Finding Syntenic Regions in Multiple Unannotated, Unaligned Genomes

Matthis Ebel[1], *Ingo Bulla*[2] *and Mario Stanke*[1]

[1] Institute of Mathematics and Computer Science, University of Greifswald {matthis.ebel, mario.stanke}@uni-greifswald.de

[2] Université Perpignan Via Domitia, IHPE UMR 5244, CNRS ingobulla@googlemail.com

UNIVERSITÄT GREIFSWALD
Wissen lockt. Seit 1456

Introduction

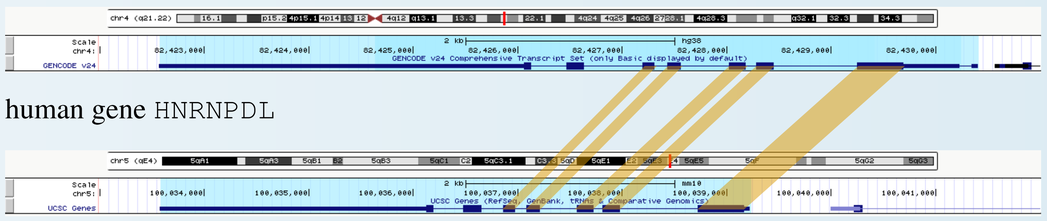
We present a new approach for finding tuples of homologous regions in multiple genomes. Our aims are

- the fast identification of orthologous genes and other genomeic elements
- without the need for an alignment

A downstream application is de novo comparative genome annotation of clades. We target clades of many related genomes that

- are alignable
- may have undergone genome rearrangements since the most recent common ancestor

The approach is based on k-mers, short sequences of length k, that have exact matches in multiple genomes. We expect that in the conserved coding regions of homologous genes, there are many such k-mers. We seek to use accumulations of shared k-mers as hints for finding homologous regions.



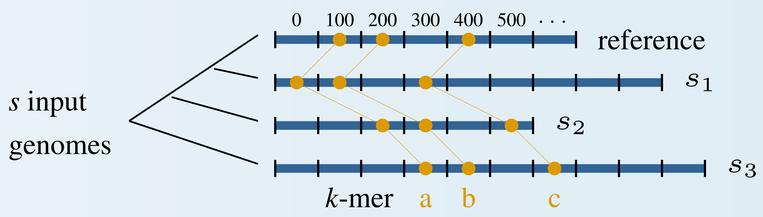
human gene HNRNPDL

mouse gene HNRNPDL

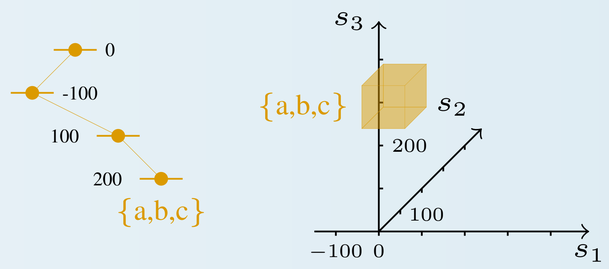
screenshot UCSC genome browser [3, 2]

II – Geometric Hashing

Geometric hashing is a technique to find recurring patterns in data that may have undergone affine transformations such as relocation [5]. With it we efficiently identify sets of k-mers that all have similar relative distances with respect to a reference genome. We use rounded positions to find related k-mers also if short indels happened in some sequences. If a k-mer has a match in s input genomes, it is mapped to a *cube* in a $s-1$ dimensional space. All k-mers that have a similar relative offset in their occurences appear in the same cube. Cubes with significant number of k-mers suggest approximate "alignment" offsets.



rounded absolute positions



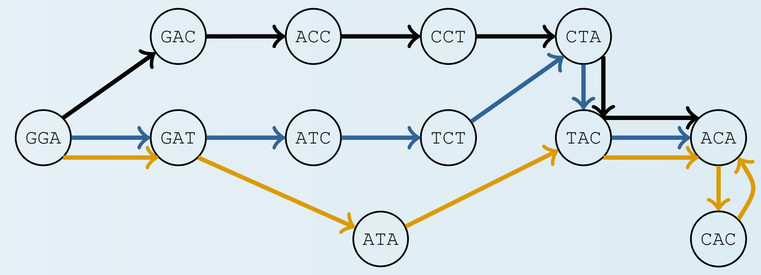
relative distances

I – Genome Graph

Mustafa et al. [4] developed a tool (based on [1]) that stores and queries a colored De Bruijn graph of multiple genomes very efficiently. Our work is based on this graph which delivers the shared k-mer sets over the input genomes from which we start. For each k-mer, it also delivers the position information for each genome it occurs in.

