SynVisio: A Multiscale Tool to Explore Genomic Conservation

A Thesis Submitted to the
College of Graduate and Postdoctoral Studies
in Partial Fulfillment of the Requirements
for the degree of Master of Science
in the Department of Computer Science
University of Saskatchewan
Saskatoon

By
Venkat Kiran Bandi

Permission to Use

In presenting this thesis in partial fulfilment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Computer Science
176 Thorvaldson Building
110 Science Place
University of Saskatchewan
Saskatoon, Saskatchewan
Canada
S7N 5C9

Or

Dean
College of Graduate and Postdoctoral Studies
University of Saskatchewan
116 Thorvaldson Building, 110 Science Place
Saskatoon, Saskatchewan S7N 5C9
Canada

Abstract

Comparative analysis of genomes is an important area in biological research that can shed light on an organism's internal functions and evolutionary history. It involves comparing two or more genomes to identify similar regions that can indicate shared ancestry and in turn conservation of genetic information. Due to rapid advancements in sequencing systems, high-resolution genome data is readily available for a wide range of species, and comparative analysis of this data can offer crucial evolutionary insights that can be applied in plant breeding and medical research. Visualizing the location, size, and orientation of conserved regions can assist biological researchers in comparative analysis as it is a tedious process that requires extensive manual interpretation and human judgement. However, visualization tools for the analysis of conserved regions have not kept pace with the increasing availability of information and are not designed to support the diverse use cases of researchers. To address this we gathered feedback from experts in the field, and designed improvements for these tools through novel interaction techniques and visual representations. We then developed SynVisio, a web-based tool for exploring conserved regions at multiple resolutions (genome, chromosome, or gene), with several visual representations and interactive features, to meet the diverse needs of genome researchers. SynVisio supports multi-resolution analysis and interactive filtering as researchers move deeper into the genome. It also supports revisitation to specific interface configurations, and enables loosely-coupled collaboration over the genomic data. An evaluation of the system with five researchers from three expert groups coupled with a longitudinal study of web traffic to the system provides evidence about the success of our system's novel features for interactive exploration of conservation.

Acknowledgements

I would like to express my heartfelt gratitude to my supervisor Carl Gutwin, for constantly offering me support both academically and personally, and guiding me towards a field of research that was fantastic to explore and learn.

I am particularly grateful to Gwen Lancaster for her support when I joined the program and am also grateful to my lab mates and all the staff members of the Department of Computer Science who have helped me along the way. Also, I would like to thank the members of my thesis committee, Ian McQuillan and Debajyoti Mondal, whose comments and suggestions have greatly improved this manuscript.

Finally, I would like to thank Canada for offering me a chance at a fresh start and I acknowledge the support of my family and friends for helping me along this journey in starting a new life.

This thesis is dedicated to my Mom and Dad for their unconditional love and support.

Contents

List of Figures

List of Abbreviations

BLAST	Basic Local Alignment Search Tool
CNV	Copy Number Variation
CSS	Cascading Style Sheet
DNA	Deoxyribonucleic Acid
DOM	Document Object Model
FASTA	Fast All
GFF	General Feature Format
HTML	Hypertext Markup Language
mRNA	Messenger RNA
RNA	Ribonucleic Acid
SNP	Single Nucleotide Polymorphism
SVG	Scalable Vector Graphics
XSS	Cross Site Scripting

1 Introduction[1]

With the emergence of new sequencing systems, genomic data is being generated at an unprecedented rate. Almost two decades back, *The Human Genome Project* took 13 years and over $3 billion dollars to sequence the entire human genome whereas the same information can be sequenced today in under an hour for $1000 dollars [64]. This rapid improvement in sequencing has improved the availability of high-resolution genomics data and has helped researchers in tackling a wide range of biological questions [66].

An essential area in biological research where genomic data is extensively used is comparative genomics. It involves comparing genomic information between or within different species to understand genetic similarity. A genome of an organism consists of its complete set of DNA as a collection of chromosomes which contain genes, where every gene is a sequence that is responsible for one or more traits in that organism [36]. Comparing genomic sequences between two different organisms can help researchers in understanding their evolutionary relationship, as sequence similarity can often mean that the genes have the same function. Such similar sequences are referred to as homologous sequences, and they indicate shared ancestry. As organisms evolve over time and diversify into different species, they retain parts of their DNA from their common ancestor. The study of these conserved homologous regions is called **synteny analysis**.

Some aspects of large-scale genomic comparison are purely computational and thus can be automated, but human judgment is still vital in comparative analysis and visualization tools can assist researchers in these tasks. The choice of visual encoding in the representation of syntenic relationships is dependent on the kind of analysis that is being done by genome researchers. Certain graphical representations like dot plots (where every conserved gene is represented as a point on a two dimensional matrix) are useful in analyzing large scale genomes in a summarized representation as shown in Figure 1.1, while other representations like parallel plots (where syntenic regions are represented as coloured ribbons connecting similar regions) are useful in performing a more in-depth analysis as the conserved regions are more visually prominent [69]. Additionally, Circos plots which use a circular ideogram layout, as shown in Figure 1.2, are also frequently used by researchers in publications as they can be aesthetically pleasing and useful at summarizing large scale patterns effectively [46]. With such varied graphical representations, arriving at the right form of visualization can be difficult, and any system that offers only a single kind of visual encoding can become limited in its usability for complex datasets with diverse use cases. Further due to the complexity of generating

[1]Portions of this thesis appeared in the following publication: Bandi, V and Gutwin, C. 2020. Interactive Exploration of Genomic Conservation. In Proceedings of Graphics Interface 2020 (GI '20). The first author carried out the large majority of the requirements gathering, design, development, and evaluation of the SynVisio system, as well as the large majority of the writing of the paper; the second author participated in the collaboration with genomic researchers to formulate design requirements, and editing of the paper.

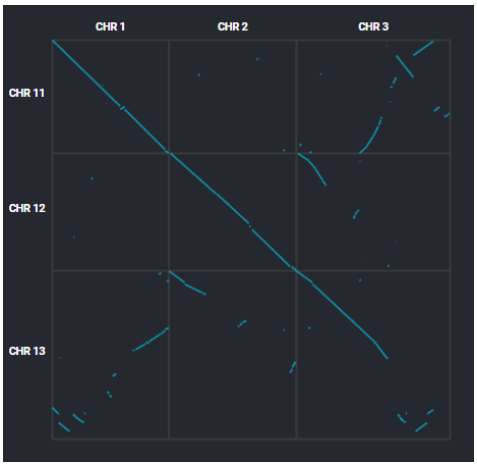

Figure 1.1: Dot plot

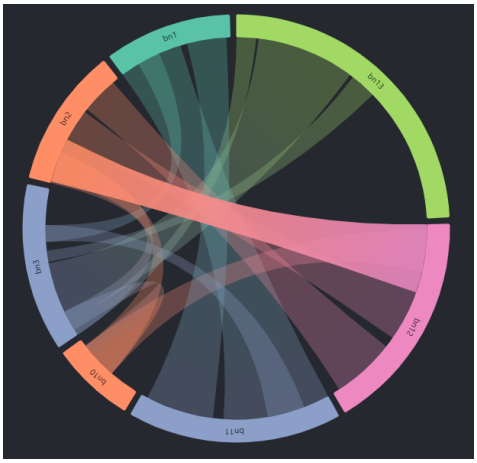

Figure 1.2: Circos Plot

visualizations of large scale genomes, current synteny visualization tools are primarily operated through command-line interfaces or are stand-alone programs limited to specific operating systems. This combined with the steep learning curve in using these systems and their limited usability, means that a broad set of these tools are beyond the reach of the wider science community. This has created a need for easy to access visualization tools that let researchers interact with their datasets and change parameters in real time to explore their results in multiple coordinated visual representations.

1.1 Problem and Motivation

The problem addressed in this thesis is: *existing genomic visualization tools have limited support for exploration, interaction, and collaboration tasks with large scale genomic datasets and are poorly integrated with existing synteny detection tools.*

Understanding genomic conservation is crucial for researchers as it has applications in a wide variety of scenarios, such as predicting whole-genome duplication events, annotating extremely large genome sequences like wheat, and classifying the proximity of different species in their evolutionary history. The increasing size and complexity of genome sequences mean that the work that genomic scientists do with their datasets is constantly evolving: genome visualization tools are now being used in diverse tasks such as evolutionary investigations of gene duplication events [100], missions to look for new medical treatments [14], and comparisons of gene expression to relate genotype and phenotype [34]. These kinds of complex tasks indicate that researchers need access to systems that can support a wide variety of exploration, interaction, and collaboration activities. This increasing need for interactivity coupled with the easy availability of datasets (e.g., through public databases such as NCBI and Ensembl) has led to a surge in the demand for computer-based support tools. However, current tools for visualizing and exploring genomic datasets have not kept pace with this increasing demand and are limited in their capabilities: they typically support only a small variety of datasets; they are not designed for investigation of complex synteny scenarios such as polyploidy

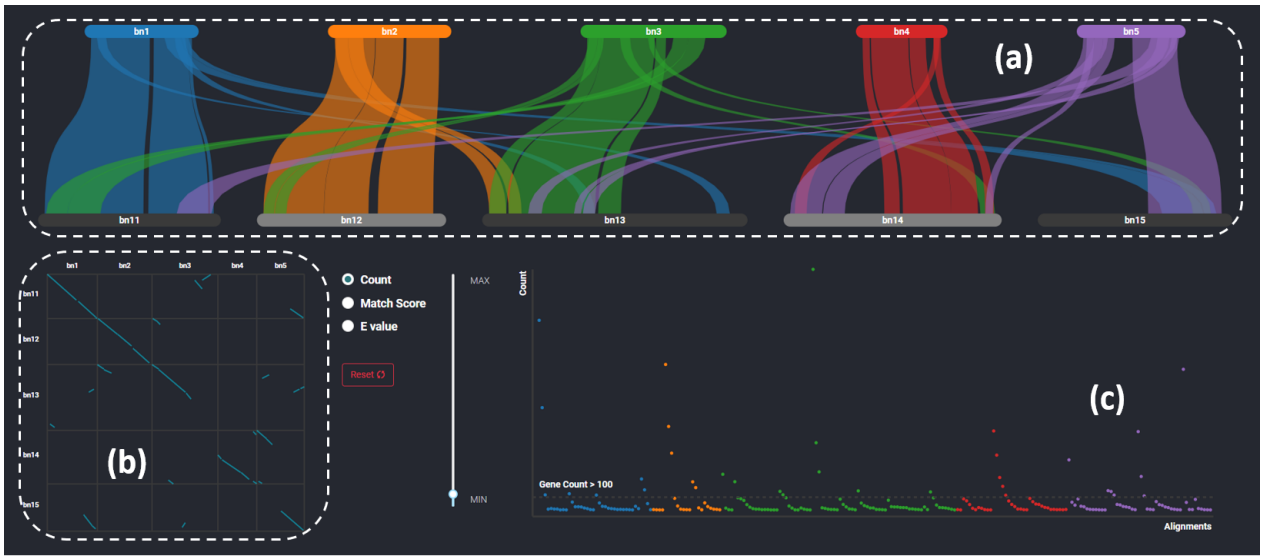

Figure 1.3: Syteny Dashboard visualizing genome collinearity in Canola *(Brassica napus)* with the following components: **a)** Parallel plot with connected ribbons representing collinear gene blocks. **b)** Dot plot where every collinear gene is represented by a point and contiguous collinear blocks are shown as lines. **c)** Filter panel representing all the collinear blocks based on the count of their genes with ability to refine results using slider to the left.

(whole-genome duplication, which is common in plants); and they often do not support visualizations at multiple genomic scales. A possible reason for these limitations is that genomic visualization tools are rarely developed in close collaboration with the genomic scientists who actually use those tools, and as a result they do not consider the kinds of genomic exploration and analysis tasks that are now performed. For example, a task such as tracing the conservation of genes across more than one species requires the ability to explore pairwise comparisons at multiple levels; similarly, refining sequence assemblies requires annotating existing visualizations with gene density plots to verify assembly quality.

1.2 Solution

To address the limitations of current visualization tools, we met with three teams of genome researchers to understand the interactive and visual requirements for current genomic investigations. The three teams all study plants, but perform very different kinds of exploration and analysis. In collaboration with these experts, we first identified the basic visual requirements of a synteny analysis tool and then supplemented this list with additional requirements for interactive genomic visualizations that are not supported by current synteny visualization tools such as: the need to refine datasets in real-time, the need to work with multiple perspectives on the data, the need for dynamic multi-resolution visualizations, the need to link secondary datasets to the genomic data, the need for new visualization of synteny across multiple genomes, and the need to support navigation and revisitation in genomic data spaces.

Based on these requirements, we designed a tool called SynVisio that has novel visualization and inter-

action capabilities to meet the needs of genomics experts. SynVisio is an open source web-based system available at `https://synvisio.usask.ca`. It lets researchers explore syntenic blocks through coordinated multiple views including parallel plots at several scales (Figure 1.3 (a)), dot plots (Figure 1.3 (b)), and a dynamic filter panel (Figure 1.3 (c)) where users can refine the display of conserved regions based on similarity and the number of contiguous genes in a conserved block.

SynVisio can directly work with the results of existing synteny detecting tools such as MCScanX [128] and DAGChainer [33] and can visualize conservation in multiple representations. SynVisio has two modes: a primary mode, and a multi-analysis mode. The primary analysis mode lets users compare chromosomes in the same genome or between two genomes, and the information is visualized as parallel plots, dot plots, or both as shown in Figure 1.3. For visualizing synteny across several genomes simultaneously, SynVisio offers a multi-genome analysis mode where synteny is visualized in stacked parallel plots or hive plots. SynVisio further offers a rich interactive experience by letting users switch views in real time and explore data from the genome level all the way down to the individual gene level. Users can do this by clicking on any two chromosomes when looking at the visualization in the genome level and then further step down from the individual chromosome level by clicking on a particular gene block to look at its constituent genes and their orientation. Additionally, users can also annotate their views with additional genomic data in the form of tracks above the genomes or chromosomes, which can be visualized as heatmaps, histograms or scatterplots.

As exploration in SynVisio is heavily based on user interactions, it offers them the ability to record these interactions as snapshots, for future revisitation. This gives users the ability to examine multiple scenarios and rapidly switch between them in real-time. The system also indexes all the conserved genes in the browser, thus letting users quickly look up genes by their gene IDs to see which conserved blocks they belong to. Finally, SynVisio offers users the ability to download all generated visualizations in transform and scale invariant vector graphics for scientific documentations, reports and research publications.

1.3 Steps to the Solution

There were several steps involved in designing a system that addressed the usability issues mentioned in the problem statement.

- **Formulate Design Requirements** - To characterize the needs from the biological research community, we primarily met with three groups of researchers studying genomic conservation through a series of structured interviews to collect their requirements. The first group we met was interested in exploring synteny in wheat while the other two groups were involved in studying canola and pulse crops respectively. All three groups were unanimous in the verdict that synteny is a critical issue to study for understanding genomic evolution and that existing tools don't meet their needs. Based on the feedback from the genomic research community all requirements can be broadly classified into either **functional** or **supplementary** requirements. Functional requirements include understanding the size, location,

and orientation of conserved sequences along with having the ability to filter sequences based on their similarity while non-functional requirements include features like the ability to download images or snapshot explorational points to revisit.

- **Identify and Explore Existing Alternatives** - Since synteny analysis is a combination of synteny detection followed by downstream analysis using visualization systems, we looked at tools that operate in both these domains. We looked at several state of the art syteny detection packages like MCScanX, DAGChainer, Cyntenator and i-ADHoRe [33,89,98,128]. We also tested the visual outputs of some that had their own downstream analysis tools. We focused our research on MCScanX and DAGChainer out of the other alternatives as they were the most popular and frequently used tools by the researchers we interviewed and had more accessible and efficient output data formats in the form of syntenic blocks or orthologue tables. We followed this by looking at the tools that worked in the second stage of the analysis pipeline by providing visualizations, like SynChro, GSV, Mizbee, VGSC, and Circos [22,59,92,138]. Of these, we found that most served as simple graphic generating systems instead of offering a platform for detailed analysis except for MizBee, which was however, limited by its accessibility due to its small variety of visual representations and availability only as a desktop application.

- **Explore Visual Design and Architecture** - To implement our solution, we decided on a web-based single page tool that would work as a part of the existing analysis pipeline by working directly on the results of existing synteny detection tools. We adopted a thick-client architecture model instead of the traditional thin client model where visualizations are generated on the server, as a thick-client model would let researchers directly upload their analysis files and see the resulting images in real time without their sensitive data being sent to a remote server. To visually represent genomic conservation, we used linear connections (parallel plots) and points (dot plots). We then encoded additional information about the size and orientation of the gene blocks through a combination of colour and shape.

- **Implement System** - We built SynVisio using a combination of React.JS [90] and D3.JS [8]. The former was used to render the visual elements on the web interface while the latter was used to calculate the positions of the graphical elements. To render the visualizations, we used both canvas and simple vector graphics (SVG) and compared their performance. We found that vector graphics based visualizations while being more resource intensive offered a better visual experience across different resolutions with greater scope for interactive features. So our system was designed to render all charts as simple vector graphics by default but can dynamically switch to canvas for rendering of large scale genomes when there are a large number of individual graphics elements. The final application was developed through several design iterations as the system was used by members of our expert user group and several additional features such as support for additional tracks and the ability to download publication ready images were added based on their feedback.

1.4 Evaluation

A stable version of SynVisio was deployed and has been available for public use since the start of 2019. Evaluation of our system was done in two ways: a log based usage study and an interview-based expert study. Firstly to determine the overall use of our system, we analyzed the web traffic logs to SynVisio for a period of one year (2019-2020). We found that it had 154 unique visitors from 18 different countries across the world using it for a wide variety of projects. Additionally, during this period, the open-sourced code for SynVisio was made available on GitHub under an MIT License and has since been adopted into several online genome databases for species such as Tea, Grape Vine, and Silkworm respectively.

To see if we had met our design requirements, we also evaluated our system through semi-structured interviews with five domain experts consisting of four genome researchers and one bioinformatician. The interviews were conducted via phone or in person and lasted around 45-60 minutes. Researchers were asked open-ended questions about synteny analysis and how it is used in their field of research. They were then asked to give their opinion on the various features of the system that were developed to improve its usability. Finally, they were also asked to rate the ability of the system to visualize genomic conservation on a 5 point scale. The feedback from the domain experts was largely positive and helped us in understanding the performance of our tool across difference scenarios. This positive feedback coupled with the broad usage of the system by researchers around the world, shows that SynVisio has been able to address the problem of limited usability of synteny analysis tools.

1.5 Thesis Outline

This thesis is organized into eight chapters, including the current chapter. Chapter 2 firstly present a discussion of the biological background behind genomic conservation and synteny in particular and looks at the different ways in which studying such conservation can assist researchers. We then explore the framework of synteny detection and some of the tools that are currently being used. Secondly, we look at the different kinds of visualization systems and techniques that are used in representing genomic data at various resolutions. We then look at visualization systems dedicated to analyzing sequence similarity and synteny and discuss their merits and limitations. Finally, we explore the various techniques that genomic visualization tools utilize to manipulate both the underlying data and the graphical representation to facilitate data exploration.

In Chapter 3, we first discuss the underlying data abstraction layer in our system by describing the properties of syntenic data and how it is computed and processed. We follow this with an exploration of the different analysis tasks that can be performed on syntenic data and organize these tasks into three basic groups according to the genomic scale at which they operate. We finally discuss supplementary requirements that can enhance user experience with the system.

Chapter 4 provides a discussion of the visual design of SynVisio. We first explore the different forms of

visual encoding used in representing genomic conservation and follow this with a description of the different layout strategies that were explored in designing SynVisio. We then discuss the interaction strategies that were adopted in our final design based on the visual information seeking mantra framework and design of multiple coordinated views. Finally, we conclude the chapter with a summary of the various steps in our iterative design cycle.

Chapter 5 presents a detailed description of our visualization system SynVisio. We first discuss the different modes of synteny analysis SynVisio offers and how they operate. We then provide a description of the various supplementary features that SynVisio provides to enhance user experience with the tool. Finally, we elaborate on the choices made in the architecture of the system, and discuss the software implementation of SynVisio as a web interface built with JavaScript.

Chapter 6 provides a detailed evaluation of our system. To quantify user engagement, web traffic to the system was analyzed for a period of one year. Examples of adaptations of open-sourced code of SynVisio in several online genome databases are also discussed. Finally, a user study was conducted with five researchers through semi-structured interviews, and their responses are summarized through three major case studies highlighting the usability of our system across different types of genomes.

Chapter 7 presents a discussion of the design choices and the insight gained from the development of our genomic visualization system. Further, it also presents the current limitations of our system and highlights possible avenues for improvement in the future.

Finally, Chapter 8 summarizes this thesis. It reiterates the problem statement and outlines our major contributions.

2 Related Work

This research builds on previous work in three major areas: genomic conservation and synteny detection; visualizations of genomic data and conserved regions; and interaction techniques to facilitate data exploration of large scale genomic data. Each of these areas are explored in detail in the following sections.

2.1 Genomic Conservation and Synteny Detection

In this section, we discuss the biological background behind genomic conservation and how analyzing it can provide answers to researchers' biological questions. We also explore synteny detection and the existing tools that are currently used in synteny analysis.

