Earmark graph approach to *de novo* genome assembly

Mikhail E. Dvorkin Alexander S. Kulikov Max A. Alekseyev

Algorithmic Biology Lab, Academic University of the Russian Academy of Sciences, St. Petersburg, Russia
http://bioinf.spbau.ru/en/

The presentation of this work was made possible by a Travel Fellowship awarded by ISCB with grant funds obtained from the Department of Energy Office of Science, the National Science Foundation Bio-Directorate, and the NIH National Institute of General Medical Sciences.

Abstract

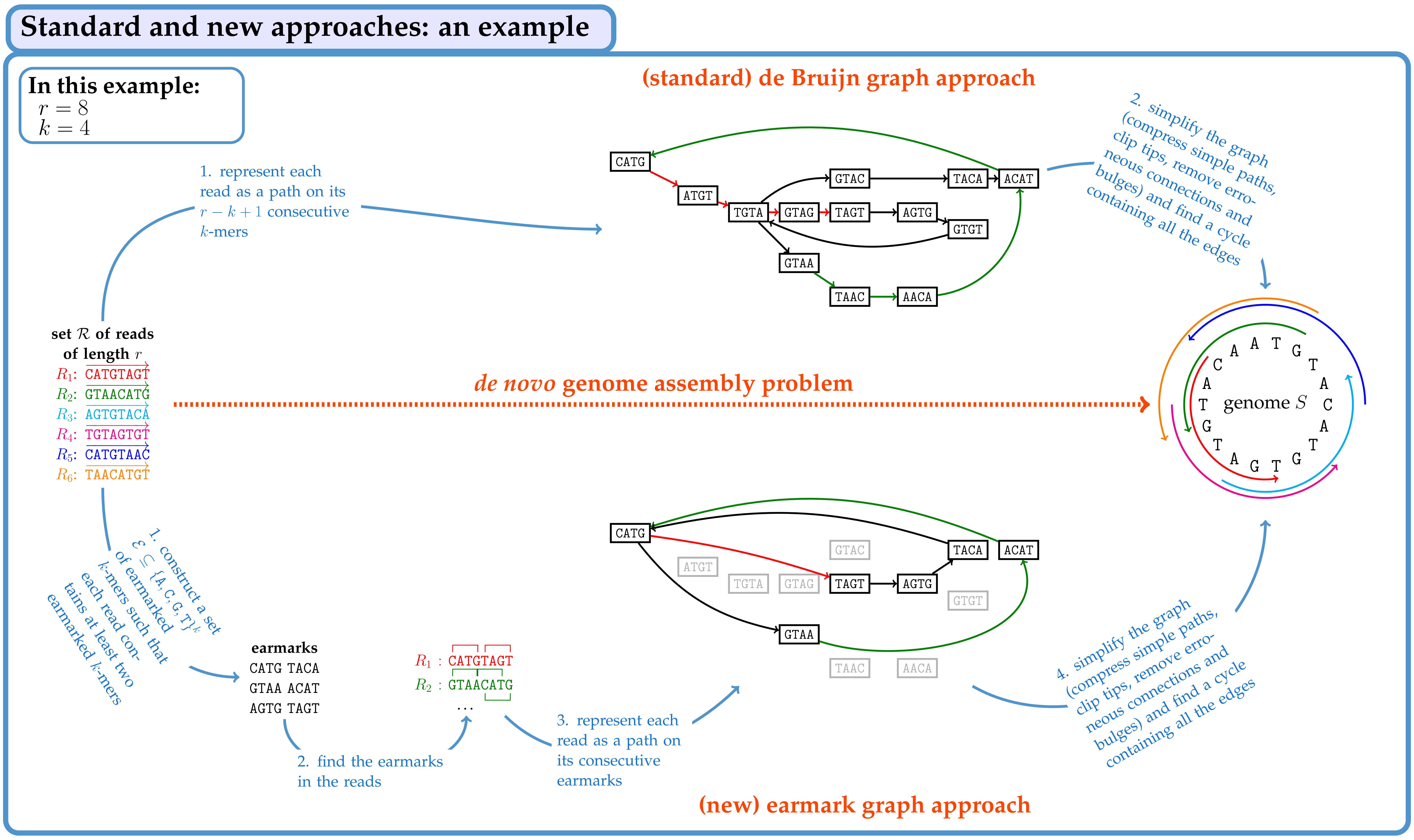
A common approach to assembling a genome from short reads is constructing the de Bruijn graph on all k-mers from the given set of reads and finding a traversal of edges in this graph. We propose a new approach that allows to decrease the graph size without losing the essential information from the input data. Instead of using all the k-mers from a read we take only a few of them (call them *earmarks*). Besides an obvious advantage of requiring less memory and time for constructing, the resulting earmark graph has several other advantages over the de Bruijn graph.