GGACCTACA
GGATCTACA
GGATACACA

input genomes



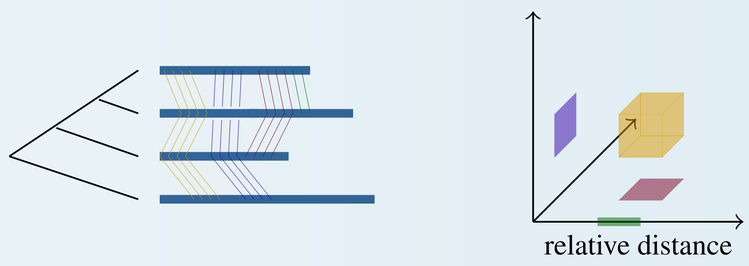
colored De Bruijn graph

k-mer	genome coordinates
ACA	coord$_1$, coord$_2$, coord$_3$, coord$_4$
...	...

coordinate graph annotation

III – Connect and Select Cubes

The cubes at which k-mers are grouped can have less than $s-1$ dimensions if a k-mer occurs only in a subset of the s species. Collecting all of these k-mers is done by applying a partial order on the set of cubes and drawing the transitive reduction, also known as Hasse diagram. This links related cubes of different dimensionality.



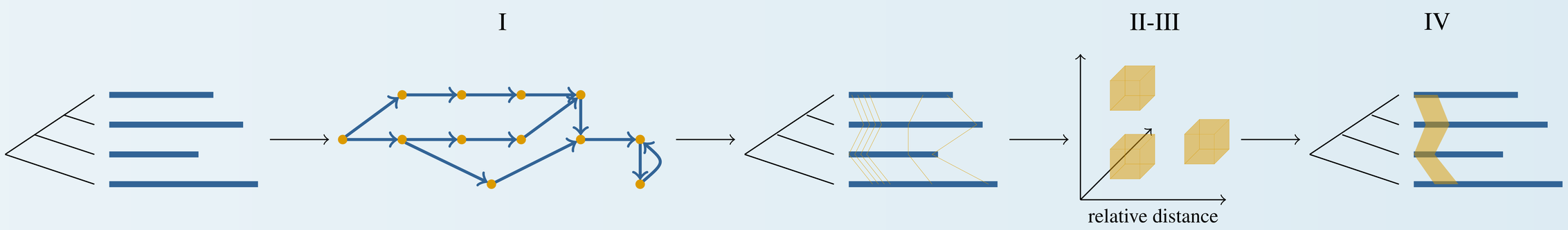
Hasse diagram

Each point inherits the k-mers from the connected lower dimensional points. From all points, a suitable subset is chosen via a scoring function. The score of a point accounts for the number of k-mers it contains and the number of genomes each k-mer covers. This subset is then used to extract homologous regions.

IV – Homologous Regions

Tuples of homologous regions are extracted from the highest scoring cubes. As there can be huge gaps between regions with high k-mer density, an algorithm needs to be applied that finds a good balance between number and size of contiguous region tuples and k-mer coverage. The details of this extaction step are work in progress.

Workflow



References

[1] Alexander Bowe et al. "Succinct de Bruijn Graphs". In: *Algorithms in Bioinformatics*. Ed. by Ben Raphael and Jijun Tang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 225–235. ISBN: 978-3-642-33122-0.

[2] Jonathan Casper et al. "The UCSC genome browser database: 2018 update". In: *Nucleic acids research* 46.D1 (2017), pp. D762–D769.

[3] W James Kent et al. "The human genome browser at UCSC". In: *Genome research* 12.6 (2002), pp. 996–1006.

[4] Harun Mustafa et al. "Metannot: A succinct data structure for compression of colors in dynamic de Bruijn graphs". In: *bioRxiv* (2018). DOI: 10.1101/236711.

[5] Haim J Wolfson and Isidore Rigoutsos. "Geometric hashing: An overview". In: *IEEE computational science and engineering* 4.4 (1997), pp. 10–21.