2.1.1 Biological Background

Genomics is the field of biology that involves the study of genomes of various organisms to understand their structure, function, and evolution. [72]. A genome is defined as the complete set of DNA of an organism, where DNA (DeoxyriboNucleic Acid) is the chemical compound containing a series of instructions responsible for the development and functioning of that organism [65]. All living organisms transfer this genomic information from one generation to the other through chromosomes in the nucleus of the cell. Humans, for example, have 23 pairs of chromosomes where one from every pair is inherited from each parent. These chromosomes are responsible for the organism's unique traits and characteristics. A chromosome structurally is a tightly packed length of DNA along with proteins that regulate its structure and activity. This DNA is made of two long strings of nucleotide bases along with sugar and phosphate groups that are wrapped around each other in a double helix structure. There are four bases: adenine (A), guanine (G), cytosine (C) and Thymine (T) with specific pairing rules between them such that adenine always pairs with thymine and cytosine always pairs with guanine. These nitrogenous base pairs collectively make up the entire genome of an organism [124]. The human genome, for instance, is made up of around three billion base pairs encoding information for 20,000-25,000 genes [15].

Genes are long segments of DNA that encode information for a specific protein, and are the basic building blocks of all organisms. Proteins are made up of long chains of amino acids where the structure and function of the proteins are determined by the order of these amino acids. Proteins are manufactured using the information encoded in a gene through a process of transcription and translation called gene expression. During transcription, the DNA present in a gene acts as a template to form an mRNA (messenger RNA)

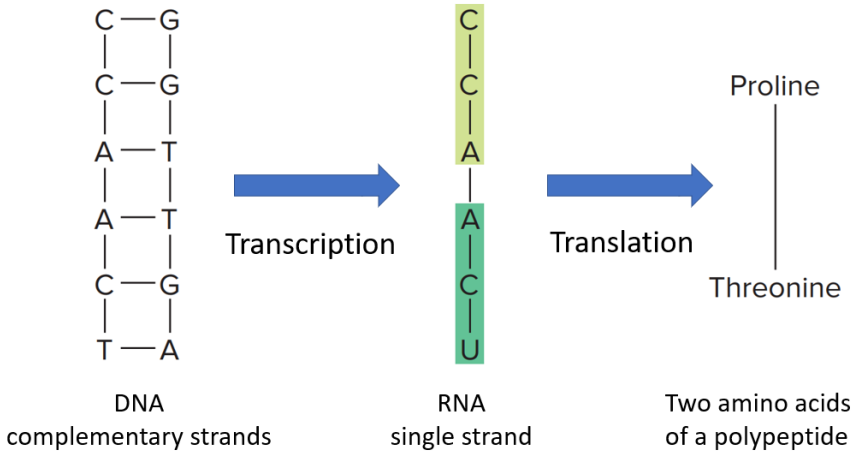

Figure 2.1: Conversion of DNA information into protein via the genetic code. Complementary bases in a DNA strand are split into a single RNA strand, which is read in pairs of three bases at a time (codon) to create a single amino acid in a polypeptide chain. Adapted from [36].

structure, which is a single-stranded structure consisting of one of every complementary base pair in the DNA. This is followed by translation where mRNA is used as a template to assemble a chain of amino acids such that each group of three bases in the mRNA (called a codon) creates one particular amino acid, as shown in Figure 2.1. Thus the order of bases in the DNA encodes for the order of amino acids in the protein and, in turn, the protein's structure and function [13].

DNA is transferred from one generation to the next in all livings organisms through the process of self-replication where the double-helix structure of the DNA comes apart, and each of the complementary strands acts as a template in the production of its counterpart forming new pairs of DNA strands [87]. Although cellular error-checking mechanisms ensure that these new DNA strands are nearly identical to the original strand, mutations can occasionally occur. This can happen when a base at one position is replaced by one of the other bases or is entirely lost. Alternatively, insertions or duplications of extended sets of base pairs can also happen. Other kinds of larger mutations such as chromosomal rearrangements can also occur, including inversions (where a large segment of a chromosome is inverted in orientation) or translocation (where parts of chromosomes swap places) [36]. While most mutations that occur during duplication do not have an effect on a gene, they can occasionally alter the gene's function. This can be detrimental, leading to diseases such as cancer in certain cases. Alternatively, such mutations can also be beneficial by offering resistance to diseases or other environmental stresses.

2.1.2 Comparative Genomics

As mutations accumulate over time, they lead to the divergence of species. Understanding how these changes could have occurred is a significant area of study in comparative genomics and has large scale implications such as discerning the role of genetic factors in human health and disease [14]. Comparative genomics, as the

name suggests, involves comparing genome sequences of different species to identify regions of similarity and difference in order to gain information about the relatedness between the species genomically and functionally. The fundamental principle in comparative genomics remains simple in that sequences that encode for proteins and gene expression should be conserved in related species, whereas sequences that are responsible for differences between species will themselves be divergent [35].

Comparative genomics can assist biologists in linking the phenotypic and genotypic properties of an organism to understand its different characteristics. For example, researchers combined the gene expression data of several plant sequences which have high gene duplication rates with evolutionary conservation data to improve gene discovery [34]. Also, the comparative analysis of genes and their regulatory pathways in the context of phylogeny (the study of evolutionary relationships) provides scientists with a better understanding of how evolution happens at the molecular level [115]. However, the questions that are addressed by comparing genomes at different phylogenetic distances can vary [35]. For example, genomic comparison between species that are separated by very long phylogenetic differences such as yeast (*Saccharomyces cerevisiae*), worms (*Caenorhabditis elegans*), and fruit flies (*Drosophila melanogaster*) reveals that their genomes encode for many of the same proteins while the order of the genes and sequences are not conserved [100]. In contrast, comparison between more closely related species like Humans (*Homo sapiens*) and Chimpanzees (*Pan troglodytes*) reveal that the overall divergence between the two genomes is only 4% and results are more oriented towards identifying the differences than the similarities [125].

Comparative genomic studies are primarily focused around the study of homologous sequences, which are gene sequences that have shared ancestry. The extent of homology is determined by sequence similarity and such similar sequences are commonly referred to as homologs. Such similarity between DNA sequences of two different species can occur either because of a speciation event (a species diverges into two separate species) leading to orthologs, or due to a gene duplication event (a gene is duplicated within the same genome) leading to paralogs [42]. Research into such similar genes, especially in eukaryotic organisms, can shed light on gene duplication events that led to the creation of gene families [100]. Gene families are defined as large groups of gene sequences that are similar to each other while also having a similar function or gene expression. Usually, when a gene duplication occurs the new gene either becomes inactive as a pseudogene or exists as a duplicate copy of the original gene performing the same function. An increased number of gene duplicates through natural selection can often lead to an increase in the protein synthesized by the gene. An example of this is the variation in gene copy number in the human salivary amylase gene (AMY1) responsible for starch hydrolysis in certain human populations [80]. A third scenario of gene duplication that occurs rarely is when the duplicated gene acquires a new function through mutations. An example of this is the trocarin D gene of the Australian rough-scaled snake that acts as a toxin by coagulating the blood of its prey. Comparative genomic analysis of the trocarin D gene revealed that it is nearly identical to the coagulation factor X gene present in the plasma of the snake responsible for blood coagulation (to prevent bleeding when injured) indicating that that the gene was recruited for a new function after a gene duplication event [94].

2.1.3 Synteny

One of the ways in which homology can be inferred for understanding large scale duplication events is through studying collinearity of several genes, where both the gene content and order are conserved [89]. Such long regions containing several genes that display collinearity in the order of kilobases (Kbase) to megabases (Mbase) are referred to as *synteny blocks* [141]. The word synteny has Greek origins with *syn* meaning "together" and *taenia* meaning "ribbon" and is used to indicate the presence of genetic loci on the same chromosome [91]. However, synteny can also occur between different chromosomes and the term is more commonly used to refer to "gene loci in different organisms located on a chromosomal region of common evolutionary ancestry" [78].

With the availability of fully sequenced genomes for several model species, synteny analysis can reveal evolutionary adaptions and also improve the transfer of knowledge to non-model organisms that have not been fully mapped [142]. Synteny analysis, particularly in angiosperms (flowering plants), can help in understanding the consequences of whole genome duplication in plant evolution [1] as shown in the analysis of polyploidy in Thale cress (*Arabidopsis thaliana*), which despite its relatively small genome size has been shown to have undergone repeated cyclical genome doubling [106, 107]. This state of polyploidy where an organism contains more than two sets of homologs is a widespread occurrence in plants due to several whole-genome duplication events at diverse temporal scales but is rare in mammals with evidence of the last whole genome duplication event occurring almost 500 million years ago [1, 74]. The increase in genome size due to polyploidy can occur either due to the inheritance of duplicated sets of chromosomes from the same species where its called autopolyploidy, or due to hybridization between two difference species where its referred to as allopolyploidy. Such changes in genome structure can often have immediate effect on the phenotype and fitness of the individual and in certain hybridization scenarios can even improve the "vigor" of the resultant polyploid compared to its parental species [73]. Synteny analysis of the genomic structure of such polyploid individuals can assist researchers in predicting the number and timings of polyploid events in the organism's evolutionary history [1].

While polyploidy in the form of whole genome duplication continues to be a driving factor in the evolution of plants, genomes of mammals show evidence of only two shared rounds of whole genome duplication specified in the 2R hypothesis [39]. However gene collinearity is conserved to a greater degree in mammals than plants thus making synteny analysis at smaller scales (called microsynteny) much more feasible [142]. In this way, synteny analysis can be adapted to a wide range of genomic scales based on the underlying question.

Apart from regular duplication events mapping syntenic regions can also help in identifying other genomic rearrangement events such as fusions, fissions, translocations, inversions, and deletions [70,133]. Chromosomal fusion can occur at both the telomeres (ends of a chromosome) or centromeres (center of a chromosome linking two chromatids) and is caused by the union of two or more chromosomes to form a single entity. Human chromosome 2 is a well known example of ancestral telomere-telomere fusion of two ape chromosomes as shown in Figure 2.2 [41]. Similarly fission is the splitting of two functional halves of a chromosome where

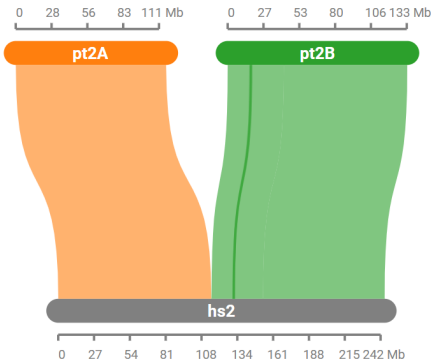

Figure 2.2: Synteny between chromosomes 2A and 2B from Chimpanzees (*Pan troglodytes*) and chromosome 2 from Humans (*Homo sapiens*) depicting an ancestral chromosomal fission event.

the break point is usually situated at the centromere (centric fission) [95]. This can lead to an increase in chromosome number and in certain instances such as the model species of yellow monkey-flowers (*Mimulus guttatus*) synteny analysis has shown that multiple centric fission events (along with partial fusion) provide a better explanation for near-doubling of chromosome numbers compared to the traditional whole genome duplication theory [27].

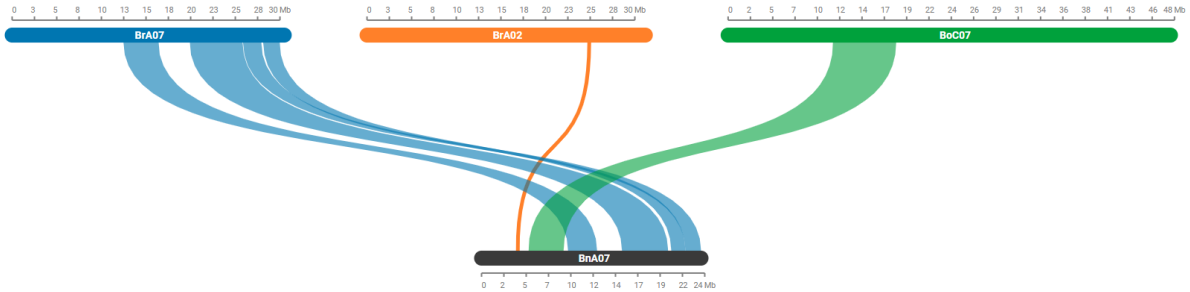

Figure 2.3: Synteny between *Brassica napus* and its ancestors *B. rapa* and *B. oleracea* showing reciprocal translocation rearrangements.

Another common type of chromosomal rearrangement is translocation which is caused by the change in the position of segments of a chromosome. These variations can be either intrachromosomal or interchromosomal. The former is a type of variation where segments are shifted from one arm of a chromosome to the other and the later occurs when a segment is transferred from one chromosome into another (transposition) or segments are mutually exchanged between different chromosomes (reciprocal) [95]. In certain scenarios both intra and interchromosomal translocations can occur together. An example of this can be seen in the synteny analysis of *Brassica napus* an allotetraploid containing the A-genome and C-genome from its progenitor species of *B. rapa* and *B. oleracea*. While 8 chromosomes were found to contain only skeletons from the corresponding chromosomes of B. rapa or B. oleracea, 11 others were composed of different chromosomal segments along with the skeletons indicating that a variety of chromosomal rearrangement events occurred after the initial duplication [11,75]. If we look at chromosome BnA07 in *B. napus* as shown in Figure 2.3, it consists of three different chromosomal sources. The first is the skeleton of BrA07 from *B.rapa*, the second is a fragment of

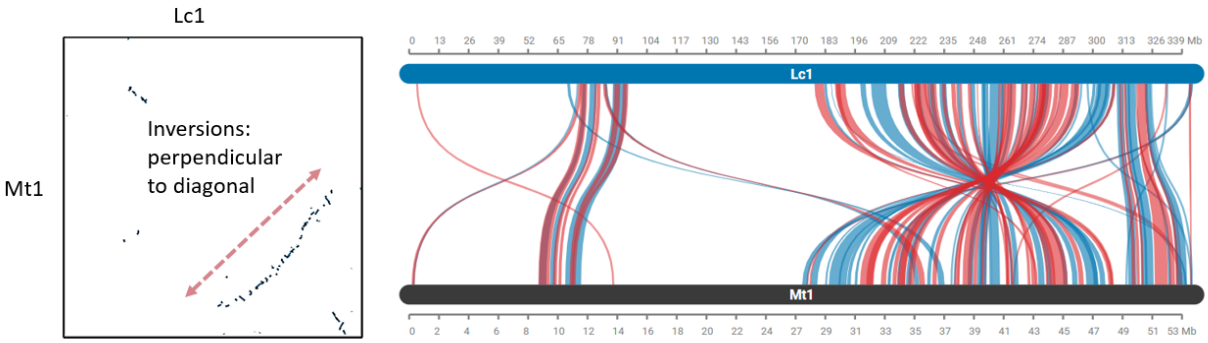

Figure 2.4: Synteny mapping between Chromosome 1 from *L. culinaris* and *M. truntula* showing large scale inversions through a dot plot (left) and a parallel plot (right). The red ribbons represent inverted syntenic regions and the blue ribbons represent regular regions.

BrA02 from *B.rapa* that occurred through reciprocal translocation and, the third is a homologous fragment of BOC07 from *B. oleracea*. Finally inversions and deletions are two other types of genomic rearrangements where the former occurs when a segment of a chromosome is reversed in orientation and put back in its place and the later occurs when a portion of chromosome is lost [95]. An example of inversion can be seen in the comparitive mapping of *Lens culinaris* and model legume *Medicago trunctula*, where a large set of inversions (colored red in Figure 2.4 (left)) are shown between chromosome 1 of *L. culinaris* relative to chromosome 1 of *M. truntula* [32]. These inversions are much more evident in dot plots (Figure 2.4 (left)) as they show up as perpendicular lines to the diagonal. In this way apart from the study of polyploidy, synteny analysis can also be used in understanding different kinds of chromosomal rearrangements.

2.1.4 Analysis Pipeline

Synteny analysis consists of three major steps: sequence alignment, synteny detection, and data visualization. Although SynVisio focuses on the last step, we will take a brief look at the other preceding steps. Before analyzing synteny between organisms their genomes need to be sequenced and assembled at least partially into scaffolds. This process is then followed by similarity detection between the two genomes through sequence alignment.

Sequence Alignment

Sequence alignment is extensively used in computational biology to assess the similarity between DNA, RNA, and protein sequences. Sequence alignment works by arriving at an optimal alignment through a scoring mechanism, where gaps are introduced in one or both of the sequences but penalized accordingly, as shown in Figure 2.5. A gap at any position in the final alignment is an indication of an insertion or a deletion and is penalized because these events are far less likely to occur than mutations. The validity of sequence alignment results is dependent on the alphabet size of the sequences; protein sequences can contain up to 20

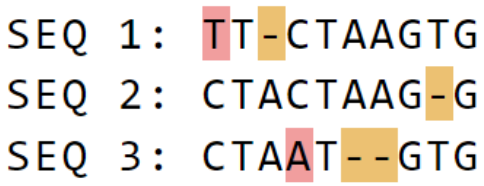

Figure 2.5: Sequence alignment of sequences 'TTCTAAGTG','CTACTAAGG' and 'CTAATGTG' with mismatches and gaps highlighted in red and orange.

different amino acids, whereas DNA sequences only contain four different bases, leading to better alignments in proteins. There are two major types of sequence alignments, and they are each used in different scenarios. The first type of sequence alignment is called a global sequence alignment, where an optimal match is found by aligning the two entire sequences end to end, and is used to compare homologous sequences. The second type of alignment that looks at smaller sections or sub-sequences is called local sequence alignment and is used to look for patterns in a sequence when comparing it with a larger set of sequences such as those in a database.

Every sequence alignment is centred around an optimization problem and early alignment techniques such as the Needleman and Wunsch method [63] used a dynamic programming approach to arrive at the optimal alignment. Dynamic programming is a computational strategy that recursively breaks larger problems into smaller sub-problems and reuses the results of previously solved sub-problems to arrive at a solution to the larger problem. A variation of the Needleman and Wunsch method for local sequence alignment is the Smith-Waterman algorithm [113] that uses a matrix-based scoring scheme for comparing sub sequences. However, the time complexity of such methods is exponential meaning that searching for sequences in large databases is infeasible. This has led to the adoption of heuristic methods to align sequences such as the FASTA (Fast-All) algorithm [52]. Although this algorithm is no longer in use, the name FASTA is still used for a popular file format in bioinformatics, that represents nucleotide and protein sequences as a series of single-letter characters.

BLAST (Basic Local Alignment Search Tool) is a popular local sequence alignment tool that acts as a direct successor to the FASTA algorithm, being more time-efficient and operating on the same file format [81]. It operates by identifying small query words that contain three nucleotides or amino acids for protein sequences in a particular order based on their occurrence along the sequence and closeness to other similar words. It then expands on these words in either direction based on searches from target databases that are rated by a special scoring matrix. BLAST by default uses BLOSUM62 (Block Substitution Matrix) as its scoring matrix which ensures that even more distantly related sequences are detected, but other matrices such as PAM250 (Point Accepted Mutation) can also be specified.

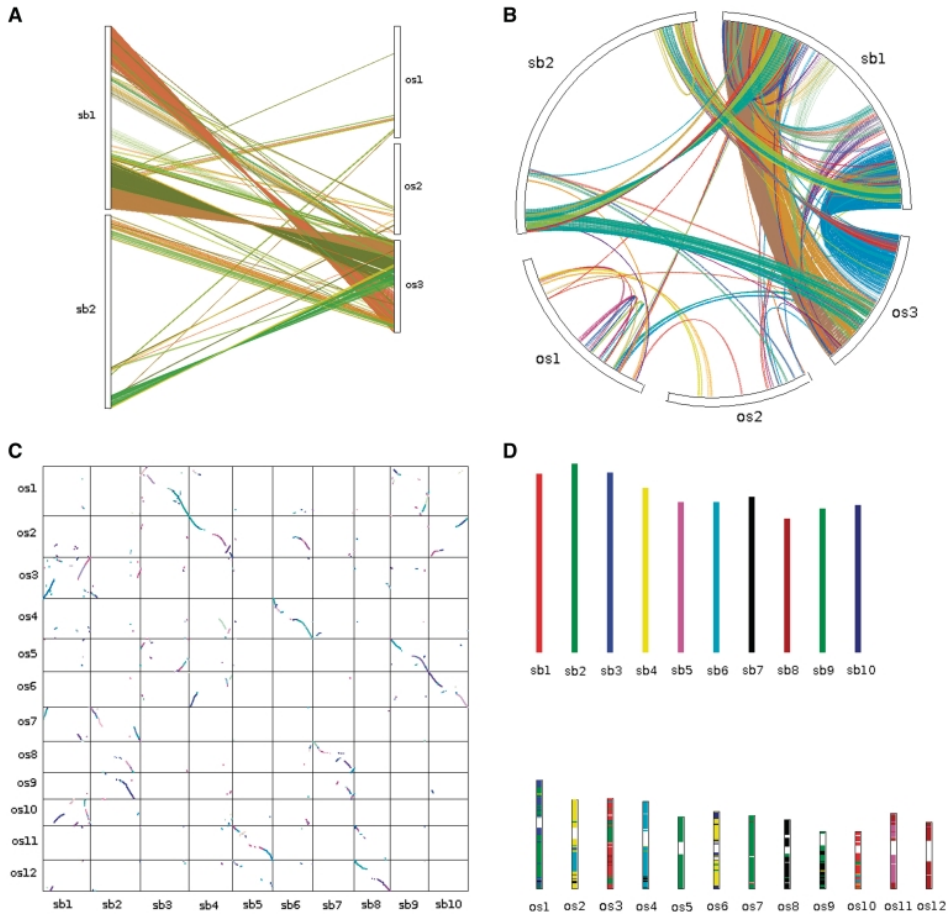

Figure 2.6: Different types of plots visualizing synteny generated by MCScanX : (A) dual synteny plot, (B) circle plot, (C) dot plot and (D) bar plot, From Wang et al. [128].