Typical genome assembly setting

Input: a set of substrings (called reads) $\mathcal{R} \subseteq \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}^r$ of an unknown circular string $S \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}^*$ (called genome).
Output: the genome S.

Complications: reads may contain errors.

Standard and new approaches: an example



In this example:
$r = 8$
$k = 4$

(standard) de Bruijn graph approach

1. represent each read as a path on its $r - k + 1$ consecutive k-mers

2. simplify the graph (compress simple paths, clip tips, remove erroneous connections and bulges) and find a cycle containing all the edges

set $\mathcal{R}$ of reads of length r

R_1: $\overrightarrow{\mathtt{CATGTAGT}}$
R_2: $\overrightarrow{\mathtt{GTAACATG}}$
R_3: $\overrightarrow{\mathtt{AGTGTACA}}$
R_4: $\overrightarrow{\mathtt{TGTAGTGT}}$
R_5: $\overrightarrow{\mathtt{CATGTAAC}}$
R_6: $\overrightarrow{\mathtt{TAACATGT}}$

de novo genome assembly problem

genome S

1. construct a set $\mathcal{E} \subseteq \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}^k$ of earmarked k-mers such that each read contains at least two earmarked k-mers

earmarks
CATG TACA
GTAA ACAT
AGTG TAGT

2. find the earmarks in the reads

R_1 : CATGTAGT
R_2 : GTAACATG
...

3. represent each read as a path on its consecutive earmarks

4. simplify the graph (compress simple paths, clip tips, remove erroneous connections and bulges) and find a cycle containing all the edges

(new) earmark graph approach

Benchmarking

		de Bruijn	earmarked
E. coli genome	# vertices	9149884	694592
	# edges	9154495	698170
+ compression	# vertices	30002	7862
	# edges	34613	11440
+ tips clipping	# vertices	8090	4142
	# edges	12701	7720
+ bulge removal	# vertices	6634	3030
	# edges	10487	5862
+ erroneous connection removal	# vertices	2852	1350
	# edges	4403	2514
+ tips clipping	# vertices	2742	1304
	# edges	4293	2468
+ bulge removal	# vertices	2442	1090
	# edges	3829	2080

E. coli genome,
reads of length $r = 100$,
filtered with Quake;
$k = 25$

Advantages of earmark graph over de Bruijn graph

Smaller size. The earmark graph requires less time and memory for construction. Note also that one can control the size of the earmark graph by varying the size of the set of earmarks. Also, it is easy to see that in an extreme case when $\mathcal{E}$ is just the set of all the k-mers of the input reads, the earmark graph coincides with the de Bruijn graph.

Simpler structure. In the example above, the de Bruijn graph contains two vertices v with $\text{indegree}(v) \times \text{outdegree}(v) \geq 2$, while the earmark graph contains only one such vertex. This corresponds to a simpler representation of repeats in the genome and simplifies the problem of finding a cycle in the graph.

Using trusted k-mers. By restricting earmarks to k-mers present in many reads ("trusted" k-mers) one can reduce the number of erroneous edges resulting from sequencing errors in the reads.

References

1. Pevzner, Tang, Tesler. *De novo* repeat classification and fragment assembly. *RECOMB*, 2004.
2. Pevzner, Tang, Waterman. An Eulerian path approach to DNA fragment assembly. *Proc Natl Acad Sci USA*, 2001.
3. Roberts, Hayes, Hunt, Mount, Yorke. Reducing storage requirements for biological sequence comparison. *Bioinformatics*, 2004.
4. Zerbino, Birney. Velvet: algorithms for *de novo* short read assembly using de Bruijn graphs. *Genome Res.*, 2008.