Improving Comparative Genomic Studies: Definitions and Algorithms for Syntenic Blocks

THÈSE N^{O} 6347 (2014)

PRÉSENTÉE LE 26 SEPTEMBRE 2014

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS
LABORATOIRE DE BIOLOGIE COMPUTATIONNELLE ET BIOINFORMATIQUE
PROGRAMME DOCTORAL EN INFORMATIQUE ET COMMUNICATIONS

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Cristina Gabriela GHIURCUŢA

acceptée sur proposition du jury:

Prof. W. Gerstner, président du jury
Prof. B. Moret, directeur de thèse
Dr Ph. Bucher, rapporteur
Dr J.-P. Kocher, rapporteur
Prof. J. Stoye, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2014

Acknowledgements

Success is not a destination but the road that you're on has it the saying, and I will add to it *and the people that you meet along it.* I would like to express my most sincere gratitude for the very precious contribution of my friends and family, who greatly contributed to my present achievements.

My utmost gratefulness to Prof. Bernard Moret, for having me part of the LCBB family; for being my priceless mentor and friend; the fascinating discussions about science, society and concepts—thank you for putting up with all my questions. I cannot thank enough you enough for all the support, patience, the valuable teaching, and the guidance that you offered me in moments of need; your confidence in my success; giving me the freedom to explore, while always keeping me on the right track! Thank you, Bernard, for being such a wonderful mentor!

My warmest thanks to the ex- and present members of the LCBB: Xiuwei Z., Vaibhav R., Yu L. and Yann C., for all the support, insightful (non-)scientific discussions, but mostly for the warm welcome they gave me when I joined the group. Your support was of tremendous help to me. Special thanks to Nishanth N. and Slavica D., fellow Ph.D. students and friends whit whom I spent so many nice moments during my time at EPFL. Sincere thanks to Min Y. and Mingfu S., for keeping the spirit of the lab high, and equally for their support.

I had the pleasure to mentor and work with Anastasiya Tychinskaya during her semester project and with Dorija Humski during her summer internship at EPFL. Thank you both for the good work that contributed to this thesis, and for the fun time that you brought along! Friendly thanks to Laura H. for the amazing, lively and lovely spirit that she brought to us during this past summer.

I thank the EPFL Doctoral School and the Swiss Institute of Bioinformatics for the scientific training that they offered over the past few years; for providing a truly diverse and creative environment, along with access to cutting-edge technology!

I sincerely appreciate and thank to Prof. Jens Stoye, Dr. Philipp Bucher, Dr. Jean-Pierre A. Kocher and to Prof. Wulfram Gerstner for the considerable effort and participation in my thesis committee, and for their valuable feedback.

As part of paving my road to this successful piece of work, I acknowledge and thank to my mentor at ETH Zürich, Prof. Gábor Székely, and Dr. Alexander Rauch, for their patience, support and guidance during my first research experience in the field of computational biology. My special thanks to the members of the BIWI, especially to Michael B., Bryn L., Gabriele F., Andreas E., Peter B., Alain L. and Benjamin H., who gave me an overwhelmingly hearty welcome to the lab, and in the same time also to Switzerland. I am deeply grateful to Dr. Bernhard Reber and Ruth Steinmann, for their kind support and for caring for me like my family!

I am sincerely thankful for all the precious friendships that I made, during my stay in Switzerland, through IAESTE and at EPFL. I could not have made it without you guys, without the rich memories and the many happy moments that you granted me. Thank you for giving me your time and for your presence in my life! Special thanks to Ludek C., Petr S., Jennifer S. and Mutaz A., for being there for me!

Sincere and warm thanks to Stefanie Uhl, for all the patience, initiative, understanding and her priceless pieces of advice, since my very beginnings in Switzerland. *Thank you, Stef!*

Iulia, my dear *sis;* thank you so much for your precious support and patience, for constantly being there for me during the past 10 years. I am so grateful to have met you!

Finally, but most importantly, I owe this success to *my parents*, Ildiko and Ioan Ghiurcuţa; I have not enough words to express my deepest gratitude for your tremendous effort and sacrifice that you put into setting (and keeping) me on the right path, and into shaping me into the person that I am today. Your unconditional love and affection, your support and understanding, your priceless teachings through all these years, made it possible to be here today. I dedicate this work to you!

Drága, páratlan es nagyon szeretett anyu és apu, e doktori sikerét nektek köszönhetem. A ti kitartásotok és erőtok hogy áthidaljatok nehéz pillanatokat szolgál erőforrásként számomra. A ti kiválló és egyedűlalló tanításotok által lettem azzá aki a mai nap vagyok és mindent amit elértem, azt a ti önzetlen szeretetetekből és támogatásotokból meritett erővel értem el. Ezért ezt a munkát nektek dedikálom!

Lausanne, September 2014
Cristina Gabriela Ghiurcuta

Abstract

Comparative genomics aims to understand the structure of genomes and the function of various genomic fragments, by transferring knowledge gained from well studied genomes, to the new object of study. Rapid and inexpensive high-throughput sequencing is making available more and more complete genome sequences. Despite the significant scientific advance, we still lack good models for the evolution of the genomic architecture, therefore analyzing these genomes presents formidable challenges. Early approaches used pairwise comparisons, but today researchers are attempting to leverage the larger potential of multiway comparisons.

Current approaches are based on the identification of so called *syntenic blocks*: blocks of sequence that exhibit conserved features across the genomes under study. Syntenic blocks are in many ways analogous to genes—in many cases, the markers are used to constructing them are genes. Like genes they can exist in multiple copies, in which case we could define analogs of orthology and paralogy. However, whereas genes are studied at the sequence level, syntenic blocks are too large for that level of detail—it is their structure and function as a unit that makes them valuable for genome level comparative studies.

Syntenic blocks are required for complex computations to scale to the billions of nucleotides present in many genomes; they enable comparisons across broad ranges of genomes because they filter out much of the individual variability; they highlight candidate regions for in-depth studies; and they facilitate whole-genome comparisons through visualization tools. The identification of such blocks is the first step in comparative studies, yet its effect on final results has not been well studied, nor has any formalization of syntenic blocks been proposed.

Tools for the identification of syntenic blocks yield quite different results, thereby preventing a systematic assessment of the next steps in an analysis. Current tools do not include measurable quality objectives and thus cannot be benchmarked against themselves. Comparisons among tools have also been neglected—what few results are given use superficial measures unrelated to quality or consistency.

In this thesis we address two major challenges, and present: (i) a theoretical model as well as an experimental basis for comparing syntenic blocks and thus also for improving the design of tools for the identification of syntenic blocks; (ii) a prototype model that serves as a basis for implementing effective synteny mining tools. We offer an overview of the milestones present in literature, on the development of concepts and tool related to synteny; we illustrate the application of the model and the measures by applying them to syntenic blocks produced by different contemporary tools on publicly available data sets.

We have taken the first step towards a formal approach to the construction of syntenic blocks by developing a simple quality criterion based on sound evolutionary principles. Our experiments demonstrate widely divergent results among these tools, throwing into question the robustness of the basic approach in comparative genomics. Our findings highlight the need for a well founded, systematic approach to the decomposition of genomes into syntenic blocks and motivate the second part of the work—starting from the proposed model, we extend the concept with data dependent features and constraints imposed by the computational power, in order to test the concept in practice and to provide a solid, structured basis for future synteny mining tools.

Keywords: homology, syntenic blocks, comparative genomics, de Bruijn graphs, multiway comparisons, whole genomes, yeast genomes, sequence similarity, markers, formal model.

Résumé

La génomique comparative vise à comprendre la structure des génomes et la fonction des regions genomiques particulieres, par le transfert de connaissances acquises découlant des génomes bien étudiés, au nouvel objet d'étude. Le séquençage rapide et peu cher à haut débit a mis à la disposition de plus en plus des séquences complètes du génome. Malgré le progrès scientifique significatif, nous manquons encore de bons modèles pour l'évolution de l'architecture génomique, donc l'analyse de ces génomes présente toujours de formidables défis. Les premières approches utilisées comparaisons par paires, mais aujourd'hui, les chercheurs tentent de tirer parti de la plus grande potentiel de comparaisons multivoies.

Les approches actuelles sont basées sur l'identification de ce qu'on appelle *blocs de synténie* : blocs de séquence qui présentent des caractéristiques conservées à travers des génomes étudiées. Les blocs de synténie sont à bien des égards analogues à des gènes—dans de nombreux cas, les marqueurs utilisés pour les construire sont des gènes. Comme les gènes, ils peuvent exister en plusieurs exemplaires, dans ce cas, nous pourrions définir des analogues de l'orthologie et de la paralogie. Cependant, tandis que les gènes sont étudiées au niveau de la séquence, les blocs de synténie sont trop grands pour telle niveau de détail—c'est leur structure et fonction comme une unité qui les rendent utiles pour des études comparatives au niveau du génome entier.

Les blocs de synténie sont nécessaires pour des calculs complexes à l'échelle des milliards de nucléotides présents dans de nombreux génomes ; ils permettent des comparaisons entre des larges gammes de génomes, car ils filtrent beaucoup de la variabilité individuelle ; ils mettent en évidence les régions candidats pour des études approfondies ; et ils facilitent les comparaisons des génomes entiers grâce à des outils de visualisation. L'identification de ces blocs est la première étape dans les études comparatives, quoique leur potentiel n'a pas été bien étudié, ni a une formalisation des blocs de synténie été proposé.

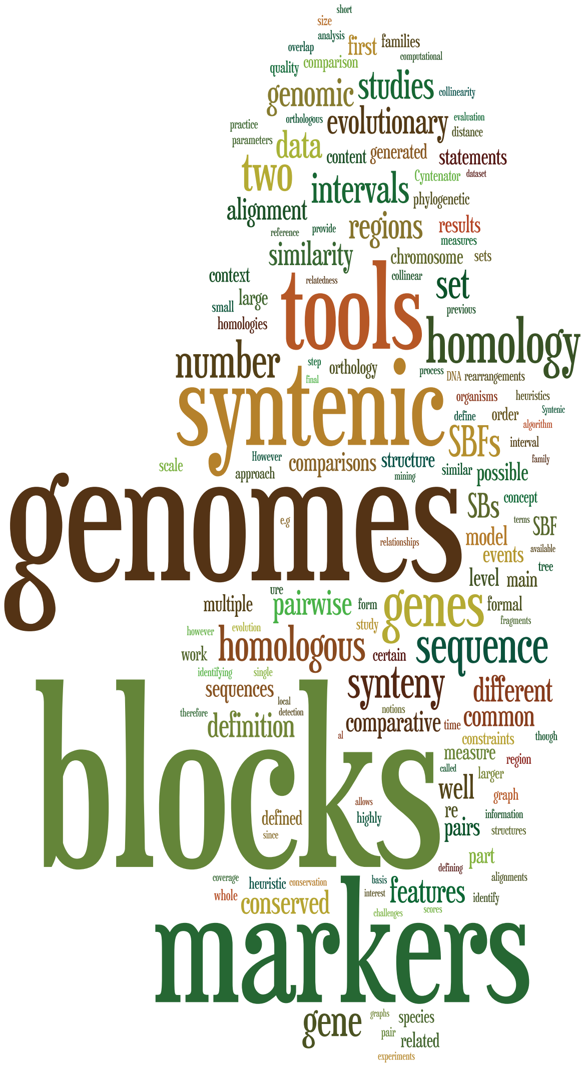
Des outils pour l'identification des blocs de synténie donnent des résultats tout à fait différents, ce qui empêche une évaluation systématique des étapes suivantes de l'analyse. Les outils actuels ne comprennent pas des objectifs de qualité mesurable, donc ils ne peuvent donc pas être comparés éux-mêmes. Les comparaisons entre les outils ont également été négligé—les peu des résultats de ces comparaisons qui sont donnés, utilisent des mesures superficielles sans rapport avec la qualité ou la cohérence.

Dans cette thèse, nous abordons deux défis majeurs, et présente : (i) un modèle théorique et une base expérimentale pour comparer des blocs de synténie, qui pourra servir à améliorer la conception d'outils pour l'identification des blocs de synténie ; (ii) un prototype qui sert comme base pour la mise en œuvre des outils efficaces d'extraction de synténie. Nous proposons un aperçu des étapes présentés dans la littérature, sur le développement des concepts et d'outils liés à la synténie ; nous illustrons l'application du modèle et les mesures en les appliquant à des blocs de synténie produites par différents outils contemporains sur des ensembles de données publiquement disponibles.

Nous avons fait le premier pas vers une approche formelle de la construction des blocs de synténie par l'élaboration d'un critère de qualité simple, basé sur des principes sons d'évolution. Nos expériences montrent des résultats très divergents parmi ces outils, en questionant la robustesse de l'approche de base en génomique comparative. Nos résultats mettent en évidence la nécessité d'une approche systématique fondée à la décomposition des génomes en blocs de synténie et de motiver la deuxième partie de ce travail—à partir du modèle proposé, nous étendons la notion de base avec des contraintes caractéristiques a des données et celles imposés par la puissance de calcul, afin de tester le concept dans la pratique et à fournir une base solide et structurée pour les futurs outils d'exploration de synténie.

Mots-clés : homologie, blocs de synténie, la génomique comparative, des graphes de Bruijn, des comparaisons multivoies, génomes entiers, génomes de levure, similarité de séquence, marqueurs, modèle formel.

Thesis Content Highlights



Contents

List of Figures

List of Tables

1 Introduction

The intricate story of evolution is encoded in the genomic data of all living organisms. Cracking evolution, implies a good understanding of the mechanisms and principles which drive biological processes; here we include the nucleotide level mutations (identified as Single Nucleotide Polymorphisms—SNPs), insertions and deletions that shape the genome at a small scale, up to the large scale events that involve the alteration of extended DNA strips—s.a. rearrangements (transposition, inversion, fusion, etc.), recombination, (whole genome) duplications, gain or loss of genomic material of different types (lateral or horizontal transfer, etc.). An evolutionary perspective that explains the diversity of life and considers such events, relies on fundamental and commonly accepted ideas that form the basis of structured and formal evolutionary models. As such, the data that is used to validate them has to come from several organisms and tissues, revealing evolution.

Literally, comparative genomics allows one to connect laboratory notebooks of clinical and basic researchers. Studying certain organisms in a laboratory setting is difficult–experiments cannot be performed on some organisms, including the main target for the application of clinical studies—humans, for ethical and practical reasons. Hence, learning about such organisms is best done by studying other ones that present a high degree of relatedness to them. In this context, comparative approaches have long been the mainstay of knowledge discovery and transfer in biology. Years of research show that tracing back evolutionary events is best done by transferring knowledge from well studied organisms to the ones that are not yet well studied. This transfer is done by identifying conserved patterns through sequence alignment and careful analysis of the DNA of highly related species. With knowledge of model organism genomes, biomedical studies of human genes can be complemented by experimental manipulations of corresponding e.g. mouse genes to accelerate functional understanding.

Depending of the aim of the study–whether it is for detecting population level genetic changes or phylogenetic studies of multiple organism, different models and data sources apply. Comparing even just two genomes raises computational and conceptual challenges. Consequently, researchers have addressed the computational problem by defining (or searching for) well conserved sequences (mostly belonging to the better understood coding regions of the genome) that are under positive selection playing a critical role in the existence of an organism. *Genes*—large sequences whose identity for relatedness and sequence similarity is not in doubt, make up a very small part the genome—5%, while most of it (90% in humans) is poorly understood and so lacks a suitable evolutionary model. For this reason, most of the studies employ genes, the main focus being the interactions between genes and the associated regulatory regions.

With the advent of inexpensive sequencing tools, pairwise sequence comparison became a major research tool; programs such as BLAST [1] are used to identify short, super conserved genomic regions in similar genome sequences, in order to study problems in genetics and genomics, by using knowledge from better characterized organisms. Such comparisons have been carried out on relatively short sequence fragments–usually up to the length of a protein transcript, i.e., a few thousand nucleotides.

Such work continues at a great pace today, but the rapidly increasing availability of complete genome sequences has led to the desire to compare entire genomes at once, the better to understand the large-scale architectural features of genomes and the evolutionary events that have shaped these features, such as segmental and whole-genome duplication, horizontal transfer, recombinations of various types, and rearrangements.

In the case of vertebrates, mammals in particular, the genome comprises over 3G base pairs (bps), which is huge in the context of what present computational methods can handle. A sparse sampling of the genome can be achieved by identifying regions of high similarity with other genomes. There exist a number of attempts to identify such regions, some approach it in a very principled way, such as Mauve [8]. The regions are then used to form large-scale patterns that can be evaluated for similarity and conservation. Such large scale patterns when used systematically, can be viewed as alternative representations of the genomes.

The simplest such representation uses the concept of *syntenic blocks* (SBs), large blocks of sequence that are well conserved (as testified by commonality of markers and similarity of high-level patterns) across the species (or within a genome). Comparative genomics relies on the structuring of genomes into SBs. SBs are required for complex computations to scale to the many similar regions shared by multiple genomes.

Working with such blocks facilitates comparisons across broad ranges of genomes: (i) it confers robustness against variability across individuals and against various sources of error; (ii) it reduces the dependence on an accepted model of sequence evolution for each region and is less likely to suffer from homoplasy–a parallel and independent evolution of analogous structures in different organisms, that was not present in the common ancestor; (iii) it reduces the complexity of the analysis of the genomic structures; (iv) it provides high-level features for further evolutionary studies; (v) it identifies specific regions of interest for detailed studies and possible bench experiments; and (vi) it facilitates whole-genome comparison through visualization tools.

However, the concept of SB remains loosely defined. Tools for the identification of SBs yield quite different results, thereby preventing a systematic assessment of the next steps in an analysis. Current tools do not include measurable quality objectives and thus cannot be benchmarked against themselves. Comparisons among tools have also been neglected–what few results are given use superficial measures unrelated to quality or consistency.

This thesis highlights systematically, the role that is played by syntenic blocks in comparative studies. The thesis has two parts:

1. **A survey on the notion of synteny and syntenic blocks present in current literature**– definitions, use in comparative studies; what are the issues that prevent the application of syntenic blocks to whole genome comparative studies; solutions to two of the problems. This is presented in chapter 2 through 4.
2. **A practical model and its possible implementation** that makes use of our formal framework proposed in the first part, that we enrich with common sense principles inspired by hands on dataset analysis.

Based on the review of the work on synteny and its evolution in the context of comparative studies, we provide solution to two of the problems that we point out throughout it.

- **A formal and principled definition for syntenic blocks**: here we address the lack of a consistent, commonly accepted and used notion for syntenic blocks; as a direct consequence, a common basis for the comparison and evaluation of such structures is not possible, thus their use in comparative studies is hindered. We promote the use of syntenic blocks in whole-genome comparative studies, and propose a principled definition that serves as an experimental basis for comparing syntenic blocks and for improved design for tools that are meant to identify such blocks. This work is described in detail in chapter 4.
- **A number of measures that are used to evaluate current synteny mining tools' output against our definition**: reflects features of syntenic blocks, the relatedness, respectively the structural differences among current syntenic blocks and the heuristics that are used in existing state of the art tools. The results and findings from the evaluation of various syntenic blocks against the same definition, as well as the challenges revealed by our findings are presented along chapter 5.

The second part of the thesis focuses on the refinement and adaptation of the formal definition to defining and identifying synteny block families on specific datasets:

- **By using weak common intervals, we propose a model for identifying syntenic blocks**: we identifying genomic regions of high similarity that are further used in the context of common intervals, to formalize and adapt the concept to datasets of interest, for which we generate syntenic blocks. This part of the thesis is presented in chapter 6 and is work in progress.

Chapter 2 introduces notions of comparative genomics. In a first part, an overview presents genomic units that have been used over time in comparative studies; while the second part of the chapter presents the methods that have been commonly used to find such genomic units. A comprehensive review of the previous work that is necessary for synteny based studies, in literature since early times up to contemporary notions, is provided throughout chapter 3. Chapter 7 is the conclusions chapter.

2 Background

The first part of this thesis provides a comprehensive review on the evolution of the notion of synteny in the context of comparative studies. The survey implicitly extends over the heuristics used by tools designed for identifying syntenic regions, from early notions up to the state of the art. This chapter provides the notions necessary for understanding the work from this thesis.

2.1 Biological concepts

Frequently the reason why the reliability of comparative studies is jeopardized, is the lack of sound understanding of basic notions of sequence similarity and evolutionary relatedness. In this sense, throughout this section, we will briefly present the notions of *homology*, *orthology*, *markers* and *synteny* and their main features.

To start with, Figure 2.1 sketches an example of hierarchy between homology and orthology, and paralogy. As the figure shows, homology is the basic form of evolutionary relatedness, while orthology designates a pair of related genes consequence of a speciation event. Paralogs are then copies of the same ancestral gene, but which reside in the same genome; unlike in the case of orthology where the association is between organisms belonging to different species. A more in detail presentation of these terms follows.

2.1.1 Homology

In evolutionary biology, two structures (character positions in a sequence, markers of various types, genes, syntenic blocks) are *homologous* if they are descended from a common ancestral structure [10]. Common ancestry is the base feature for homology, but in the same time such structures present highly similar or identical base composition, therefore sequences that are homologous are also called conserved. To be noted though, that such structures do not necessary fulfill similar functions.

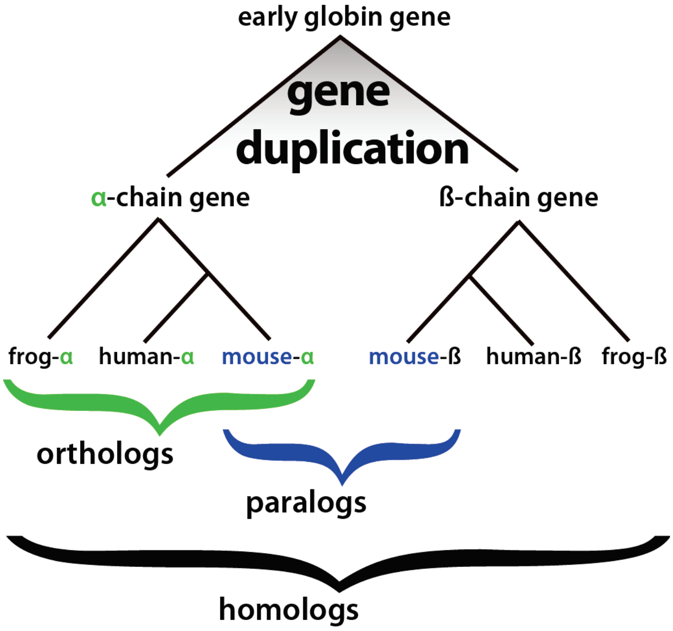


Figure 2.1 – A visual representation of the evolutionary events and thus the presence of different copies of the same gene in variously related species, that makes the difference between the degree of relatedness of these copies.

Homology cannot be observed, but only inferred. It is not a measurable quality of two structures—there exists no quantification for the degree of relatedness. Due to this feature, homology is transitive, a unique property that distinguishes it from the other forms of evolutionary relatedness. It is an equivalence relationship and, as such, determines equivalence classes, the families of homologous structures.

Sequence similarity alone can be explained by various phenomena, other than true homology. High sequence similarity can occur by chance, by *convergent evolution*–independent evolution of similar features in species that belong to different lineages and that were not present in the common ancestor. Such an evolution creates analogous structures–also known as *homoplasy*. Yet in practice, homology for markers and genes is determined on the basis of sequence similarity, using tools such as BLAST. For this reason, inferred homologies are neither symmetric nor transitive in practice, as they depend on similarity thresholds.

2.1.2 Orthology

It is another form of genomic structure relatedness, mostly used to identify genes. It relies on homology, but in addition to it, the structures are *orthologous* if the branching at the last common ancestor was a speciation. Orthology, in contrast with homology, depends on the speciation point—when a species diverges into two separate species, and so is context-dependent; in particular, it is generally not transitive. (For instance, two gene duplicates within the same genome cannot be orthologous, but these two duplicates and a homologous gene in another species are orthologous if the duplication followed the speciation.) Instead, orthology must be specified through hierarchies structured through the phylogeny as pointed out by Gabaldon in [12].

In this context, orthologs are indispensable for species' evolution as they serve as milestones for speciation events. Until recently, another strong statement about the role of orthologs was their role played in functional information transfer from experimentally already characterized genes to the ones in newly sequenced genomes. However, the strong belief that orthologous copies of the same ancestral gene belonging to different species are the closest related, has been challenged by stating that copies of the same gene in the same species (paralogues) are functionally more closely related than orthologs. A comprehensive overview of the topic is given in [12]. Such findings call for a thorough approach in studies that use orthologous copies, and throw into question the reliability of their results.

Like homology, orthology is also inferred. It is also initially determined through sequence similarity, but often verified through phylogenetic analysis or by ascertaining functional similarity. This makes orthology, a stronger relationship than homology; it is often preferred, at least for pairwise comparisons, as it may provide higher-quality markers.

However, only rarely is position along the genome taken into account—exceptions are the database OrthoDB, which also provides a hierarchy of orthology relationships, and the orthology tool MSOAR [11]. In practice, therefore, identifying homologies is much easier than identifying its particular cases.

2.1.3 Markers and Anchors

From medical research, but for comparative studies as well, genetic markers are very practical as they can help link an inherited disease with the responsible gene. A genetic marker is a short—even single nucleotide (identified as a SNP), yet precisely conserved DNA sequence across genomes; it has a known physical location on a chromosome; it is long enough to make its conservation statistically significant. DNA segments close to each other on a chromosome tend to be inherited together, as they are in *linkage.* The genetic marker itself may be a part of a gene or may have no known function. Genetic markers are used to track the inheritance of a nearby gene that has not yet been identified, but whose approximate location is known. A short review and presentation of the use and origin of markers and their application in population genetics is given in [31].

Genomic alignment uses markers as *anchors*, that is, fixed references in the alignment. It may use however, a richer pool of markers, such as scaffold data, maximum unique matches (perfectly conserved sequence fragments of maximal length), genes, and even assembly contigs. Those that use markers in the sense of highly conserved sequence fragments define markers through a variety of criteria, such as Bayesian statistics in Pecan [17] or sequence similarity iterated through a refinement pipeline in ProgressiveMauve [8].