Synteny Detection Tools

The next step after detecting the similarity between two sequences is the actual synteny detection, as alignment results are only pairwise between sequences and need to be grouped into larger blocks to look for patterns. Although synteny detection tools differ in their operating file formats and computational efficiency, they broadly work by combining positional information of genes along a genome sequence with pairwise BLAST results to construct chains of collinear gene pairs. Grouping neighboring gene pairs that match is one way of detecting syteny [128] that is implemented in tools such as OrthoCluster [141], TEAM [55] and ADHoRe [89]. These tools are, however, outdated and are not efficient in detecting syntenic blocks with conserved gene order, especially in scenarios that might include chromosomal rearrangements and tandem duplications [128]. A new class of synteny tools such as MCScanX [128], DAGChainer [33], and CYNTENATOR [98], that utilize a dynamic programming approach to create chains of pairwise collinear genes around anchor genes are much more efficient at detecting collinear syntenic blocks. Some tools such as MCScanX even offer downstream analysis tools with static visualization results, as shown in Figure 2.6.

2.2 Genomic Visualizations

With the advent of rapid genome sequencing systems, genomic data is being generated at a rapid pace that is not being equalled in terms of innovations in data analysis systems that can help researchers in understanding these ever-increasing data streams. Visualization systems can play a critical role in bridging this gap in data exploration as humans are intuitively good at finding visual patterns. As research in this field becomes increasingly data-driven, visualization systems can aid researchers in generating a hypothesis and iteratively refining it by encoding genomic information through visual cues in the form of shapes and colours [69]. Genomic visualization systems can be used in several scenarios, such as analyzing sequence data at different resolutions and browsing annotations and reference tracks or in comparing sequences from different organisms [66]. However a major challenge in this field remains in determining the right graphical representation based on the genomic context and data under exploration. In this section, we first explore the different kinds of visualizations systems and techniques that are used in representing genomic data at the sequence and genome level. We then look at systems that are used in exploring sequence similarity at different resolutions and finally give a brief overview of current synteny visualization systems and their merits and limitations.

2.2.1 Sequence and Genome Browsers

Visualizing genomes at the sequence level is primarily done by representing sequences as a series of letters organized on a linear scale from left to right. Additionally, sequences that are longer are stacked vertically in a scrolling window. This ordering of bases or amino acids is meant to aid researchers in identifying discrepancies by quickly scanning down the sequence along its length, especially in scenarios involving comparison of read alignments. Visual cues such as emphasis through colours are further used to highlight erroneous bases in some sequence analysis tools [24,30]. Other tools like IGV (Integrative Genomics Viewer) [121] and Hawkeye [103] opt for a simpler representation by only visualizing discrepancies. Sequence visualization systems are also used to interpret and refine the results of sequencing systems where visual cues are provided to highlight gaps, mismatches, and the order of repeats [7,30]. In some systems such as Consed [30], information is visualized in pairs that are colour-coded so that structural variations such as insertions, deletions, and inversions are also considered, and additional information is provided in the form of annotations of amino acid translations to identify mis-assemblies. Finally, some tools like the ABySS-Explorer provide an assembly graph overview for high-level inspection of the assembly instead of focusing on the local sequence mismatches [67].

Once the short sequence reads are assembled into a single large genome, a different set of visualization tools are required that are focused around identifying particular regions of interest. A genome sequence in its entirety can act as a reference against which several features such as gene densities, SNPs, and repeats can be mapped. This form of analysis is increasingly being used in a wide range of browsers that were developed to disseminate information and provide a platform for the exploration of several genomes that are sequenced for

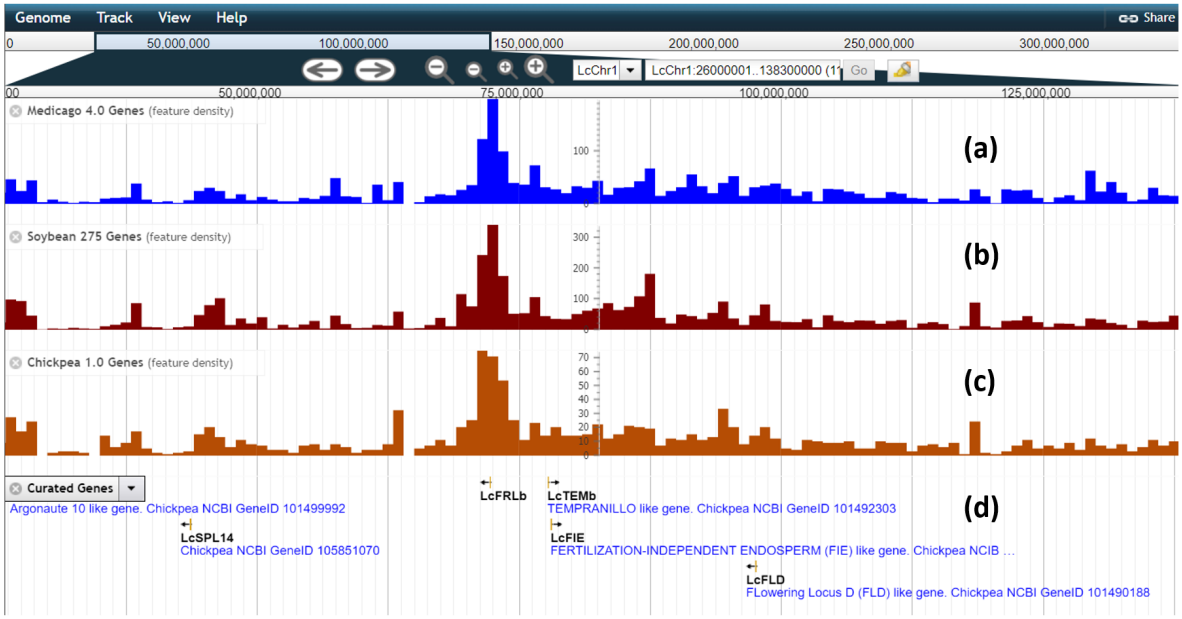

Figure 2.7: JBrowse is used to compare gene densities between (a) Barrel Medic (*Medicago truncatula*), (b) Soybean (*Glycine max*) and (c) Chickpea (*Cicer arietinum*) in relation to a set of curated genes encoding for prominent phenotypes (d) generated through KnowPulse [102].

model organisms. These tools include GBrowse [118], Ensembl Genome Browser [117] and UCSC Genome Browser [44]. These browsers work by displaying a requested portion of the genome with several annotation tracks stacked in rows vertically along a reference axis. The annotation tracks can contain different kinds of information such as positions of single nucleotide polymorphisms, regions with regulatory elements, or location of important genes, as shown in Figure 2.7 (d) and can be toggled on or off depending on their role in the analysis. The information is visualized at several resolutions from hundreds of base pairs all the way up to tens of thousands with the ability to move along the genome horizontally and zoom in and out of a particular region. Some genome browsers even offer the ability to search for a particular gene by looking up its position in the underlying database [66].

Finally, additional visual representations such as summary views (e.g. copy number variations) in the context of biological pathways are also provided in certain browsers like UCSC Cancer genomics browser [44], allowing researchers to associate clinical features with genomic data directly. This form of a genome overview that preserves global context while still allowing researchers to explore smaller chromosome level entities is also present in Gremlin, a tool that offers a novel visual model for exploring structural variants and rearrangements [71].

With an increase in the volume of data being generated, genome browsers are beginning to use a decentralized model where processing power is distributed between the server and the client's browser. Information is retrieved only when needed for the region that the user is interested in, and visualizations are generated

dynamically at the client-side. This reduces the load on the server substantially and modifies it to serve purely as a database to look up information when necessary. This form of rendering in the client can offer a smoother interactive experience while navigating a genome through panning or zooming. It also ensures that there are no disruptions to a user's sense of location, by averting incongruous transitions that occur when large data sets are being loaded and traversed. An example of this approach is JBrowse, which offers visualizations as shown in Figure 2.7, along with significantly reduced server overhead compared to other genome browsers [111]. Certain genome browsers take this decentralized model a step further by offering connectivity to data stored locally on a user's computer [44, 101]. This can provide a personalized experience when dealing with sensitive data in situations where storing data on a remote server may not be viable.

2.2.2 Comparative Genome Browsers

With the ability to sequence multiple genomes within a short span of time, a new field of research has emerged that focuses on comparing genomes instead of looking at them in isolation; this field is called comparative genomics, as discussed earlier in Section 2.1.2. Regardless of the data type or domain, comparison is a common task in data analysis and visualization, when there is a need to understand the relationship between a given set of items. [29]. Visual comparison has been shown to improve the understanding of data in several charts and designs explored by Tufte [123] along with specific examples centred around Playfair's use of line graphs to demonstrate the change in stock prices in relation to wars [17]. In the field of genomics, comparison can aid biologists in a diverse set of tasks such as identifying functional elements, studying large scale rearrangements and genome evolution, and refining results of genome assembly systems through reference genomes [66]. Several systems have been built to address each of these tasks through visual comparison at different genomic scales as sequences can be compared at the nucleotide level all the way up to the whole genome level.

At the nucleotide level, researchers compare sequences to identify the location of mutations, insertions, and deletions; most visualization tools designed for such analysis achieve this by representing alignments on a linear scale. Visual cues are provided by colouring each of the four nucleotides with a categorical colour scale, and the sequences are presented in a linear layout, usually in a stacked arrangement. Some of these tools, such as JalView (JV2) [132], are enhanced with interactive features that let users sort, filter, highlight, and edit multiple sequences in real time. Jalview also lets users overlay sequence features onto the alignment and render extra positional features through transparent or opaque shading over specific regions of an alignment. AliView is a similar tool that works for extremely large datasets through an indexing process and offers support for multiple file formats [49]. JalView and AliView, however, are limited in usability being desktop applications but recent tools such as MSAViewer [139] and JSAV (JavaScript Sequence Alignment Viewer) [57] have been designed to work across the internet as web applications with a similar set of features.

When comparing sequences at higher levels, researchers look for large scale rearrangements. Due to the higher resolution of data, genomic features are grouped into contiguous blocks on each chromosome called

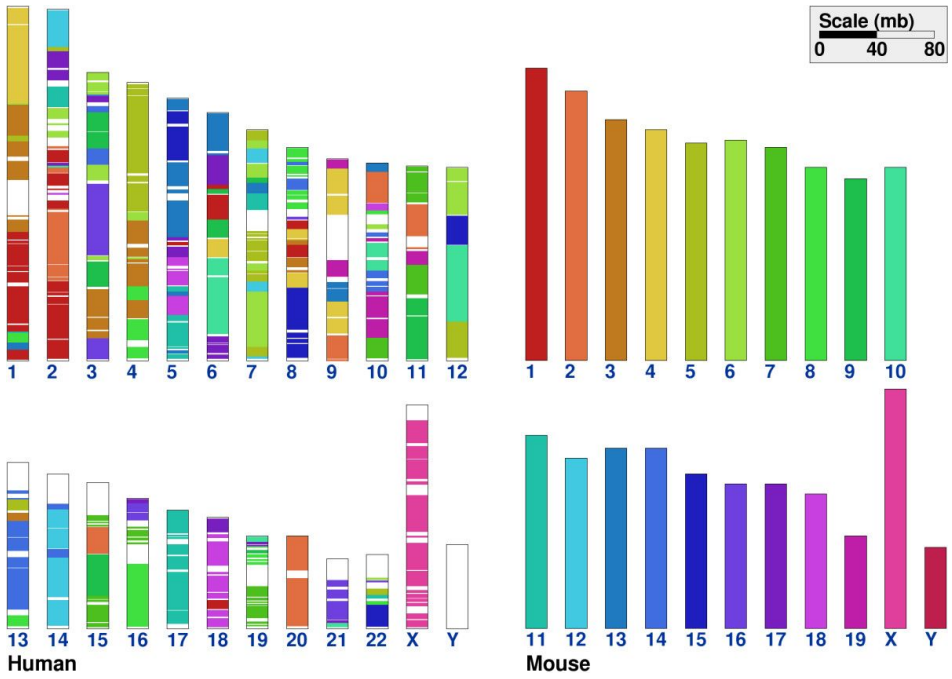

Figure 2.8: Visualization of synteny between human and mouse genomes shown by a pill-based design in Cinteny. Image extracted from [110].

syntenic blocks where conservation is implied through the similarity and relatedness between these blocks. Several strategies have been explored to graphically represent syntenic blocks both at the chromosome and the whole genome level. The earliest examples for representing synteny involve the adaptation of dot plots used for comparing local alignments for larger sequences. Most of these tools primarily perform the actual genome level comparison and present their results through dot plots for closer inspection such as DAGChainer [33] or the VISTA plots of the MUMmer alignment tool [47]. Dot plots are two dimensional representations where genomes of two organisms are presented along the x and y axes. Gridlines are used to show chromosomal boundaries, and every similar gene is represented as a point, implying that large collinear blocks of genes are shown as lines. Such matrix-based representations are extremely good at identifying genome rearrangements, as duplications show up as secondary lines parallel to the diagonal and inversions end up as straight lines that are perpendicular or inclined away from the diagonal. While these plots offer effective genome level summaries of alignments, they cannot be extended for multi-way comparisons between several genomes or used to identify smaller rearrangements at the chromosome level.

Another representation of synteny that has been used extensively in tools such as Cinteny [110], Sybil [18] and MEDEA [69] is the pill-shaped ideogram design of chromosomes. In this design, chromosomes of the source genome are represented as pill-shaped rectangular blocks that are colour coded on a categorical scale and chromosomes in the target block are represented as similar pills with varying sizes based on their genomic sizes. Syntenic regions in the target are then represented through colour coded bands where the colour is determined by the source chromosome that the alignment belongs to, as shown in Figure 2.8. The choice

of representing a chromosome as a rectangular pill is based on a karyogram representation often used for chromosomes in biological literature [79], but information about the position of the centromere is omitted, and the "X" or "V" shaped design is adopted into a single cylinder shaped like a small pill. While the use of colours makes it easier to quickly identify similar regions and their distribution in the target genome in comparison to a dot plot, this representation loses some information such as the orientation of the aligned blocks and their relative position in the source genome. Tools like Apollo partially solve this problem by extending the representation by also linking the coloured segments in the reference genome with their corresponding loci in source chromosome through lines, and by interleaving the source and target regions [50]. Mauve follows a similar approach but uses a linear layout and stacks the sequences parallel to each other, using connections to encode conservation [19]. A significant problem with this representation, along with other designs that rely extensively on colour is that it cannot be extended for a large number of chromosomes as humans cannot intuitively distinguish beyond ten colours [123]. Further, the choice of colours is extremely important as colours need to be visually distinct unlike the colours chosen in tools like Cinteny [110] as shown in Figure 2.5. The colours used in chromosomes 6-9 and 12-15 are very close to each other in the green and blue spectrum and it is hard to distinguish them when the coloured bands in the target are small and close to each other.

A third form of representation that has recently become popular due to its aesthetic appeal is the Circos style plot [46]. In these plots, genomes are represented as arcs presented in a circular layout. Syntenic regions are shown as lines connected through the middle of the circle. Additional tracks representing other information such as copy number variants and SNPs are presented through outer circles along the genome. The circular arrangement is meant to reduce visual clutter that can arise in a linear stacked arrangement when multiple linked regions are represented through connections that cross each other excessively. Tools like Circa that generate these plots can be configured to use a diverse set of colour schemes and visual representations like histograms, heat maps, line charts, and scatter plots for the outer tracks [62].

Mizbee is an example of a standalone synteny browser that combines the circular layout with a linear arrangement to present syntenic information at different scales. Mizbee presents genomic conservation through a combination of connections and colour encoding in three linked views that are presented next to each other, as shown in Figure 2.9. In the genome view, chromosomes are presented as arcs in two concentric rings. The source chromosomes are presented in the outer ring, and the inner ring contains the target chromosomes arranged around a copy of the selected source chromosome. Conservation is then encoded through links connecting the collinear blocks inside the inner ring with additional encoding in the form of colour. The circular layout reduces visual clutter, which is further addressed through edge bundling of contiguous blocks [143] that go to the same destination chromosome. Data in the inner ring can be explored by selecting a particular region which is then presented in a chromosome view in the middle of the display, as seen in Figure 2.9. The colour coding at this level is similar to the genome view but connections are presented in a vertical layout that supports precise spatial analysis. The final view, called the block view, presents

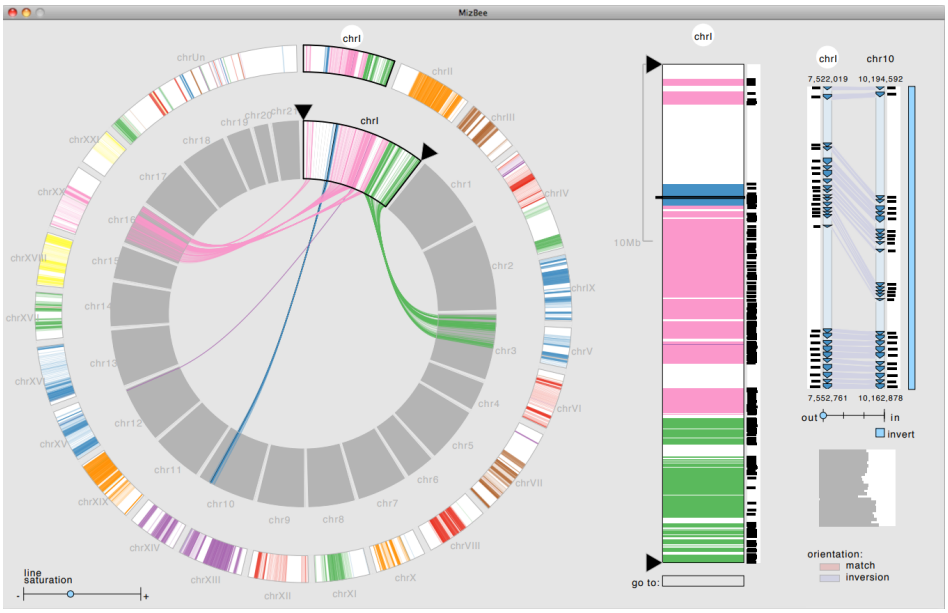

Figure 2.9: Synteny visualization as shown by Mizbee. Image extracted from [59].

the individual genes in a particular collinear block through connected ribbons along with their orientation encoded as directed triangular blocks. Mizbee is the first of many browsers that have been developed to go beyond simple chart generation and act as a complex analysis tool that can present conservation through a combination of multiple visual representations. Mizbee, however, doesn't perform the actual synteny detection and relies on a formatted input dataset. This requires researchers to first detect synteny through a detection tool like DAGChainer, MCScanX or iADHoRe and then modify the output to match the input format of Mizbee. Mizbee also doesn't supplement the generated visualizations with tracks for additional information that provide biological context, and is limited in its usability being a standalone desktop tool.

mGSV (Multi Genome Synteny Viewer) is a synteny viewer that works similarly to Mizbee by visualizing synteny through a combination of visual representations, but is available as a web-based tool [92]. It lets users upload pre-computed syntenic data in a tab-delimited format and also accepts an extra annotation file to show additional genomic features as an annotation track. The system, however, works in a distributed model where syntenic information is stored in a remote database and charts are generated at the server and fetched based on user interactions in the browser. This server-based model can cause data security issues and also add unnecessary network delay into the analysis, making it a time consuming process. SimpleSynteny is another web based tool that represents information in a horizontal linear layout using a combination of coloured bars and connected ribbons for representing genomic conservation [127]. It however accepts FASTA files as input for the genomes and uses BLAST [2] on a remote server to align the sequences into collinear blocks. While this does improve the usability of the tool, synteny detection is a resource intensive process that can place a heavy load on the server, especially for large genomes, and is best done on desktop machines that are computationally powerful.

