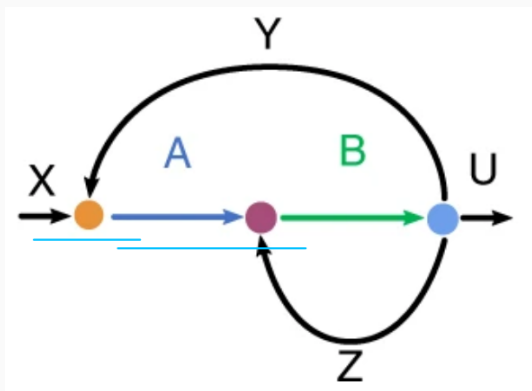
Segmental Duplications [4]

- 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.

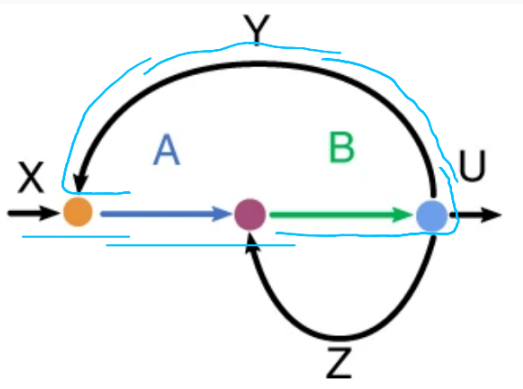
## Disjointig creation



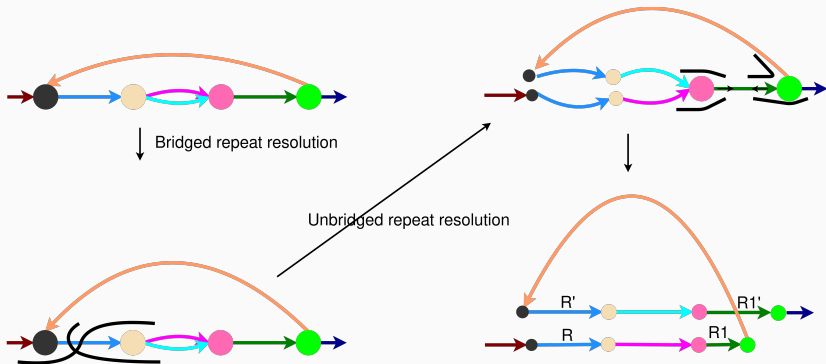
## Disjointig creation



## Disjointig creation



# Repeat resolution



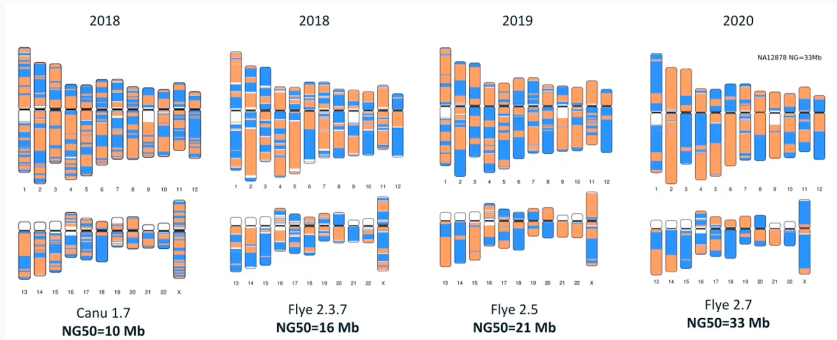
Repeat resolution of the given example [2]

## Repeat resolution

"Pilon is a software tool which can be used to automatically improve draft assemblies or to find variation among strains, including large event detection. Pilon requires as input a FASTA file of the genome along with one or more BAM files of reads aligned to the input FASTA file. Pilon uses read alignment analysis to identify inconsistencies between the input genome and the evidence in the reads. It then attempts to make improvements to the input genome, including" [5]:

- Single base differences
- Small indels
- Larger indel or block substitution events
- Gap filling
- Identification of local misassemblies, including optional opening of new gaps

# Contigity improvement



Contigity improvements (M. Kolmogorov, personal communication, 29.07.2021)

- colors are contigs → change in color means fragmentation