Considering markers individually, means working at the base pair level of the sequence. We are not much interested in the individual movement and reconstruction of markers, as they form only a small part of the genome. We have to think of them rather as in their original context and use them to characterize what is in between such markers. Taken as a collection of markers, their varying positioning across genomes shows that major events took place—trying to explain the event chain that lead supports the present genomic architecture being the main challenge in phylogenetic studies. A comprehensive review on how non-coding nuclear DNA markers of various origins (internally and externally transcribed spacers (ITS/ETS), transposable elements (TE) microsatellites, etc.) are used for phylogenetic reconstruction, is provided in [30].

In order to use them in alignments or to study synteny, markers have to be formalized. *Syntenic blocks* are one such formal context.

2.2 Computational Approaches

Based on the notions from the previous section, there are four distinct concepts necessary to effective synteny mining, that will be discussed in details in the following sections:

1. **Homology tools** that identify short sequences of local similarity in the context of pairwise genome comparison.

2. **Databases** used in the analysis to be presented and that served as source for genomes and as resource for homology/orthology.

3. **Genomic alignment** tools that extend comparative studies beyond looking for local, short sequences of similarity, offering a broader perspective on the possible evolutionary scenarios.

4. **Common intervals** as a formalized frame for working with markers towards discovering genomic patterns that encompass large scale evolutionary events.

2.2.1 Mining For Homologs

Tools that are widely used for the detection of various types of homologs, are sequence alignment tools that provide short local matches, as a result of comparing a query sequence to several possible matching target sequences, part of a database. Such tools that are widely used and accepted in the community are the earlier mentioned BLAST and FASTA [18]—adequate for searching DNA and protein databases and for evaluating statistical significance from randomly shuffled sequences.

2.2.2 Databases

Gene data for running experiments has been downloaded from various sources. Databases used included OrthoDB [28], NCBI [36] and YGOB [5].

2.2.3 Genomic Alignment

Early comparative studies (CS) were carried out at the level of short DNA sequences. This allowed for the prediction of pairwise homologies, limited orthology and paralogy relationships within gene families; however, such comparisons allowed only for the detection of edit events such as insertions, deletions and substitutions, limiting the scope of pairwise local alignments to gene-sized alignments, which are without doubt still very important as they form the basis of most evolutionary studies, for they allow translating functional information between genomes.

However, there is more to evolution than just basic genome edit events. Once that sequencing techniques improved, whole genomes (WG) became available, making it possible to leverage phylogenetic information as CS would be carried out at the genome level instead of only locally. A straightforward result is the incorporation and the possibility to study various evolutionary *mechanisms* (s.a. natural selection, genetic drift and hitchhiking, etc.—all important genetic indicators), reconstruct events (s.a. speciation, duplication, loss, the large scale events like rearrangements, lateral gene transfer and many more), assess more carefully or proof check the type of homologous relationships between related sequences. In [32] Dewey provides more details on WG alignments, issues, purpose and use.

Even though using phylogenetic information improves CS considerably, handling duplications—establishing the correct orthology/paralogy relationships, still remain significantly challenging. The genome level alignment combines local alignment-level similarity with events that are detected at a higher, block-level analysis. Genomic regions identified as part of higher-level abstract features, are related to other such regions that are approximately similar to each other. This approximate similarity reduces the computational effort for large genomes, s.a. the eukaryotic ones. This comes consequently with new challenges, leaving room for creativity—defining sets of parameters that are minimal, general and as little as possible empirical to consolidate evolutionary models to be able to deal with large scale rearrangements. As a result, determining syntenic blocks remains a very hard task, hence the many tools (GRIMM-Synteny [38], ADHoRe [27], Cinteny [37], FISH [6], OrthoCluster [40], etc. to name a few—more detailed description of their main features is given in chapter 3).

Just as most work on defining syntenic blocks focuses on two genomes at a time, so is whole-genome alignment usually done pairwise. Biologists have long known that multiway comparisons provide more information than pairwise comparisons, especially multiway comparisons within a phylogenetic context.

Multiple sequence alignment (MSA) builds upon local pairwise alignment and makes

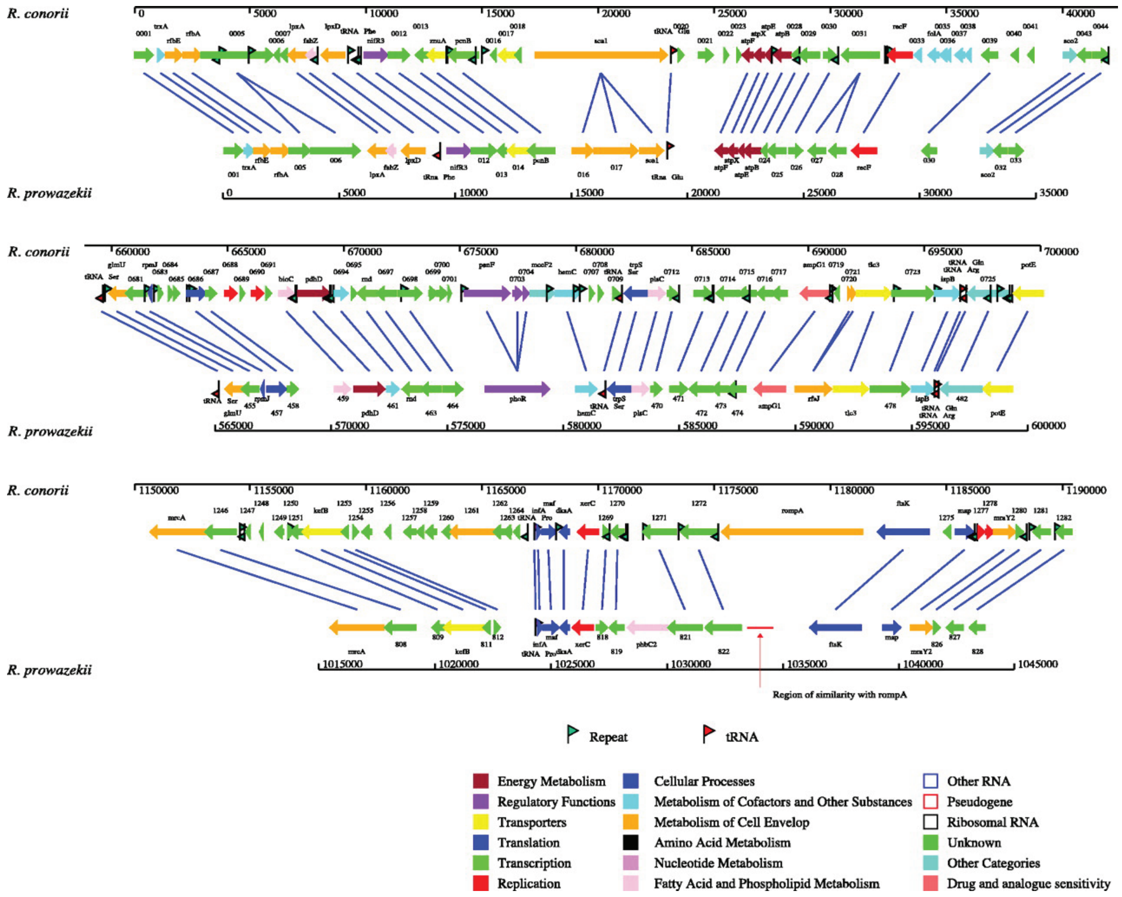


Figure 2.2 – *Example use of sequence alignment in comparative studies. Illustration for gene content alteration in the intergenic regions of R. prowazekii. Different types of genome alteration are presented: split genes in R. prowazekii (top) and in R. conorii (middle), partially conserved gene in R. prowazekii (bottom) through the pairwise alignment of* R. conorii *with* R. prowazekii *[66].*

use of phylogenetic information, enhancing the quality of the final alignment. However, in the same time it comes with new problems: finding good markers that are present in all, or almost all, genomes; choosing or inferring a number of parameters related to attributes difficult to measure, such as the level of evolutionary divergence among the genomes or the quality of the genome sequences used; assigning one-to-one correspondences among similar blocks so as to minimize the number of evolutionary events needed to explain the architecture of the modern genomes; whether to insist on the transitivity of relationships such as homology and orthology (among markers, among genes, among syntenic blocks, etc.); and many others.

Figure 2.2 illustrates how sequence alignment is used for annotation and functional transfer—each of the three pairwise alignments between the genomes of R. conoriiand R. prowazekii, depict an event that altered the sequence of R. prowazekii. In the top example, the region around position 20000 is split in the R. prowazekiias opposed to how it shows in R. conorii. The second alteration shown in the middle alignment, is the mirror example for the previous case, while the bottom alignment shows the decay for the *rompA* gene in the genome of R. prowazekii. The arrows that make up the two genomes from the illustration stand for genes, and the conserved adjacency between them is referred to as *collinearity*. Collinearity constitutes a milestone notion, that will be discussed in more detail throughout subsequent chapters, as it played and still does an important role in synteny detection.

What can be efficiently solved for pairwise comparisons, remains intractable for more than two genomes. As a direct consequence, all methods attempt to reduce MSA to a series of pairwise comparisons, hence the progressive pairwise alignments that either use phylogenetic trees as a guide for the merging process or consider a reference genome which is then used on the position of an ancestral genome. It must be mentioned though, that at this level too, handling complex events is still far from being solved—besides point mutations and indels, e.g. determining the right orthologous pair has not yet become a basic task that would yield accurate and reliable results without the need to curate them. For this reason, most of the multi wise comparisons are most successful for closely related strains of the same species, larger taxonomic groups with a short history (e.g. vertebrates).

2.2.4 Common Intervals

The study of rearrangements led to the definition of *common intervals*. The concept is used to compute the evolutionary distance between the species in question, as given by the number or reversals necessary to transform one genome into another one; common intervals stand for conserved regions of a chromosome within which the same set of genes can be observed, albeit not necessarily in the same order. Taking the mathematical model introduced by Uno in [39], Heber et al. in [35] places the concept into a biological context, pointing out ways to detect functional associations between genes. If the genomes are modeled as permutations of genes, finding co-occurring genes translates into finding common intervals—all on the premises that genes occurring in different genomes, but existing in eachother's neighborhood, tend to encode proteins that interact at the functional level [67], [68].

The formal definition for common intervals, as presented in [35], is as follows: given a set $S = \{1, \dots, n\}$ and the signed permutation π of its elements: denote by $\pi[i] = j$ the ith element of π. For $x, y \in S, x \leq y, [x, y] = \{x, x+1, \dots, y\} \subseteq S$ and $\pi[[x, y]] = \{\pi(i) | i \in [x, y]\}$ is called an *interval* of π. Let $\Pi = (\pi_1, \dots, \pi_k)$ be a family of k signed permutations of S. Assuming w.l.o.g. that $\pi_1 = id_n = (1, \dots, n)$. A k-tuple $c = ([l_1, u_1], \dots, [l_k, u_k])$ with $1 \leq l_j < u_j \leq n$ for all $1 \leq j \leq k$ is called a *common interval* of Π if and only if

$$\pi_1([l_1, u_1]) = \pi_2([l_2, u_2]) = \ldots = \pi_k([l_k, u_k]). \tag{2.1}$$

The set of all common intervals of Π is denoted by C_Π—the definition excludes common intervals of size one; it allows for nested intervals.

Example Let $S = \{1, \dots, 9\}$ and $\Pi = (\pi_1, \pi_2, \pi_3)$ with $\pi_1 = id_9$ (the identity permutation for the first 9 elements), $\pi_2 = (9,8,4,5,6,7,1,2,3)$, and $\pi_3 = (1,2,3,8,7,4,5,6,9)$. $C_\Pi = [1,2],[1,3],[1,8],[1,9],[2,3],[4,5],[4,6],[4,7],[4,8],[4,9],[5,6]$.

Given the nature of the data that common intervals are built on, the actual nucleotide sequences of these genes as well as the precise locations on the genome are not longer taken into account. The definition is given in terms of families of non-duplicated genes (or other families of unique sequences) and their ordering; such a formal definition captures many of the properties informally associated in the literature with syntenic blocks.

Permutations, and the reversal operation, are useful tools for estimating evolutionary distances in comparative studies. In [3], Bergeron et al. projects the reversal distance problem into a solution for sorting sets of similar genes, with different orientation and ordering. If the genome is seen in terms of sets of ordered genes per, organized on chromosomes, where the genes have the their orientation associated to a $+/-$ sign; as the gene content of a chromosome is shuffled and pruned or enriched by evolutionary events, comparing two sets that belong to different species, that are located on corresponding chromosomes, yields a signed permutation, which is then used to estimate the evolutionary distance between the two species.

Selective pressure will impose a strong conservation on the gene order and content of the genomic material of certain regions. However, for practical cases when comparing distantly related species, through shuffling and various evolutionary events that act upon the genomes, this framework becomes less feasible for most of the relevant cases. Therefore, the *approximate* version of the concept is developed based on distance metrics—a maximum distance between the consensus gene set and the allowed approximate sets is defined, in order to be able to cluster intervals with varying content. An efficient algorithm that implements this extension is presented in [13].

In [33], Dörr introduces a model that is not based any more on an *a priori* defined family of homologous genes, that is the case in the previous work; instead it is inferred from gene sequence level similarity, for the dataset in question. The advantages and features of the family-free comparative studies are presented in detail in [46]; a particularly important feature is that it fends off issues like the fact that gene families are most of the time inferred in an unsupervised manner; consequently such inference methods can yield false biological gene families, which at their turn will generate incorrect results for subsequent analyses. This latest approach relies only on gene similarity measures, and makes possible the inference of several features such as conserved structures, rearrangements, ancestral genome reconstruction, gene family prediction instead of use and many more.

The combination of this latest gene family free approach with the one of common intervals is introduced in [34], under the name of *context intervals*. As the nature of the combined heuristics suggests, the new concept has promising potential to serve as basis for defining syntenic blocks, that can additionally be organized into a hierarchical structure.

There are of course limitations to the generic form of common intervals. A straightforward example is the impossibility to handle a variable gene content, as the reference set and permutation is the identity one. In order to deal with e.g. duplication and therefore the induced marker content change, the set of markers of a generic common interval would become a *multiset* instead of set, while the reference permutation would change as well.

Then the latest form of *common intervals*, presented in [65], introduce the concept of *weak common intervals*. This concept addresses the shortcomings and the limitations previously mentioned, by recasting the problem of common intervals on *sequence profiles*.

3 Synteny: Evolution From Concept To Implementation

This chapter provides an overview of the concepts, tools and heuristics developed to date in what concerns synteny. The chapter is structured into the following parts:

- *The concept of synteny*: early notions and studies introducing sinteny related concepts.
- *Syntenic blocks (SBs), markers and their inference*: a few remarks on the underlying structures that serves as a *scaffold* for building syntenic blocks.
- *(Pairwise) Synteny mining tools and their underlying heuristics*: this part presents the tools that implement notions on synteny used in pairwise studies.
- *State-of-the-art in synteny mining in the context of multiple genome comparison*: presenting the most recent tools that were used as well in the experimental part of the thesis, and that are capable of handling multiple genomes in a comparative study.

3.1 The Concept Of Synteny

The first mention of synteny as it is understood today was in 1971 in an article of Renwick [23] on human chromosome mapping, in the context of linkage studies in population genetics. Genetic linkage is observed between two loci (i.e., the *coordinate* of the location of a gene or DNA sequence on a chromosome) when the recombination rates between the specific chromosomal regions are so low, that the loci are inherited together over generations. Renwick introduces the term *synteny* out of necessity to denote *collocation of markers on the same chromosome* while testing for linkage, as opposed to the clear cases where related loci are relocated onto different chromosomes. Thus, in theory, all linked loci are syntenic, but not all syntenic loci are necessarily linked. According to this definition, loci found on the same chromosome are syntenic, regardless of whether this relationship can be established by experimental methods such as DNA sequencing/assembly, genome walking, physical localization or hap-mapping ([43]). In this context, it makes sense to look at synteny in multi-chromosomal organisms e.g. eukaryotes, as opposed to e.g. bacterial genomes where all genes are according to this definition, in synteny.

In 1984 Nadeau and Taylor extends the so far existing notion on synteny in [16], into a concept for two or more pairs of homologous genes occupying the same chromosomal segment, where homologous loci are based on similarity of function of the products of the corresponding genes. Their informal definition of *syntenic segments*, is based on conservation of sequence. They carefully distinguished synteny from such conserved segments, by calling for conservation of function, rather than of sequence. In this same work they gave a list of features viewed as supporting inclusion of markers in an SB, a list that includes conserved orientation, conserved adjacency, and conserved position of homologous markers associated with the corresponding mapped chromosomes, a collection of features that loosely defines what is more commonly called today *collinearity*. This work became the most cited reference by researchers concerned with synteny.

3.2 Markers, Syntenic Blocks And Genomic Alignment

Identifying SBs and aligning whole genomes both rely on identifying markers. SB construction uses subsets from the set of markers: if a sufficiently dense region is identified in most of the genomes, those regions can be viewed as SBs. It is defined both through families of homologous markers and through placement within the genome. Homology is inferred from sequence level similarity, therefore the initial sources of homologous markers for a synteny study relies on (local) alignment tool produced regions, which are inferred from the sequence level of the genome. Representative tools in this context, which will not all be presented in detail, being Mauve [47], MUMmer [44], Multi-LAGAN [48], AVID [49] or WABA [50] to just name a few. The mentioned tools were primarily developed for local sequence alignment, however the byproducts of the alignment process can serve as basic units for synteny mining.

Identifying SBs, in addition to prior knowledge of homologies, requires the examination of rearrangement and duplication events, which disperse the members of a homologous family throughout the genome. (Conversely, of course, producing SBs makes direct statements about the evolutionary history of the genomes by ruling out some of the possible scenarios.) As a result, in principle, the identification of SBs should proceed from homologies (which have little direct dependence on location) rather than from orthologies inferred without regard to location. Computing gene clusters, for instance, is best done based on families of homologous genes instead of relations derived from orthologous groups [13].

More recent work has typically used conservation of sequence rather than conservation of function, but has also made use of orthology. As inferred homologies lose their symmetric and transitive properties in practice, orthology is often preferred, at least for pairwise synteny, for it simplifies the task through its earlier mentioned features. Some synteny finders simply transform orthologous relationships into bijections, in spite of the fact that orthology is a many-to-many relation.) When moving from pairwise to multiway syntenies, orthologies become problematic: the more diverse the group of genomes, the more difficult it becomes to identify orthologies. In consequence, therefore, synteny tools rely on both homology and orthology, viewed largely as different degrees of sequence similarity, with orthology as a dominant preference presumably because it is viewed as a stronger indicator of conserved function than homology.

Most SB finders use genes as markers; for a fixed value of k, a *de Bruijn* graph is defined on k*-mers* and path through such a graph represent contiguous sequences. (*de Bruijn* graphs [19] are widely used for genome assembly—see Compeau [7] for an excellent introduction in this context. In such a graph, every k-mer found in the input sequences is represented by an edge connecting two vertices that are the $k-1$ prefix and $k-1$ suffix of the k-mer. Thus a path of j edges through such a graph corresponds to an assembled sequence of length $k+j-1$ formed by ordering j k-mers, with each consecutive pair presenting a perfect overlap of length $k-1$; in particular, an Eulerian path through the graph corresponds to an assembly of all k-mers into a single sequence.) Genomic alignment may use a richer pool of markers, such as scaffold data, maximum unique matches (perfectly conserved sequence fragments of maximal length), genes and even assembly contigs (overlapping DNA fragments that together form a consensus DNA region). Those that use markers in the sense of highly conserved sequence fragments define markers through a variety of criteria, such as Bayesian statistics in Pecan [17] or sequence similarity iterated through a refinement pipeline in ProgressiveMauve [8].

For finding SB on the other hand, k*-mers* are replaced by genes; DRIMM-Synteny uses the concept previously presented on k*-mers*, to define an *a Bruijn* graph on them. In such a graph, every vertex is a gene found in the input sequences. The edges connect contiguous successions of genes in the input genomes. Thus a path of j edges through such a graph corresponds to a SB.

3.2.1 Principled Marker Detection

In the multiple sequence alignment tool Mauve [47] introduced a principled approach to detect and then use markers in sequence alignment. The model integrates both large scale evolutionary event detection (i.e., rearrangements and inversion in conserved regions) and breakpoint analysis, producing a final multiple sequence alignment. The package has been developed for bacterial genome analysis, which present several particularities (i.e., small genome size, significant repetitive regions, etc.). Such a highly repetitive structure can quickly explode the universe of possible homologous associations among multiple genomes, without the right additional data. The strategy adopted by the authors to address this issue, is: (i) identify highly conserved, sufficiently long sequences (anchors) throughout a concatenated multi-chromosomal genome, to constitute an initial boundary for a certain genomic region (called Multiple Maximal Unique Matches—multiMUMs—that are unique matches in at least two genomes; have length k); (ii) each subsequent region that exceeds a certain length—thus is potentially significant, is searched through the refinement process, for (recursively) shorter (than k) additional anchors. This recursive refinement continues until the anchor coverage has reached a sufficient density or the heuristic cannot retrieve any additional anchors.

The core idea supporting the heuristic is to identify so called *Locally Collinear Blocks* (LCBs), which by definition allows for partial matches between subsets of genomes, thus the strict collinearity among all genomes is not required anymore. LCBs are then attributed a weight as a measure to quantify the level of confidence in the rearrangement that generated the LCB. Thus, the maximum weight LCBs are a collection of anchors such that each collinear subset of anchors meets some minimum-weight criteria.

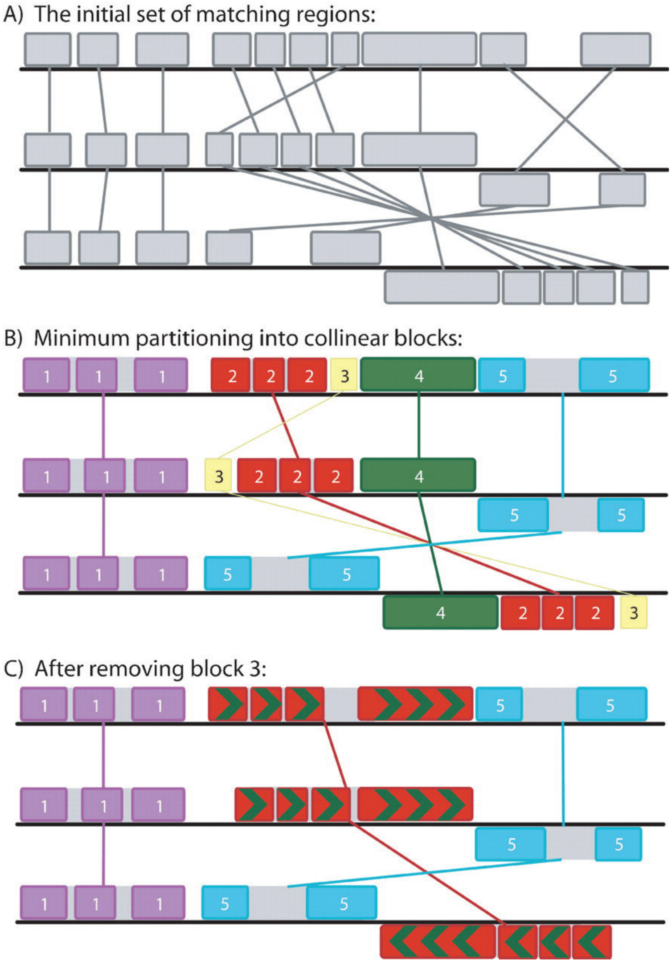

A) The initial set of matching regions:
B) Minimum partitioning into collinear blocks:
1
2
3
4
5
C) After removing block 3:

Figure 3.1 – *Mauve: Representation of greedy breakpoint elimination in three genomes, for LCB detection from multiMUMs. Illustration of the process that identifies collinear blocks of multisMUMs and how removing a low-weight collinear region can eliminate a breakpoint. The resulting collinear sets of anchors delineate the LCBs that are used to guide the remainder of the alignment process. (A) The algorithm begins with the initial set of matching regions (multi-MUMs) represented as connected blocks. Blocks below a genome's center line are inverted relative to the reference sequence. (B) The matches are partitioned into a minimum set of collinear blocks. Each sequence of identically colored blocks represents a collinear set of matching regions. One connecting line is drawn per collinear block. Block 3 (yellow) has a low weight relative to other collinear blocks. (C) As low-weight collinear blocks are removed, adjacent collinear blocks coalesce into a single block, potentially eliminating one or more breakpoints. Gray regions within collinear blocks are targeted by recursive anchoring [47].*

In summary, the main steps of the algorithm are:

1. based on the multiMUMs, a phylogenetic guide tree is computed (and not recomputed anymore throughout the process);
2. a subset of the multiMUMs is selected to form LCBs through breakpoint analysis [51] (illustrated in figure 3.1);
3. the recursive anchor detection is performed;
4. a progressive alignment produces the final result based on the guide tree;

Two significant features of the tool worth highlighting for future reference are that (i) it performs a recursive anchor detection and (ii) the anchors are based on homology.