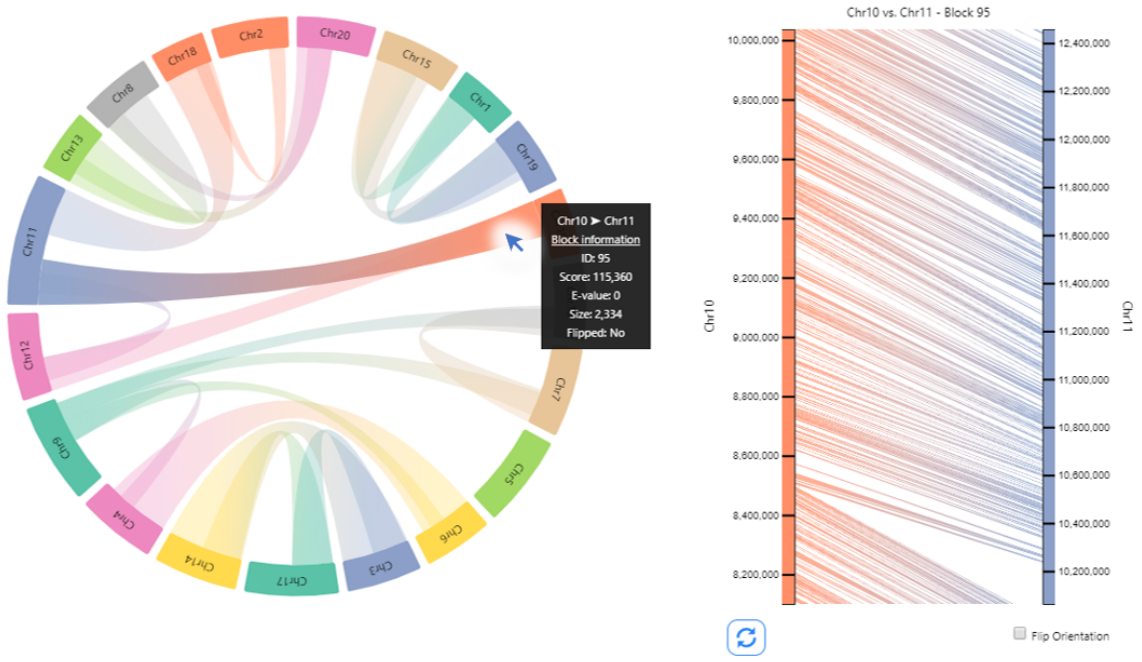

Figure 2.10: Multi View Synteny Exploration in AccuSyn [68] showing conservation in *Camelina sativa* with a single collinear block highlighted between Chromosomes 10 and 11.

A recent set of synteny browsers that are web based and also let users download visualizations as image files are Synteny Portal, MultiSyn, and AccuSyn. Synteny Portal uses alignments that are pre-built and stored in the UCSC genome browser database but cannot be extended for custom sequences [51]. MultiSyn [3] is similar to SimpleSynteny but relies on only protein sequences as its input and detects synteny on the server using MCScanX [128], a popular synteny detection tool. AccuSyn [68] is a recent tool that lets users upload synteny results of MCScanX and generates visualizations in real-time on the client. It presents conservation similar to Mizbee in two linked views with a Circos style layout with connected ribbons for the genome level and a vertical layout with connected glyphs for the block level, as shown in Figure 2.10. AccuSyn also lets users upload several annotation tracks with the ability to customize the colour scale and the visual representation of the track. Finally, AccuSyn attempts to minimize overlaps in the connecting lines by rearranging the chromosomes with a simulated annealing algorithm.

2.3 Interaction Techniques in Genomic Visualizations

Visualization for data exploration and analysis primarily involves graphical representation of the data, but user interaction still plays a major role in both navigation and assessment of the generated visualizations. In this section, we first explore the different techniques that are used in genomic visualization tools to manipulate both the underlying data and the graphical interfaces for data exploration, and then discuss how data analysis can be improved with support for revisitation and tracking user interactions.

2.3.1 Multiple Linked Views

Although visual encoding of information through static visualizations can assist users in simple data analysis tasks, the usability of this approach falls short when it is utilized for complex tasks and activities [20, 83, 84, 104, 122, 140]. The efficacy of such a system can be considerably improved by providing interaction mechanisms that can modify the graphical representations based on the different tasks, users, expertise, and other contextual factors [104]. Complex analysis tasks go beyond basic visual perception needed for simple tasks, and often require extended cognitive processing from the user. Interaction mechanisms are important in these scenarios because engaging user actions can enhance cognitive processing [104]. Research has also shown that viewing the same underlying data in multiple representations can help users in forming an accurate mental model of the data [48, 105, 119, 129].

In the context of comparative genomic visualizations, some of the earliest examples of enhancing basic charts with interaction techniques include support for zooming into a two dimensional dot plot for closer inspection of a particular region in Dagchainer [33]. Tools like SynMap2 [37] belonging to the CoGe (Comparitive Genomics) [85] web platform also allow exploration of dot plots through mouse-based scrolling and panning similar to modern mapping platforms like Google maps. This form of zooming for exploration is also used in several other tools and is part of the larger design principle proposed by Shneiderman [109] called the visual information seeking mantra. The principle summarizes the essential elements of interaction with visualization systems as: "overview first, zoom and filter, then details-on-demand". In synteny visualizations an overview can often mean presenting comparative relationships at the whole genome scale by grouping collinear blocks. At this level visual encoding is used for distinguishing the different chromosomes the collinear blocks belong to and their size, proximity and position. Mizbee is an example of a tool that follows all four essential steps of the visual information seeking mantra [59]. It presents overview level information in the circos style plot, as shown in Figure 2.9, and specific sections of the genome can be zoomed and filtered through markers placed on the overview plot. Additional details of individual gene blocks that make up larger collinear blocks are brought up on-demand through user interactions at the higher level either in the Genome View or the Chromosome View.

mGSV (Multi Genome Synteny Viewer) is another tool that provides a summary view showing genomic conservation at the genome level in a Circos style plot [92]. Unlike Mizbee, which lets users explore using multiple linked views, mGSV provides two operating modes: a pairwise view mode and multiple view mode. In the pairwise mode conserved regions are shown between adjacent genomes and interactions are centered around selecting specific regions of the different genomes and reordering the genomes. In the multiple view mode, conserved regions connecting all visible genomes are shown and interactions are focused around toggling the visibility of genomic regions to ensure they do not overlap over other genomes in the stacked layout. mGSV also employs a heuristic algorithm to optimize the layout of the genome order based on the size of the conserved regions (to minimize visual clutter). However, this can often create layouts that while being visually clear may not provide the right biological context. AccuSyn solves this problem through a novel

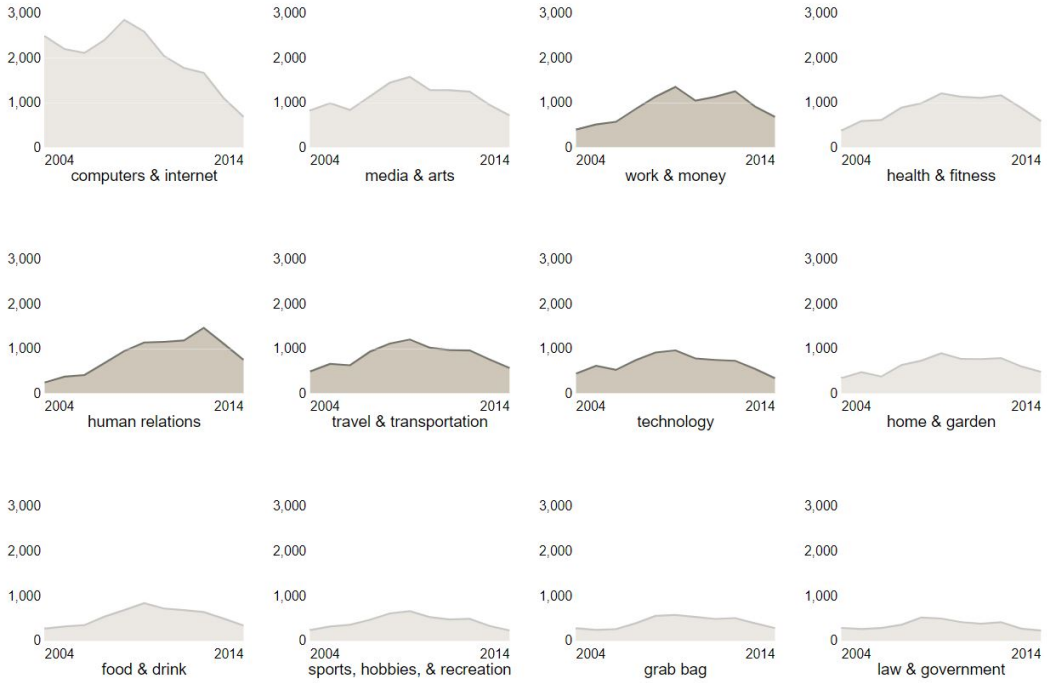

Figure 2.11: Example of Hindsight [26] system that visualizes interaction history by making visited charts appear darker.

Human-in-the-loop methodology [68]. It uses a simulated annealing heuristic to arrive at the optimal layout and also takes into consideration the position of chromosomes set by the users through manual dragging and flipping operations. This can ensure that as users explore the syntenic relationship between the genomes, they can tune the algorithm to arrive at an uncluttered layout that also has meaningful insights.

2.3.2 Interaction History and Revisitation Support

Exploring genomic information at different resolutions can be problematic due to a dataset's volume, complexity, and due to limitations in the availability of visual space but these problems can be addressed through effective interaction techniques. However, relying on interaction mechanisms to navigate visual spaces can come with its own set of problems, one of which is memorability. Memorability is the degree to which users can retain information about the position of objects and markers in a visualization and revisit them. Revisitation is important in data exploration as it can help users retrace their steps and is part of the *history* stage of the information seeking mantra presented by Schneiderman [109]. However, revisitation can be affected by interaction techniques that require users to switch between graphical representations or zoom into a particular region and cause them to lose context of their previous position. Humans primarily rely on spatial cognition to remember the locations of objects in information workspace tasks but context-switching visual systems can disrupt this ability [97]. Similarly, interactive techniques such as fisheye views that work by

distorting the original visualizations, can also impair spatial memory as shown by Skopik and Gutwin [112].

One of the methods by which this problem can be addressed is by storing the interaction history of the system and presenting it as a graphical abstraction either in the visual system itself or as an external panel. Interaction history can show the historical actions performed by a user and can also provide information on the actions performed so far that have led to the current state of the visual system. This interaction history can be provided indirectly through a history widget panel or by direct encoding into the visualization through the notion of *readwear* or *visitwear*. An example of direct interaction history encoding in a visualization system is the visitwear mechanism developed by Skopik and Gutwin for fisheye views to highlight visited nodes. *HindSight* is a similar direct encoding design framework that proved that users were able to visit more data points and recall novel insights by using their framework [26]. It encoded interaction history by making visited charts in a multi plot system appear darker, as shown in Figure 2.11, or relied on the existing visual encoding in line-charts and made visited lines slightly larger.

The second indirect encoding method for interaction history stems from research in visual analytics for supporting *provenance* - which is the history of steps that led to a particular result in a data analysis workflow [28, 31]. The concept of provenance of visual history is centered around systems like the *VisTrails* tool [6] that let users save visual outputs and revisit earlier states in the data analysis process. The notion of the earlier states termed as *graphical histories* is explored further by Heer et al. [38] in their graphical history interface for the Tableau visualization system. In the context of genomic visualizations to the best of our knowledge direct encoding of interaction history has not been explored but some recent synteny visualization tools like AccuSyn have offered partial support for revisitation through indirect encoding by letting users manually capture states of the system and revisit them through a history panel [68].

3 Data and Task Abstraction

SynVisio was developed over multiple iterations as the necessary data and requirements were refined through continuous consultation with our genome research collaborators. To understand the design choices for visual encoding and interaction that we adopted in developing SynVisio, we need to first elaborate on the underlying data and task abstractions. We start with the data abstraction, where we explain the different characteristics of the syntenic data and how it is computed using synteny detection tools and further processed by our system. We then explore the different visual analysis tasks that can be performed on syntenic data, and finally, we discuss additional interactive and usability requirements of SynVisio that can arise when users explore complex datasets and biological scenarios.

3.1 Data

This section starts with the description of the structure of a genome and its constituent elements and follows through with an explanation of how syntenic data is generated and represented.

3.1.1 Genome Structure and Scales

The genome of every organism is unique and can be defined as the complete set of DNA needed to build and maintain that organism. Structurally, genomes are broken down into smaller sections called chromosomes, where every chromosome is a long strand of DNA coiled up along with various proteins. Each chromosome is made up of several genes, which are the basic functional units of heredity and which code for a specific protein. Genes can then be further broken down into nucleotides which are the smallest building blocks of DNA. For analyzing genomic conservation, researchers also look at a collection of collinear genes that are called blocks. Thus the genome structure can be ordered into the following five levels from top to bottom in terms of genomic size : *Genome→Chromosome→Block →Gene→Nucleotide.* However, to analyze large scale genomic conservation we only look at the first four levels. The structural data of a genome describing its constituent entities is provided in the form of a *GFF (General Feature Format) File.* It contains the start and end position of every gene on a linear scale in a chromosome, the gene identifier, and the reference name of the parent chromosome in a three column tab-delimited format. A partial GFF file can be seen in Figure 3.1. Information on several genomes belonging to different species can be presented in the same file; each species is distinguished by the two-character key present in the reference name of the chromosome. The data in the GFF file is processed by SynVisio to get the genomic size (number of nucleotides) of the different

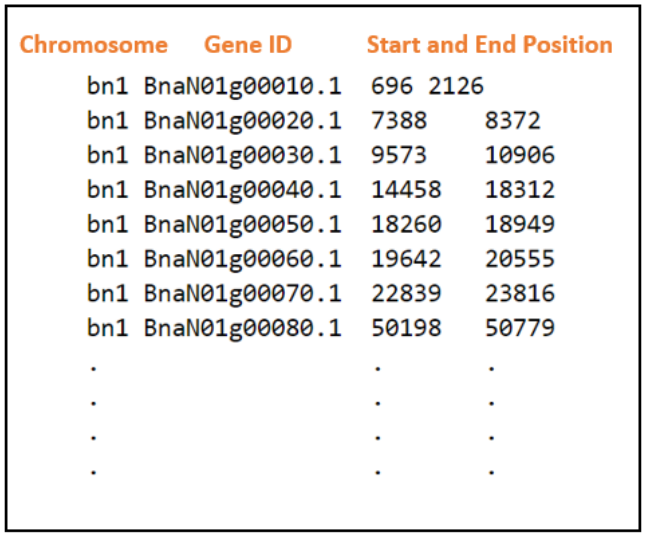

Figure 3.1: Partial GFF file describing structure of a genome.

genomes and their constituent chromosomes along with the size and position of all the genes within these chromosomes. This data gives us a precise structural map of every genome and its different sub-elements and so can be used to visualize a genome over multiple scales and levels.

3.1.2 Conservation Data

At the smallest level, conservation between two genes can be inferred by looking at sequence homology, which is the similarity between nucleotide sequences. Larger genomic conservation events can be studied by looking at blocks of such homologous genes and grouping them together based on their chromosomal positions to identify collinearity. To arrive at this data, we first identify all the homologous genes between two genomes using a local alignment tool such as BLAST (Basic Local Alignment Search Tool) [2]. Different synteny detection tools then construct collinear blocks of these homologous genes by either clustering neighbour matching gene pairs (ADHoRe, OrthoCluster) [89, 141] or by constructing chains around an anchor gene (MCScanX) [128]. These collinear blocks are referred to as syntenic blocks, and are the primary source of data for SynVisio and are provided as a collinearity file. A partial sample of a collinearity file can be seen in Figure 3.2. Every block of collinear genes has a corresponding similarity score indicating the quality of match; an expect value (*E-value*) indicating the probability that the match may have been due to chance; the count of genes; the names of the source and target chromosomes that the block of genes belong to; and the orientation of the block (forward or reverse). Finally, every block of data also consists of a list of the homologous gene pairs in that block and their statistical significance (*E-value*). This data, combined with the information about the structure of the genome can be used to associate every region of a genome with its homologous regions in the other genome or within itself depending on the type of synteny under investigation.

```
# Alignment 1: score=371.0 e_value=1.3e-13 N=8 at1&at1 plus
 1-  0:  AT1G13730 AT1G69250   6e-48
 1-  1:  AT1G13740 AT1G69260   8e-58
 1-  2:  AT1G13830 AT1G69295   6e-20
 1-  3:  AT1G13920 AT1G69325   1e-14
 1-  4:  AT1G13940 AT1G69360   1e-82
 1-  5:  AT1G13950 AT1G69410   1e-79
 1-  6:  AT1G14010 AT1G69460   3e-79
 1-  7:  AT1G14040 AT1G69480      0
# Alignment 2: score=345.0 e_value=2.8e-13 N=8 at1&at1 plus
 2-  0:  AT1G59730 AT1G69880   9e-30
 2-  1:  AT1G59830 AT1G69960  3e-174
 2-  2:  AT1G59890 AT1G70030   7e-37
 2-  3:  AT1G59910 AT1G70140  2e-146
 2-  4:  AT1G59970 AT1G70170  2e-109
 2-  5:  AT1G60050 AT1G70260  4e-126
 2-  6:  AT1G60140 AT1G70290      0
 2-  7:  AT1G60170 AT1G70400   1e-58
# Alignment 3: score=336.0 e_value=4.6e-14 N=8 at1&at1 plus
 3-  0:  AT1G52150 AT1G79840   1e-20
 3-  1:  AT1G52240 AT1G79860  3e-170
 3-  2:  AT1G52315 AT1G79910   5e-41
 .   .   .         .           .
 .   .   .         .           .
```

Figure 3.2: Partial collinearity file with a single block highlighted.

Chromosome	Start and End Position		Data
chr1	6400001	6800000	100
chr1	6800001	7200000	100
chr1	7200001	7600000	92
chr1	7600001	8000000	92

Figure 3.3: Sample track file.

3.1.3 Auxiliary Track Data

Since genomic data is represented as a collection of linear sequences, additional information can be provided in the form of tracks parallel to the original gene sequence structure. These tracks can contain information about genomic features such as gene density, copy number variations (CNVs), and single-nucleotide polymorphisms (SNPs). The data is provided to SynVisio in a BedGraph file format consisting of four tab-delimited values: chromosome identifier, chromosomal start position, chromosomal end position, and a data value. This information can be hierarchically grouped at the block or chromosome levels in the same way as genomic sequences and can be used to annotate the corresponding sequence structure.

3.2 Tasks

In this section, we look at the different questions researchers have when analyzing genomic conservation. We then collate all the design requirements into a series of analysis tasks and group them based on the conservation relationship they address. Finally, we elaborate on additional interactive requirements that improve the usability of the system and assist users in performing complex analysis tasks.

3.2.1 Requirement Gathering Phase

To formulate our design requirements, we met with three groups of researchers working in different fields of biological research. The meetings were conducted over several iterations to refine our initial requirements. The sessions broadly revolved around understanding the basic tasks researchers perform when analyzing genomic conservation and looking at the shortcomings of the existing synteny analysis tools. Although all three groups were primarily interested in analyzing synteny, their individual use cases varied, providing us with a diverse set of user scenarios.

Our first research group was involved in investigating genomic conservation in the *brassica* genus as it offers an ideal model to study polyploid evolution, which is responsible for genetic variations that are advantageous from an evolutionary perspective [53, 56]. In particular they were interested in understanding genomic conservation within an allotetraploid species *Brassica napus* (AACC), an important oilseed crop, and also in comparing it to the closely related diploid species *Brassica rapa* (AA) and *Brassica oleracea* (CC) that belong together in the classical triangle of U [61]. The requirements from this research group were focused around having access to a system that could let them visualize the conserved relationship between the different diploid species and also within the chromosomes of a single allotetraploid (i.e. self-synteny).

The second research team was interested in looking at genomic conservation between *Lens culinaris* and *Cicer arietinum* to improve various agronomic traits, as these are both widely grown legume crops. The requirements here were largely focused on cross synteny rather than self synteny. A unique trait of this particular dataset is that while *C. arietinum* has a genome size of 740 Mbp, *L. culinaris* has a genome size of 4 Gbp. This large difference in the sizes of the two genomes makes visualizing synteny at the whole genome level difficult and researchers must hence rely on comparing individual chromosomes of *L. culinaris* one at a time or use a visualization that can have variable scales between the source and the target genomes.

The final research group works on sequencing wheat, and the researchers in this group were interested in understanding the genomic conservation between the three subgenomes of the hexaploid bread wheat genome. Researchers from this team wanted a system that could visualize synteny between three different genomes instead of a single source and target, while taking into consideration the extremely large size of the wheat genome. They were also interested in adopting a novel network-based visualization called a Hive Plot [45], which maps nodes onto radially distributed linear axes, for exploring multi-way genomic conservation.

3.2.2 Tasks

The requirements gathered from our research collaborators can be grouped into primary visual tasks for observing conservation and interactive system tasks for exploring complex datasets and biological scenarios. Further based on the underlying data, the primary visual tasks can be ordered into three basic groups according to the genomic scale at which they operate.

Primary Visual Tasks

Genome Level

Q_1. What is the level of conservation that exists between two or more sets of genomes?

Q_2. How does the density of conservation change across the genomes, and are there any gaps?

Q_3. How does the ordering of chromosomes based on conservation change between a given set of genomes or within a single genome? (possibility of detecting whole-genome duplication or genome reversal)

Q_4. If unmarked scaffolds exist, which regions of the target genome do they share similarity with?

Q_5. Which chromosomes are sparsely or entirely unaligned, and how does the level of conservation change when these are ignored?

Chromosome Level

Q_6. What is the level of conservation between a specific subset of chromosomes?

Q_7. What is the level of conservation between a single chromosome and an entire target genome or several other chromosomes (detecting unaligned regions within a chromosome)?

