SyntenyFinder: A Synteny Blocks Generation and Genome Comparison Tool

Ilya Minkin[1], Nikolay Vyahhi[1], and Son Pham[2]

[1] St. Petersburg Academic University, St. Petersburg, Russia
[2] University of California, San Diego, USA

Introduction

We propose *SyntenyFinder* – a tool for finding synteny blocks in genomes represented as nucleotide sequences. Our approach is based on de Bruijn graph and can be applied to closely related genomes.

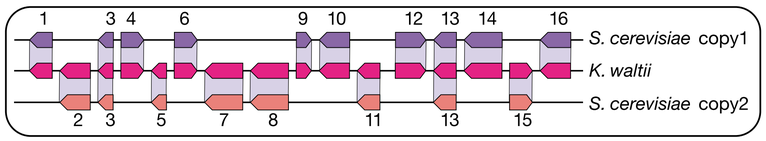


Figure 1: Rectangles with arrows depict synteny blocks in two yeast genomes (Kellis2004)

Recent advances in sequencing and genome assembling technologies are resulting in many finished genomes. The comparison of these genomes has been emerging as a powerful tool for genome interpretations. These tasks often require genomes to be decomposed to a collection of synteny blocks – regions of conserved DNA.

De Bruijn graphs in synteny block mining

Given a string S and a natural number k we construct k-dimensional de Bruijn graph $G(k)$ as follows:

- For each unique k-substring add a vertex to $G(k)$
- For each $(k+1)$-substring w in S add to $G(k)$ an edge that connects k-prefix of w with k-suffix of w
- Label edges with positions of the corresponding $(k+1)$-mers

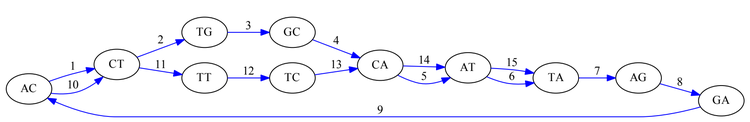


Figure 2: De Bruijn graph $G(2)$ built from string *"ACTGCATAGACTTCATA"*

In this graph we allow only paths that have consecutive labels on edges. Graph $G(k)$ has following properties:

- Each valid path in the graph spells a substring from S
- Non-branching paths in the graph indicate exact repeats in S
- Variations in repeats create *bulges* in $G(k)$
- Bulges are formed by two valid paths with shared ends

We remove bulges with size larger than some predefined constant δ to obtain long non-branching paths. This process is called *simplification*.

Double strandness issue

Remember that a DNA molecule has two strands and synteny blocks can be located on both. We resolve this by following:

- Build graphs for both strands separately
- Label edges in these graphs with two different colors
- Merge two graphs and work with the resulting graph

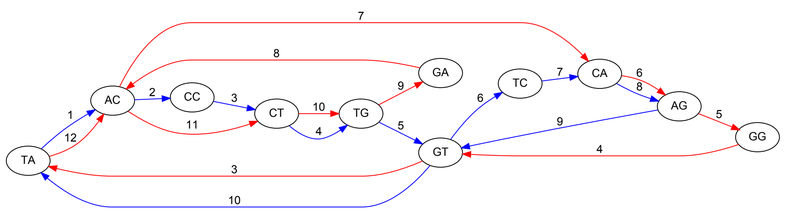
SyntenyFinder algorithm

Given two numbers k and δ and a set $S = \{S_1, S_2, \ldots, S_n\}$ of chromosomes represented as nucleotide strings, our algorithm works as follows:

- Concatenate all chromosomes in S into the supergenome $\hat{S}$
- Construct graph $G^+(k)$ from $\hat{S}$ and color all it's edges *blue*
- Construct graph $G^-(k)$ from reverse-complementary of $\hat{S}$ and color all it's edges *red*
- Obtain $G(k) = G^+(k) \cup G^-(k)$
- Change $\hat{S}$ so that $G(k)$ doesn't contain bulges having size $< \delta$
- Output non-branching paths

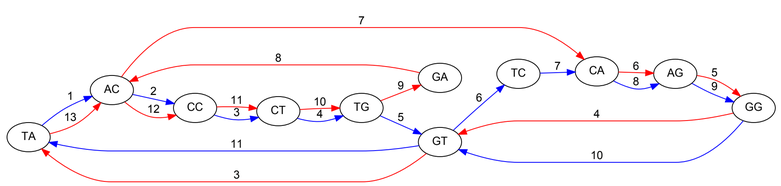
At any moment during simplification there is a one-to-one correspondence between the string and the graph – any changes in $\hat{S}$ are immediately reflected in $G(k)$.

5' TACCTGTCAGTA 3'
3' ATGGACAGTCAT 5'



(a) De Bruijn graph $G(2)$ built from $\hat{S}$ = *"TACCTGTCAGTA"*

5' TACCTGTCAGGTA 3'
3' ATGGACAGTCCAT 5'



(b) The same graph after collapsing the bulge.

Figure 3: Illustration of *de Bruijn* graphs and simplification process

Results

We present benchmarking of our tool on two datasets:

- Four strains of the bacteria *Pseudomonas aeruginosa*
- Two different yeast species: *K.waltii* and *S. cerevisiae*

Dataset	Total size	k	δ	Multiplicity	Count	Coverage
Bacteria	27 MBP	1000	5000	4	140	90%
Yeast	23 MBP	1000	20000	3	270	64%

For the yeasts we used local alignments from (Kellis2004) to enrich number of common k-mers in conserved regions. Our blocks share 80% of their basepairs with the blocks described in (Kellis2004).

Discussion

SyntenyFinder can be efficiently used for finding synteny blocks in closely related genomes represented as nucleotide sequences. Benchmarks show that with some modifications our method can be applied to more distant genomes. Our near plans include:

- Release version for closely related species (Fall 2012)
- Extend our tool to a wider range of genomes
- Incorporate it into genome rearrangements analysis tools

Acknowledgments

We thank Pavel Pevzner, Steve O'Brien, Alla Lapidus, Matt Schultz and Dinh Diep for their for their help and contributions to many useful discussions.
This work was supported by the Government of the Russian Federation (grant 11.G34.31.0018) and the National Institutes of Health (NIH grant 3P41RR024851-02S1).

References

Idury RM, Waterman MS
A new algorithm for DNA sequence assembly.
Journal of computational biology, 1995 Summer; 2(2):291-306.

Kellis M, Birren BW, Lander ES
Proof and evolutionary analysis of ancient genome duplication in the yeast Saccharomyces cerevisiae.
Nature, 2004 Apr 8; 428(6983):617-24.

Pham KS, Pevzner PA
DRIMM-Synteny: decomposing genomes into evolutionary conserved segments.
Bioinformatics, 2010; 26(20):2509-2516.