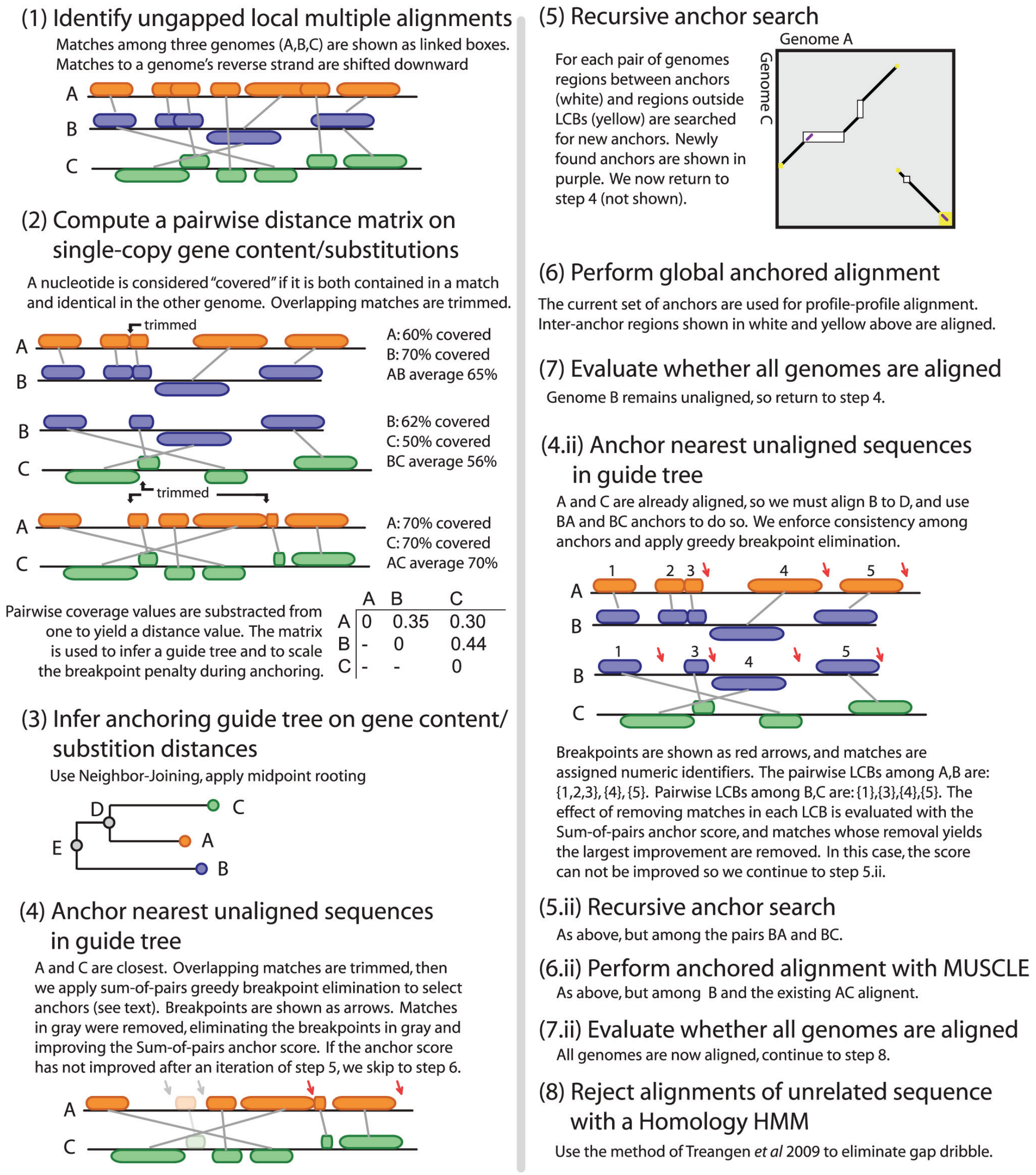
Later on, in [8] Darling *et al.* improves further the concepts implemented in Mauve by including positional homology information. *ProgressiveMauve* improves on the anchor detection by computing them progressively according to a guide tree—which is not the one used for alignment and which is build as shown in figure 3.2. The tool allows for a larger number of genomes to be compared, it increases the quality of the alignment for the regions shared by only a subset of the genomes. This is done through algorithmic innovations related to breakpoint scoring, optimization of the anchor set for the alignment, and a homology HMM model that is used to reject unrelated, erroneous alignments in regions presenting unequal gene content.

The multiMUMs from the previous version, are generalized and replaced by local multiple alignments (LMAs) that will be part of the final whole genome alignment. By including approximate matching, by using a palindromic spaced seed pattern, unique seeds common to at least two genomes are extended for defining ungapped alignments of unique subsequences. The use of such seeds allows for mismatches and relaxes the homologous structures, part of the final alignment, as well as increases the tool's sensitivity to indels and the large scale segmental gain and loss.

(Progressive)Mauve was designed as an alignment tool, not a synteny tool, but it generates a list of homologous, locally collinear regions that can be used as a basis for defining syntenic blocks.

3.3 Early Synteny Tools and Heuristics

Around the same time with the formalization of common intervals, in response to the complete sequence of the human genome becoming available, the Mouse Genome Sequencing Consortium produces a high-quality draft sequence of the mouse genome, and presents the results of the comparative study between the human and the mouse genome in [26]. In the form of such comparative analyses, they propose SBs as sets of adjacent *syntenic segments*



(1) Identify ungapped local multiple alignments

Matches among three genomes (A,B,C) are shown as linked boxes. Matches to a genome's reverse strand are shifted downward

(2) Compute a pairwise distance matrix on single-copy gene content/substitutions

A nucleotide is considered "covered" if it is both contained in a match and identical in the other genome. Overlapping matches are trimmed.

A: 60% covered
B: 70% covered
AB average 65%

B: 62% covered
C: 50% covered
BC average 56%

A: 70% covered
C: 70% covered
AC average 70%

Pairwise coverage values are substracted from one to yield a distance value. The matrix is used to infer a guide tree and to scale the breakpoint penalty during anchoring.

	A	B	C
A	0	0.35	0.30
B	-	0	0.44
C	-	-	0

(3) Infer anchoring guide tree on gene content/ substition distances

Use Neighbor-Joining, apply midpoint rooting

(4) Anchor nearest unaligned sequences in guide tree

A and C are closest. Overlapping matches are trimmed, then we apply sum-of-pairs greedy breakpoint elimination to select anchors (see text). Breakpoints are shown as arrows. Matches in gray were removed, eliminating the breakpoints in gray and improving the Sum-of-pairs anchor score. If the anchor score has not improved after an iteration of step 5, we skip to step 6.

(5) Recursive anchor search

For each pair of genomes regions between anchors (white) and regions outside LCBs (yellow) are searched for new anchors. Newly found anchors are shown in purple. We now return to step 4 (not shown).

(6) Perform global anchored alignment

The current set of anchors are used for profile-profile alignment. Inter-anchor regions shown in white and yellow above are aligned.

(7) Evaluate whether all genomes are aligned

Genome B remains unaligned, so return to step 4.

(4.ii) Anchor nearest unaligned sequences in guide tree

A and C are already aligned, so we must align B to D, and use BA and BC anchors to do so. We enforce consistency among anchors and apply greedy breakpoint elimination.

Breakpoints are shown as red arrows, and matches are assigned numeric identifiers. The pairwise LCBs among A,B are: {1,2,3}, {4}, {5}. Pairwise LCBs among B,C are: {1}, {3}, {4}, {5}. The effect of removing matches in each LCB is evaluated with the Sum-of-pairs anchor score, and matches whose removal yields the largest improvement are removed. In this case, the score can not be improved so we continue to step 5.ii.

(5.ii) Recursive anchor search

As above, but among the pairs BA and BC.

(6.ii) Perform anchored alignment with MUSCLE

As above, but among B and the existing AC alignent.

(7.ii) Evaluate whether all genomes are aligned

All genomes are now aligned, continue to step 8.

(8) Reject alignments of unrelated sequence with a Homology HMM

Use the method of Treangen *et al* 2009 to eliminate gap dribble.

Figure 3.2 – *ProgressiveMauve heuristic. Overview of the alignment algorithm and recursive anchor detection, using three example genomes A, B, and C. The algorithm starts by identifying unique matched among the three genomes (1) and computes the relatedness distance between pairs of these genomes (2), distance that is used for inferring the guide tree. As part of the refinement suite, the anchor set is extended if possible by searching between already identified anchors (5); the final anchor set is used to compute the global alignment between closely related genomes that become further profiles (6-7) [8].*

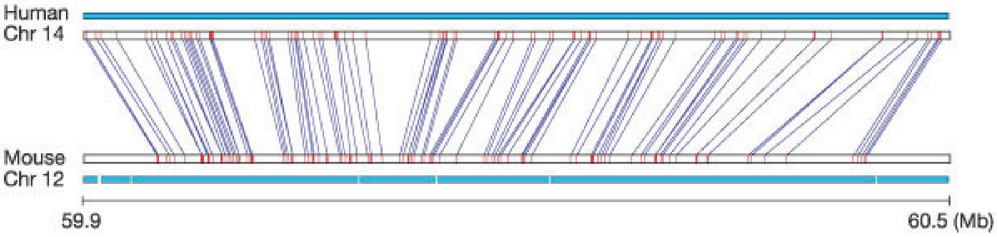


Figure 3.3 – *Syntenic segment between human and mouse. Example illustration of the conserved synteny between human and mouse—courtesy of the Mouse Genome Sequencing Consortium.*

(possibly shuffled in order and orientation) belonging to the same chromosome; a syntenic segment consists of markers arranged in a conserved order, on a single chromosome in both species. In this view, syntenic fragments obey the collinear property, whereas SBs need not do so. An example illustration of syntenic segment as identified by the work of the Consortium is given in figure 3.3.

3.3.1 Statistical Validation Based Tools: FISH And ADHoRe

Unlike in [26], where the focus is on genome assembly and specifically on comparing the human-mouse genome pair, Calabrese *et al.* in [6] addresses the problem of finding *segmental homologs* in highly divergent genomes. The proposed probabilistic model is implemented in FISH (Fast identification and statistical evaluation of segmental homologies) synteny tool. The concept is exploiting linear sequences of homologous features; it is capable of handling duplications but produces only pairs of homologous segments that share multiple homologous features, in a rough collinear order. The objective of the framework is to distinguish blocks that are likely significant and did not appear by chance. For this reason, a null model is defined for individual features that are homologous, yet not part of a segmental homology; the randomness of the structures is based on the computed p-value of the block. Mismatches are quantified through probabilistic values.

The following pipeline is used to generate segmental homologs:

1. establish pairs of homologous features for single chromosome genomes;
2. place the homologous pairs into a matrix;
3. progressively explore overlapping neighborhoods of homologous features, to define *clumps* of closely located homologous points that are set roughly in a diagonal.

The ordering of features belonging to two homologous segments needs not be strictly conserved, some variation being allowed; strandedness is also disregarded.

To generate all the maximum sized clumps representing segmental homologies, a classical dynamic programming (DP) algorithm is used. The illustration of the neighborhood

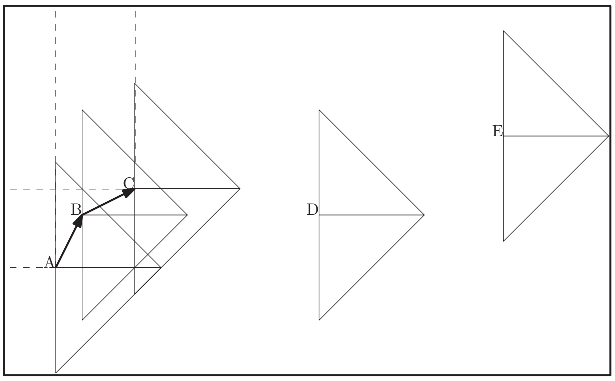


Figure 3.4 – *Illustration of the neighborhood search for segmental homologs. A 3-clump (containing points A, B and C) draws a segmental homology in the space delimited by the matrix of pairwise homologous features. The neighborhood of A contains point B and the neighborhood of B contains point C, but D and E are not part of the neighborhoods of C. Neighborhoods are defined by Manhattan distance. The neighborhood of C is restricted by the top and that of E by the right boundary of the matrix [6].*

concept used to define the clumps, is shown in figure 3.4. The tool is capable of dealing with indels, however the use of DP does not allow for the detection of major inversions—small ones being covered by the neighborhood concept of a clump.

In contrast to previous work, Van de Peer *et al.*, authors of the ADHoRe (Automatic Detection of Homologous Regions) tool [27], chooses to emphasize collinearity and to break larger blocks into smaller, yet statistically significant collinear blocks, such that they maintain the collinear property. The tool was developed for analyzing plant genomes—characterized

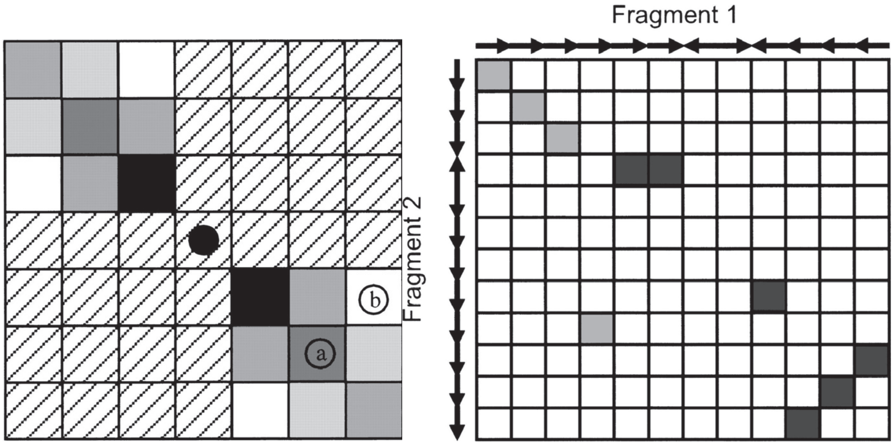


Figure 3.5 – *Illustration of the matrix used by ADHoRe for clustering; crossed squares are masked, the dark squares are at distance 1, while the white ones are at distance 5: (a) Graphical representation of the diagonal pseudo distance (DPD) function. Every rectangle represents a cell of the matrix. The central dot corresponds to an element of a cluster. Example positioning in the matrix, where the DPD defines dot* a *as being in a closer proximity to the central cluster dot, than* (*b*)*; (b) Matrix representation of homologous genes. Arrows indicate the orientation of the genes on the two genomic fragments compared. Homologous genes with the same orientation are colored in gray; homologous genes with an opposite orientation are in black [27].*

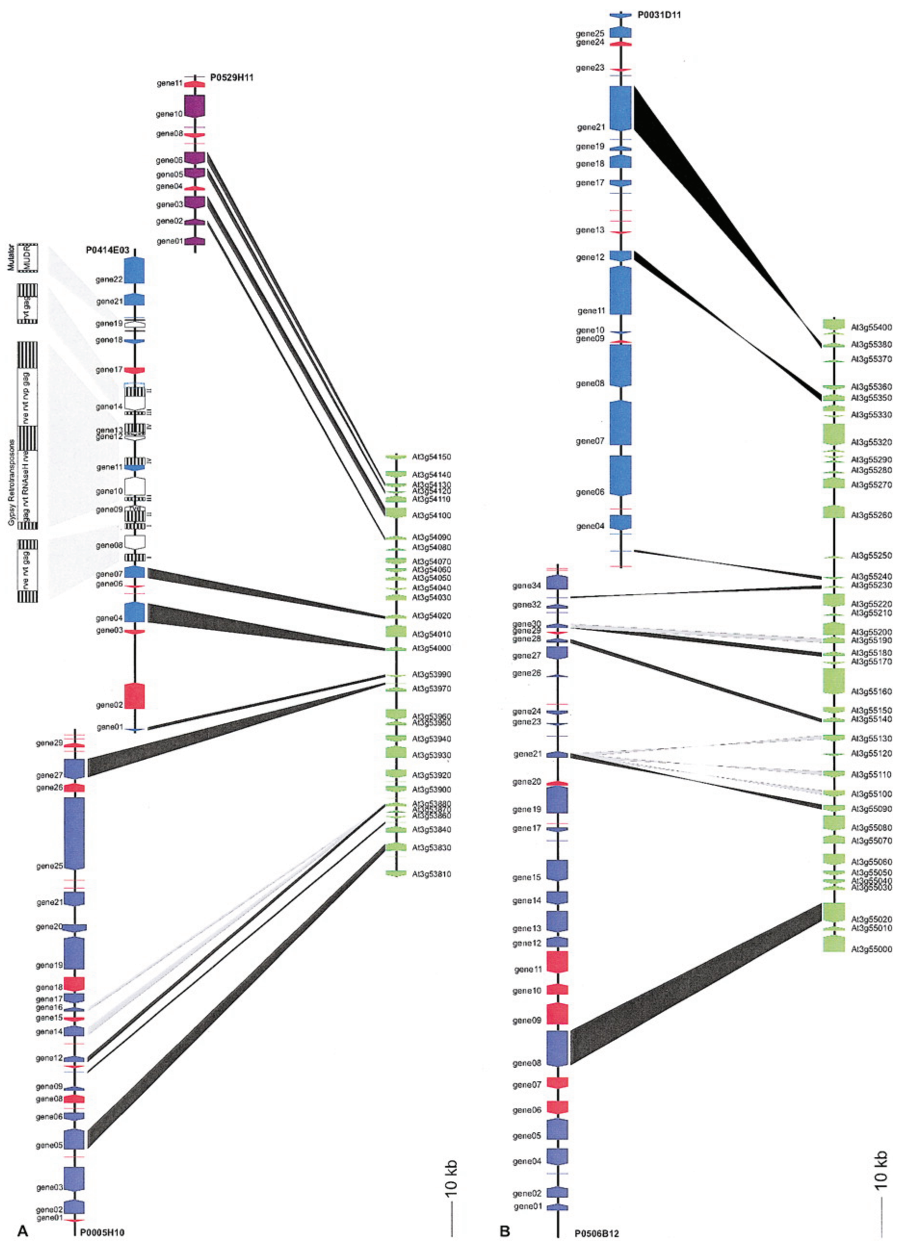


Figure 3.6 – *Examples of collinearity as defined by ADHoRe between rice BAC and A. thaliana (A.t.). Examples of collinear regions found between overlapping rice BACs and segments of the A.t. genome. Two (figure A and B) collinear segment examples between rice BACs and part of the A.t. chromosome 3. Arrows indicate genes present on the genomic segment (black line), black bands connecting A.t. and rice genes indicate anchor points (homologs), whereas gray bands indicate a tandem duplication. Genes probably erroneously predicted in rice are indicated in red [27].*

by at least one but could be as well several whole genome duplications, setting them apart from the genomic architecture of e.g. vertebrates. The concept of collinearity found between a plant genome (A. thaliana) and a rice bacterial artificial chromosome (BAC) is depicted in figure 3.6.

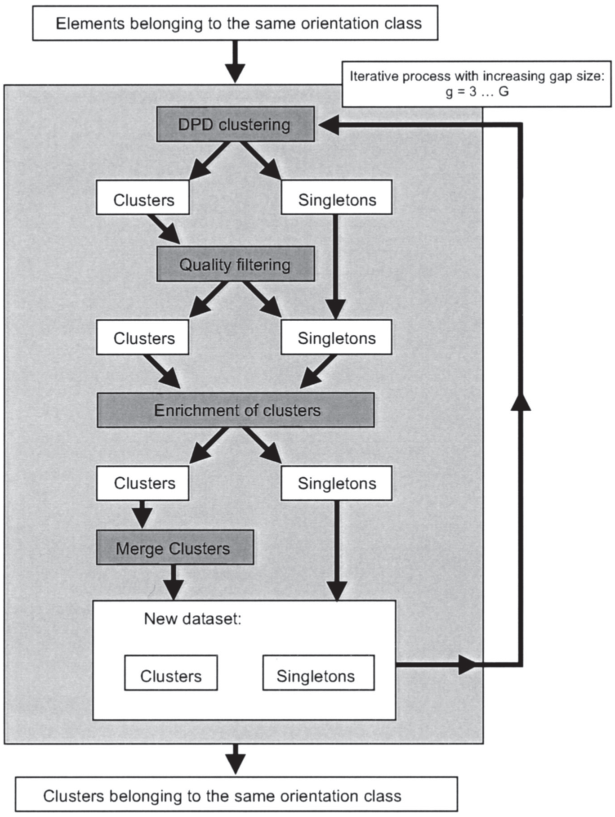

Elements belonging to the same orientation class
Iterative process with increasing gap size: g = 3 ... G
DPD clustering
Clusters
Singletons
Quality filtering
Clusters
Singletons
Enrichment of clusters
Clusters
Singletons
Merge Clusters
New dataset:
Clusters
Singletons
Clusters belonging to the same orientation class

Figure 3.7 – *ADHoRe flowchart for the general heuristic. White boxes represent data items, gray boxes represent routines and arrows indicate the dataflow for the core algorithm [27].*

The main three steps based on which ADHoRe produces SBs are as follows:

1. irrelevant points are eliminated, while all occurrences of tandem duplicates, with the same orientation, are collapsed into the representative tandem duplicate;
2. cluster genes with the same strandedness into blocks, based on the distance measure $d((x_1, y_1)), (x_2, y_2)) = 2max(|y2 - y1|, |x2 - x1|) - min(|y2 - y1|, |x2 - x1|)$, where x and y denote the coordinates of a gene on a chromosome; the matrix representation of the homologous genes is shown in figure 3.5.
3. statistical evaluation of the clusters, by calculating the probability of the collinear fragment with the given gap size vs. the null model given by the sampled and shuffled data set.
4. merging the two strand dependent groups

The heuristic flowchart and the one for detecting collinearity highlight (figures 3.7 and 3.8) the iterative refinement part of the algorithm, responsible for the convergence of parameter values. This approach becomes particularly important for the heuristics which

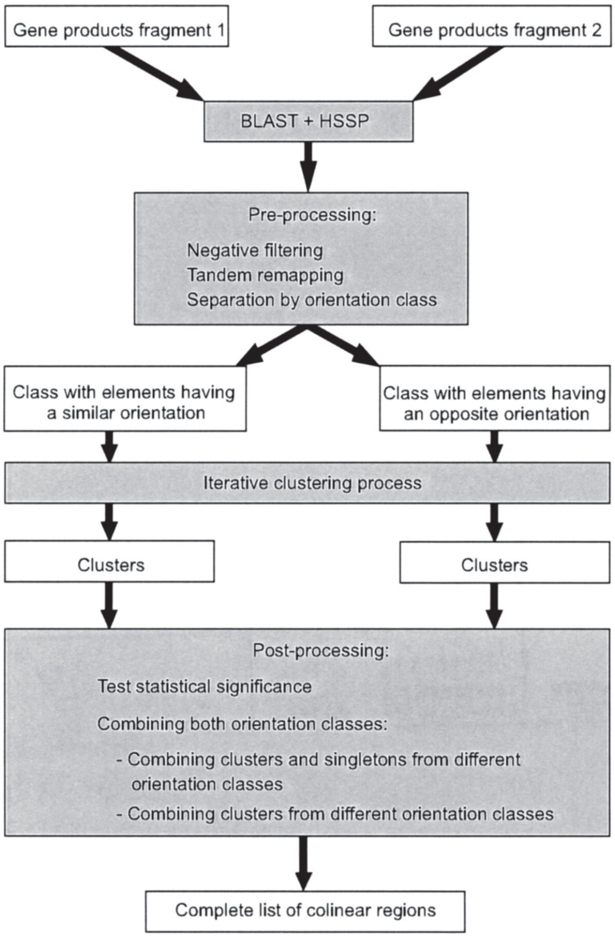

Gene products fragment 1
Gene products fragment 2
BLAST + HSSP
Pre-processing:
Negative filtering
Tandem remapping
Separation by orientation class
Class with elements having a similar orientation
Class with elements having an opposite orientation
Iterative clustering process
Clusters
Clusters
Post-processing:
Test statistical significance
Combining both orientation classes:
- Combining clusters and singletons from different orientation classes
- Combining clusters from different orientation classes
Complete list of colinear regions

Figure 3.8 – *ADHoRe flowchart for defining collinear regions between genomic fragments. White boxes represent data items, gray boxes represent routines and arrows indicate the dataflow for the heuristic [27].*

involve many parameters, for they need fine tuning or a certain convergence. It is designed to work on a pair of genomes and not multiple ones.

3.3.2 Rearrangement Study Based: GRIMM-Synteny And Cinteny

As the complete human and mouse genomes became available at the time, studying genomic rearrangements between human and mouse came into the focus of many research groups. The reference work of Hannenhalli and Pevzner [52] on sorting by inversion, served as a basis for many works attempting to reconstruct genomic rearrangement based evolutionary scenarios. The fundamental question to answer here being: given two permutations of the same set, compute the minimum number of inversion operations necessary to transform one permutation into the other (reference) one.

In this sense, an early and relevant work that actually uses the mathematical concept in comparative genomic studies, is presented by Pevzner in [20]. He introduces a parsimony based approach to define SBs, concept that is implemented in GRIMM-Synteny. Here the genomes are represented as signed permutations (i.e., each *gene* is represented by a (signed) number in a set ordered by the occurrence of the gene along the genome; the sign represents the position of the gene on one of the two strands of the double helix structure of the DNA). The goal is then to compute the minimum number of operations necessary to transform one genome into the reference one. SBs are defined in terms of conserved segments that can be disrupted by internal microrearrangements—such rearrangements the authors found to be far more common than previous studies revealed and that therefore had to be largely ignored in constructing SBs. The lack of precision in previous studies is then attributed to the low resolution of the comparative maps that were used for certain regions of the genome. At this stage, the heuristic computes pairwise comparisons only.