Q_8. How large are the collinear blocks relative to neighbouring chromosomes?

Q_9. What is the orientation of collinear blocks between two given chromosomes? (regular or inverted)

Block Level

Q_{10}. What is the level of conservation between the set of genes in a collinear block?

Q_{11}. What are the different genes contained in a block?

Q_{12}. What is the size of a gene relative to the size of the collinear block?

Q_{13}. Are there large gaps between genes in collinear blocks?

Q_{14}. Answer Q_{10}-Q_{13} when a collinear block is reversed?

There are also several other analysis tasks that researchers perform that go beyond the block level and down to the nucleotide level. However, these tasks are beyond the scope of this research work and there are several other systems specifically designed to investigate collinearity at the nucleotide level, such as JBrowse [111].

Interactive System Requirements

R_1. **Dynamic refinement of visualizations**. Synteny analysis focuses on identifying conserved regions in specific parts of the genome with the ability to focus on distant or close matches from an evolutionary perspective. Researchers need to be able to filter the generated visualizations in real time based on features of the conserved region such as the level of match and its chromosomal position.

R_2. **Multiple perspectives on the dataset**. Researchers who undertake complex analysis tasks require multiple coordinated views that show different visual representations, each focusing on a particular primary feature of the dataset such as the orientation (dot plot) or the location (parallel plot). Researchers integrate the different perspectives in different ways as they carry out their tasks.

R_3. **Dynamic visualizations of multi-scale data**. Genetic conservation is often explored at several levels (genome, chromosome, or gene block), and the focus of analysis can be different at every genomic scale. Visualization systems should therefore allow users to switch scale quickly and easily, and should provide capabilities and interactions that adapt based on the scale of the investigation.

R_4. **Augmenting visualizations with secondary data**. Insights in synteny analysis can be gained by looking at conservation in the context of gene density or the positions of genetic anomalies (single nucleotide polymorphisms). Therefore, systems should offer researchers the ability to add layers of visual information onto the basic visualizations, using the main representation as a reference frame.

R_5. **Visualizations of multiple genomes**. Multi-way visualizations can let researcher trace conservation across several genomes, thus offering a novel way to visualize synteny combined with phylogeny (i.e., the evolutionary relationship between two species).

R_6. **Navigation and revisitation support**. Synteny analysis is often used in the hypothesis-generation stage of research, requiring that genome scientists explore several scenarios through the analysis tool. This can be problematic when genomes are large (e.g., the wheat genome is 17 Gbases – six times larger than the human genome), because researchers can easily lose context of their location in the genome (particularly in polyploid organisms where genes are duplicated multiple times). In addition, a complex visualization system also presents a large "parameter space" requiring that users remember the settings and navigation actions that brought them to their current viewpoint. Analysis tools therefore need to support navigation and record provenance in order to enable communication between collaborators and to enable revisitation of potentially-interesting locations during exploration.

4 Visual Design

Visualizing syntenic data is a multi-faceted problem as not only can the visual representation change based on the underlying biological question but also the resolution at which the information is being visualized. In designing a solution for this problem, the taxonomy of design space created by previous synteny visualizers like Mizbee [59] was adopted and further enhanced with our recommendations for representing synteny in multi-way comparisons. This taxonomy of design space is centred around using visual variables to highlight the location, size and orientation of the conserved regions. In designing SynVisio we used two primary forms of visual representations, a parallel plot and a dot plot, and modified them to work across multiple resolution levels. We also created hybrid plots based on our parallel plot design to represent synteny in multi-way comparisons. In this chapter, we first discuss the different forms of visual encoding used in our system. We follow this with a description of the different layout strategies that were explored. Then we elaborate on the various interaction strategies we adopted based on the visual information seeking mantra framework and design of multiple coordinated views. Finally we end the chapter with a discussion on our iterative design process through four major development cycles.

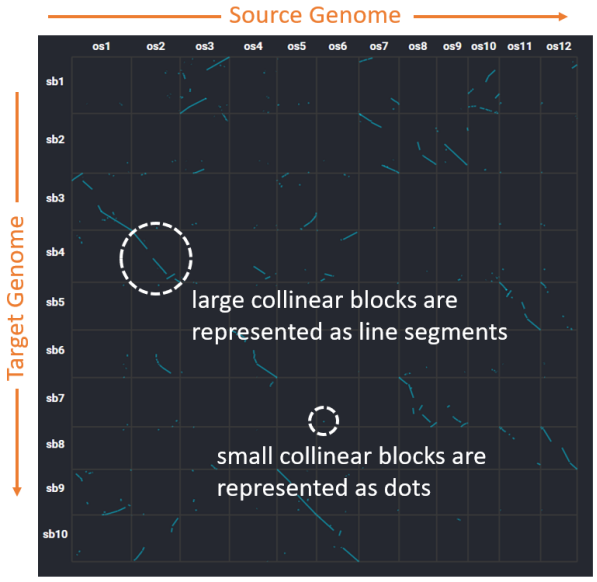

Figure 4.1: Dot plot showing whole genome synteny between Rice *(Oriza sativa)* and Corn *(Sorghum bicolor)* with grid-lines added for chromosomal boundaries.

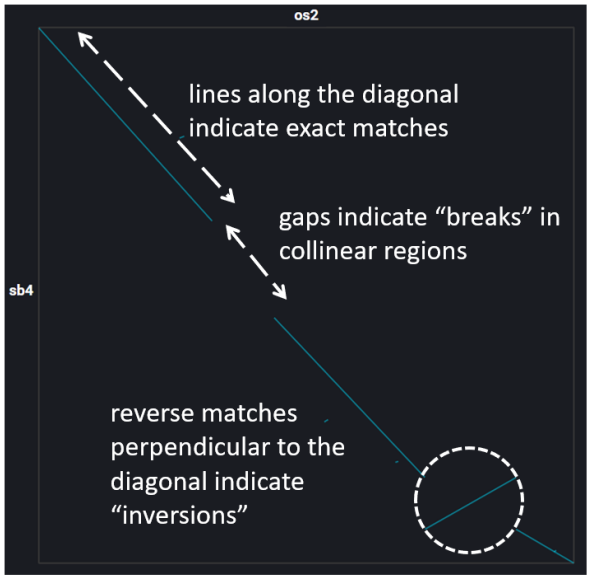

Figure 4.2: Dot plot showing breaks, inversions and duplication events between chromosome 2 and 4 of Rice *(Oriza sativa)* and Corn *(Sorghum bicolor)* respectively.

4.1 Visual Encoding

A common way to represent sequence alignment or similarity is to visualize it as a two-dimensional ‘dot plot’ [10, 116] through positional encoding. We adopted this strategy for our first visual representation by placing the source and target genomes along the x and y axes, respectively, and marking gene alignments with dots as shown in Figure 4.1. Grid-lines were then further added to the plot to indicate chromosomal boundaries.

This plot can also be adopted for other resolutions by changing the genomes along x,y axes to either individual chromosomes or smaller gene blocks. Such matrix-based representations are very good at providing an overview of the dataset and can be used to highlight breaks, inversions, and duplications, as shown in Figure 4.2. However, being a relatively primitive visual representation dot plots are often found to be visually unappealing and complex to understand without the proper background context, making them unsuitable for a variety of exploration and communication tasks.

For our other primary visual representation, we adopt a design that represents synteny through a combination of positional encoding for genomic distances and connected lines for similarity. In this approach, genomic sequences are stacked horizontally parallel to each other, and similar genes are connected through lines to indicate similarity. However, unlike dot plots that use the same visual encoding across all genomic sizes, for this visual representation, we adopt a different secondary encoding based on the resolution of the genomic sequences being visualized.

There are three basic levels in which synteny can be visualized starting from the gene block level, which is the smallest unit at which syntenic data is reported. A gene block is a collection of collinear genes in the

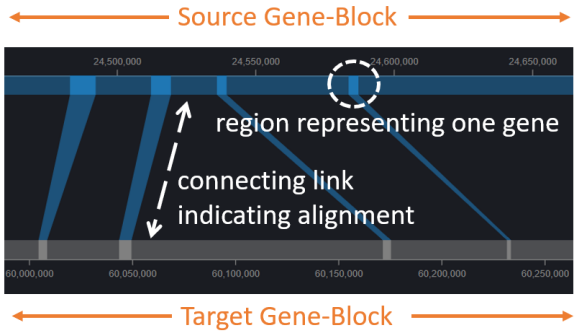

Figure 4.3: Parallel plot at the gene-block level

source genome that are aligned to a group of collinear genes in the target genome. To encode conservation at this level, we use two gene blocks that are represented by line segments, and are stacked parallel to each other. Similar genes within the blocks are then connected with ribbons, as shown in Figure 4.3. The connecting ribbons are four sided polygons whose edge widths are dependent on the number of gene pairs in the collinear block at each edge. The source and target gene blocks are annotated with numeric tracks corresponding to their position in the chromosome and are coloured in distinct colours to distinguish them. The individual genes are represented as rectangles highlighted with a deeper shade of the base colour of the track for easier reference.

At the next level, individual chromosomes are considered since a collection of gene blocks form a chromosome. Visualizing synteny at this level involves encoding information related to the location, size and orientation of conserved regions. To achieve this, chromosomes are stacked parallel to each other and their lengths are encoded to reflect their genomic size. So chromosomes with more base-pairs in them show up as wider line segments. Conserved regions in the chromosomes are then connected through ribbons from their positions on the chromosome to indicate similarity. This encodes both the location and the size of the conserved regions as the width of ribbons changes based on the genomic size of the linked gene blocks.

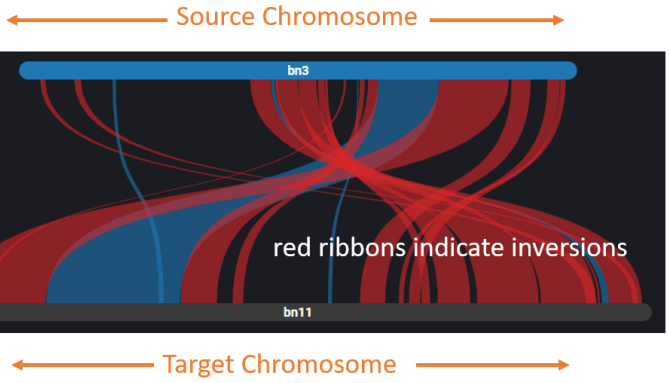

Figure 4.4: Link Plot at the Chromosome level where the blue coloured ribbons represent forward matches and the red coloured ribbons represent reverse matches (inversions).

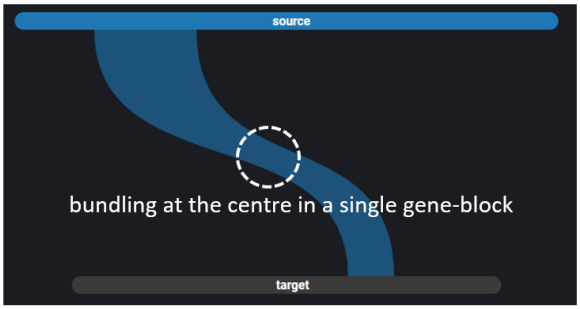

Figure 4.5: Ribbon bundling to reduce visual clutter with the control points set towards the centre indicated in a single gene-block.

To encode the orientation of the gene block, secondary encoding in the form of colour is adopted to visually distinguish gene inversions, as shown in figure 4.4. So forward matches are coloured in blue, and reverse matches are coloured in red. Unlike the gene block level, at the chromosome level several bands can overlap and cross each other due to multiple gene translocation and inversions events and can cause visual clutter. To mitigate this problem, complex polygons are used instead of rectangular ribbons and are generated through **B**-spline curves [86] with control points set to bundle the curves towards the centre [143]. The control points are adjusted to be vertically in the middle of the parallel blocks to ensure that the original size of the ribbons remain undistorted at regions where they join the chromosome as they visually represent the size of the conserved region as shown in figure 4.5.

Finally, at the whole genome level where synteny is observed between several chromosomes at once, the chromosomal source of the collinear regions is given higher priority, and so secondary encoding in the form of colour is used to distinguish different chromosomes. A layout similar to the parallel stacking at chromosome level is adopted, however, instead of having a single connected unit for the entire genome, chromosomes are separated from each other with gaps serving to indicate the start and end of each chromosome. Chromosomes in the source layer are assigned a unique colour, while chromosomes in the target layer are assigned colours through an alternating gray and black colouring scheme. Ribbons are then linked between conserved regions to represent syntenic gene-blocks and are assigned a colour based on their source chromosome. This form of encoding location information about the source in the connection through colour has been used earlier in other synteny visualisation systems and has been proved effective [59]. We adopt the aforementioned bundling strategy of using **B**-spline curves [86] to improve visual clarity but set the control points independently for every chromosome to group all the gene blocks emerging from each chromosome into a single bundle. Finally, inverted regions at this level are represented through ribbons that are wide at the extremities and pinched to be only a pixel wide at the centre which gives the visual impression of a flipped ribbon.

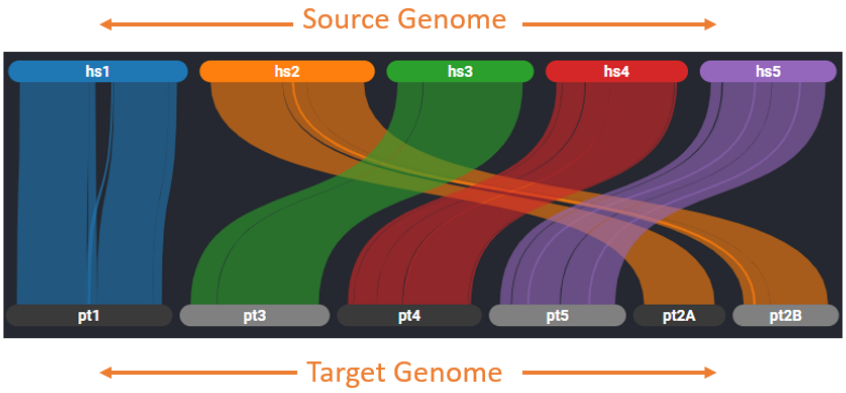

Figure 4.6: Visual encoding at the chromosome level with connecting ribbons coloured based on the source chromosome they are linked from.

4.2 Layout Strategies

A common strategy that is used among all the three parallel stacked representations is the vertical separation between the source and the target to visually distinguish the two regions. This is easy to implement at the gene-block level, and the chromosome level as the source and target regions are single continuous entities but requires minor adaptations at the genome level. The genome is a combination of several chromosomes, so each chromosome had to be individually distinguishable while still being represented as a part of the whole source group and different from the target group. To achieve this grouping, we use the visual law of proximity from Gestalt principles [136] which states that proximity can override other visual similarities (shape, size, colour) to differentiate a group of objects. and represent each chromosome as a pill-shaped region and then lay them out end to end horizontally with small gaps between them. The gaps between the chromosomes achieve the task of making the chromosomes look distinct and also being smaller than vertical gaps between the two genomes, clustering the source and the target regions into two separate groups visually.

In arriving at the optimal layout strategy, we looked at several different alternative ways of arranging the chromosomes. In the popular synteny browser MizBee [59] the authors provide a taxonomy of the different synteny layouts and broadly classify them into two categories: contiguous and discrete. In the former, the chromosomes are presented adjacent to each other either in a linear or a circular layout, and in the later, chromosomes are treated as distinct elements, and presented either in segregated groups or interleaved with each other. In our design we go for the contiguous scheme, but we omit the circular layout as it has already been explored in AccuSyn [68] and instead look at possible linear layout strategies where conservation is encoded through connections as shown in Figure 4.7. In the vertical (a) and horizontal layouts (b) the underlying approach is similar except for the orientation of the two parallel layers. However the number of chromosomes in a genome can be numerous as in the case of humans who have 23 and can cause the vertical layout to be quite long. This makes it sub-optimal for the horizontal "landscape" aspect ratio of most computer monitors. Therefore of the two, the horizontal parallel layout is the preferred mode of encoding

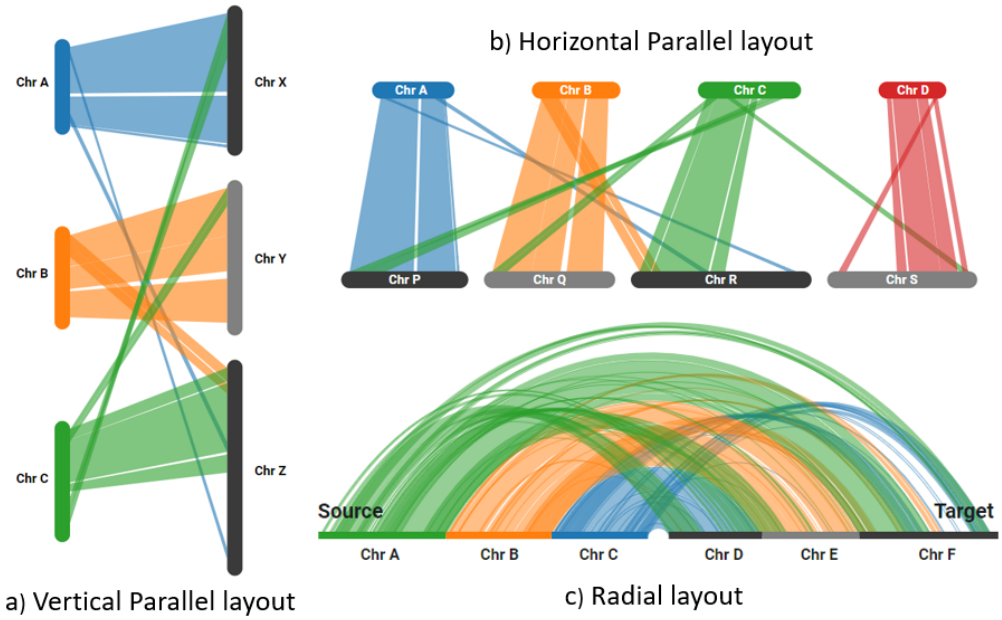

Figure 4.7: Different layout strategies at the genome level with conservation being encoded as connections.

synteny. A common advantage of these two layouts is that they can be stacked in multiple layers such that chromosomes at every level are linked to both chromosomes above them and also the chromosomes below them, as shown in the layout (a) in Figure 4.8. This stacked layout strategy is used to represent synteny in the form of a tree view chart and can be particularly useful in scenarios where conserved regions need to be traced across several evolutionary levels.

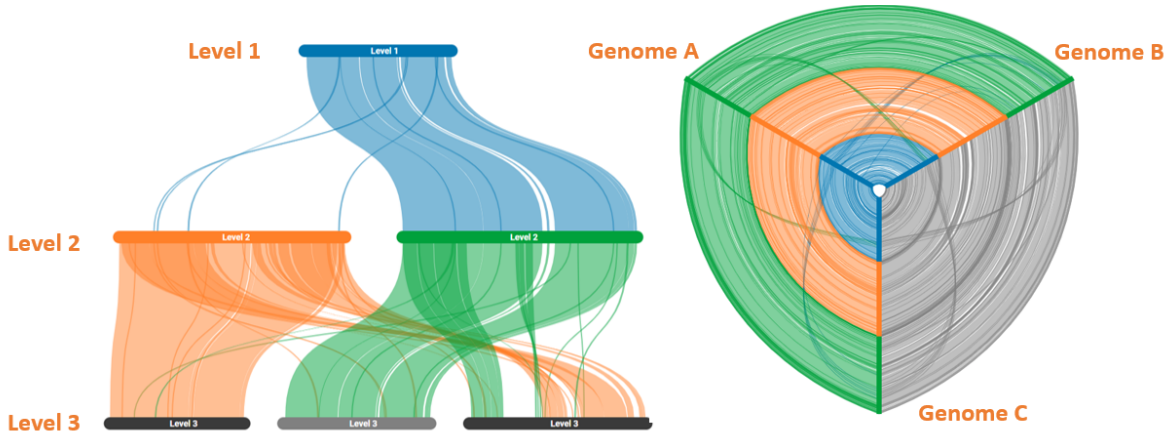

Figure 4.8: Multi-level layouts: Parallel layout (left) and Radial layout (right).

The bi-directional linking strategy is however, unavailable in the parallel layout scheme in pairwise comparison scenarios but can be utilized by moving the two layers adjacent to each other, as shown by the layout (c) in Figure 4.7. In this layout, the chromosomes are in the same level, making it possible for conserved re-

gions to be linked in two directions either from the top or from below. This allows us to include an additional layer of encoding. For example, if we had to represent the orientation of the conserved regions, we could link all forward matches through connections from the top, and all reverse matches through links from the bottom. The disadvantage of this layout is the high number of crossing between the connections. This can be made worse in scenarios where there is a high degree of collinearity between the two genomes due to the ordering of the chromosomes as every connection between the first chromosome in the source and the first chromosome in the target is crossed by all other connections emerging from the rest of the chromosomes in the source. An alternative approach to solve this problem includes reversing the layout of one of the layers or arranging the chromosomes in a radially outwards fashion in both the layers. This layout can also be extended to express synteny in multiple levels by merely increasing the number of radial layers such that each layer is connected to both the layer on its right and the layer on its left, as shown in the layout (b) in Figure 4.8.