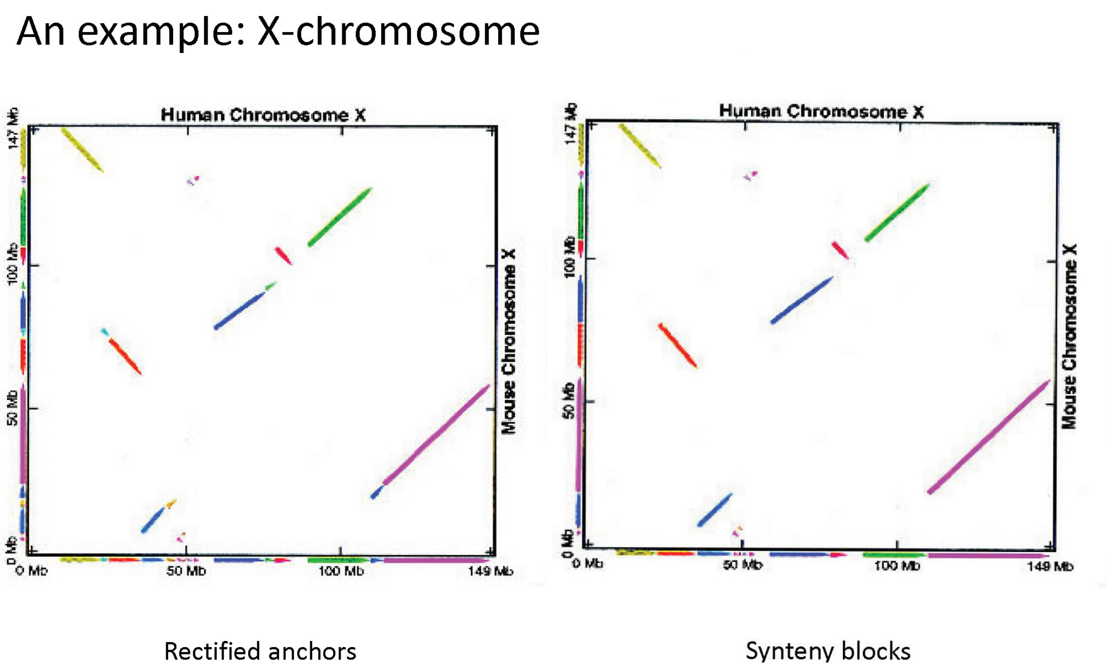
Even though the main focus is to find the most parsimonious transformation, generating SBs from this pipeline boils down to finding the connected components of a multigraph of the alignment that has been constructed based on the homology relation between genes. The latter is established using the Manhattan distance, such that a pair of genes is homologous if they are close to each other up to a certain threshold value. Given two pairs of homologous genes (g_1, g_2) and (g'_1, g'_2), where r_i denotes the position of the gene on the genome, the Manhattan distance between the pairs is $d_M(((g_1, g_2)), (g'_1, g'_2)) = |r'_1 - r_1| + |r'_2 - r_2|$.

Later Bourque *et al.* [4], also working on GRIMM-Synteny, improves the tool further by merging the pairwise comparisons produced by the previous version of GRIMM-Synteny, according to phylogenetic information (tree based), and applies it to the the human-mouse-rat genome trio. This improvement targets the reconstruction of the ancestral mammalian karyotype, by using phylogenetic information. In figure 3.9 we can see the illustration of the pairs of anchors as lines 3.9a, as well as the corresponding SBs 3.9b for the clusters of anchors that are found by the algorithm. The fact that SBs are defined as clusters bigger that a threshold size, implies indirectly that the blocks are allowed to overlap.

Sinha *et al.* [37] proposes a new heuristics for mining SBs that are given by *walks* through a (ternary) search tree (TSTs). The heuristic is implemented in the tool *Cinteny*. The proposed heuristic is designed to deal with several genomes at a time as opposed to pairwise comparisons only. It takes various types of markers as input (in theory, but in practice works on genes). The tree structure holds in each node a gene, while homologous groups are determined by a vertical walk to the leaf level. A horizontal walk will yield the linear order in which each gene appears in the original genome, per chromosome. The "walks" at the leaf level that yield SBs are illustrated in figure 3.10.

The main principles underlying the heuristic are:

1. identify blocks of conserved markers without any disruption of order and orientation (i.e., two signed permutations representing these blocks being identical up to one reversal operation);



An example: X-chromosome

Figure 3.9 – *GRIMM-Synteny generated SBs. The pipeline for generating syntenic blocks (SBs) (a) in the form of clusters of anchors; (b) returned by the heuristic of GRIMM-Synteny [4].*

2. the perfectly collinear blocks are aggregated into larger (non-overlapping) blocks by ignoring smaller blocks resulting from micro-rearrangements.

When inferring phylogenetic information, dealing with *paralogs* remains an issue. Cinteny offers the *user* somewhat arbitrary strategies to deal with this issue: (i) use a paralog which lies within the most conserved region (i.e., the largest SB); (ii) use a random paralog; (iii) ignore all genes which have paralogs. Note that markers that are observed only in some genomes but not all, are automatically filtered out when multiple genomes are used.

An important feature of the algorithm is that it produces blocks of different sizes, thus of various levels of abstraction (i.e., the aggregation level dictates to what size micro rearrangements are deleted). An example of different levels of *aggregation* (coarse-graining) is given in figure 3.11. The aggregation concept is particularly useful when it comes to using SBs for phylogenetic studies (where the large scale events are of interest therefore large blocks are desired and their content is less under the microscope, but more of a black box) or for analyzing targeted genomic regions (where the detailed block content is in focus). This feature introduces however, *user* dependent parameters (e.g., minimum length of SBs), to define the extent to which small divergences are tolerated within extended SBs. While some parameters are user-defined, the heuristic remains rigid when it comes to strandedness and ordering, these features are not subject to the choice of the user.

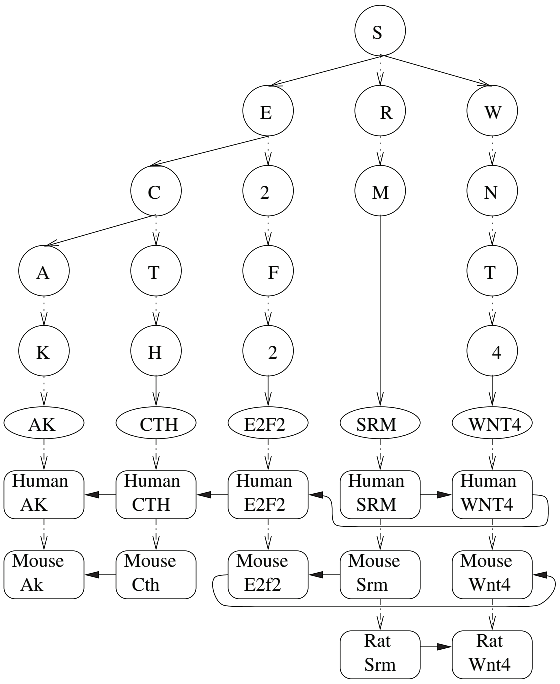


Figure 3.10 – *Cinteny: TST based representation of "genomic walks". Example of a TST used by Cinteny, with several genes from the human, mouse and rat genomes as well as the corresponding walks. The TST is constructed with the gene symbols (nodes of the TST are represented by round circles in the figure, e.g., S, E, etc.) and the leaf nodes (shown as ovals, e.g., AK, CTH, etc.) represent each homologous group (a unique string representing the name of a gene representing that group). The individual genes belonging to each homologous group are connected below the leaf node as meta nodes (shown as rounded rectangles, Human AK, Mouse Ak, Human CTH, etc.). Linear walks are formed by connecting meta nodes based on the order in which the markers appear on a chromosome, as shown by arrows connecting the meta nodes, e.g.,* Mouse Srm ⟶ Mouse E2f2 ⟶ Mouse Wnt4*. The strandedness (orientation) of a gene is stored in the node using an additional variable, and thus the whole genome is formally represented as a signed permutation [37].*

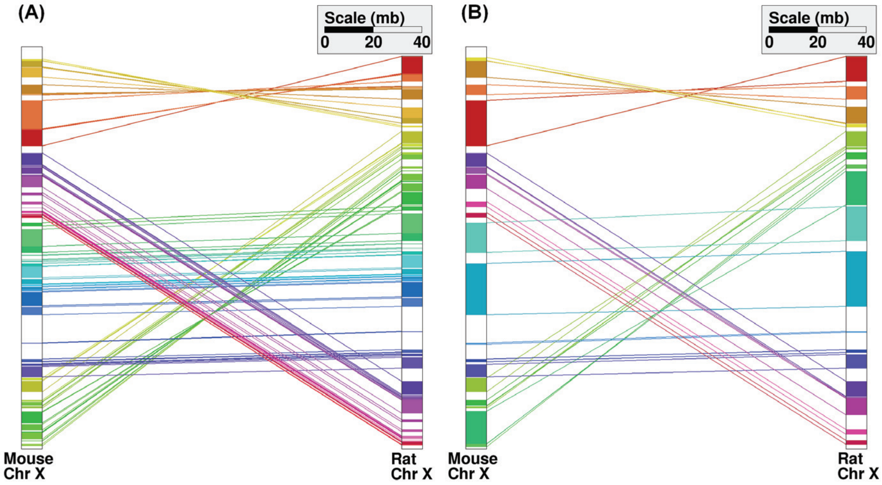


Figure 3.11 – *Cinteny: comparison of the X chromosomes of the mouse and the rat genome (a) before and (b) after aggregation.*

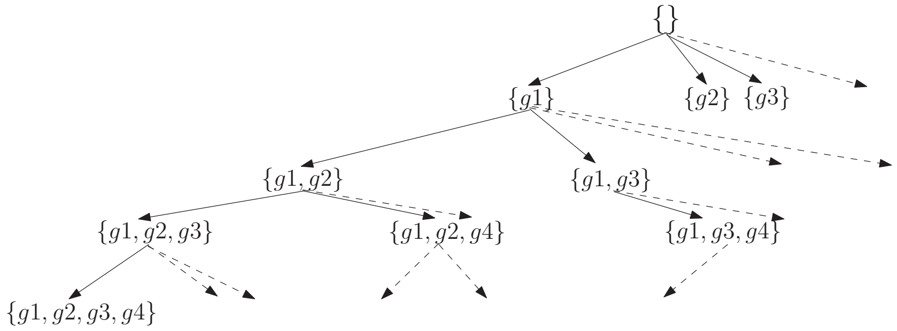


Figure 3.12 – *OrthoCluster: A Set Enumeration Tree. Each node in the tree represents a subset of (orthologous) genes; each node is obtained by adding one gene to its parent. The descendants of node g1 in the tree consists of all subsets having g1, the descendants of node g2 consists of all subsets having g2 but not have g1, etc. [40].*

3.4 State-Of-the-Art Synteny Tools And Heuristics

The problem with tools that use matrices or dot plots to visualize blocks and a Manhattan(-like) distance to cluster related elements found on the diagonal, is that they are not suitable for comparing more than 2 genomes; consequently, handling one-to-many marker relatedness is also hindered; strandedness is not exploited, therefore inversion detection is poor as well. Such features are indispensable for studying large-scale events, as well as for leveraging the extra knowledge that is *reachable* through multiway comparisons.

Modern tools all attempt to handle the loss of collinearity, in recognition of the fact that collinearity (absence of rearrangements) is unlikely to be observed in collections of genomes of any significant size or degree of divergence. Equally important and still challenging is the ability to deal with varying marker (most often gene) content: given reasonably divergent genomes, markers will have been variously lost or acquired over time.

Whole genome sequences become available with a fast pace, while tools remain still fairly limited when it comes to dealing with duplicates, with multiple genome comparison or improving underlying evolutionary models. In this context promising progress has been done, the state-of-the-art being presented in the followings.

Zeng *et al.* developed *Orthocluster* [29], a representative tool both for being able to handle multiple sequence comparison and for proposing a principled and explicit statement for the structure of the expected SB. The tool handles large-scale genomic events such as reciprocal *translocation* (the exchange of genetic material between chromosomes), *transposition* (transfer of genetic material between organisms other than by vertical gene transfer), *indels*, *inversions* and *duplication*. It is based on gene orthology though, which means that the applicability of the tool is limited to closely related organisms.

The core idea for generating SBs is to enumerate all the possible combinations of orthologs that are present in all the genomes to be compared. The subsequent results are generated via a set enumeration tree illustrated in figure 3.12. This process can become very time consuming and computationally demanding, since enumerating all subsets for n genes, requires 2^n steps. For this matter, the authors employ three tree pruning techniques:

1. sliding window: to reduce the count of putative blocks, by limiting the (number of genes) size of the window, to a user defined value;
2. iterative refinement: SBs are generated though extension by an iterative evaluation of the fitness of each gene in the reference block; constraints are given through *user-defined* parameters;
3. pruning by in-between genes: to filter out possible situations where genes present in one genome could disrupt the content wise consistency of a block defined over another genome;

The heuristics that complement the main heuristic, introduce many parameters such as the maximum/minimum number of genes in a block, the allowed number of genes per block without orthologs in other genomes from the block (*in-map*, number of genes without any ortholog counterpart in the block (*out-map*), number of singletons allowed in the block, strandedness awareness, order-preserving.

OrthoCluster is a solid tool, for several features s.a. capability to handle multiple genomes, to deal with various large scale evolutionary events, to generate blocks based on pertinent hypothesis. However, in practice, it does not scale for distantly related genome comparisons; moreover, handling multiple genomes becomes highly resource consuming. The main drawback to it is that the input must be formatted such that only orthologs shared by all the genomes are considered. Even though one-to-many relations are accepted, when the correspondence file defining the orthology relationships contains too many entries, the performance of the tool is radically hindered, to the point of not performing anymore at all.

Cassis [2], also based on orthology relationships, prunes considerably the list of orthologous gene pairs provided as input, eliminating those that disrupt collinearity. The remaining pairs are one-to-one orthologous relationships used to form non-overlapping blocks based on a statistical evaluation of their match to the collinear model. The tool is designed for analyzing whole-genome comparisons, in the light of evolutionary breakpoint regions (EBRs). The algorithm infers the possible location of EBRs though establishing a reference genome, and then comparing a related breakpoint region to the reference genome. Using the position of orthologous genes as markers, it generates a list of ordered and non-overlapping blocks which are used to identify breakpoints; then by means of sequence alignment, (denoted by S_{oA} and S_{oB} in figure 3.13) the EBR coordinates are defined more accurately.

The breakpoint operations are evaluated only for a certain size of genomic span upwards, for the authors consider it to be a safe mean for avoiding false positives, however this limits the tool's capability to deal with duplications and deletions. The whole heuristic pipeline is presented in the work of Lemaitre *et al.* [53]. The segmentation or refinement step proves to

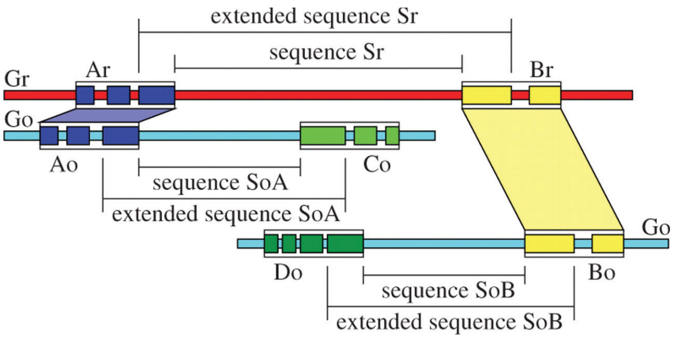


Figure 3.13 – *Cassis: illustration for detecting SBs. Sequence S_r is defined by the boundaries of two consecutive syntenic blocks A_r and B_r on the genome G_r. S_{oA} (S_{oB}) is defined by the boundaries of the orthologous block A_o (B_o) and of the previous/next synteny block (according to the orientation of the blocks) in the genome G_o. To perform the segmentation (second step), the tool considers the extended version of the sequences S_r, S_{oA} and S_{oB} which includes the first/last genes of the SBs [2].*

be harsh on the fragments of S_r which are not covered by neither S_{oA} or S_{oB}–they are deleted from S_r. The correct partitioning of S_r into three sub units representing (i) a match between S_R and S_{oA}; (ii) a match between S_R and S_{oB}; and (iii) a match between S_R, S_{oB} and S_{oA}—that all three are retrieved from Blastz [54] alignments, is established by testing of an iterative manner, for the right position in the breakpoint region, where the (breakage point) sum of squares of the deviations of the data from the model, is minimized.

In contrast to clustering based heuristics, Cassis imposes two structural constraints on SBs: strict collinearity is required among the markers of two blocks, thus blocks do not overlap, and since it uses breakpoints to define start/end points for the blocks, the markers belonging to the two genomes that frame the block are required to be orthologs, this way assuring that they can be aligned in a future step.

Cyntenator [25] uses genes as markers and is based on a progressive alignment of profiles of gene-order data. For the detection of conserved gene orders, thus SBs, the tool represents first the genome with partial order graph (POG) and then aligns the graphs by using dynamic programming (DP). The bit score quantifying the sequence similarity of homologous genes is obtained from BLASTP searches. Further on, all pairwise alignments are generated using Smith-Waterman local alignments [55] (DP). The so produced local alignments are then placed in a POG, where the nodes are either a pair of orthologous genes from the two genomes, or a single gene (evidence for insertion/deletion). The edge between two nodes means that at least in one of the genomes the nodes are consecutive. Before continuing with the progressive alignment of POGs, the intermediary results are refined such that only homologs found in all species are retained for searching for collinear regions shared with the next genome.

The core heuristic improves the earlier work from [24], by making use of phylogenetic information. Thus, the progressive alignment is done based on a phylogenetic tree. The phylogenetic information improves also the use of gap and mismatch penalties, the goal being to penalize more the loss of a homologous pair when the genomes are closely related. A sketch

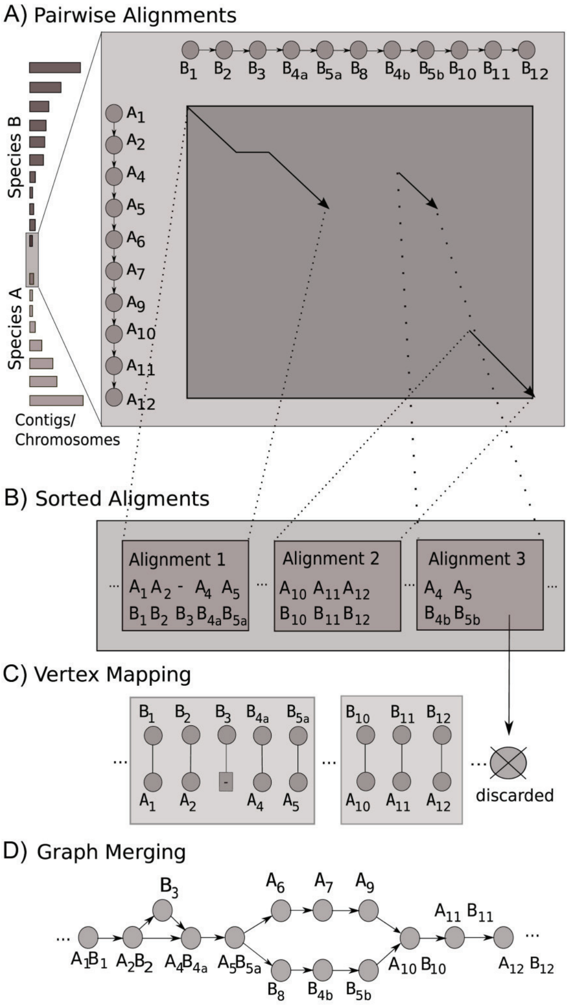


Figure 3.14 – *Cyntenator: pairwise partial gene order alignment of two genomes. During the pairwise alignment, all contiguous sequences are compared. Alignments having a lower score that the user-defined threshold are eliminated and by a greedy approach, genes are assigned to one another, based on the sorted alignments. Finally the chain graph are merged into a POG [25].*

of the core heuristic as implemented in Syntenator, is given in figure 3.14.

The tool in this setup allows limited gene duplication analysis (not internal duplication) and genomic loss. As is the case for most such tools, the blocks identified by Cyntenator are not formally characterized, but indirectly defined through the algorithm.

i-ADHoRe 3.0 [22] also uses genes as markers; it includes heuristics to deal with rearrangement and duplication. This version of the tool, is the result of several iterations of improvement to the original tool, presented earlier, ADHoRe. This version is however capable of aligning multiple sequences. The tool produces profiles of collinear regions based on homology maps

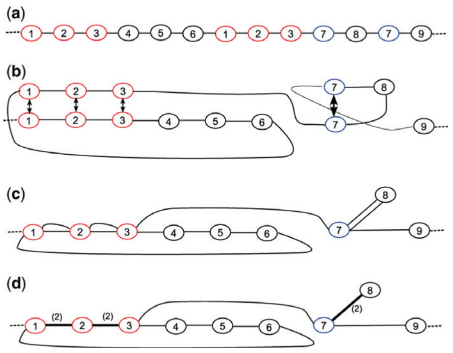


Figure 3.15 – *The genomic decomposition procedure for DRIMM-Synteny. The decomposition of the genome into conserved segments is presented in three steps: (a) the genome is represented as a path of 13 genes—a gene adjacency becomes an edge of the graph; (b) matching the vertices together before collapsing them into the final A-Bruijn graph; (c) construct the A-Bruijn graph for the path, by collapsing the vertices with the same label; (d) a final graph is obtained, that has weighted edges by the number of their occurrences across the genomes [21]. Syntenic blocks correspond to paths through the graph.*

of pairs of genomic regions and uses a greedy, graph-based aligner that employs a network flow heuristic [56] to resolve the pairs of genes in the graph that cannot be aligned. The statistical validation of the final clusters has been improved by also considering the background density of the matrix. During the clustering process, once at least three homologous gene pairs are found, multiple hypothesis testing correction is done via either Bonferroni or False Discovery Rate [58] [57] method. The tool provides three constraint models for generating syntenic blocks: collinear (conserving both order and orientation), cloud (conserving neither order nor orientation, but content) and a sequential mixture of the two.

DRIMM-Synteny (Duplications and Rearrangements In Multiple Mammals) [21], the multiway successor of the pairwise *GRIMM-Synteny*, is, like most synteny tools, based on genes, but follows an entirely different approach, as it is based on de Bruijn graphs. A somewhat different version of de Bruijn graphs, called A-Bruijn graphs [19], is used to account for the different facets to using gene orders rather than overlaps. Depending on the use of the graph (sequence assembly, synteny detection, etc.) there exist several application specific filters and improvements when working with A-Bruijn graphs. The latest tailoring of A-Bruijn graphs for SB generation that is implemented in DRIMM-Synteny, is the sequence modification problem, which basically eliminated small cycles from the graph structure, in order to reveal the true SBs. The cycles that hide the SBs are the so called *bulges*.

In a similar vein to an RNA bulge structure, a bulge in the A-Bruijn graph is defined as a short cycle that contains both forward and reverse edges [19]. SBs on the other hand, are non-branching paths in the graph with multiplicity larger than 1. The core heuristic for constructing an A-Bruijn graph is presented in figure 3.15. SBs are then generated by finding a maximum spanning tree through the graph, then iteratively the edges that disturb the path structure such as short paths that disturb the linear structure of the MS path, e.g. short cycles are eliminated.

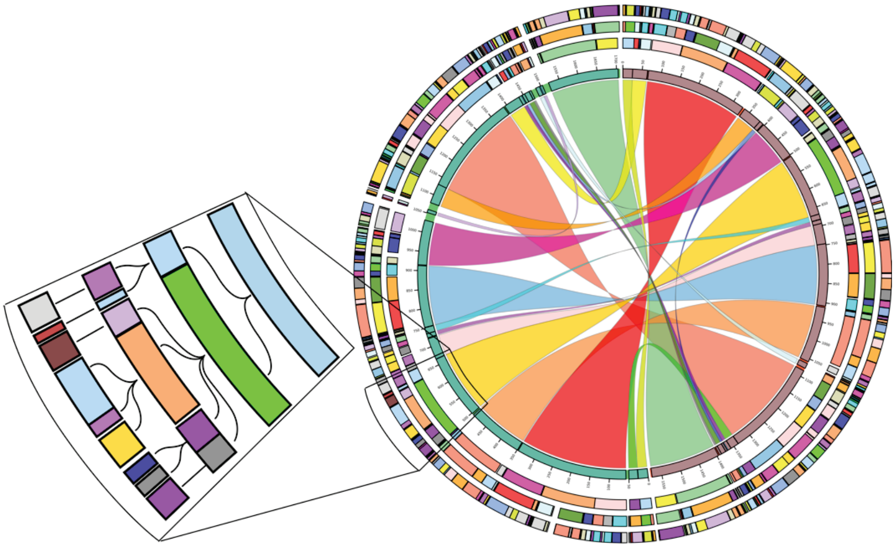


Figure 3.16 – *Sibelia generated decomposition of the genome into pseudo-hierarchical blocks. The decomposition of the genome into conserved segments generated for increasing lengths of the k-mers [15].*

Sibelia [15] follows up on DRIMM-Synteny, in that it is also based on de Bruijn graphs, but, being designed for bacterial genomes, it works directly from sequence data and so builds standard de Bruijn graphs from sequence k-mers. It also adds an iterative refinement procedure that provides a range of granularity for the final blocks. The pipeline is executed for a range of successive, increasing values of k-mer sizes, until the output block is the whole genome. At each iteration, a different set of blocks is generated and is placed as a node into a tree structure, with the root of the tree corresponding to the whole genome. To evaluate the performance of Sibelia against other tools such as Mauve, Multiz [42] and Mugsy [41] (all of them being alignment tools), the genomic coverage is computed from the genome fragments covered by a block, respectively via the F-score between two tools, given as $2(PR)/(P+R)$, where P is the fraction of nucleotides in the blocks generated by one one tool that overlap with blocks from the second tool, and R the opposite overlap count.

The particular feature of the tool is that it proposes a *pseudo-hierarchical* structuring of the blocks that are generated for k-mers of different lengths. However, this is not a true hierarchical organization, for there is no correspondence established between the different levels of the hierarchy, unlike shown in figure 3.16.

3.4.1 Observations and Lessons Learned

Over the past decade, technological development imposed the direction for developing tools for comparative genomics. Initial access to the coding region of the genome provided a narrow insight into the evolutionary history of organisms. However, it stimulated its thorough exploration and thus yielded a *good* understanding of the interdependence between genes and their regulatory networks, the role that certain genes play in pathological cases, as well as a better understanding of relatedness between organisms—based on reliable studies on conserved patterns of the aforementioned genomic regions.

As the first human draft genome becomes available, there was an increasing number of studies that tried to relate the human genome to model organisms that facilitate lab experiments. This entails the mass development of alignment tools that focus on comparing whole genomes at a larger than pairwise scale, thus stimulating the use of phylogenetic data. The milestone at this stage, being the inference of large scale evolutionary events and then the use of rearrangements for working with higher level genomic units than only genes—this includes the early notions of synteny. The technological advance broadened the spectra of challenges in comparative genomics, by making available an increasing amount of (more accurate) data. This lead to a computational bottleneck, which call for formal, structured models for partitioning genomes into meaningful large scale units which can be later on analyzed, in order to address biologically significant problems. This context highlights the role that syntenic blocks play in defining genomic regions of interest for targeted studies.

That blocks generated from the same data by different tools may differ enormously, is mostly due to the lack of a formal definition for syntenic blocks: with no verifiable constraints and no measurable optimality criterion, one cannot meaningfully compare two collections of syntenic blocks for the same data. In part, the lack of such constraints and criteria can be attributed to the very different uses to which syntenic blocks are put. For instance, using syntenic blocks to pinpoint a region of interest in the genomes works best if the blocks are small and highly conserved, whereas using syntenic blocks to study the evolution of the architecture of genomes does better with larger blocks and can tolerate much larger divergence in any given block among the genomes. (Indeed, the larger the evolutionary divergence, the larger and sparser the syntenic blocks should be, to account for the lower number of high-quality markers.)

When large-scale (segmental or whole-genome) duplications are present, multiple instances of the same syntenic block will be found within the same genome, as well as throughout other genomes—that is, syntenic blocks, like genes, can be grouped into families of homologs. Identifying orthologies among the markers or genes is thus intertwined with identifying syntenic blocks—arguing for a simultaneous construction, which can take into account positions, rearrangements, and duplications and losses of markers and of blocks all at once. Thus homology is at the root of any principled definition of syntenic blocks: the process of constructing syntenic blocks is simply the process of extending homologies among markers to homologies among blocks under a suitable model of evolution. OF such a manner, partitioning the genomes into syntenic blocks defines the necessary higher-level homology relationships that relate blocks within and across genomes.

Since all genomes share a common ancestor, every single genome is trivially a syntenic block by itself, albeit with a very low degree of conservation across a collection of genomes. At the other extreme, if we had available a detailed history of all evolutionary events at the sequence level, we could construct syntenic blocks consisting of a single nucleotide position. In a similar vein, two or more adjacent syntenic blocks can be viewed as single, larger syntenic blocks, presumably at the cost of some loss in conservation. In other words, *granularity* is an important attribute and one can construct a hierarchy of decompositions into syntenic blocks, by taking the form of a rooted directed acyclic graph where the trivial decomposition into a single block sits at the root and the equally trivial decomposition into individual nucleotide positions sits at the single leaf. Children of a node in this dag are associated with decompositions of finer granularity than that associated with the node itself. Under some mild constraints, this dag is in fact a lattice (or partially ordered set).

It is important to note that the lattice is determined by constraints resulting from the definition of a syntenic block, but the selection of a particular node in the lattice (a particular decomposition into blocks) is driven by other criteria (such as granularity) and thus determined by the application. (Of all the various tools reviewed here, only Sibelia makes explicit mention of a hierarchy of syntenic blocks.)

A concise overview of the features that the most recent synteny mining tools rely on, is presented in table 3.1. The presence of certain features and the lack of other ones is a slight indicator of the need for a principled and rigorous definition for SBs, that is presented in the next chapter.

Table 3.1 – *Major features or constraints of various synteny tools; presence is denoted by +, absence by -, and options by o.*

	Prog.Mauve	*OrthoCluster*	*Cyntenator*	*i-ADHoRe*	*DRIMM*
Collinearity	-	o	-	o	-
Framed blocks	+	-	-	-	-
Overlapping content	-	+	+	+	-
Selective content	-	+	-	+	+
Across chromosomes	+	+	-	+	o
Duplicated regions	-	+	+	+	+

4 A Formal Definition For Syntenic Blocks

As seen so far, little has been done towards a formal definition for syntenic blocks, towards a common sense concept for synteny in general; nor have developers of tools for finding syntenic blocks given any quantifiable goals. Instead, identifying syntenic blocks has been a matter of application-dependent heuristics, lacking any serious attempt at evaluating the quality of the approaches—something that in any case would have proved difficult in the absence of quality criteria.

Here we propose a fundamental constraint on the makeup of syntenic blocks, based on an evolutionary perspective. We first formalize that constraint for pairwise synteny, then extend it to multiway synteny. We also propose a second constraint, which provides added refinement for bacterial genomes and also helps narrow searches when looking for conserved regions of interest.

Our definitions are made in terms of markers and homology statements among them. Thus we regard each genome as a multiset of markers—a multiset rather than a set, as the same marker may occur more than once in the same genome. Associated with each marker is a set of homology statements relating that marker to its homologs in other genomes or in its own genome; a homology statement is just an unordered pair of markers. Ideally, these homology statements define an equivalence relation on the set of markers; in practice, of course, these statements come from a variety of sources (databases, direct analysis of sequence similarity, etc.) and are unlikely to obey all the requirements of an equivalence relation.

Viewed abstractly, identifying syntenic blocks is a clustering problem: how do we partition the multiset of markers into smaller multisets, so as to maximize the similarity (as attested by multiple homology statements) between some of the smaller multisets, while minimizing their similarity to others? Because our definition rests on homologies rather than orthologies, we expect to find homology statements connecting related syntenic blocks as well as some connecting unrelated syntenic blocks—by and large, the first are more likely to be orthologies, while the second are more likely to be paralogies. Our main constraint, then, is that, in order for two blocks to be homologous syntenic blocks, they must be connected through homology statements and that neither includes markers that, while unconnected in this manner to anything in the other blocks, are connected to markers in unrelated syntenic blocks.

We now formalize our definition for the basic version of syntenic blocks: syntenic blocks for two genomes, in which we restrict each to be a contiguous range of positions within a chromosome.

Definition 1 *We are given two genomes, G_A with a set A of n_A markers and G_B with a set B of n_B markers; the markers of G_A are ordered along the chromosomes, as are the markers of G_B. Let H be a set of pairs of distinct elements of $A \cup B$—the homology statements. We assume that every marker in A and B is part of at least one homology statement.*

Let S_A be a set of contiguous markers on one chromosome of G_A and S_B a set of contiguous markers on one chromosome of G_B. We say that S_A and S_B are homologous syntenic blocks *if and only if, for any marker $x \in S_A$, there exists a marker $y \in S_B$ such that $\{x, y\}$ is a homology statement, and, for any marker $u \in S_B$, there exists a marker $v \in S_A$ such that $\{u, v\}$ is a homology statement.*

We can further require that the two end markers form a conserved *frame*, thereby setting well defined boundaries on the range of positions forming a syntenic block.

Definition 2 *Let S_A and S_B be homologous syntenic blocks as per Def. 1. If the first marker of S_A is a homolog of one of the two end-markers (the first or last marker) of S_B and the last marker of S_A is a homolog of the other end-marker of S_B, we say that S_A and S_B are (homologous)* framed *syntenic blocks.*

Many of the existing tools require that the homology between markers respect the ordering of the markers along the blocks—a property usually referred to as *collinearity*. Because genomes are subject to rearrangements, we do not require collinearity, but we can define it as follows using our notation.

Definition 3 *Let S_A and S_B be two homologous syntenic blocks as per Def. 1. We say that S_A and S_B are* collinear *syntenic blocks if the following condition, stated in the direction from S_A to S_B, holds in both directions: for any markers x and y in S_A with x appearing before y, there exist markers u and v in S_B, with u appearing before v, such that both $\{x, u\}$ and $\{y, v\}$ are homology statements.*

Our requirement that each block be fully contained with a chromosome may require that some evolutionary events, such as translocation, fusion, and fission, all of which can move genomic material between chromosomes, be treated as block-splitting events. For instance, if prior to such an operation, we would have identified regions A and B as homologous syntenic blocks, but the operation moved part of region A, call it A_t (tail) to another chromosome,

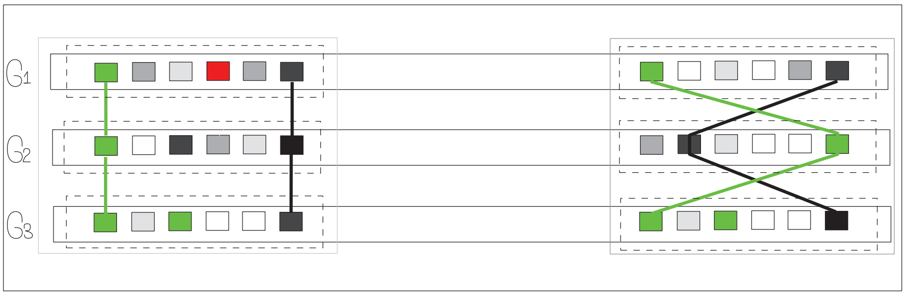


Figure 4.1 – *A cartoon for syntenic blocks among 3 genomes G_1, G_2, and G_3. The horizontal strips correspond to the genomes; small colored boxes denote markers; each syntenic block is framed by a dashed rectangular outline; and homologous syntenic blocks are aligned vertically and enclosed in a thin solid box. Colored lines between horizontal strips connect markers and denote selected homology statements. Shown are an SBF of 3 framed homologous syntenic blocks (on the left) and, using the same homology statements, an SBF of 3 ordinary homologous syntenic blocks (on the right).*

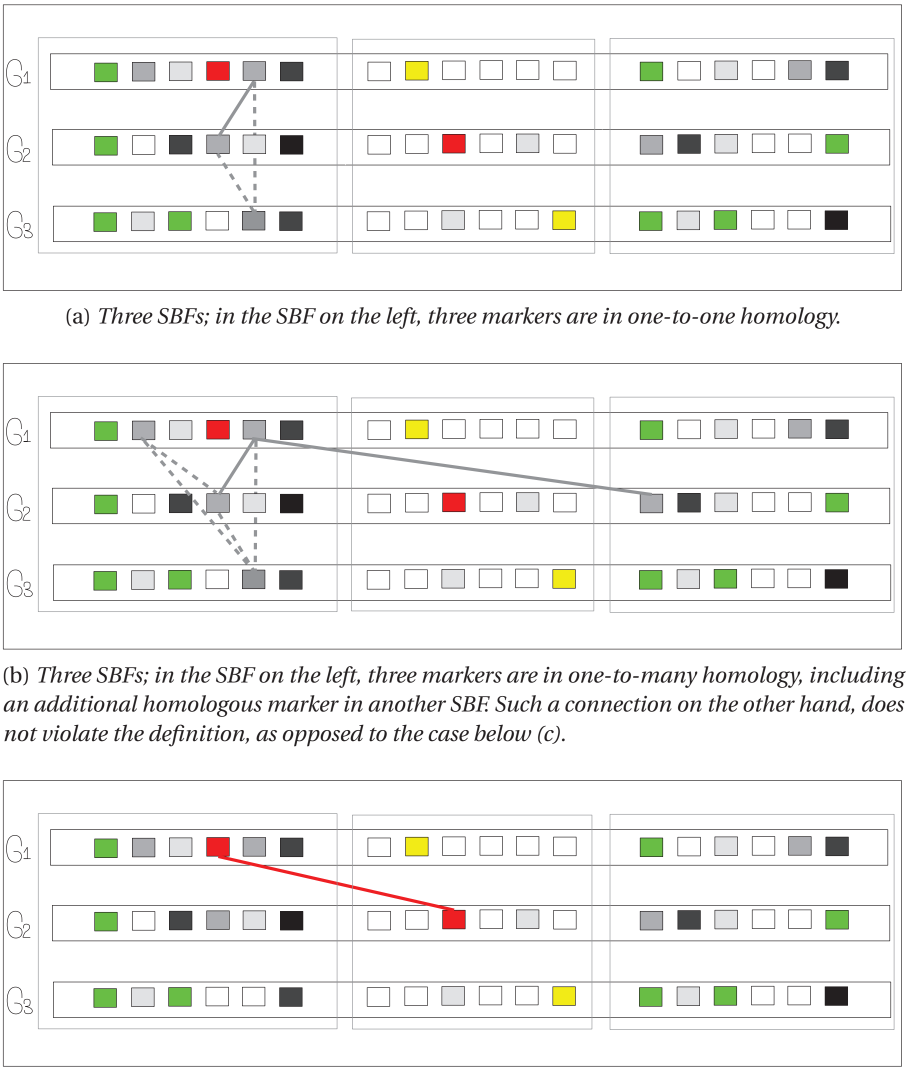
leaving only A_h (head) in the original location, then after the operation we may be unable to associate either of A_h or A_t with B, but we may be able to associate A_h with a first subregion B_h of B and A_t with a second subregion B_t of B, thereby producing two pairs of smaller syntenic blocks.

We extend pairwise synteny to multiway synteny by taking advantage of the transitive nature of true homology: we simply require transitive closure of pairwise relationships.

Definition 4 *We say that blocks $A_1, A_2, \dots, A_k$ are homologous syntenic blocks if and only if, for any i and j, $1 \le i < j \le k$, A_i and A_j are pairwise homologous syntenic blocks.*

This definition is unambiguous whenever our set of homology statements defines an equivalence relation, since this property ensures transitivity. In practice, however, neither transitivity nor symmetry will hold: our set of homology statements will typically be incomplete as not all homologies among markers are detectable and homology defined through sequence similarity (the most common type in practice) need not be symmetric.

The output of a synteny tool is a collection of families of homologous syntenic blocks (SBFs), each family tied together with homology statements. We illustrate our definitions with a few cartoons. Figure 4.1 shows the additional structure granted to SNFs by the frame feature of related syntenic blocks. Figure 4.2 illustrates the main characteristics used in our definitions. The first two cartoons in the figure show SBFs defined through one-to-one (figure 4.2a) and one-to-many (figure 4.2b) homology statements. Homology statements may connect markers in non-homologous syntenic blocks, as long as other homology statements connect these markers to markers in homologous syntenic blocks. The third cartoon (figure 4.2c) gives an example of invalid blocks: the red marker has a homolog in a non-homologous syntenic block, but none in the putative homologous syntenic blocks.



(a) *Three SBFs; in the SBF on the left, three markers are in one-to-one homology.*

(b) *Three SBFs; in the SBF on the left, three markers are in one-to-many homology, including an additional homologous marker in another SBF. Such a connection on the other hand, does not violate the definition, as opposed to the case below (c).*

(c) *Three putative SBFs; as shown, the red marker violates our definition, since it has a homology statement, but that homology connects it to a marker in a different SBF, while there is no homology connecting it to any marker within its own putative SBF.*

Figure 4.2 – *Cartoons illustrating syntenic block structures on three genomes. Alike colored markers form families of homologous units. The solid grey lines represent homology statements of interest, while the dashed ones highlight the existence of alternative relations, which are nonetheless not required to fulfill the conditions of the presented formal synteny concept.*

5 Measures For Syntenic Block Families

Our goal throughout this chapter is to exemplify evaluation and comparison means for the various decomposition schemes of the genomes into syntenic blocks, as produced by 3 state-of-the-art synteny mining tools (DRIMM-Synteny, i-ADHoRe 3.0, Cyntenator). Such evaluations and comparisons have mostly been missing and, when present, have typically been limited to aspects such as coverage of the genome or number of blocks, that might directly impact the quality of the resulting SBFs in terms of the data quality, but neither of which provides much insight into the structure, possible use, and purpose of the SBF.

Our first step was to propose formal constraints that any decomposition into syntenic blocks should satisfy, that is we proposed a formal definition as presented in the previous chapter. These constraints are not likely to be met except in ideal cases, so our second step is to measure compliance with the constraints, which is to say, to measure quality. We therefore assemble a dataset of whole genomes to use in testing various methods; devise specific measurements of compliance with our definitions; and provide other insights and measures regarding the various tools tested.

Two sets of experiments have been performed. A briefer version is presented in [45]. These experiments were performed on a set of eight yeast genomes that were downloaded from the YGOB database. The second set of experiments which provide complementary insights into the structure of the syntenic units as produced by the three tools, were obtained from a subset of six out of the eight genomes, that were retrieved from the NCBI database.

5.1 Experimental Design

5.1.1 The Data

YGOB Dataset

Because we chose to include DRIMM-Synteny in our evaluation, but could not reproduce its authors' results, we decided to use their results directly. Of the datasets used in the DRIMM-

Synteny study, only the yeasts combined complete results from the authors and public availability of the genomic data. We thus used the gene data from the *Yeast Gene Order Browser* [5] (version of April 2009) for the following eight yeast genomes: C. glabrata (c), E. gossypii (g), K. lactis (l), L. thermotolerans (t), S. cerevisiae (s), Z. rouxii (r), K. waltii (w), and S. kluyveri (k). The _genome.tab files were used to retrieve the complete list of genes for each of the organisms and the associated NT.fsa file was processed in order to retrieve the sequences for these genes. Table 5.1 summarizes the characteristics of the data. All four tools require a list of homology statements—orthology statements for OrthoCluster. We used Fasta36 [18], with a cutoff of 10^{-5}, to compile homology statements for each gene, reflecting common practice. We discarded any gene for which no homology statement was produced and, because Cyntenator does not scale well with large gene family sizes, we retained only the 10 best matches (homolog candidates) for each gene. Computational constraints imposed by the tools meant that the number of markers could not be too large; moreover, a number of tools assume that the markers are genes; thus we used genes as markers.

Table 5.1 – *Characteristics of the data from YGOB. The "genes" for K. waltii are often contigs with various functions (ORFs, short complements with intron/exon annotation), which explains their abnormally high number.*

genomes	*genes/genome*	*homolog pairs*
C. glabrata	5211	106291
E. gossypii	4725	104817
K. lactis	5086	113075
L. thermotolerans	5111	94262
S. cerevisiae	6600	140851
Z. rouxii	5006	135707
K. waltii	10825	194234
S. kluyveri	5340	166835

NCBI Dataset

Initial evaluation of the selected tools was done on a subset of the data set mentioned above. The main motivation being the quality of the data thus the choice of data source from NCBI [36], as well as the fact that the experiments with DRIMM-Synteny can be reproduced on a smaller data set. The 6 yeast species used in the preliminary evaluations are:

- C. glabrata_CBS138_uid12376
- E. gossypii_uid10623
- K. lactis_NRRL_Y1140_uid12377
- L. thermotolerans_CBS_6340_uid39575
- S. cerevisiae_uid128 (S. cerevisiae ATCC 204508 S288c)

- Z. rouxii_CBS_732_uid39573

OrthoDB Dataset

Complementary data to the NCBI one was necessary, to provide the orthology relations required for running OrthoCluster. This data was downloaded from OrthoDB [28], release number 6. The protein data was derived from OrthoDB's protein data, version 5.5. The resulting data set was normalized to the list of proteins provided by the NCBI repository—the ones absent from the repository were eliminated.

The similarity scores were retrieved from nucleotide-level comparison by using Fasta36, as well as crosschecked with the orthology relations as provided in OrthoDB.

5.1.2 The Tools

We used the results of the *DRIMM-Synteny* study and ran OrthoCluster, Cyntenator, and i-ADHoRe 3.0 on the yeast dataset. We had chosen DRIMM-Synteny because it represented a very different approach to the problem (using de Bruijn graphs) and chose the other 3 because all were of recent design and well maintained, all support multiway comparisons, and all have reasonably clear statements about their design in the respective original publications.

We ran *Cyntenator* with the parameter setting used by the authors in the original article: gap=0.3, mismatches=0.3, threshold=2, and filter=10000. The final output depends on, in effect, a guide tree (a phylogeny of the 8 species), as it is obtained by running the tool on pairs of intermediate results—the tool ran well on pairs, but not so well on triples, and almost never on larger subsets of genomes. We eventually settled on the pattern described by the tree ((r, (w, (g, (k, (c, s))))), (l, t)).

We ran *i-ADHoRe 3.0* in collinear mode, with the following parameters: gap size=15, cluster gap=35, q value=0.9, probability cutoff=0.001, anchor points=3, gg2 heuristic, no level 2 only, and FDR as multiple hypothesis correction.

OrthoCluster has a limited comparison power, since it accepts only equal gene content, shared across all the query organisms. For a large number of orthologous markers, the tool cannot generate syntenic blocks, due to algorithmic limitations. The compromise suggested by the author is to eliminate the most popular genes that form large gene families, i.e., keep only 1-to-1 correspondences among the six genomes and relax the stringency of synteny blocks, by allowing for a higher percentage of in- and out-map mismatches; however this would mean a drastic loss of information.

Unfortunately, in spite of prompt support from the developers, *OrthoCluster* [29] could not run within reasonable time on our dataset without removing so many genes and homology statements as to invalidate the exercise, so we had to exclude it from the comparison study on the 8 yeast genome sets. (We ran the tool for 2 weeks on a 48-core, 256GB Dell Poweredge 815 without results.)

5.1.3 The Output

The output of all 3 tools is in the form of families of homologous syntenic blocks (SBFs), where each family has at most 8 blocks, each belonging to one of the 8 genomes under comparison. That we get no more than 8 is due to the use of genes as markers: a large fraction of the genes are singletons (have no homolog within their own genome), thereby making it highly unlikely that a particular block structure would be found repeated within the genome. A family has fewer than 8 blocks when no homologous syntenic block in that family can be identified in a particular genome.

Of a similar fashion, the syntenic block families produced by the three tools on the dataset of the 6 genomes are families of at most 6 genomic fragments.

5.2 Results

A main set of results comes from the evaluation of SBFs against our formal definition, means of the four *measures* on the homology statements that we define. A second set of results come from measuring marker content (in terms of the size of the blocks) and reuse, as well as the overlap among blocks. The third insight into SB quality and features is a rather visual representation that gives a feeling for the size diversity of the produces SBFs by the three tools on the genome of *C. glabrata*. This result highlights more the common structure shared by SBFs as produced by the three tools.

5.2.1 Measures For Assessing SBF Robustness

The robustness of SBFs is measured in terms of the quality of the homology statements between its SBs. From this point of view, we devise and present 4 measures that highlight the relations among markers within SBFs with respect to the requirements of definition 1.

- **Block incompleteness**. This first measure is derived directly from the indirect requirement of definition 1, that is *syntenic blocks belonging to the same SBF must not contain markers that relate only to markers outside the respective SBF, while not related at in any way to any of the markers part of the SBF that it belongs to.* Since SBFs as produced by the evaluated tools, do not present an ideal structure, this requirement represents a *pre-screening* that gives a first impression on what is the composition of the SBFs as produced by each of the tools.
- **One-in-one homology**. We are interested in the nature of homology statements that is associated to SBFs. Therefore we analyze this feature in two steps. Unlike the previous

Table 5.2 – *Characteristics of the SBFs generated by the tools.*

	SBFs	*w/o homologs in the SBF*	*content overlap*	*selective content*
DRIMM-Synteny	509	509	0	455
Cyntenator	1106	583	39	0
i-ADHoRe 3.0	8088	278	2	7247

measure, here we assess the ratio of markers within a SBF that have a homologous counterpart in its own SBF. This is a weak statement fot the homology requirement of definition 1.

- **One-in-each homologies**. This is the second step in assessing the quality of the homology statements that link together homologous SBs that form an SBF. Here is of interest the percentage of markers that have homologs in every other SB that is part of the same SBF as the marker in question.

- **Shared similarity**. As the previous measures reveal different features for the SBFs produced by the different tools, this measure complements the previous findings, by highlighting the common behavior of the tested tools. This is illustrated through the SBFs that cover the same genomic area and have highly similar size in terms of gene content, for the genome of C. glabrata.

5.2.2 Evaluating SBFs Against Our Definitions

The primary constraint of definition 1 is unlikely to be satisfied in practice, so we relax the transitivity requirement and measure compliance with the resulting weakened constraint.

With the **block incompleteness** we compute the distribution of the SBFs based on the percentage of markers within each SB, that has no homolog within the SBF, yet it relates to a marker in another SBF. This count is reported in the second column of Table 5.2. Since this measure tolerates failures in transitivity, the number of SBFs not in perfect compliance with our definition may be much larger.

This first measure is an absolute count, although different tools produce different numbers of SBFs; moreover, it counts an SBF as a failure no matter how many markers in that SBF fail the test. To address the first issue, we compute the percentage of “failing" markers in an SBF—that is, markers that have homologs in other SBFs, but none in their own SBF. We use two different base counts for normalization, to reflect fundamental differences between the tools with respect to selective use of markers: the first count is the total number of markers present in the SBF as generated by the tool, denoted $E(X)$, while the second is the total number of markers present in the genome within the coordinates of the generated blocks, denoted $E(X')$. Because DRIMM-Synteny and i-ADHoRe 3.0 eliminate markers from within syntenic blocks (within the coordinates of the block), something that Cyntenator does not do, the values of

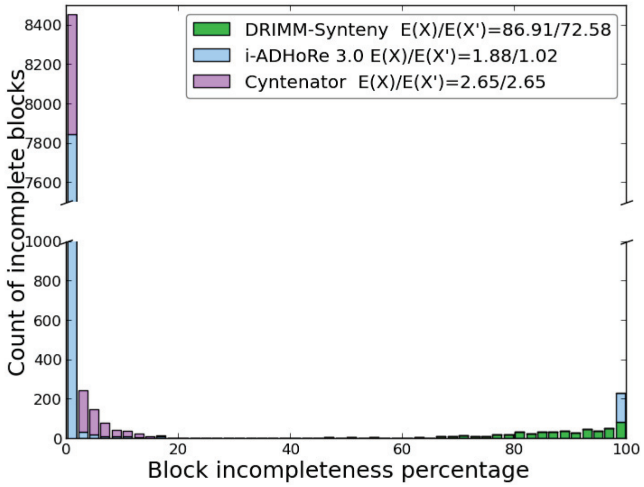


Figure 5.1 – *Histogram showing the percentage of markers from an SBF that do not have any homolog in that SBF. The percentage is computed with respect to the total number of markers present in the SBF as generated by the tool and is supplemented by the $E(X)/E(X')$ ratio.*

$E(X)$ for DRIMM-Synteny and i-ADHoRe 3.0 may be significantly smaller than those of $E(X')$. Figure 5.1 shows that i-ADHoRe 3.0 generates more, and Cyntenator fewer, blocks with a very small fraction of markers lacking any homolog within their own SBF.

The **one-in-one** and **one-in-many** measures are formalized through the following definitions.