Irrespective of the ordering or the layout of the chromosomes, they all have unique positions in a visual representation. This prevents them from being occluded by other items in that representation. This however is not the case for visual encoding of conservation through lines or ribbons that can overlap significantly in some instances. The Z-axis position of the ribbons (i.e., the stacking order in the view) decides which ribbons occlude others and is dependent on the rendering order of the ribbons. To address this problem, we sort the conserved genes in every chromosome based on their gene count. This places smaller gene blocks at the end of the list ensuring that these blocks are rendered last, making them appear higher on the Z-axis and thus above the bigger blocks. To further solve this problem, connected ribbons are rendered with 75% transparency, which ensures that even if ribbons end up overlaying other connected ribbons, they are still visible on the screen.

4.3 Visual and Interaction Design

Having discussed the different ways in which syntenic data can be encoded, we can summarise that the usefulness of a particular representation depends on the syntenic relationship under investigation. This has created a need for an adaptive system that can be used under a wide range of scenarios spanning investigation of micro synteny all the way up to high-level genome duplication events. Syntenic data also goes beyond the basic location, size and orientation of the conserved regions and includes additional information such as the match score and the E (expect) value, which indicates the level of similarity and the probability of a match occurring by chance. This inherent complexity in the dataset means that exploring it becomes challenging as the volume of the data increases. Thus in designing our synteny exploration interface, we build on the framework of "visual information seeking mantra", which offers standard visual design guidelines for developing information visualization applications [109]. The framework can be used to break down synteny analysis broadly into the following tasks: overview, filter, zoom, details-on-demand, history, and extract.

We present information in a top-down tiered approach in three distinct scales starting with the whole genome followed by stepping down into an individual chromosome and finally ending on the gene block level. Users are given the ability to start their investigation of the data at any particular level and pick either a dot plot, or a parallel plot, or a combination of both. Users are then given the option to interact with the visualizations in real-time to either filter the chart to look at conserved regions in a particular chromosome or drill down into the dataset all the way down to an individual gene in a conserved region as shown in Figure 4.9. Additional details about the syntenic blocks are available on-demand through hover-based mouse interactions either with the ribbons or the dots based on the type of representation.

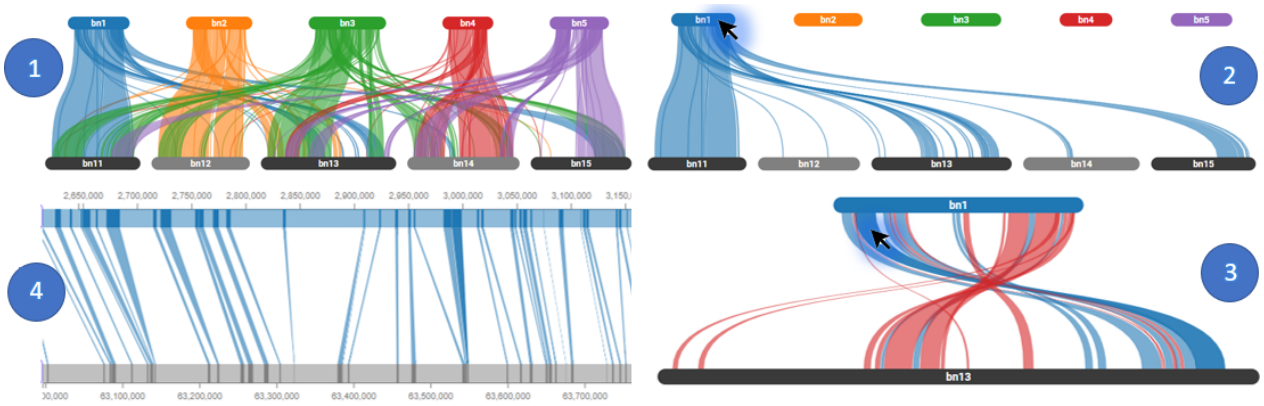

Figure 4.9: User interactions in exploring conserved regions in a top down approach through four steps pictured in clockwise fashion.

Our design also incorporates a dashboard for exploration where instead of relying on a single visualization, information about conservation at every level is presented through coordinated multiple views. This is built on the underlying premise that users have a better understanding of their data if they interact with the given information and view it through different representations [96]. In our design of multiple distinct views that support the investigation of a single entity, we followed the design guidelines set by earlier research into multiple coordinated views in information visualization systems [130]. We present the following three distinct views, each highlighting a unique facet of the dataset: parallel plot, dot plot and a simple scatter plot that acts as a filter toggle. The parallel plot offers position and location information about the conserved regions at a glance while the dot matrix plot can easily highlight reversals and deletions within the conserved regions. Both the views are linked to each other to ensure that users don't lose context of their interaction. This is done by mirroring all actions between the two views. The final scatter plot is used to present the measure of similarity of the conserved regions and has a slider built into it that can be used to filter conserved regions based on their match score or E value. The filter works synchronously with the other two views and as the user moves the slider, the conserved regions are filtered in real time in the other two views. Finally, in keeping with the visual information seeking mantra framework, user interactions are recorded with users having the ability to store all their interactions in arriving at a particular analysis point as a visual stamp that can be revisited or reset back to in-case of further analysis.

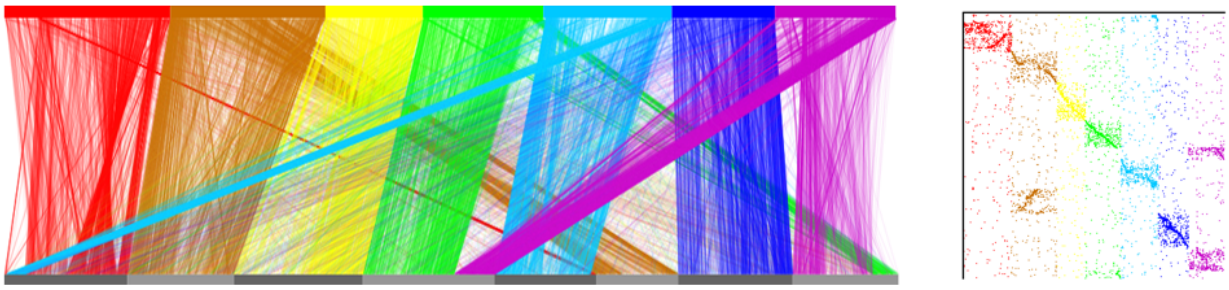

Figure 4.10: Visual representations after the first development cycle consisting of a parallel plot (left) and a dot plot (right).

4.4 Iterative Development Process

By following the guidelines of a standard design study methodology [60], our system design occurred iteratively over four main development cycles. At every stage, we presented our system to a panel composed of genome researchers and information visualization experts to gather feedback and look at possible avenues of improvement. After the initial requirement gathering phase we sketched a prototype of a linear parallel plot and a dot plot both visualizing conserved regions at the genome level. For our first development cycle, we used basic colours for distinguishing different chromosomes and encoded every collinear gene as a connecting link in the parallel plot and as a single point in the dot plot, as shown in Figure 4.10. Although this approach helped in highlighting large scale patterns in genomic conservation, encoding every single gene meant that there was significant noise in the visualization that made it difficult to see the smaller conserved regions. Feedback was also directed at the choice of colour and and the layout of the lines in the parallel plot that caused them to visually occlude the lines right below them.

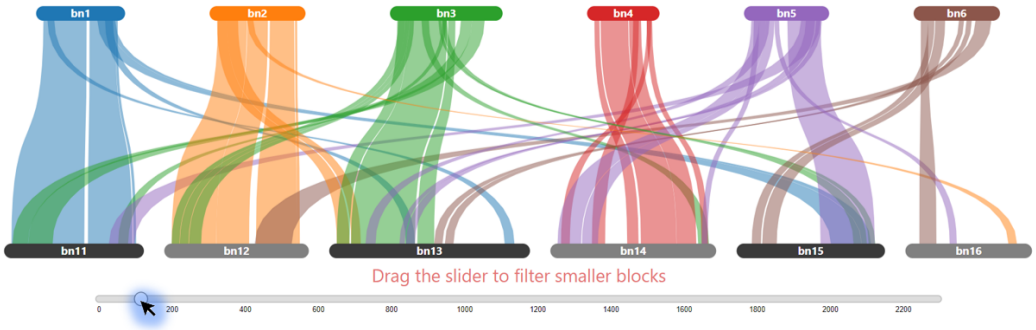

Figure 4.11: Design after the second development cycle with a slider filter.

For our second development cycle, we adopted two major changes: we divided the syntenic data into three levels and limited top level visualizations to just representing collinear gene blocks instead of their constituent genes and, we added interaction features that let users select the chromosomes they wanted to observe instead of looking at the entire genome. Although grouping collinear genes into larger blocks helped

reduce visual occlusion, this problem was further addressed by an updated colour palette and converting connecting ribbons into curved B-spline ribbons instead of straight rectangular strips. Since dot plots are based on positional encoding we removed colours to identify chromosomes in the dot plots and instead relied on grid lines to act as chromosomal boundaries. New levels at the chromosome and the gene-block level were also added to visualize conservation at smaller scales. Finally, a simple slider interface was added to filter conserved blocks based on their gene count. Feedback at this level was largely directed towards the filter feature which despite being helpful was counter-intuitive to synteny exploration as the relevance of conserved regions is based on a combination of the level of similarity , the probability of the match occurring by chance and its constituent genes and not just the gene count. This is because filtering conserved regions is highly dependent on the subject under exploration and hence it needs to be contextually adaptive.

Development in our third iteration was structured around building a system that could be adapted for use in a wide range of scenarios. We developed a composite dashboard with coordinated views and a context-aware filter to help users in making a better-informed decision when they are filtering syntenic blocks. To improve usability, we also added multi-level comparison charts such as hive plots and stacked parallel plots to represent synteny beyond simple pairwise scenarios. For our fourth and final development iteration, we added features that were designed to improve user engagement with our interface, such as gene search and addition of secondary visual encoding in the form of heat-map or histogram tracks to the chart. Visual encoding for the gene search feature was primarily done through colour and vertical positioning. A gene block that contained the gene being searched for was coloured in white and brought to the front above the other gene blocks in terms of on-screen rendering in order to highlight it.

5 SynVisio

Our most significant contribution in this research was the development of SynVisio, an online platform to explore synteny by mapping syntenic blocks that are highly conserved and long enough to be significant between a given pair of genomes or within a single genome. In this chapter, we first discuss the different modes SynVisio offers for synteny analysis and how each one operates. We then explore the various features SynVisio provides to enhance user experience with the tool. Finally, we discuss the software implementation of the tool and then elaborate on the choices made in the web architecture of the system.

5.1 System Overview

SynVisio is a multi scale genome browser that can be accessed through the web to explore genomic conservation. It lets researchers upload output files of a synteny detection system of their choice and generates visualizations from the information in these files. It offers two analysis modes: a primary analysis mode and a multi genome analysis mode. In the first mode, users can compare genomes two at a time through a dashboard where synteny is visualized as both a dot plot and a linked parallel plot. The charts are accompanied by a filter panel where the conserved genomic blocks can be filtered based on features such as the degree of similarity. In the second mode, researchers can compare several genomes at a time through multi-level representations such as hive plots and stacked parallel plots. To aid researchers in their visual exploration of synteny, SynVisio lets them annotate the generated charts with additional tracks in the form of histograms, heat-maps and other basic plots. Additional features are also provided, such as a gene search panel to look for specific genes by gene ID and the ability to export generated charts for publication.

5.2 Analysis Mode

Gene sequences can be compared in different ways, depending on the underlying biological question. This means synteny analysis can vary from visualizing simple pairwise matches between two genomes to performing multi-way comparisons across several genomes at once. The availability of datasets and their inherent quality also plays into the kind of analysis that can be done. Whole-genome alignment, for example, is usually done pairwise as looking for matches can be faster when the subset of available matches is low. Additionally, in the context of synteny detection, which is anchor-based (centered around genes), identifying common markers between multiple genomes is difficult [128]. However when the data is available, multi-way comparisons can

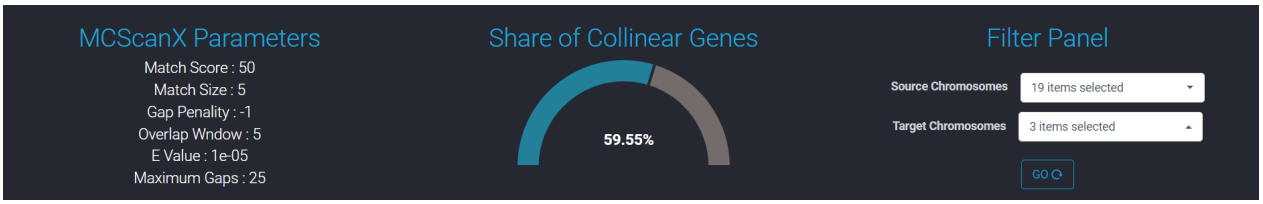

Figure 5.1: Synteny detection parameters and level of collinearity presented along with toggles to select source and target chromosomes.

offer better insights and tackle more significant questions like pan-genome synteny. Thus SynVisio offers both a primary analysis mode for simple pairwise comparisons within a genome or between two genomes and a multi-genome analysis mode for analysing conservation across several genomes, depending on the researcher's choice and the availability of data.

5.2.1 Primary Analysis Mode

This is the default mode in which SynVisio operates and is meant for exploratory tasks as it presents the collinearity between a selected set of chromosomes in two different visual representations and lets users filter the collinear blocks in real time. Although our system operates as a dashboard with multiple representations in coordinated views, we also offer users the ability to look only at one particular representation through the configuration page. This is meant to make our system unopinionated in the choice of visual representation and let users decide on how they want their data to be visualized.

The first step involved in using the dashboard is providing an input dataset; for this, users can either upload their own datasets or use existing sample files. We have already processed several datasets depicting genome conservation on a wide range of species. These are available on the homepage of our application and are updated on a regular basis. Some of the examples include self synteny in *Brassica napus* (canola), cross synteny between *Oriza sativa* (rice) and *Sorghum bicolor* (broom-corn) and cross synteny between *Arabidopsis thaliana* (thale cress) and *Vitis vinifera* (grapevine). After the initial data uploading and processing stage is complete, basic information about the parameters used in the synteny detection process is presented along with the overall collinearity present in the files accompanied with toggles to select the source and target chromosomes as shown in Figure 5.1. If outputs of synteny detection systems other than MCScanX are uploaded, the parameters tab and the percentage share of collinear genes chart are replaced with a textual panel describing the features of the data on file including a list of all unique chromosomes and the total number of collinear blocks. The list of chromosomes is ordered alphanumerically to divide the different species into distinct groups and make it easier to pick chromosomes sequentially. Additional buttons are also provided to either *Select All* or *Clear All* in both the drop-down lists intended for choosing chromosomes. The dashboard operates in three views based on the level of genomic resolution, and each of these stages are described individually below.

Genome View: This view is chosen by default when a user selects more than one chromosome in the

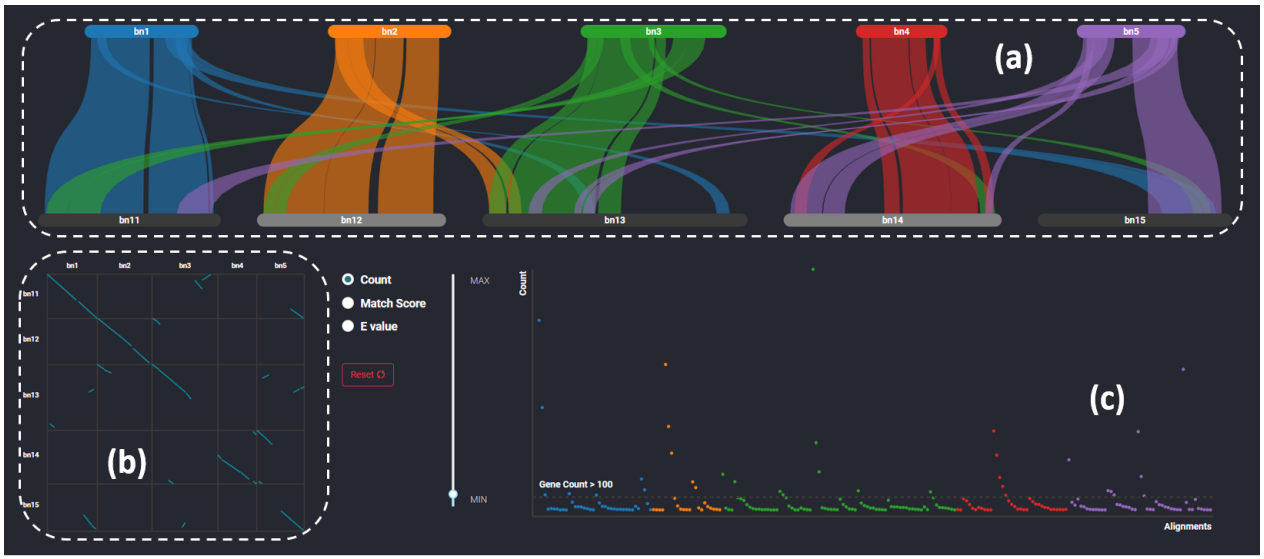

Figure 5.2: Genome View in the primary analysis mode with the following components: **a)** Parallel Link Plot **b)** Dot plot and **c)** Filter panel

source or target selection drop-down and is intended for observing large scale patterns at the genome level. The first visualization presented at the top is a parallel link plot where syntenic collinear blocks are connected by coloured ribbons, as shown in Figure 5.2. The source chromosomes are laid out on the top and the target chromosomes are spread out at the bottom. The size of the chromosomes are calculated based on the genomic sizes of the chromosomes and the available screen width to ensure that the visualizations are accurate across different screen sizes. Chromosomes in the source layer are coloured using a chromatic 10 point colour scale derived from ColorBrewer [9] and are set to repeat after every 10 chromosomes as humans often find it hard to differentiate beyond a dozen colours [131]. The connecting ribbons represent collinear blocks with the colour of a ribbon representing its source chromosome. These ribbons can have varying widths at either end due to the size of the collinear block they represent. Although collinear blocks have the same gene count at either side, the width of the block in terms of base pairs can be quite different at either side due to variable gap sizes between individual genes. This scaled representation of connected ribbons can also mean that certain ribbons can end up being smaller than a single pixel in width due to their small genomic size. Therefore, we clamp our scale at the lower end to two pixels to ensure that extremely small ribbons are still visible.

While the parallel link chart is designed to take half of the available vertical space on a standard 1080p screen, the other half is made up of a dot plot and an adaptive filter panel consisting of a scatter plot. The dot plot, as explained in the visual design chapter, uses positional encoding and represents collinear blocks as either dots or lines in a 2-dimensional matrix. To ensure that small collinear blocks are still represented on the screen, we limit them to single pixel wide dots on the chart while larger conserved blocks are encoded as lines. The dot plot works in a coordinated manner with the parallel link plot, and so any user action such as selecting a single source chromosome to highlight all conserved regions present in it, is also reflected in the other plot, as shown in Figure 5.3. Since the dot plot is always meant to be square, it has a fixed aspect ratio,

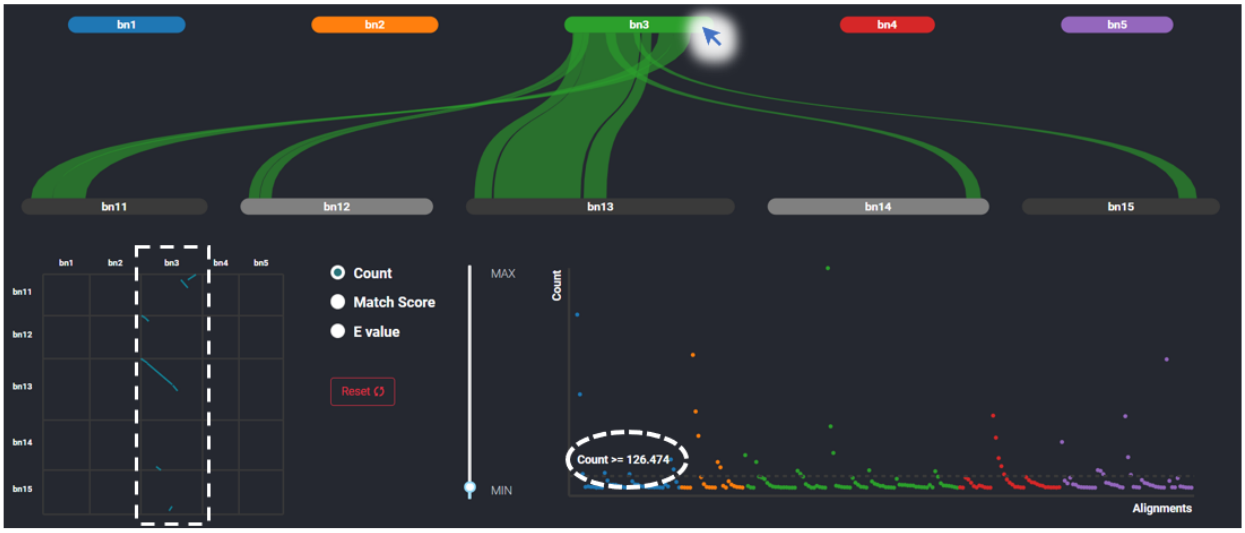

Figure 5.3: Genome View in the primary analysis mode with *Chromosome 3* selected demonstrating the coordinated action being replicated in the Dot plot and the filter panel active with a target gene count set using the slider.

and thus the filter panel expands to fill the remaining horizontal space. It provides filtering through three parameters: Gene count, Match Score, and E (expect) value. It is set to filter using gene count by default but can be changed using the radio buttons provided to the left. To offer users context into the parameter being filtered, its values across all the collinear gene blocks are visualized as a simple scatterplot. Every collinear gene block is represented by a single dot irrespective of its size and is colour coded to correspond to the source chromosomes in the parallel plot. The scale of the scatter plot is adaptive and automatically changes based on the parameter in question. Gene count and Match Score correspond to the number of genes in a collinear block and the alignment score assigned to that block, respectively, and are represented in a linear scale. E-value or expect value is the measure of the probability that a match has occurred by chance, and owing to the wide range in which this value can be reported it is represented in a logarithmic scale. Researchers can use the slider to control the visibility of collinear blocks they see in the other two views. The position of the slider on the chart is represented with a dashed line that is annotated with the value at which the charts are currently being filtered as shown in Figure 5.3.