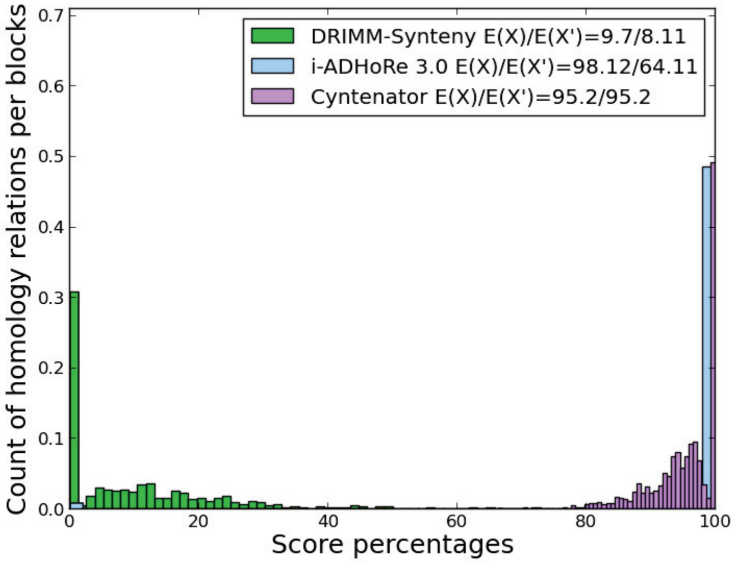
Definition 5 *We define two scores, the first more forgiving than the second.*

Relaxed Scoring *uses a pairwise view of syntenic blocks; for each block from an SBF, it counts the number of markers in that block that have* at least one *homolog within the SBF and normalizes it by the total number of markers present in the SBF.*

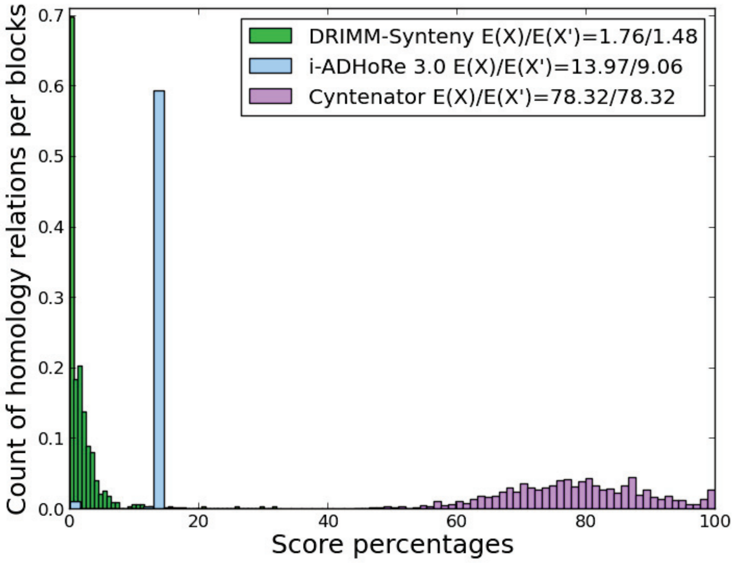
Weighted Scoring *attempts to quantify the deviation from our formal definition; for each block in an SBF, we count the number of markers in that block that have at least one homolog*in each of the other blocks *in the SBF and normalize this result by the number of blocks (minus* 1*) in the SBF and again by the total number of markers present in the SBF.*

A perfect weighted score is 1, yet an SBF of n blocks with a weighted score of $1/(n-1)$ gets a perfect relaxed score. These scores allow us to estimate the robustness of the homology statements, as they show how densely interconnected the syntenic blocks are through their homology statements. A reduction from the first score to the second indicates that the tool has removed markers (to place them in other blocks) that fell within the block—so that the block produced is not contiguous.

Figure 5.2 gives histograms of the two measures for our experiments. Since i-ADHoRe 3.0



(a) relaxed scoring



(b) weighted scoring

Figure 5.2 – *Histograms of the two scores of definition 5, illustrating the refinement over the simple score used in figure 5.1.*

explicitly produces non-contiguous blocks, its two scores predictably differ significantly (by a third). Like i-ADHoRe 3.0, DRIMM-Synteny ignores many markers within a block, but in most cases it does not use them elsewhere—instead, it eliminates them from the list of markers it uses. As a result, its two base counts remain very close, but its two scores are very different.

Cyntenator and DRIMM-Synteny yield similar distributions in both cases, but i-ADHoRe 3.0, which scores nearly perfectly under relaxed scoring, scores poorly under weighted scoring. i-ADHoRe 3.0 does not place much emphasis on multiway homologies: it keeps markers in its blocks even if these markers have just one homology with one other block. In contrast, Cyntenator progressively eliminates markers with few homology statements, therefore yielding blocks with strongly related markers. DRIMM-Synteny has much the same behavior under both scoring schemes, but its score drops by 80% when moving from pairwise to weighted scores, due to its dropping large numbers of markers from its working list. That DRIMM-Synteny scores poorly under both schemes, however, is due to a different set of goals: as stated by the authors, DRIMM-Synteny aims at maximum genome coverage and simply ignores discordant homologies and other conditions that would cause Cyntenator or i-ADHoRe 3.0 to break a block.

The yeast dataset contains several genes and ORFs that overlap. Such overlaps are discarded by DRIMM-Synteny, but not by the other two tools; consequently, Cyntenator and i-ADHoRe 3.0 occasionally output syntenic blocks with overlapping content (see table 5.2).

Although we do not require collinearity, it remains desirable because it greatly simplifies the interpretation of the blocks. Cyntenator makes this a formal constraint; in contrast, most of the blocks produced by DRIMM-Synteny and i-ADHoRe 3.0 are interrupted intervals—between the leftmost marker and the rightmost one, both tools “pick and choose" what to keep in the block. The last column of table 5.2 indicates the number of blocks affected by this selection. The high proportion of blocks with selected content explains in part the good scoring of i-ADHoRe 3.0. In contrast, the very high proportion of such blocks, together with the 100% rate of homology violation, in DRIMM-Synteny confirm the very different aim driving the tool. A related issue is the handling of inter-chromosomal blocks: since genomic recombination of various types can move parts of a conserved region to a different chromosome, one has to decide whether to split the conserved region into two syntenic blocks or to keep it as a single block. Our definition requires a split, since it assumes that each block is contained within a chromosome; DRIMM-Synteny and Cyntenator do the same, but i-ADHoRe 3.0 allows blocks to span multiple chromosomes.

The **shared similarity** measure is based on pairwise comparisons of the tools. A higher dimensional comparison was not possible due to visualization constraints. We chose to emphasize the similarity between pairs of tools. Figure 5.3 gives an overall feel for the results of the study, showing how the blocks from one tool map onto those of another. A very clear mapping pattern can be observed from both Cyntenator and DRIMM-Synteny to a specific, small subset of the blocks generated by i-ADHoRe 3.0, as highlighted by the dark blue section

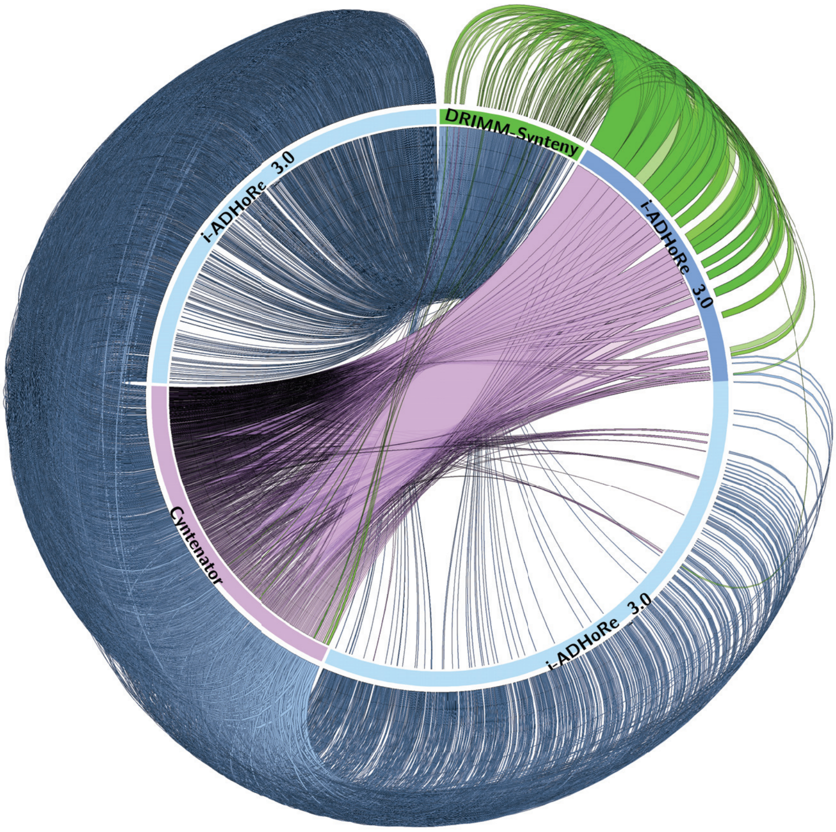


Figure 5.3 – *SBFs defined by Cyntenator (purple), i-ADHoRe 3.0 (blue), and DRIMM-Synteny (green), mapped to each other in terms of gene content. Each link bears the color of the tool, the output of which is mapped through the link onto the outputs of the other tools. There are* 6 *pairwise comparisons between the SBFs produced by the* 3 *tools. The thickness of a link shows the level of similarity, measured by the overlap between the gene content of two SBFs relative to the SBF being mapped. Each sector of the diagram is an ordering by size of all blocks generated by the corresponding tool.*

on the ring of i-ADHoRe 3.0. The number of blocks generated by i-ADHoRe 3.0 is considerably higher than those generated by Cyntenator or DRIMM-Synteny, so the blocks are smaller and the (blue) links thinner. (This kind of mapping also illustrates the lattice concept discussed earlier: the thin links bind smaller blocks to a larger block made of these smaller blocks.)

5.2.3 Quantitative Feature Evaluation For Syntenic Block

Comparing the blocks to each other is difficult, since explicit features of the blocks have not been defined *a priori* for any of the tools. We chose to focus on three features: genome coverage in terms of used markers (the one measure commonly used in the original papers), overlap of blocks for each tool, and agreement among blocks in terms of marker content. We define marker coverage as the ratio of the total number of markers present in the blocks generated by a tool to the total number of markers present in the input within the generated block boundaries. Figure 5.4 illustrates (qualitatively, not quantitatively) how the blocks generated by each tool cover a certain genomic area. (figures 5.3 and 5.4 were generated using Circos [14].) The 3 inner rings correspond to the 3 tools; each genome from our dataset corresponds to a cone in the figure, as indicated by the thin, labeled color indicator enclosing the diagram. Block boundaries are drawn in thin black lines, so that dark areas represent short marker sets, thus small blocks and highly fragmented coverage. Uncovered areas are white.

Our definition does not preclude using overlapping syntenic blocks, since it sets conditions on one SBF at a time. In the lattice of decompositions into SBFs, one may then choose to impose additional conditions to select good blocks. DRIMM-Synteny does not generate overlapping blocks, because it does not reuse markers, whereas Cyntenator and (especially) i-ADHoRe 3.0 do, which allows them to flag regions with ambiguous homologies or complex evolutionary histories. Figure 5.5 illustrates the degree to which markers are reused by Cyntenator and i-ADHoRe 3.0. While Cyntenator just reuses a few markers and not more than twice, i-ADHoRe 3.0 reuses several of them up to ten times, as depicted by the shape of the

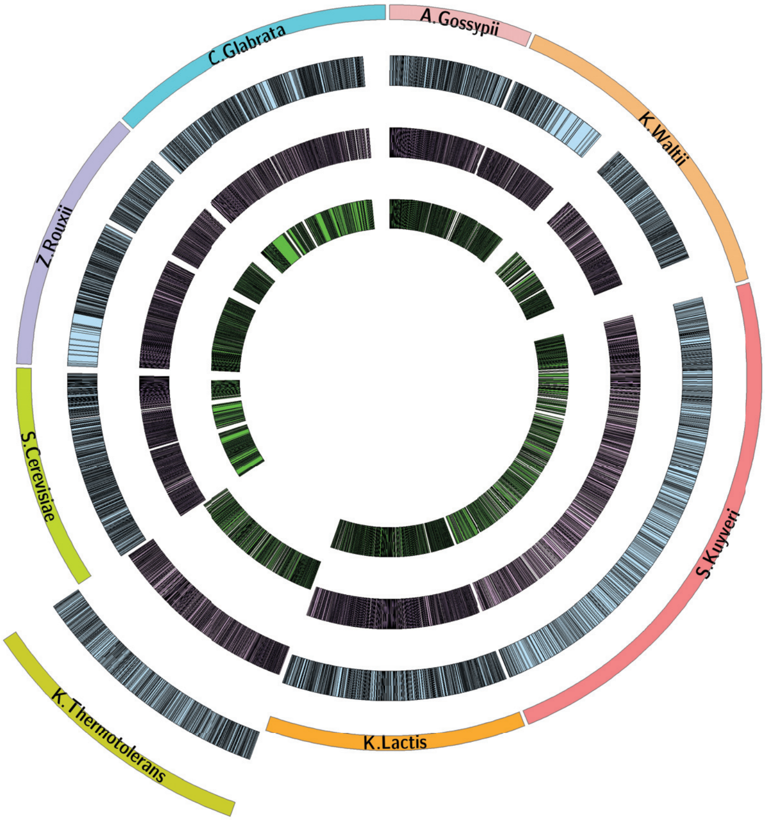


Figure 5.4 – *Syntenic blocks generated by DRIMM-Synteny (inside ring), Cyntenator (middle ring) and i-ADHoRe 3.0 (outside ring). Each ring segment is a yeast genome. Dark regions include many block boundaries—these syntenic blocks have few markers—while white regions have no identified syntenic blocks. Note the many contrasting outcomes from ring to ring: where one tool breaks a region into many small blocks, another produces a single block.*

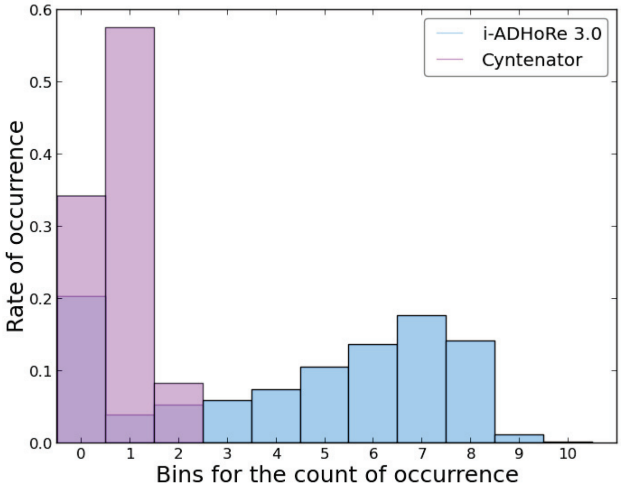


Figure 5.5 – *Histogram of the reuse rate per marker for Cyntenator (violet) and i-ADHoRe 3.0 (blue) on C. Glabrata. The x-axis shows the number of times a marker is reused and the y-axis shows the corresponding rate.*

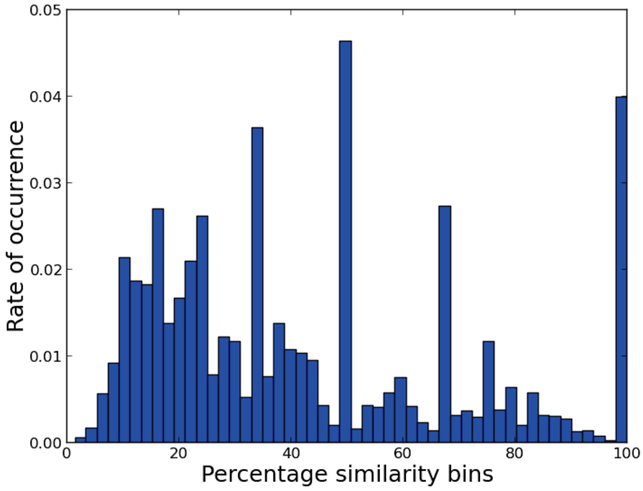


Figure 5.6 – *Distribution of the similarity values for all pairwise comparisons between the SBFs generated by the 3 tools.*

histograms.

We compute block similarity based on marker content: the markers of an SBF as generated by each tool are viewed as a single set and we compute the ratio between the overlap of two such sets relative to each of the sets, thereby yielding an asymmetric measure and six comparisons among the 3 tools. It Illustrates that the distribution is skewed towards small values—most SBFs have a small overlap with other families, thus explaining the types of links seen in figure 5.3: most of the weight of the distribution is in the 10–40% region, corresponding to overlaps with the many small blocks produced by and thus to the thin blue links of figure 5.3, while the same small blocks are also responsible for the large spike at 100%, since many will completely overlap with the larger blocks.

5.2.4 Visual Representation For Syntenic Block Size

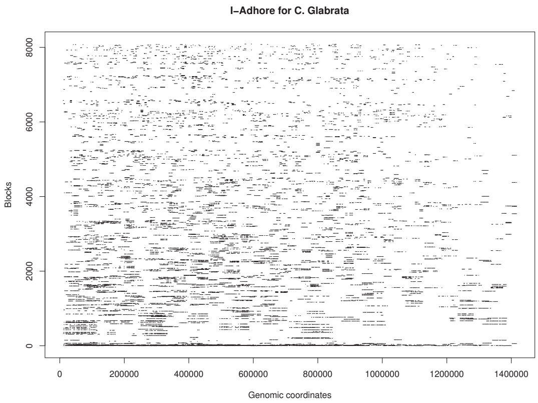
This last experiment is meant to provide a better feel for the different data sets produced by each of the three tools. The results come from experiments that were run on the NCBI dataset. Figure 5.7 gives an overall feel for the size and the number of syntenic blocks as produced by each tool, for the genome of C. glabrata. Each gene part of a syntenic block, is represented by a dot. Contiguous lines are formed in dense marker content genomic regions. i-ADHoRe 3.0 produces the highest number of SBFs, while DRIMM-Synteny and Cyntenator produce considerably less block families. Each block is represented on a different line on the y axes. The projection on the x axes is not equivalent to the gene content overlap of the blocks.

5.2.5 Observations On The 3 Tools

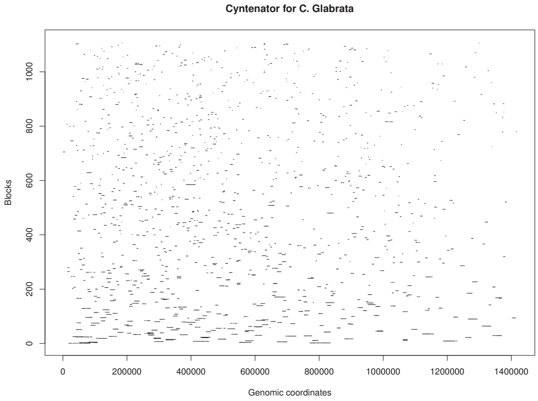
Throughout this chapter, we have presented the potential of definition 1 to reveal features of SBfs as produced by each of the three tools. The 3 tools do not implement any specific rules that would predict the final outcomes, therefore definition 1 is intended to be an impartial measure for the features that is promotes. Nonetheless, we still gained valuable insights into the structure and composition of SBFs, insights that allow us to preferentially choose a tool over another, according to the desired block structure.

The evaluation has not been done with a specific study in mind. Nonetheless, it is possible to summarize the observations and insights presented throughout this chapter. The authors of **DRIMM-Synteny** chose to pursue a maximal genome coverage as general goal. This makes the final SBFs sparse and less interconnected through homologies, which is at the main requirement of definition 1. This feature distinguished the tool from the other 2, in both weighted and relaxed scoring context (figures 5.1, 5.2). The tool is also unique in this comparison, as it does not produce blocks that overlap; this fact is emphasized by its absence from 5.5. Finally, the tool not only produces non-overlapping blocks, but it also produces a small (the smallest) set of blocks; this is illustrated best as a combination of figures 5.7 and 5.3.

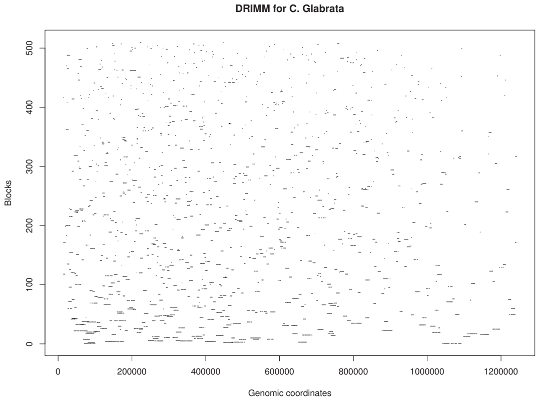
i-ADHoRe 3.0 together with Cyntenator somewhat complement DRIMM-Synteny in the aforementioned aspects. **i-ADHoRe 3.0** is promoted in the context of plant genome analysis (unlike DRIMM-Synteny that has been developed for mammalian genomes mostly). It produces SBFs that cover genomic regions that are more densely populated with genes unlike those of DRIMM-Synteny. The two scores from figure 5.2 give a more precise impression on the strength of homology statements that interconnect related SBs. Two particular features of the tool are emphasized with the presented measures: the homology statements from SBFs are weak and thus, they allow for many options. In consequence, the second feature of the



(a) i-ADHoRe 3.0



(b) Cyntenator



(c) DRIMM-Synteny

Figure 5.7 – *Each of the figures shows the blocks generated by the respective tool over the genome of C. glabrata. Each position on the y axes corresponds to one single block and their projection onto the x axes is not relevant for the marker content overlap between the blocks. Note that the most varied set of block sizes is produced by i-ADHoRe 3.0, while Cyntenator produces slightly fewer blocks than DRIMM-Synteny.*

tool is to produce a large variety and thus number of SBFs—this is depicted through both of figures 5.7 and 5.3.

Cyntenator in contrast to i-ADHoRe 3.0 and DRIMM-Synteny, produces a relatively small number of blocks (much closer to the set produced by DRIMM-Synteny, as shown in table 5.2). Its final SBFs are nonetheless, very robust from the homology statements p.o.v., as it scores relatively high even under the weighted measure. The reuse rate of the markers is very low as well.

6 Practical Design For Syntenic Block Detection

Chapter 3 presents a succinct overview of the notions, heuristics and tools developed for synteny discovery. The overview also provides insight into how research trends are influenced mostly by constraints imposed by technological development. As there is an abundance of whole genome assemblies available, the lack of data is not an impediment in comparative studies anymore. Instead, now researches try to leverage the potential of the non-coding region of the genomes at hand, in order to improve the existing models and our understanding of the encoded evolutionary patterns.

The results revealed in the previous chapters, still represent the tip of the iceberg, when it comes to having a comprehensive overview and understanding of synteny applied in comparative studies. Further steps are necessary to relax and to adapt the model presented in chapter 4, the measures from chapter 5 and the general concepts that have been introduced, such that they scale to practical cases of whole-genome data analysis. The limited understanding that we have about the evolutionary history of the genomes, the existing, simplistic, models are extended and tailored to the goals of specific studies through heuristics (data specific constraints, user-defined parameters, sometimes even an experienced eye of the scientist). The goal here is then, to adapt the formal definition presented in chapter 4 to analyzing whole genome sequences, that is, go beyond the coding region of the genome. This chapter presents some preliminary results in this direction.

The main challenges to be addressed still remain the computational ones. We tackle these by using notions from the theory of common intervals. We keep the model data sensitive, by using a minimum number of user-define parameters, thriving instead to infer characteristics from the data itself; this is highly desired as defining studies that back up choices for constraints (s.a. the size of the block, the number of mismatched markers, the quality of markers, etc.) for syntenic block construction are very few. Nonetheless, study specific features provide some of the necessary constraints for defining syntenic blocks with consistent structure and content.

The main challenges that we seek to address in this part of the work are (i) leveraging the extra knowledge provided by complete genome sequences and (ii) refining the theoretical formal model, with reasonable adjustments for practical use. By leveraging the availability of whole-genome sequences, we mean using more than just signed permutations of gene identifiers as input to the synteny-finding process. Conserved non-coding regions and other well conserved markers of sufficient length can provide a better coverage of the genome.

The two main requirements of definition 1 are inspired from the structure of genomes of high impact in comparative studies (e.g. bacterial genomes would conform the framing requirement). Nonetheless, while SBFs defined over several genomes can be bounded by a conserved frame, there is little chance that the evolutionary events that shuffled the genomic content, would have left intact, the homology relations among all the markers belonging to an SBF. This aspect is addressed though the refinement from (ii) of the definition, where the nature of popular datasets is used as constraints. More insight into each approach is presented in the subsequent sections.

6.1 The Data

Data from the NCBI repository was used; namely the complete genomes for four yeast species: C. glabrata (c), E. gossypii (g), K. lactis (l), S. cerevisiae (s). Our main and strongest requirement for detecting homologous syntenic blocks relies on homology. Therefore, we rely further on, on using homologous markers and not exclusively related genes. Based on sequence similarity, we identify homologous markers between the pairs of genomes of interest. The maximum bit score between bidirectional pairwise comparisons is retained as measure of similarity, which is necessary for further computations. The sequence similarity is determined of a similar fashion to the previous experiments, presented in chapter 5. Table 6.1 summarizes the characteristics of the data. We used Fasta36 [18], with a cutoff of 10^{-1}, to compile homology statements for each marker, reflecting common practice. We discarded any marker for which no homology statement was produced and was shorter than 20bps. Singletons were also discarded from this study. A possible integration of such markers can take place e.g. in a final step of the algorithm, when the syntenic blocks have been already generated.

Table 6.1 – *Characteristics of the data from NCBI. The first numeric column presents the number of genes as available in the YGOB database, and is used to highlight the contrast between this value and the number of possible markers. The second numeric column presents the number of markers that are longer than 20 bps, as identified by ProgressiveMauve for each genome. The last four columns present the number of homolog pairs in total and pairwise between genomes, that serve as input for generating common intervals.*

genomes	*YGOB genes/genome*	*markers/genome*	*homologs/pair*	*E. gossypii*	*K. lactis*	*S. cerevisiae*
C. glabrata	5211	17989	14128	3616	4011	6501
E. gossypii	4725	12322	8225	-	2401	2208
K. lactis	5086	14287	10190	-	-	3778
S. cerevisiae	6600	18542	12487	-	-	-

6.2 Formalization Based Syntenic Block Identification

Different data and computational resources are already available for enriched comparative studies. Integrating pathway and association analysis (e.g., [59] presents the potential of such additional knowledge) further enhances the quality of such studies; through the conservation and interaction patterns between the previously mentioned non-coding regions; though gene onthology data, and many more. However, such sources are already defined on a function related feature basis.

The core idea for this part of the work is to integrate different pieces of information of a progressive, systematic manner. We use a family-free approach to assign markers to related genomic fragments. This is achieved through the concept of *weak common intervals* (wCIs). As briefly introduced in chapter 3, the concept of wCIs is formally and precisely defined and captures many of the properties informally associated in the literature with syntenic blocks. The definition for common intervals is given in terms of families of genes (or other families of unique sequences) and their ordering. It does not take into account precise locations on the genome, nor the actual nucleotide sequences of these genes. In the case of wCIs, of interest to us is the possibility to identify genomic fragments that present also mismatched sub-segments. Moreover, the underlying formalization for common intervals is based on ranked-intervals (explained in [60]), that provide a certain hierarchical organization for the final interval pairs–that is, it provides a basis for granularity specific studies.

6.2.1 Approximate Common Intervals Between Pairs Of Genomes

The markers represent an intermediary dataset–they are used to build wCIs as described in [65]). A few notes on wCIs. The threshold on the number of mismatched markers is set by the δ parameter. An interval pair $([i, j]_S, [k, l]_T)$ in a similarity graph $B(S, T, E)$ has distance d, if the number of nodes with zero degree in the induced subgraph $B([i, j]_S, [k, l]_T, E') = d$. Thus, given $\delta \geq 0$, and two genomes S, T by their marker sets, and the bounding markers $i \leq j$ and $k \leq l$, where $i, j \leq |S|$ and $k, l \leq |T|$: the interval pair $([i, j]_S, [k, l]_T)$ with distance d in a gene similarity graph $B(S, T, E)$ is called $\delta - consimilar$ if $d \leq \delta$ and the nodes i_S, j_S, k_T, l_T are connected in the induced subgraph $B([i, j]_S, [k, l]_T, E')$. The SBFs that we analyze further on are built from such pairs of intervals.

In this context, we examine the nature of the interval-pairs, and observe the followings:

- It is possible to determine a sweet spot (optimal value) for the right threshold value of δ, by choosing the desired ration between mismatched markers and the overall marker content of an SBF (figure 6.1).
- The nature of the dataset is homogeneous; that is, based on the highly similar features that all 6 pairwise comparisons show, results can be generalized without the need of exhaustively testing all 6 cases, before concluding results.

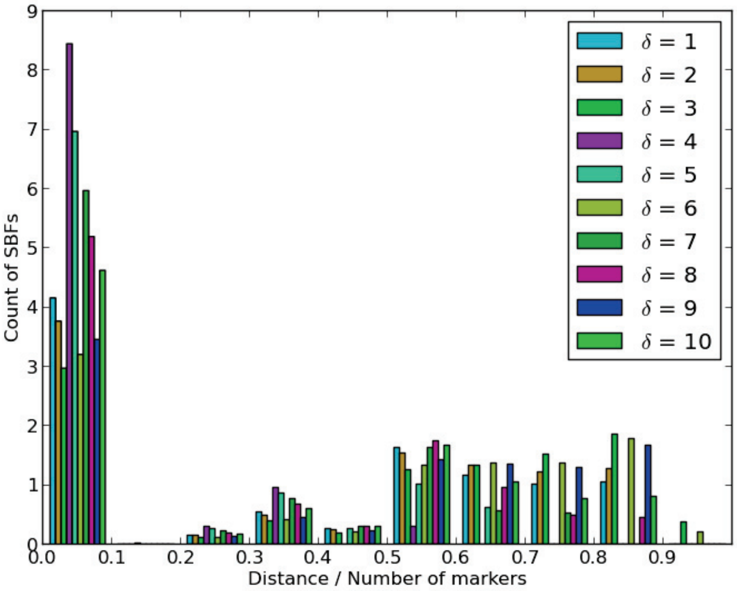


Figure 6.1 – *Interval-pair distribution based on the ratio of non-homologous markers in a SBF between C. Glabrata and E.gossypii;* $\delta \in \{1,\dots,10\}$.

In figure 6.1 we exemplify the first point. The interval pairs are grouped by the number of non-homologous markers normalized by the total number of markers from the SBF (figure 6.1). This visual examination can be easily transposed into a computational condition and guide parameter setting. For example, SBFs with a good ratio of *non-homologous markers per interval size* will be on to the left in the graph. There, homologous SBs will be strongly interconnected through homology at the marker level. The trend that we see is that most of the blocks are *good* from this quality measure p.o.v., only for certain values for $\delta = \{4,5\}$, while for values 3, 6, and 9 much more blocks end up on the right end of the histogram, bearing a considerable number of unmatched markers in their composition, relative to the total makers part of the SBF. A possible explanation for this behavior is that the genomes present multiple similar regions, which e.g. underwent evolutionary events that fragmented them to a high extent. However, this observation alone might not be enough to draw conclusions with a high confidence.

A supporting result for possible interpretations comes from the fact that the genome set is highly homogeneous. Most of the interval pairs are strongly interconnected through homology. We sample only a subset of the possible distance values, in order to discover the nature of the data. With little deviation, all pairs of genomes present the same interval-pair structure. We chose to show this trend for $\delta = 10$ non-matched markers (figure 6.2), where the algorithm chooses to identify rather well connected intervals, than to cover the genome with long marker sets that are mostly not homologs.

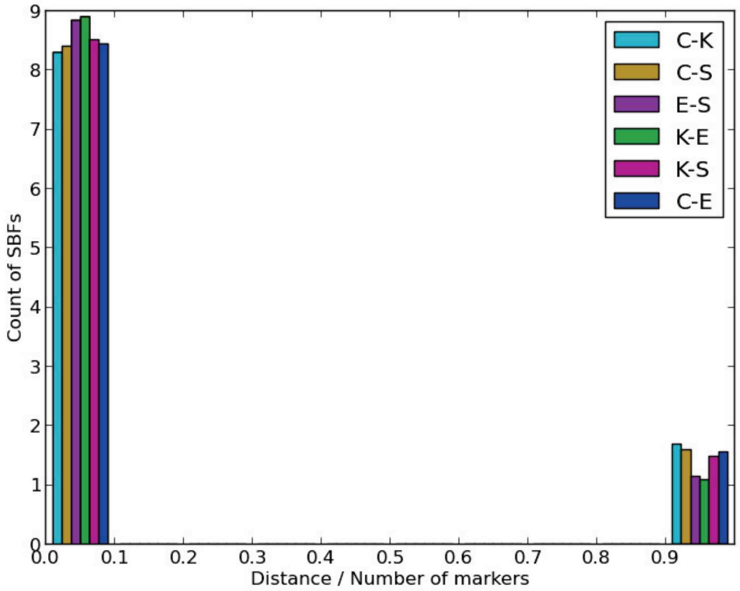


Figure 6.2 – *Similar trend in the interval-pairs generated for* $\delta = 10$*, for all 6 pairs of genomes shows the homogeneity of the dataset.*

6.2.2 Building SBFs Over Multiple Genomes

SBFs can be seen from two perspectives:

- *Structural features* provide a better understanding of marker placement throughout the genome, marker clusters based on their density within SBs and a few more. These results will be presented in the following section.
- SBF are seen as *functional units*, that are used in targeted studies. Example applications are the study of the *Hox* complex of orthologous genes and transcription factors [61]; studies over genes in linkage disequilibrium as presented in [62]; functional assignment of paralogs to syntenic regions in [63]; vertebrate *ultraconserved* elements in [64] to just name a few. However, this is a further step that has not been addressed in the current work.

In the merge and break process, we rely on a reference genome, that is used as guide for the merging process.

We build SBFs over the genomes of interest, by merging interval-pairs to obtain a maximal mutually closed interval-pair. The approach that we present is a first attempt to merge pairwise common intervals. The aim here is to produce preliminary SBFs over multiple genomes, by extending progressively the interval-pairs. SBFs are generated in a few main steps, as follows:

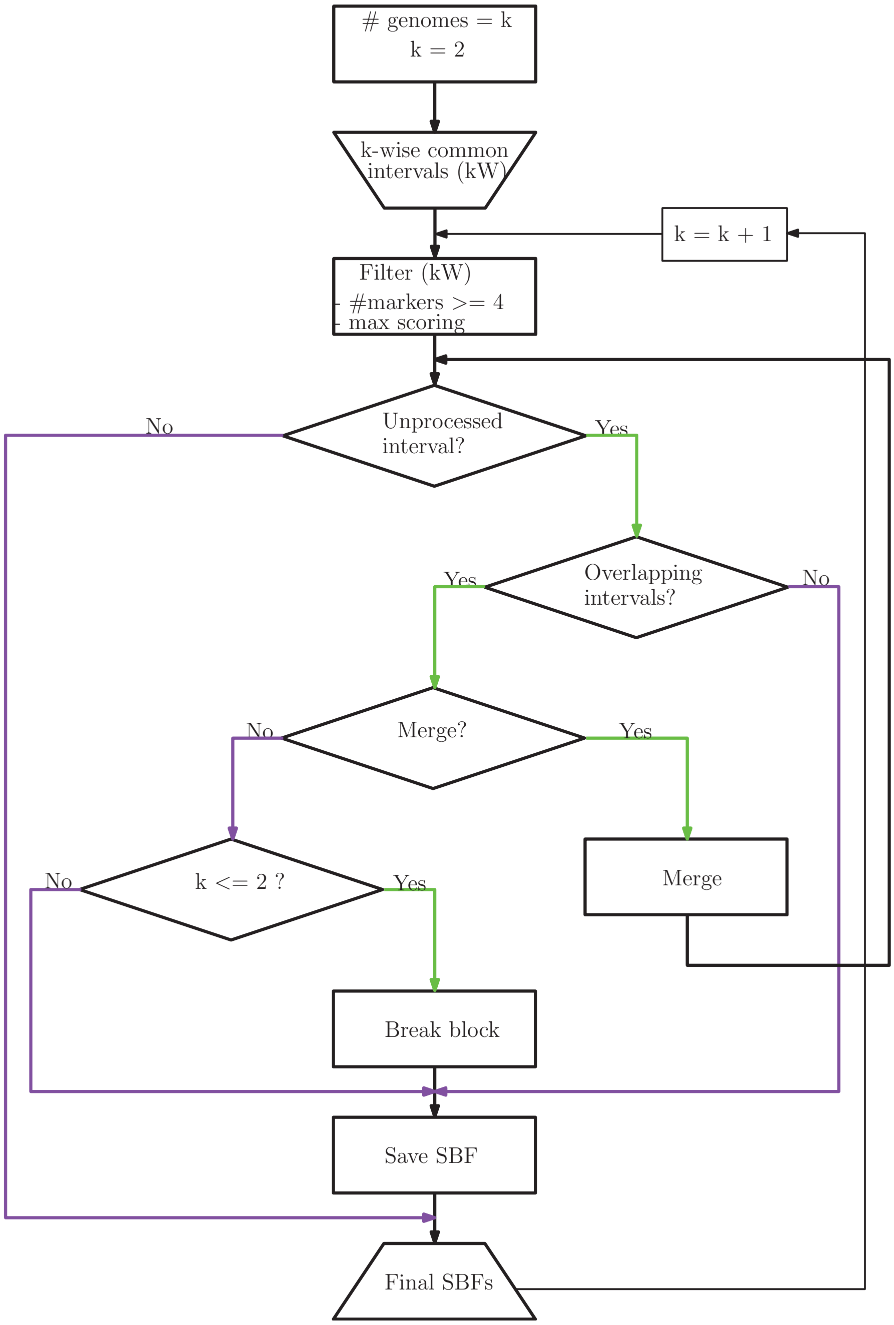


Figure 6.3 – *Code flowchart for generating SBFs over 4 genomes.*

- The approximate common interval implementation of [65] is used to generate pairs of intervals for all pairs of genomes, such that the intervals contain a given number of non-homologous markers set by the δ parameter. For the subsequent experiments, we chose $\delta = 5$ for the unmatched markers.
- Maximal intervals are then retained as intermediary outputs as a result of merging nested intervals, for maximal coverage—the main requirements for the merge are (i) that the intervals are consecutive both in the reference and the second genome; (ii)that the distance between the resulting interval pair remains under the chosen threshold value. When any of these conditions is not obeyed, the intervals are segregated.
- Finally, of an iterative manner, genomes are merged progressively thus one at a time, with the maximal intervals from the previous step.

Fig. 6.3 presents the generic code flowchart for the pipeline used to generate SBFs.

6.2.3 Features Of SBFs

Based on 6.3, we examine the features of the generated output. Three main features of interest are:

- Framing of the resulting SBFs.
- Hierarchical organization of the nested structures for the pairwise case.
- Robustness of the final SBFs from the perspective of homology relations that connect related SBs within a SBF.

Framed SBFs. We examine the number of framed intervals in the set of generated SBFs. Table 6.2 summarizes the characteristics of the output blocks that account for common genomic regions shared by the 4 genomes. A brief comparison between the percentage of framed intervals produced by the wCI version from [65] and the intervals resulting from the merging process are presented in figure 6.4.

Table 6.2 – *Characteristics for framed SBFs defined over the complete genome sets. The first column gives the number of SBFs that we obtain after merging the initial pairwise set, denoted by 2-wise*. The second column represents the number of families that are framed by homologous markers found on the same extremities of the blocks. The total number of blocks is given in the 3rd column, while the percentage of the framed blocks is the last value in the table.*

	framed	*total SBFs*	*%*
*2-wise	178164	262789	67.79
2-wise	16712	22572	74.03
3-wise	2	138	1.44
4-wise	0	7	0

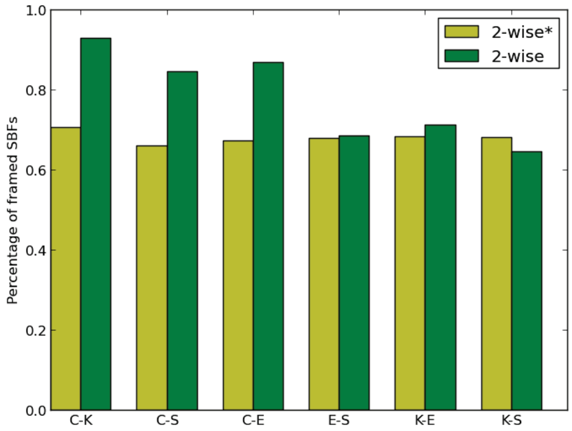


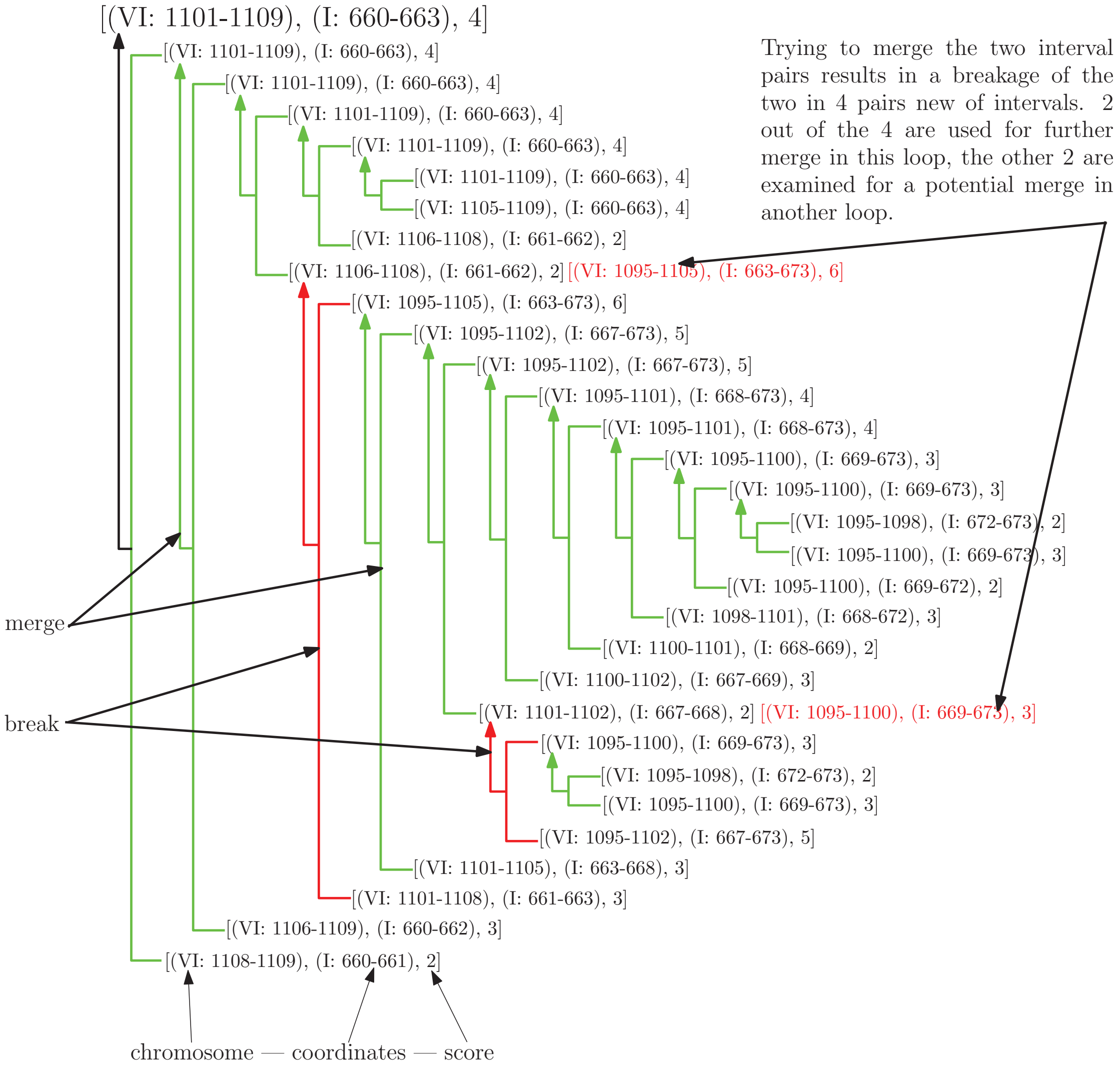
Figure 6.4 – *A brief comparison between the percentage of framed SBFs as produced by the algorithm introduced in [65] and the ones resulting from the merge process.*

The *initial* interval-pairs are already framed SBFs (denoted by **2-wise*). However, not all the initial intervals are maximal, which motivates our merge step. Thus after processing all the pairwise intervals, we obtain more framed pairs.

Hierarchical organization. As mentioned previously, common intervals are a subset of rank-intervals and therefore they present a certain hierarchical ordering; that is, the resulting intervals are nested and form a tree structure, where at the leaf level intervals have the smallest rank and at the top of the tree, we find high ranking intervals; the rank basically accounts for the position of the first occurrence of an interval (number of intervals that appear before the interval in question) in a set of intervals that are ordered according to a reference set [65].

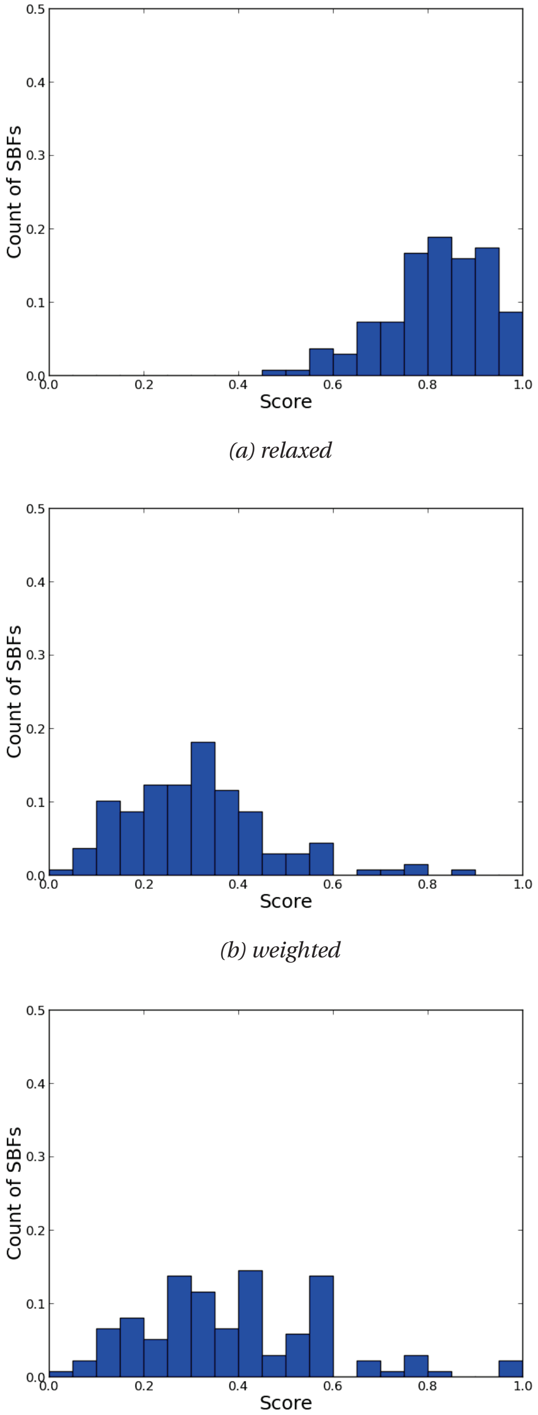
A possible hierarchical decomposition of a genomic region and its marker set is given in fig. 6.5. Here we can see that the merge process either produces a super-interval for the two intervals in each of the organism pairs (interval marked in green), or it breaks the interval according to the algorithm from 6.3 (interval marked in green and red). In this example 2 merge attempts end up in a break, such that there is no overlap in any of the resulting interval-pairs. One of the interval pairs is used further for the merge process, while the other one is *recycled* in a later cycle. After a few levels of merge, a final and maximal interval-pair is generated.

Robustness of homology relations. In this context, we evaluate the resulting intervals with the relaxed and the weighted scores that we introduced in the previous chapter, and the *selective* measure—a complementary measure for the weighted score, that highlights the impact of the mismatching markers within an SBF, on its robustness. Figure 6.6 presents the histograms for the three measures for SBFs on the genomes of C glabrata, E. gossypii and S. kluyveri. The blocks are relatively well interconnected by the weak homology requirement. This is highlighted through the relaxed score. On the other hand, the shift in the center of


[(VI: 1101-1109), (I: 660-663), 4]
[(VI: 1101-1109), (I: 660-663), 4]
[(VI: 1101-1109), (I: 660-663), 4]
[(VI: 1101-1109), (I: 660-663), 4]
[(VI: 1101-1109), (I: 660-663), 4]
[(VI: 1101-1109), (I: 660-663), 4]
[(VI: 1105-1109), (I: 660-663), 4]
[(VI: 1106-1108), (I: 661-662), 2]
[(VI: 1106-1108), (I: 661-662), 2] [(VI: 1095-1105), (I: 663-673), 6]
[(VI: 1095-1105), (I: 663-673), 6]
[(VI: 1095-1102), (I: 667-673), 5]
[(VI: 1095-1102), (I: 667-673), 5]
[(VI: 1095-1101), (I: 668-673), 4]
[(VI: 1095-1101), (I: 668-673), 4]
[(VI: 1095-1100), (I: 669-673), 3]
[(VI: 1095-1100), (I: 669-673), 3]
[(VI: 1095-1098), (I: 672-673), 2]
[(VI: 1095-1100), (I: 669-673), 3]
[(VI: 1095-1100), (I: 669-672), 2]
[(VI: 1098-1101), (I: 668-672), 3]
[(VI: 1100-1101), (I: 668-669), 2]
[(VI: 1100-1102), (I: 667-669), 3]
[(VI: 1101-1102), (I: 667-668), 2] [(VI: 1095-1100), (I: 669-673), 3]
[(VI: 1095-1100), (I: 669-673), 3]
[(VI: 1095-1098), (I: 672-673), 2]
[(VI: 1095-1100), (I: 669-673), 3]
[(VI: 1095-1102), (I: 667-673), 5]
[(VI: 1101-1105), (I: 663-668), 3]
[(VI: 1101-1108), (I: 661-663), 3]
[(VI: 1106-1109), (I: 660-662), 3]
[(VI: 1108-1109), (I: 660-661), 2]
merge
break
chromosome — coordinates — score

Trying to merge the two interval pairs results in a breakage of the two in 4 pairs new of intervals. 2 out of the 4 are used for further merge in this loop, the other 2 are examined for a potential merge in another loop.

Figure 6.5 – *An example illustration for a possible hierarchical structure as generated from the interval merging process presented in the code flowchart. The structure of the hierarchy is given by the rank based left-maximal property of the approximate common intervals derived from rank-intervals.*



(a) relaxed

(b) weighted

(c) selective

Figure 6.6 – *The three scores that help quantify SBF robustness according to the first requirement of the formal definition, that is the homology based interconnection between homologous SBs. Even with a certain number of allowed mismatched markers, the blocks are relatively high scoring, as shown in figure (a); figures (b) and (c) show a slight shift of the graphic towards the center end of the graphic, that is the mismatched markers have a low impact on the robustness of the final SBFs.*

the histogram towards the center, from the weighted score to the selective one, suggests that mismatched markers have a small impact on the overall robustness of the final SBFs, for the given distance value. However, the drastic shift of the scores from the relaxed to the weighted histograms, is possibly due to the markers that form the intervals, which are not necessarily always the same when it comes to merging several genomes together. This implicitly calls for a devising a different measure for the robustness of various markers, that accounts for the fact that markers in an interval need not relate to markers in all other SBs, but instead having selective match could be an allowed feature.