Chromosome View: This view can be triggered in two ways, either by selecting a single source and a target chromosome using the drop-down selectors, or by clicking on a source and a target chromosome in the genome view. This acts as the logical second stage in exploring a genome where researchers can focus on a particular pair of chromosomes. At this stage, the layout of the visual representations remains the same however the visual encoding of colours to identify chromosomes is replaced with the orientation of the conserved regions with regular conserved regions represented as blue ribbons and inverted conserved regions represented in red. The adaptive filter panel also functions in the same way as in the genome view, but the dots in the scatter plot are not colour coded anymore to represent their chromosomal source. In this stage

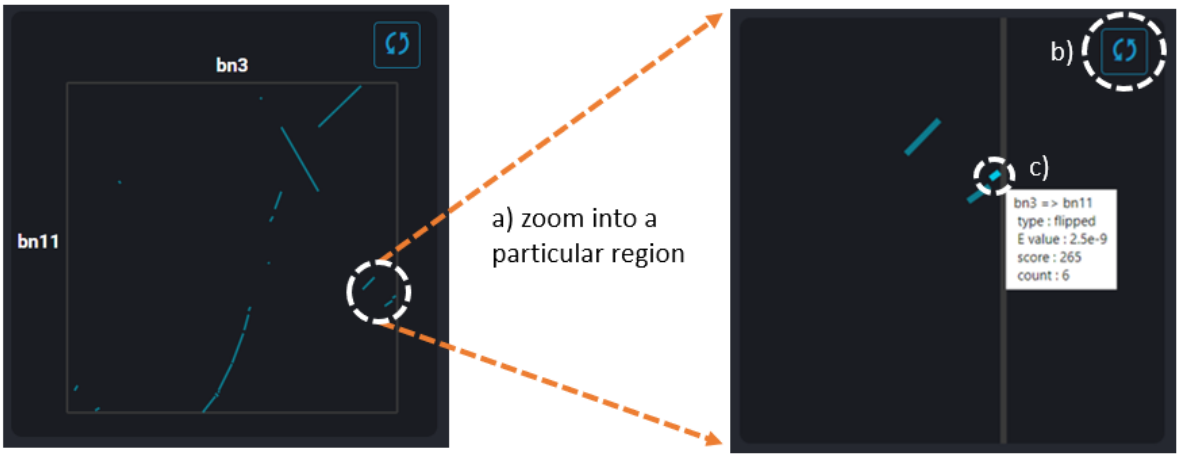

Figure 5.4: Dot plot in the Chromosome View showing the ability to zoom into a particular region of interest (a), reset the zoom to the original state (b) and view additional information about a conserved block (c).

users are allowed to explore small scale patterns in conservation and thus have the ability to zoom into a particular part of the chart using mouse-based interactions, as shown in figure 5.4. This feature is available to both the parallel plot and the dot plot, and the charts are provided with a reset button to readjust the scale of the charts to their original state. Finally, hovering the mouse over a conserved region in either plot will let users see additional information about that gene block in an on-screen tool-tip.

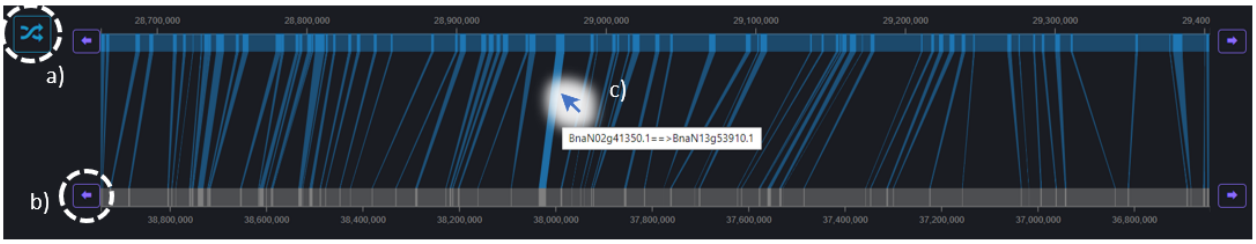

Figure 5.5: Visualization in the Gene-Block View: a) Toggle button to flip the target gene block when exploring reverse matched gene blocks. b) Buttons to move the tracks horizontally along any one direction. c) On-screen tool-tip invoked by a mouse hover showing the source and target *gene IDs* for a particular gene link.

Gene-Block View: This is the final view in the exploratory dashboard and can be accessed from the chromosome view by double-clicking on a particular gene block in either of the two plots that are available at that level. It offers an in-depth view of each of the constituent genes in a collinear gene block and only has a single parallel plot representation. Every connected link represents a single gene and the size of the link is based on the number of base pairs in that gene. This plot can be zoomed in just like the plots in the chromosome view, however, the zoom is applied to the horizontal scale, which means after zooming in at a point, users can then pan the chart either to the left or right look at adjacent genes. Users can zoom in to a large cluster of genes to look at every single gene pair and get their *gene IDs* by hovering over a connected

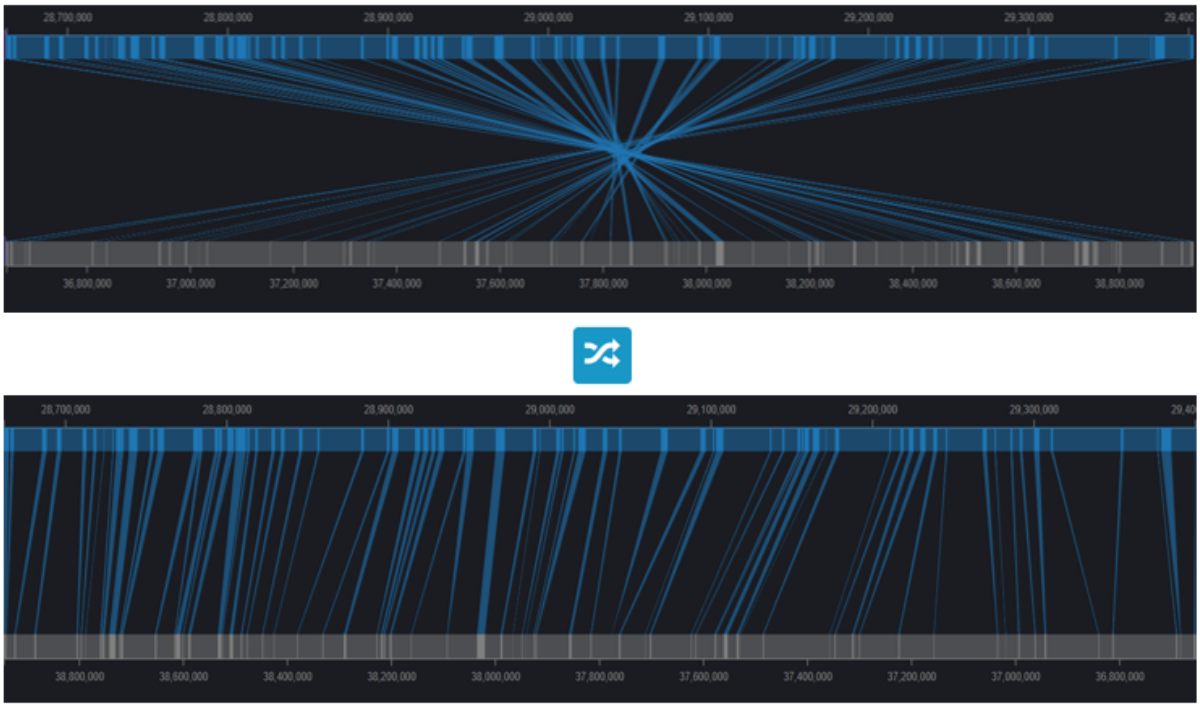

Figure 5.6: Conserved regions that have undergone reversals (top) can be flipped along the target genome using the toggle button to provide an uncluttered representation (bottom).

link. The horizontal zoom however, can cause distortion when linked genes are far apart, causing them to stretch when zoomed in. To address this issue, we provided the tracks with two buttons on either side of both the source and target tracks. These can be used to shift the scale in the required direction to ensure that the gene pairs under investigation are not distorted and instead correctly lined up. Another common issue that can occur in this view is the visual occlusion problem due to a high number of crossing in an inverted gene block. Gene pairs in an inverted gene block are linked across the opposite ends of the gene block and so any zooming into the gene block at this stage would cause the genes to stretch further apart. To address this issue, we provide a toggle button next to the chart whenever a gene block is identified to be a reverse matched conserved region. This button selectively flips the target (bottom) genome scale, causing the gene pairs to line up without any crossings, as shown in figure 5.6.

5.2.2 Multi Genome Analysis Mode

This mode of SynVisio is used to trace and analyze conserved regions between several genomes at once. It can be enabled through the configuration tab of SynVisio and offers two kinds of visualizations: Tree plot and Hive plot.

Tree Plot: This view is an extension of the primary two-axis parallel plot that is used in the primary analysis mode (Figure 5.2), but with additional rows that show pairwise synteny across multiple genomes.

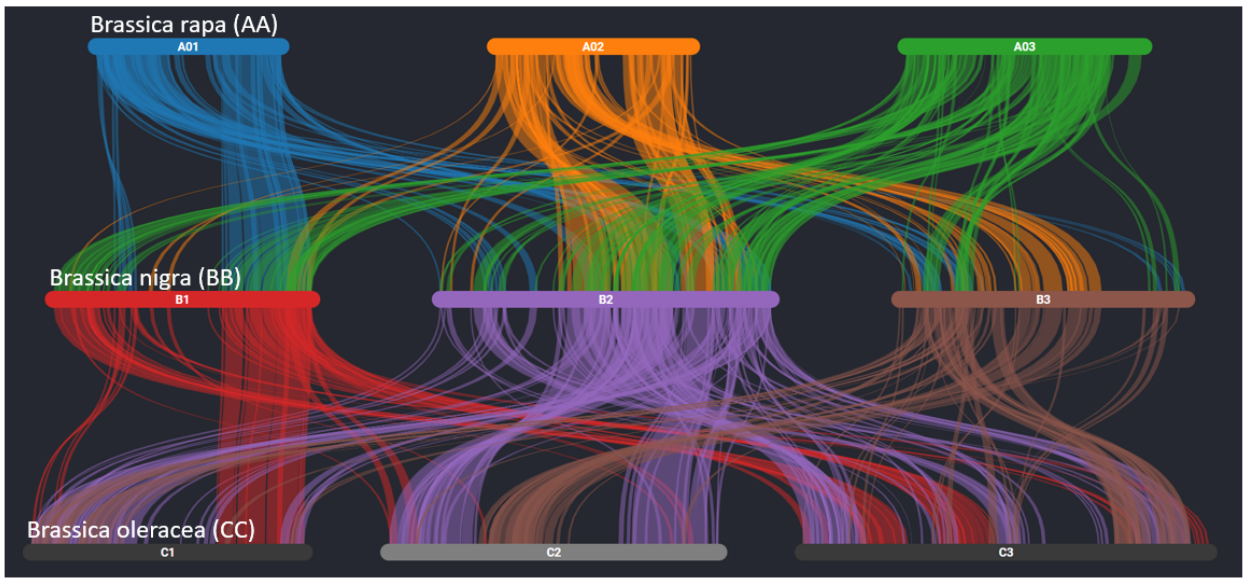

Figure 5.7: Tree plot showing multi genome synteny between the three ancestral genomes from *Brassica* genus.

Users are first given the option to select the number of rows they need, and they then choose the chromosomes they need in each row individually. Chromosomes are stacked in multiple rows and conserved regions in the chromosomes in every row are linked bidirectionally (Figure 5.7). Starting from the top layer, every chromosome acts as a parent node and is linked to the chromosomes in the row directly below it if conservation exists, thus forming a tree-like pattern that can be used to look for conserved regions in ancestral species. Figure 5.7 shows synteny between the first three chromosomes of three ancestral genomes of the *Brassica* genus from the **Triangle of U** evolutionary model [12]. Users are given the option to filter the conserved regions by clicking on a chromosome to visualize just the conserved blocks emerging from it at every layer. A dual filter toggle is also provided to ensure that chromosomal filtering occurs bidirectionally (i.e., conserved links that both emerge from a chromosome into the the layer below and the conserved links that join into it from the layer above it are both filtered).

Hive Plot: These plots have recently been used in large scale network visualizations such as gene regulatory networks due to the high degree of perceptual uniformity that they offer [45]. They have also been demonstrated as good alternatives for Circos [46] style plots in representing three-way genome alignments. Hive plots are based on a linearized network layout where nodes are placed in radially-oriented axes, and edges are drawn between the nodes to encode additional information. In our tool, the nodes represent chromosomes, and they are ordered sequentially based on their order in the genome. The radial angles between the axes are chosen based on the number of selected axes. Conserved regions between the chromosomes are then linked through connecting ribbons which are drawn using Bezier curves with edge bundling to reduce visual clutter [143]. Unlike the pairwise comparison scenarios, hive plots do not have a single source axis as all axes are uniform in a multi-way comparison scenario. Therefore, connected ribbons are not coloured to

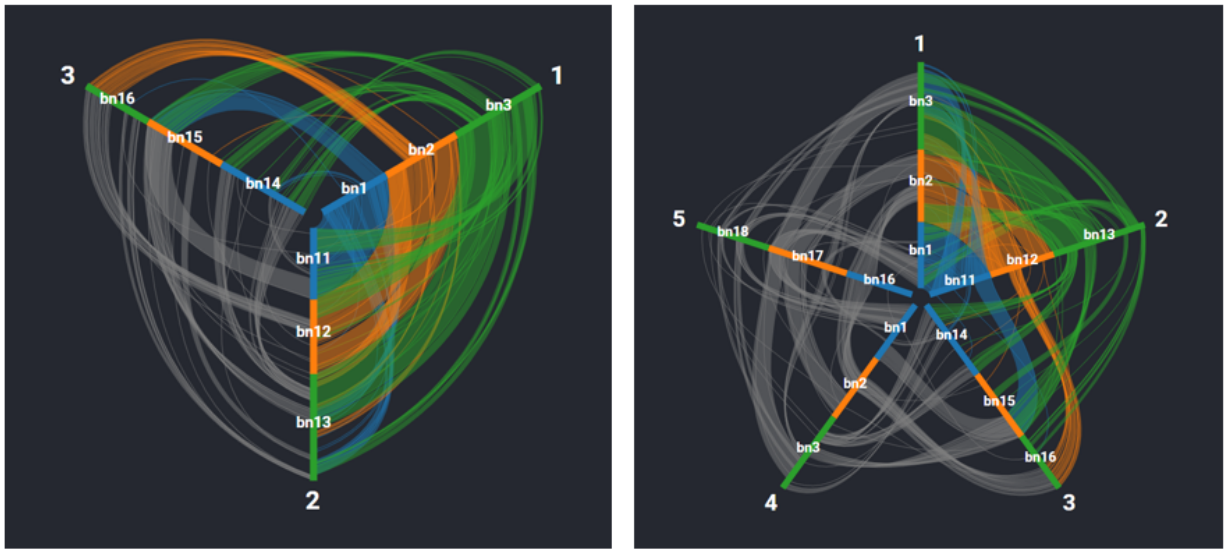

Figure 5.8: Hive plots showing 3 way synteny (left) and 5 way synteny (right) in *Brassica napus* respectively.

represent the chromosomes they emerge from and are instead left to be translucent gray. User interactions with the hive plot are used to select the source axis. When a user clicks on a particular radial axis, all the connected ribbons emerging from it are coloured based on the chromosomes they belong to in that axis. This form of variable encoding based on user choice can be useful in selectively identifying patterns for every genome represented in one radial axis. To use the hive plot, users first select the number of radial axes they need and then select the chromosomes to be encoded in each axis. The generated hive chart can then be annotated with chromosomal labels (hidden by default but can be toggled on/off using a checkbox present in the filter panel). The scales for the radial axes are calculated based on the genomic size of all the chromosomes included in the chart. This ensures that chromosomes that are small in terms of the number of base pairs show up as smaller edges. This can further mean that the sizes of the radial axes of the hive chart can vary depending on the chromosomes that are represented in them. Our interface also provides a *Normalized length* checkbox that can be toggled to make the hive chart axes equal in "visual length" increasing uniformity. This is achieved by using a variable linear scale for every axis while keeping its total length constant.

5.3 Usability Features

As SynVisio developed into a full-scale application, we were provided with several non-functional requirements from our research collaborators. For each of these requirements, we added features that do not change the existing visual encodings used in the system but rather enhance the overall user experience.

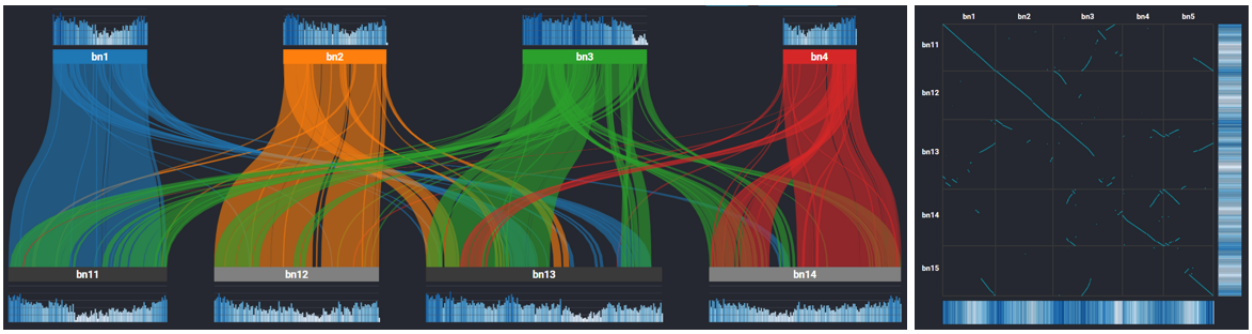

Figure 5.9: Additional tracks showing gene count as a histogram in the Parallel plot (left) and as a heat-map in the Dot plot (right).

5.3.1 Track Annotation

The ability to annotate visualizations with additional data in the form of tracks is a feature that is available in many genome browsers such as JBrowse, GBrowse, and UCSC Genome Browser [21, 43, 111]. However, it has not seen widespread adoption in the existing synteny analysis tools except for Mizbee, AccuSyn, and GSV [59, 68, 93]. Showing annotation tracks can help researchers better understand the data under investigation because tracks such as gene counts and SNP (single nucleotide polymorphism) variations can highlight regions of interest in the entire genome. In our system users can upload a track along with their initial data file using a BedGraph file (a standard format used in representing continuous-valued data as tracks). Our system parses the data in the track file and automatically groups it based on the chromosomal widths into distinct regions for each chromosome so that the data can be loaded on demand when a user selects a particular set of chromosomes for their visualization. We also calculate the maximum and minimum values in the file and use them to generate a linear scale. We limit the number of tracks to prevent overcrowding in the visual representation.