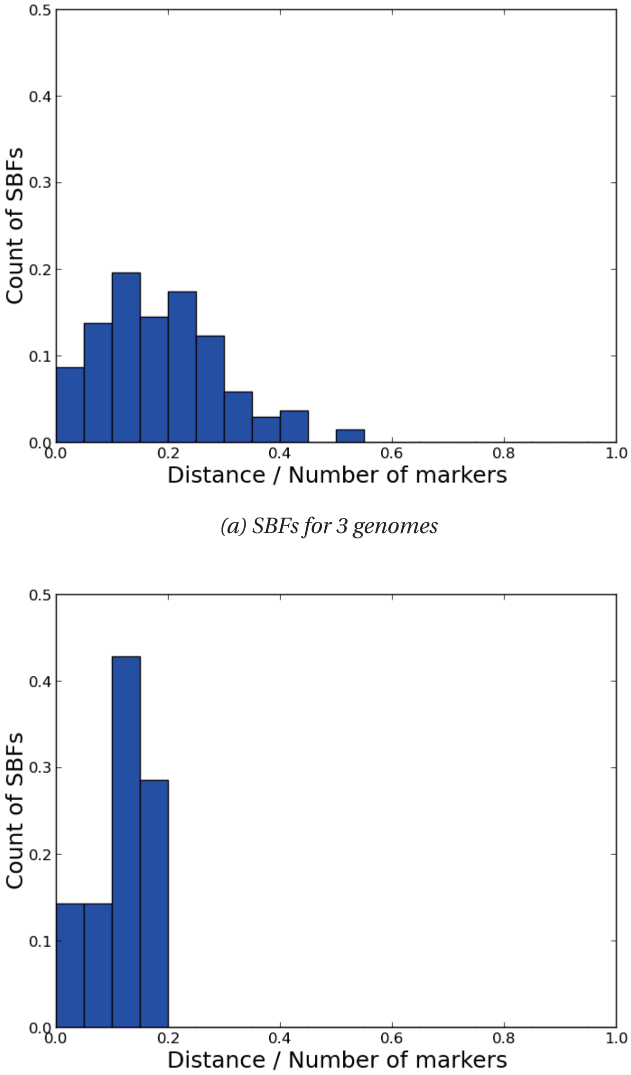
The distance measure between blocks is kept constant, therefore we observe a decrease in the number of SBFs as genomes are added. Figure 6.8 illustrates this trend. In figure 6.7, we see that the SBFs that contain distantly related blocks, are eliminated when the last genome is added to the family. The skewness of both histograms suggests on the other hand, that the blocks are reasonably well balanced in terms of the ratio between total and mismatched number of markers. Such behavior, together with the skewness of the histograms (towards better scores) as shown in fig. 6.6, sets a potential direction for choosing the right value for the δ variable or for the quality of the resulting SBFs.

The dynamics for the marker content in 3- and 4-wise SBFs is presented in fig. 6.8. Without a dynamic adjustment of the scoring function to the increasing number of genomes added to the comparison, we get a similar behavior to defining synteny in terms of strict collinearity, that is, the number of blocks decreases just as shown in table 6.2, the total number of blocks; in the same time the number of markers in a block increases.

6.2.4 Future Directions And Challenges

The so far presented approach for generating SBFs and the measures, reveal more insights into the structure of SBFs as based on wCIs. However, such insights are still far from enough for defining functional SBFs. This combination provides nonetheless, inspiration for future directions and possible improvements. It is clear that the structure of SBFs is best shaped by dynamic scores that take into account additional features of the genomes and of the algorithm, when establishing the number of mismatching markers (e.g. size of the genomes; the density of the markers for a certain genomic region; the quality of markers; etc.).

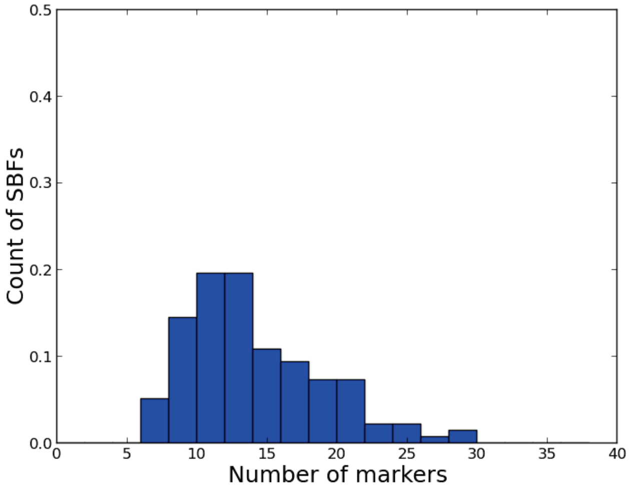
So far we have evaluated only SBFs that are present in all the genomes that are being compared. This is a restricted set of the possible blocks, however, it already gives an insight into the influence of various parameters over the features of the block families. This simplification is directly related to the computational bottlenecks at storage, ordering, coverage detection that arises when working with the entire set of possible SBFs. Another convention that needs improvement is the dependence on a reference genome in multiple comparisons. The absence of such a convention, would explode the number of possible blocks. A hierarchical organization offers then a convenient ordering for the various decomposition schemes of the genomes into SBFs, and implicitly addresses other issues s.a. the storage and handling of the



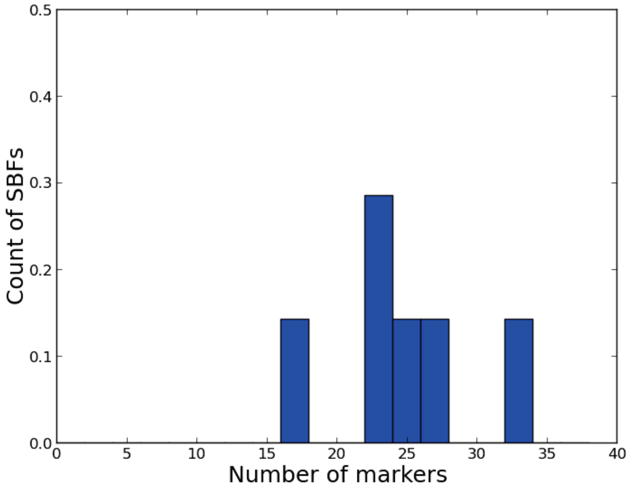
(a) SBFs for 3 genomes

(b) SBFs for 4 genomes

Figure 6.7 – *The constant distance measure for an increasing number of genomes produces SBFs of higher quality, with the price of decreasing number of SBFs as shown in table 6.2, the total number of SBFs columns.*



(a) SBFs for 3 genomes



(b) SBFs for 4 genomes

Figure 6.8 – *A distribution for the size of SBFs in terms of marker content.*

multiple blocks. Moreover, combining this organization with an appropriate visualization method, facilitates the choices of the user, when it comes to selecting the desired SBF structure. A pertinent and sensibly chosen set of parameters, together with the right formalization of the definition and the underlying structure have the potential to be a *silver bullet*.

7 Conclusions

We presented a review of the work to date on the definition and construction of syntenic blocks, pointing out the lack of a formal definition of syntenic blocks as well as the lack of clear objectives for the tools designed to construct these blocks. The latter prevents us from evaluating each tool in terms of its own performance; the former prevents us from establishing a gold standard for evaluating the quality of syntenic blocks.

We evaluate the quality of a decomposition into syntenic blocks by defining new quality measures applicable to all decompositions into syntenic blocks. We apply them to the output of several synteny tools run on a dataset of 8 yeast genomes. This evaluation revealed very different behavior, as well as some reassuring commonalities, among the tools on the same dataset.

To remedy this situation, we proposed a simple set of homology-based criteria that syntenic blocks should satisfy. These criteria do not identify unique solutions—we argued that a range of solutions should remain, since the specifics of the application should influence the selection of good blocks. We based our definitions on homologies, because syntenic blocks are aimed at decomposing a genome into conserved regions (one of the few points on which all researchers agree) and conservation is embodied in homologies.

Almost all existing synteny tools use genes as markers. Not only does such a choice restrict the usable range of granularity, but, at least in the case of most eukaryotic genomes, it discards most of the sequence data (close to 98% in the case of the human genome). A sequence-based approach to the identification of markers, in the style of progressiveMauve or Sibelia, makes more sense in today's data environment. Among choices that a user should be able to make are: (i) permissible degree of overlap of blocks; (ii) acceptable percentage of dropped markers; and (iii) granularity. In addition, since the level of confidence in markers will vary, these choices should be further refined by taking into account the contribution of each shared, dropped, or included marker.

A first step towards integrating such features and capabilities in future synteny mining tools, we take the formal definition a step further, and investigate a subset of 4 genomes for possible syntenic structures. We implement an algorithmic framework for detecting syntenic regions, that is based on a variety of common intervals. This provides (i) structure for SBFs; (ii) hierarchical organization for various levels of decomposition; (iii) addresses computational challenges though its formalism (e.g. marker relatedness in a family-free context). To have a functional definition, we relax certain constraints (i.e. by allowing a certain number of mismatched markers in SBFs). The effect of the new constraints we test on SBFs over the 4 genomes and present promising results for identifying SBs with a minimal parameter set, that are in the same time strongly interconnected through homology.

We have shown that the hierarchical organization of the SBFs addresses computational challenges, offering concomitantly a convenient repertoire of genomic coverage by various levels of SBFs for targeted comparative studies. Clearly, then, the next generation of tools needs a hierarchical organization of blocks, a measure of significance for blocks based on strong connections between markers in the same SBF, and user-defined (and application-motivated) constraints and parameters, defined of a sensible manner.

Bibliography

[1] S. Altschul et al. Basic local alignment search tool. *J. Mol. Biol.*, 215:403–410, 1990.

[2] C. Baudet et al. Cassis: Detection of genomic rearrangement breakpoints. *Bioinformatics*, 26(15):1897–1898, 2010.

[3] A. Bergeron, S. Heber, and J. Stoye. Common intervals and sorting by reversals: a marriage of necessity. In *Proc. 1st European Conf. Comput. Biol. ECCB'02, Bioinformatics*, 18:S54–S63, 2002.

[4] G. Bourque, P. Pevzner, and G. Tesler. Reconstructing the genomic architecture of ancestral mammals: lessons from human, mouse, and rat genomes. *Genome Res.*, 14(4):507–516, 2004.

[5] K. Byrne and K. Wolfe. The Yeast Gene Order Browser: combining curated homology and syntenic context reveals gene fate in polyploid species. *Genome Res.*, 15(10):1456–1461, 2005.

[6] P. Calabrese, S. Chakravarty, and T. Vision. Fast identification and statistical evaluation of segmental homologies in comparative maps. *Bioinformatics*, 19, Suppl 1:i74–i80, 2003.

[7] P. Compeau, P. Pevzner, and G. Tesler. How to apply de Bruijn graphs to genome assembly. *Nature Biotech.*, 29:987–991, 2011.

[8] A. Darling, B. Mau, and N. Perna. progressiveMauve: multiple genome alignment with gene gain, loss and rearrangement. *PLoS One*, 5(6):e11147, 2010.

[9] R. Deonier, S. Tavare, and M. Waterman. *Computational Genome Analysis: An Introduction*. Springer, 2005.

[10] W. Fitch. Homology: a personal view on some of the problems. *Trends Genet.*, 16(5):227–231, 2000.

[11] Z. Fu et al. MSOAR: a high-throughput ortholog assignment system based on genome rearrangement. *J. Comput. Biol.*, 14(9):1160–1175, 2007.

[12] T. Gabaldon and E. Koonin. Functional and evolutionary implications of gene orthology. *Nature Reviews Genetics*, 14:360–366, 2013.

[13] K. Jahn. Efficient computation of approximate gene clusters based on reference occurrences. *J. Comput. Biol.*, 18(9):1255–1274, 2011.

[14] M. Krzywinski et al. Circos: an information aesthetic for comparative genomics. *Genome Res.*, 19(9):1639–1645, 2009.

[15] I. Minkin et al. Sibelia: A scalable and comprehensive synteny block generation tool for closely related microbial genomes. In *Proc. 13th Workshop Algs. in Bioinf. (WABI'13)*, vol. 8126 of *LNCS*, 215–229. Springer Verlag, Berlin, 2013.

[16] J. Nadeau and B. Taylor. Lengths of chromosomal segments conserved since divergence of man and mouse. *Proc. Nat'l Acad. Sci. USA*, 81(3):814–818, 1984.

[17] B. Paten et al. Sequence progressive alignment, a framework for practical large-scale probabilistic consistency alignment. *Bioinformatics*, 25(3):295–301, 2009.

[18] W. Pearson. Empirical statistical estimates for sequence similarity searches. *J. Mol. Biol.*, 276(1):71–84, 1998.

[19] P. A. Pevzner et al. De novo repeat classification and fragment assembly. *Genome Res.*, 14(9):1786–1796, 2004.

[20] P. Pevzner and G. Tesler. Genome rearrangements in mammalian evolution: lessons from human and mouse genomes. *Genome Res.*, 13(1):37–45, 2003.

[21] S. Pham and P. Pevzner. DRIMM-Synteny: decomposing genomes into evolutionary conserved segments. *Bioinformatics*, 26(20):2509–2516, 2010.

[22] S. Proost et al. i-ADHoRe 3.0–fast and sensitive detection of genomic homology in extremely large data sets. *Nucleic Acids Res.*, 40(2):e11, 2012.

[23] J. H. Renwick. The mapping of human chromosomes. *Ann. Rev. Gen.*, 5:81–120, 1971.

[24] C. Rödelsperger and C. Dieterich Syntenator: multiple gene order alignments with a gene-specific scoring function. *Algorithms Mol. Biol.*, 3:14, 2008.

[25] C. Roedelsperger and C. Dieterich. CYNTENATOR: progressive gene order alignment of 17 vertebrate genomes. *PLoS One*, 5(1):e8861, 2010.

[26] The Mouse Genome Sequencing Consortium. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420(6915):520–562, 2002.

[27] K. Vandepoele et al. The automatic detection of homologous regions (ADHoRe) and its application to microcolinearity between Arabidopsis and rice. *Genome Res.*, 12(11):1792–1801, 2002.

[28] R. Waterhouse et al. OrthoDB: the hierarchical catalog of eukaryotic orthologs in 2011. *Nucleic Acids Res.*, 39:D283–D288, 2011.

[29] X. Zeng et al. Orthocluster: A new tool for mining synteny blocks and applications in comparative genomics. In *Proc. 11th Conf. Extending Database Technology EDBT'08*, 656–667. ACM Press, New York, 2008.

[30] M. Calonje et al. Non-coding nuclear DNA markers in phylogenetic reconstruction. *Plant Syst. and Evol.*, 282:257–280, 2009.

[31] R. L. Cann DNA markers and human evolution. *eLS*, 2006.

[32] C. N. Dewey Whole-genome alignment. *Methods Mol Biol*, 855:237–257, 2012.

[33] D. Dörr and A. Thévenin and J. Stoye Gene family assignment-free comparative genomics. *BMC Bioinformatics*, 13(19):S3, 2012.

[34] D. Dörr et al. Consimilar Intervals. *Recomb-CG*, 2013.

[35] S. Heber and J. Stoye Finding all Common Intervals of k Permutations. *Combinatorial Pattern Matching. Lecture Notes in Computer Science*, 2089:207-218, 2001.

[36] K. D. Pruitt et al. RefSeq: an update on mammalian reference sequences. *Nucleic Acids Res.*, 42:D756–D763, 2014.

[37] A. U. Sinha and J. Meller Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. *BMC Bioinformatics*, 8:82, 2007.

[38] G. Tesler and G. Bourque Computational tools for the analysis of rearrangements in mammalian genomes. *Methods Mol. Biol.*, 422:145–170, 2008.

[39] T. Uno and M. Yagiura Fast Algorithms to Enumerate All Common Intervals of Two Permutations.
Algorithmica, 26:290–309, 2000.

[40] I. A. Vergara and N. Chen Using OrthoCluster for the detection of synteny blocks among multiple genomes. *Curr. Protoc. Bioinformatics*, Chapter 6:Unit 6.10 6.10.1–Unit 6.10 6.1018, 2009.

[41] S. V. Angiuoli and S. L. Salzberg Mugsy: fast multiple alignment of closely related whole genomes. *Bioinformatics*, 27(3):334–342, 2011.

[42] M. Blanchette et al. Aligning multiple genomic sequences with the threaded blockset aligner. *Genome research*, 14(4):708–715, 2004.

[43] Wikipedia Synteny — Wikipedia, The Free Encyclopedia. 2013

[44] A. Delcher et al. Alignment of whole genomes *Nucleic Acids Res.*, 27(11):2369–2376, 1999.

[45] C. G. Ghiurcuta and B. M. E. Moret Evaluating synteny for improved comparative studies. *Bioinformatics*, 30(12):i9–i18, 2014.

[46] M. D.V. Braga et al. The Potential of Family-Free Genome Comparison. In *Models and Algorithms for Genome Evolution, Computational Biology*, vol. 19, chap. 13, pp. 287–307, Springer Verlag, Berlin, 2013.

[47] A. Darling et al. Mauve: multiple alignment of conserved genomic sequence with rearrangements. *Genome Res.*, 14(7):1394–1403, 2004.

[48] M. Brudno et al. LAGAN and Multi-LAGAN: efficient tools for large-scale multiple alignment of genomic DNA. *Genome Res.*, 13(4):721–731, 2003.

[49] N. Bray et al. AVID: A global alignment program. *Genome Res.*, 13(1):97–102, 2003.

[50] D. Baillie et al. WABA success: a tool for sequence comparison between large genomes. *Genome Res.*, 10(8):1071–1073, 2000.

[51] M. Blanchette et al. Breakpoint phylogenies. *Genome Informatics*, 1997:25–34, 1997.

[52] S. Hannenhalli et al. Transforming cabbage into turnip: polynomial algorithm for sorting signed permutations by reversals. *Journal of the ACM*, 46(1):1–27. ACM Press, New York, 1999.

[53] C. Lemaitre et al. Precise detection of rearrangement breakpoints in mammalian chromosomes. *BMC Bioinformatics*, 9(1):286, 2008.

[54] S. Schwartz et al. Human–mouse alignments with BLASTZ. *Genome Informatics*, 13(1):103–107, 2003.

[55] T. F. Smith et al. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147(1):195–197, 1981.

[56] J. Fostier et al. A greedy, graph-based algorithm for the alignment of multiple homologous gene lists. *Bioinformatics*, 27(6):749–756, 2011.

[57] Y. Hochberg and Y. Benjamini More powerful procedures for multiple significance testing. *Statistics in Medicine*, 9(7):811–818, 1990.

[58] S. Dudoitet al. Multiple testing. Part I. Single-step procedures for control of general Type I error rates. *Statistical Applications in Genetics and Molecular Biology*, 3:1, 2004.

[59] L. Luo et. al Genome-wide gene and pathway analysis. *European Journal of Human Genetics*, 18(9):1045–1053. Nature Publishing Group, 2010.

[60] G. Didier et. al Character sets of strings. *Journal of Discrete Algorithms*, 5(2):330–340, 2007.

[61] D. Duboule The rise and fall of Hox gene clusters. *Development*, 134(14):2549–2560. The Company of Biologists Limited, 2007.

[62] O. Simakov et al. Insights into bilaterian evolution from three spiralian genomes. *Nature*, Nature Publishing Group, 2012.

[63] C.H.E. Chen et al. The dynamics of functional classes of plant genes in rediploidized ancient polyploids. *BMC Bioinformatics*, 14(Suppl 15):S19, 2013.

[64] J. Reneker et al. Long identical multispecies elements in plant and animal genomes. *National Academy of Sciences*, 109(19):E1183–E1191, 2012.

[65] D. Dörr et al. Identifying Gene Clusters by Discovering Common Intervals in Indeterminate Strings *BMC Genomics*, Proc. of RECOMB-CG 2014.

[66] H. Ogata et. al Mechanisms of evolution in Rickettsia conorii and R. prowazekii. *Science*, 293(5537):2093–2098, 2001.

[67] X. Ling et al. Detecting gene clusters under evolutionary constraint in a large number of genomes. *Bioinformatics*, 25(5):571–577, 2009.

[68] R. Overbeek et al. The use of gene clusters to infer functional coupling. *National Academy of Sciences*, 96(6):2896–2901, 1999.

Cristina Gabriela GHIURCUTA

Rte de Chavannes 27C/17
Lausanne, CH-1007
+41 78 604 32 83

http://www.cristinaghiurcuta.com
cristina@cristinaghiurcuta.com
EU (HU and RO) citizen
Valid B work permit

Autonomous by nature, I have a strong drive to achievement. I seek challenges and with a proactive mindset, I embrace pragmatic problem solving. I enjoy collaborative work.

Employment and Training

Ph.D. Candidate — EPFL (Swiss Federal Institute of Technology Lausanne)
2009 – Present — Laboratory of Computational Biology and Bioinformatics

Thesis focus: improving whole genome comparative studies through large scale conserved patterns across genomes. I conducted a detailed examination of existing concepts, identified failure points and as a solution, I proposed a formal model that offers a structured, rigorous definition. This milestone work sets the baseline for conducting quantitative and qualitative synteny based comparative studies, of a systematic manner.

The results were presented at ISMB 2014, a World premier conference in Computational Biology, with a top acceptance rate (19.6% acceptance rate per 204 submissions). I was also granted a performance based travel award.

Recently, I established a collaborative follow-up project for the above presented work.

I supervised a M.S. semester project and three summer interns. I taught and proactively guided them towards the desired project outcomes. I managed 4 times and participated in teaching the advanced algorithms, M.S. level course.

Software Q.E. — Continental Automotive Romania (part time employment)
2008 – 2009 — R&D Department, Chassis & Passive Safety Team
(Sept. – Jan.)

As a software quality engineer (Q.E.), I managed several international projects, by performing milestone reviews according to the company's internal process web, technical (SPICE, MISRA) standards, process improvement (CMMI) and quality management standards (ISO 9000).

Jr. Software Dev. — Siemens VDO Romania (internship)
2007 Sept. — R&D Department, Commercial Vehicles and After Market Team

As a junior software developer I gained practical experience of working with the software module responsible for data acquisition and transmission from the temperature controller of the vehicle.

Education

M.S. Thesis — ETHZ (Swiss Federal Institute of Technology Zurich)
2009 (April – Sept.) — Computer Vision Laboratory

I successfully developed and tested a stochastic mathematical model for simulating the mechanism of nuclear migration in S. Cerevisiae (budding yeast). Through this project I established a valuable cooperation between the life sciences representative and the technical module's developer, by adapting the model to the new biological constraints.

2004 – 2009 — B.S. (M.S. equivalent) Polytechnic University of Timisoara (UPT), Romania.
Dipl. Eng. in Computer Science

Technical Skills

Advanced: Java, C/C++, Python, MySQL, LaTex, Shell script, MatLab, MS Office
Intermediate: R, Perl, Visual Basic for Excel, Circos
Operating Systems: Linux, Windows, Mac OS

Language Skills

Hungarian and Romanian – Native
English – Fluent (C1)
French – Working proficiency (B2/C1)
German – Limited working proficiency (B1/B2)

Extracurricular Projects

2013 (May – Oct.)	Fix the Leaky Pipeline – a career building program for women in science
2012 April	How to rebuild a country — Haiti in the aftermath of the 2010 earthquake McKinsey & Company, case study workshop
2010 (Sept. – Dec.)	What a Young Entrepreneur Should Know, 2nd prize IVE – Institute for Value Based Enterprise, business plan writing workshop

Community Involvement

2009 – 2012	Event organizer for Local Committee Lausanne of the International Association for the Exchange of Students for Technical Experience (IAESTE)
2007 / 2010	Short term volunteering camps in Portugal and Iceland with Service Civil International (SCI)

Interests and Hobbies

I train my systematic and structured thinking skills by practicing rock climbing and rowing within the frame of the university offered facilities as well as through the Swiss Alpine Club's. These activities equally consolidated my sense of responsibility as a team player, improved focus, perseverance and task-mastery through individual practice. I regularly dedicate time to organize get-togethers with friends or to discover my environment though various activities. Throughout the year, I plan several trips to enlarge my horizons by traveling and discovering new cultures and landscapes, via conferences (latest e.g. ISMB) or individual ones (e.g. Asia trip 2013). I practice sports regularly for a good mind-body balance.

Appendix

Publications

Ghiurcuta, C.G., and Moret, B.M.E., Evaluating synteny for improved comparative studies, Proc. 22nd Symp. on Intelligent Systems for Mol. Bio. ISMB'14, Bioinformatics (2014) 30 (12): i9-i18.

Presentations, Conferences and Workshops

2014 July	International Society for Computational Biology (ISMB) Boston, MA, USA. Travel award granted by NSF USA (84/202 awards & >1300 participants). Oral presentation in proceedings.
2014 Feb.	Swiss-French Meeting on Bioinformatics and Evolutionary Genomics (ALPHY/PhyloSIB) Geneva, CH. Oral presentation.
2013 July	54th Annual Short Course on Medical and Experimental Mammalian Genetics Scholarship award for participation costs. The Jackson Lab, Bar Harbor, ME, USA. Poster presentation.
2013 May	Statistical Genomics and Data Integration for Personalized Medicine Ascona, CH. Poster presentation.
2012 Oct.	Advanced Algorithms M.S. level course EPFL (Fall 2012) Guest lecturer for "Average case analysis of randomized algorithms."
2012 June	Symposium of Society for Molecular Biology and Evolution Dublin, IR. Poster presentation.
2011 Sept.	Scientific research plan writing, Swiss Institute of Bioinformatics Chandolin, CH.
2011 June	[BC]2 Conference Basel, CH. Poster presentation.
2010 Aug.	Quantitative imaging and modeling of biological processes NBIC/SIB summer school, Amsterdam, NL.