When SynVisio detects additional track data in the system, it automatically provides a new button next to the chromosome selection panel called *Toggle Tracks*, which controls the visibility of tracks in the visualization. We offer four visual representations for tracks: heatmap, histogram, linechart and scatterplot. The heatmap uses a sequential colour scale to encode values in increasing order from white to dark blue. The same colour scale is also used for the histogram and the scatter plot as the default encoding scheme. The linechart alone does not use any colour encoding and is represented in a single base colour. All the graphical positions in the charts are rendered using the previously calculated linear scale which is also used to generate five equidistant grid lines for the tracks. For the parallel plot (Figure 5.9 left), the tracks are added on the outer side of the visualization with one track sitting above the source genome and the other track sitting below the target genome. Since the tracks are accurately mapped to the chromosomes, the pill shaped design of the chromosome similar to karyograms (discussed in Section 2.2.2) is replaced with rectangles to ensure the start and the end of the chromosomes are consistent with the start and the end of the tracks. For the dot plot, the tracks are annotated along the *x,y* axes on the opposite side of the chromosome labels (Figure

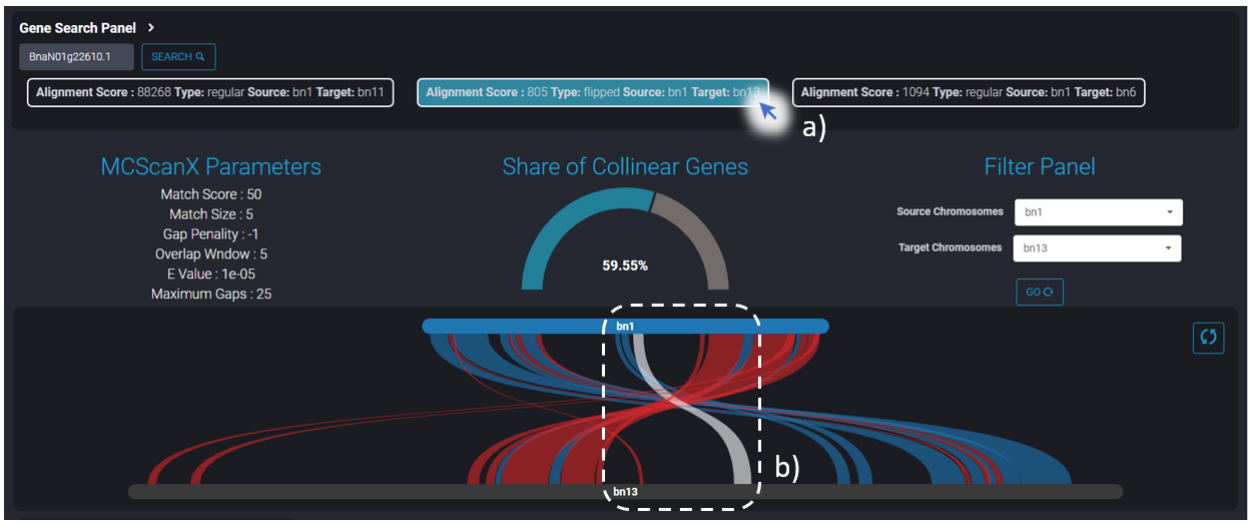

Figure 5.10: Gene Search Panel in SynVisio, with matching alignments present as clickable buttons (a) that when clicked highlight the corresponding alignment (b).

5.9 right).

5.3.2 Gene Search Panel

SynVisio maintains an in-memory collection of all the genes present in the genome or multiple genomes under investigation in the interface. This can be used to quickly look up conserved regions that contain a particular gene of interest. This feature is presented in the *Gene Search Panel* situated at the top of the dashboard (Figure 5.10). Users can type in a gene ID and click on the search button. The system then checks the collection of genes and presents all matching alignments as clickable buttons along with information on the source and target chromosomes of the alignment and its orientation. Clicking on any alignment launches the chromosome mode of SynVisio between the pair of chromosomes that contain the alignment block, and the alignment block itself is highlighted in a pale white colour, as shown in Figure 5.10.

5.3.3 Support to Map Unplaced Scaffolds

Genome assembly is the process in which a genome is pieced together into a large number of contigs from randomly sequenced reads (DNA/RNA segments) [40]. These contigs are then assembled into longer scaffolds which are in turn further assembled into chromosomes. However, due to lack of sufficient mapping information the position and orientation of certain scaffolds remains unknown and also many genome assemblies only assemble data to the scaffold level [23]. This makes it impossible visualize synteny as most synteny tools only offer mapping based on chromosomal order. SynVisio attempts to solve this problem by letting users visualize the synteny between unplaced or unlocalised scaffolds. By default SynVisio ignores scaffolds as they can often be numerous causing cluttering of the visualization, however users can opt out of ignoring them. This will give users a list of scaffolds to select in the source and target chromosome list in the filter

panel at the top of the dashboard. This feature is particularly important to users who would like to visualize collinearity between unplaced scaffolds and known sequenced genomes to improve their genome assembly results.

5.3.4 Image Export

Most synteny browsers such as MultiSyn, Synteny Portal, and GSV [3, 51, 92] have the option to export images; an exception is Mizbee [59] which is designed more for analysis than image generation. SynVisio can work both as an analysis tool and can also generate high-quality publication ready images when required. SynVisio renders visualizations on screen as SVGs (a transform and scale-invariant vector graphics format), and these can be downloaded using the *export toggle* provided as a floating button situated in the bottom corner of the screen. Furthermore, SynVisio exports visualizations based on the current settings of the system as users interact with it in real-time (changes such as adding tracks, filtering out low similarity score blocks and highlighting a particular chromosome are all retained in the exported image). A final advantage of exporting images in SVG format is that these images can be edited and the colours of the individual elements changed using vector graphic editors like Inkscape and Adobe Illustrator.

5.3.5 Revisitation Support

Multi-scale visualization systems are effective at exploring large datasets because they help researchers follow the visual information seeking mantra of *overview first, zoom and filter*, followed by *details on demand* [109]. They achieve this by changing the visual representation of the data at different levels of abstraction as the user zooms in for closer inspection [120]. This process however, can cause the user to lose an overall context of their position in the visual space as they constantly need to switch between different kinds of visual representations. Although humans are good at leveraging spatial cognition to remember locations of objects in information workspace tasks [97], context switching can disrupt this ability. To assist the user in navigating to previous states of the system, SynVisio lets users keep track of their actions through a visual snapshot feature. It preserves the sequence of actions that led to the current state of the visual interface in a snapshot. Users can store this snapshot by clicking on the floating camera button at the bottom left corner of the interface. Snapshots are stored sequentially and are available for revisitation at the *Snapshot Store Panel*. Clicking on any snapshot in the panel automatically recreates the stored visual state of the system. The snapshots, however, are stored only as an in-memory collection in the system and so do not persist over page reloads or when users switch the input data files.

5.4 System Architecture

Recent trends in software development have shown an increase in the development of internet-based web applications that are built using HTML5, JavaScript and CSS3. This is due to the availability of web

applications and their device-agnostic design that ensures that they run independently of operating system or device type. Although some synteny browsers have been developed as native applications (Mizbee and SyMAP [59,114]), most recent synteny browsers like MultiSyn, mGSV and Synteny Portal [3,51,92] all adopt a web-based approach. Following this trend, we developed SynVisio as a web-based application that can be accessed for free through the internet at `https://synvisio.usask.ca`.

A choice must be made between two main design patterns for web applications: single-page or multi-page application. The choice is based on the content being served. Single-page applications request content markup and data independently and render pages dynamically in the browser through JavaScript. This makes them fast and responsive despite the initial delay in loading all content in a single bundle during page load. There are no subsequent page reloads and all further requests are purely for data which has a significantly smaller payload. Multi-page applications, on the other hand, follow a more traditional approach with changes and interactions being sent to the server, which responds back with a new page to be rendered by the browser. This additional communication between the browser and the server can make these systems cumbersome to use due to added delay. However multi-page applications are more secure compared to single-page applications which can be susceptible to cross-site scripting (XSS) attacks. Although users aren't concerned with software architecture, as they focus on tasks and not the structure of a system, it can be a determinant for the usability of any system. Proper information architecture should offer users logical structures that can aid them in navigating towards the right answers and completing their tasks [99]. Thus for SynVisio, we adopt the single-page architecture design and render visualizations in the browser instead of having them shipped from a remote server, as this allows for users to interact with them in real-time and offers a smooth interactive experience [66].

This architecture model where data processing is managed locally in the client machine is called a thick client model. It ensures that all the data files that researchers upload to our website remain secure within the same browser instance and are not sent to any other remote server. However, processing data files in the web browser comes with its own set of complications as JavaScript has a single threaded environment and any intensive data processing can block the main thread limiting user interactivity in the page. To address this issue we use the *web workers* API to spawn background scripts that can handle computationally intensive tasks without blocking the main user interface [137]. Every user-uploaded file is processed in an independent thread through a web worker and the processed results are then combined to form an in-memory dictionary of all the genomic links, classified based on the chromosomes present in the genome. This dictionary is then used whenever a user selects a pair of chromosomes to query the required set of genomic links for visualization.

The web interface of SynVisio is built using two JavaScript libraries: React.JS [90] to handle user interactions and render content, and D3.JS [8] to generate visualizations. React is a popular JavaScript library maintained by Facebook [25] that is used in building user interfaces in single-page applications. It can efficiently manipulate the representation of a webpage which is called the document object model (DOM). It does this through a process of reconciliation with an in-memory representation of the actual DOM called

the virtual DOM. This is particularly useful in our scenario as all visualizations are rendered through vector graphics and so are part of the actual DOM structure. Thus as the size of the data being visualized increases there is a corresponding increase in the number of visual elements in the DOM. However by using React we can handle large DOM networks efficiently by deferring updates only when necessary. The second major advantage to using React is that it follows a component-based model where every part of the interface is built using a set of reusable components that render differently based on the data being passed to them. This enables us to rapidly switch the generated visualizations on the screen by simply modifying the underlying data provided to the component. This is particularly useful in implementing interactive real-time filtering as any filtering done to the underlying dataset is reflected onto the actual visualization. This feature is also used in providing support for the revisitation feature discussed in Section 5.3.4. Every saved snapshot is essentially a data object that is stored and passed to the graphical component to recreate the required visualization.

All visualizations in SynVisio are rendered as scalable vector graphics (SVG) with the coordinate values of the underlying graphical elements being calculated using D3.JS. These include the mathematical calculation involved in the interpolation of points that connect conserved regions and scale transformations from genomic distances to pixels. Finally, SynVisio can switch to rendering using the HTML Canvas (an immediate-mode pixel-based drawing surface) instead of vector graphics when there are more than 20,000 graphical elements in a visualization, as SVG performs poorly at these scales. This also limits the user interactions that are possible with the visualization (for example, detecting mouse interactions can be slow), and a warning message is shown to the user to select a smaller subset of the dataset or an alternate representation with fewer visual elements. This adaptive mode of SynVisio was built to ensure that even extremely large datasets can be viewed in the system without a reduction in performance.

6 Evaluation

SynVisio was made publicly available to use for free on the Internet on September 2018. A stable version was deployed in the start of 2019 and since then, it has been used by several researchers across the world in exploring genomic conservation in a wide variety of organisms. Evaluation of the system was done through a combination of user study based on semi-structured interviews, and analysis of web traffic to the system. Domain experts from the field of biology were consulted for the user study and their feedback of the system is summarized through three case studies presented in the sections below. To provide evidence on the effectiveness of our system in the wild we also explored user activity logs on the website for a period of 12 months through Google analytics. Finally to demonstrate the open-ended design of SynVisio we provide examples of genome databases for silkworm and two other species that were extended to show synteny visualizations using the open-sourced code of SynVisio.

6.1 Methods

The user study to evaluate our system was conducted in the form of semi-structured interviews with five domain experts from three major research groups we were collaborating with; one of the experts was a bioinformatician who worked across all three research groups. The interviews were conducted either through phone or in person and lasted around 45-60 minutes. All domain expesearchers were asked to rate SynVisio on a scale of 1 (very bad) to 5 (very good) for its ability to represent genomic conservation.

6.2 Case Studies

6.2.1 Wheat *(Triticum aestivum)*

Wheat is one of the most widely cultivated crops in the world and plays an important role in human nutrition. Being a common cereal, wheat genomes are highly diverse and spread over a large geographic range. The genome is capable of tolerating mutations and extensive hybridization which is why it has been able to adapt to such a wide variety of environmental conditions [16, 88]. Wheat is also one of the *neolithic founder crops* that were the first to be domesticated almost 10000 years ago. Bread wheat (AABBDD) is a hexaploid genome and is the result of series of hybridization events between three ancestral genomes A Donor *(Triticum urartu)*, B Donor *(Aegilops speltoides)* and D Donor *(Aegilops tauschii)* which makes it an interestingerts had considerable exposure to SynVisio and were familiar with the different features provided by the system. This

was further verified before the interviews were conducted. Researchers were first asked about the relevance of synteny visualizations in their field of research and then asked to give their opinion on the different modes of analysis that SynVisio offered through 3 open ended questions with one question targeting each genomic resolution (genome, chromosome and gene level). They were then asked to give feedback for the different interactive features provided in the system. Finally all r subject for synteny analysis. Our collaborators were part of a research team involved in sequencing a high quality version of the wheat genome. Since the wheat genome is extremely large and complex, synteny analysis can help researchers in assessing the quality and contiguity of the genome assembly though alignments between the sub genomes (A, B, and D).

Our collaborators relied on visualizations generated by SynVisio to present and summarize their sequence assembly results - *"the images have been used in presentations, academic meetings such as the international wheat congress and also at the plant-animal genome conference." (R1).* While they used Circos style plots for research publications earlier, they mentioned that the multi level representations in SynVisio were far more useful for genomes like wheat with many chromosomes - *"This tool is better than a Circos plot, especially when comparing multiple genomes, circos can be limited because you are seeing too many chromosomes in one circle and so are losing information ... a stacked layout like yours is easier to see." (R4).* One collaborator in particular also appreciated the system for its ability to handle large datasets like wheat - *"this is really neat. this is also very useful...a single wheat chromosome is vast and wheat has 21 of those (chromosomes) placing stress on an analysis pipeline in terms of computational complexity...it is also very repetitive..." (R1).* Feedback provided by this research team was also used in adding support for hive plots which offer an intuitive representation to compare multiple sub genomes in crops like wheat (Figure 6.1). The Appendix provides supporting material describing the process needed to generate this hive plot using SynVisio for this particular dataset. Our collaborators in this team plan on publishing images generated using SynVisio and have already used it to create a portal for researchers to compare 12 different wheat cultivars for the 10+ Wheat Genomes Project. Users can use this portal to select any two varieties from 12 different wheat cultivars and then compare genomic conservation between them using SynVisio [4, 88].

6.2.2 Lentils *(Lens culinaris)*

Lentil is an important legume crop that is grown globally as a valuable source for dietary protein. It also plays a crucial role in food security in developing countries along with other legume crops like Chickpea *(Cicer arietinum)* [126]. Lentils can be made more resistant to diseases and weed infestations by increasing the genetic diversity of the genome through hybridization between disease resistant wild varieties. This however requires mapping the traits through molecular markers to assess their diversity. Our collaborators relied on comparative genomic mapping to leverage information from a model legume species like Barrel Medic *(Medicago truncatula)* onto less studied crop species like lentils and chickpea which lack common markers.

Unlike the wheat genome, synteny analysis requirements for this project were centered around cross synteny between species rather than self synteny. Due to the large size difference in the genomes between

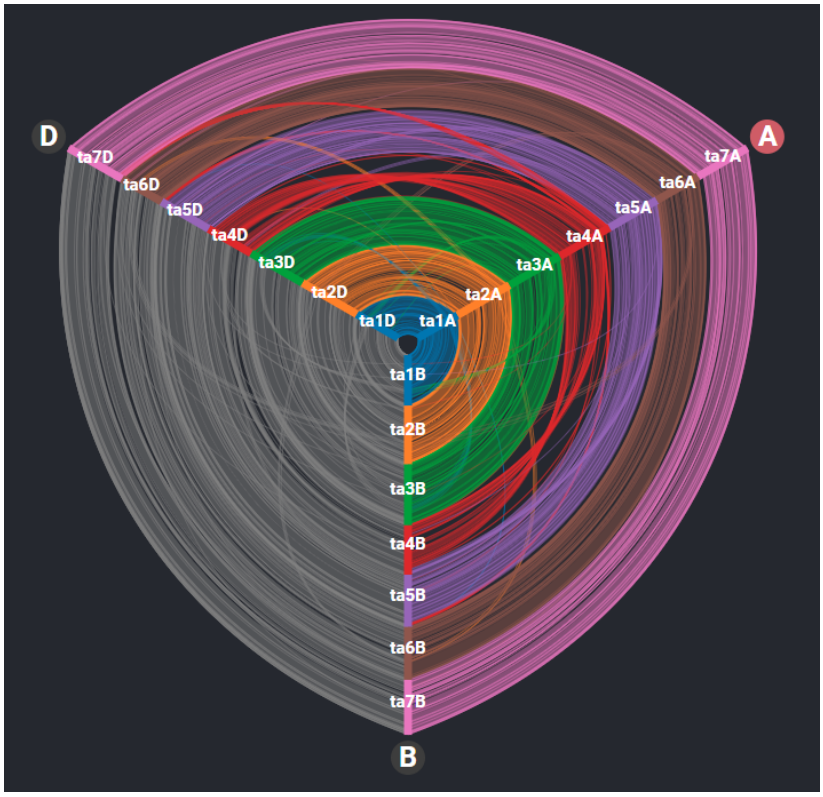

Figure 6.1: Genomic conservation between the three sub genomes A, B and D of Wheat (Chinese Spring Variety) shown through a Hive plot in SynVisio.

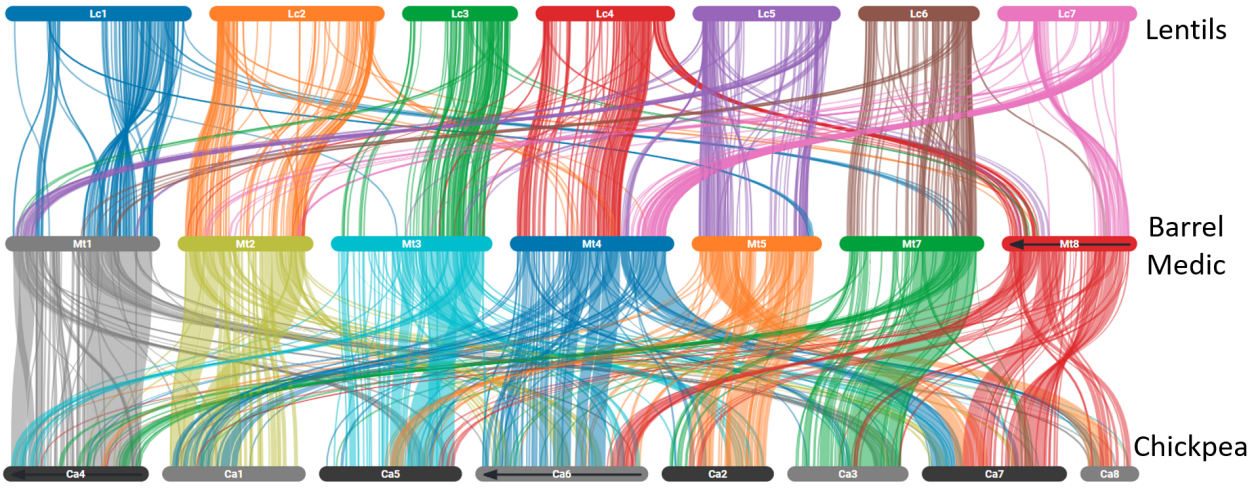

Figure 6.2: Collinearity between Lentils (Lc), Barrel Medic (Mt), and Chickpea (Ca) presented through a Tree view plot. The ordering (Ca) and orientation (Mt8, Ca4, and Ca6 - flipped) of some chromosomes have been changed to reduce visual clutter.

Lentils (4Gbp) and Chickpea (740Mbp) the first version of SynVisio was not able to generate legible charts as the Lentil chromosomes were extremely wide compared to chromosomes from the other species and so a special feature was added to have variable scales at different levels. Our collaborators were pleased with the updated view and also remarked on the multiple visual representations provided in the genome view - *"I think it's quite good, I do really like that there's also the dot plot, in the corner, so that if anything is a little bit unclear, from the parallel view, you can kind of refer back to that." (R5).* Because this was a cross synteny analysis between several genomes, researchers mentioned that the Tree view was particularly helpful in summarising large scale chromosomal rearrangements and inversions while still keeping the different genomes visually distinct as shown in figure 6.2. They also compared it to circos plots and remarked on its usability - *"It's like the circos plots are beautiful but you can't do anything with it. Whereas this, the tree-view in particular, is very aesthetically pleasing and that's the kind of thing that you can show to your collaborators and you can also understand it, at the same time, and then the interactive nature of it helps too..." (R5).* Researchers from this group have also used SynVisio to study genomic conservation in other legumes like the Tepary Bean *(Phaseolus acutifolius)* and are planing on using it to generate images for their research publications in future.

6.2.3 Canola *(Brassica napus)*

Canola is an important oil seed crop in the world as it is an excellent source for both animal feed and high quality edible oil [108]. It is an allotetraploid (4 copies - AACC) species that was formed through interspecific hybridization between diploid ancestors Brassica rapa (A Donor) and Brassica oleracea (C Donor) [76]. Studying this genomic conservation can help researchers in looking at genetic variations that are advantageous from an evolutionary perspective in polyploids like Canola. Our collaborators from this research group were particularly interested in using comparative mapping to understand the level of genome duplication in modern brassica cultivars and the occurrence of genomic rearrangement in the evolution of these varieties from a common ancestor. This meant that they needed to visualize both self synteny between Canola itself and also cross synteny between canola and its closely related species - *"...in polyploid plants where there are many genomic rearrangements, visualization is really useful because there is lots of information and its really complicated for us to understand without an overview... " (R2).*

SynVisio also helped researchers in this team at refining their assemblies - *"Our assembly got better when we upgraded our sequencing from short read to long read sequencing technology as more regions are assembled. This tools helps us visualize that improvement ..." (R2).* Regarding the visual representations one collaborator remarked that the parallel plot representation was better at showing genomic conservation than dot plots - *"We have always used dot plots but these (parallel plots) are visually more intuitive...when chromosomes start breaking apart its much more difficult to follow where things are going in that big square and in this its easier to play around...Its much easier to trace things and work out where you are..." (R3).* Researchers from this team have used visualizations generated by SynVisio at several conferences such as

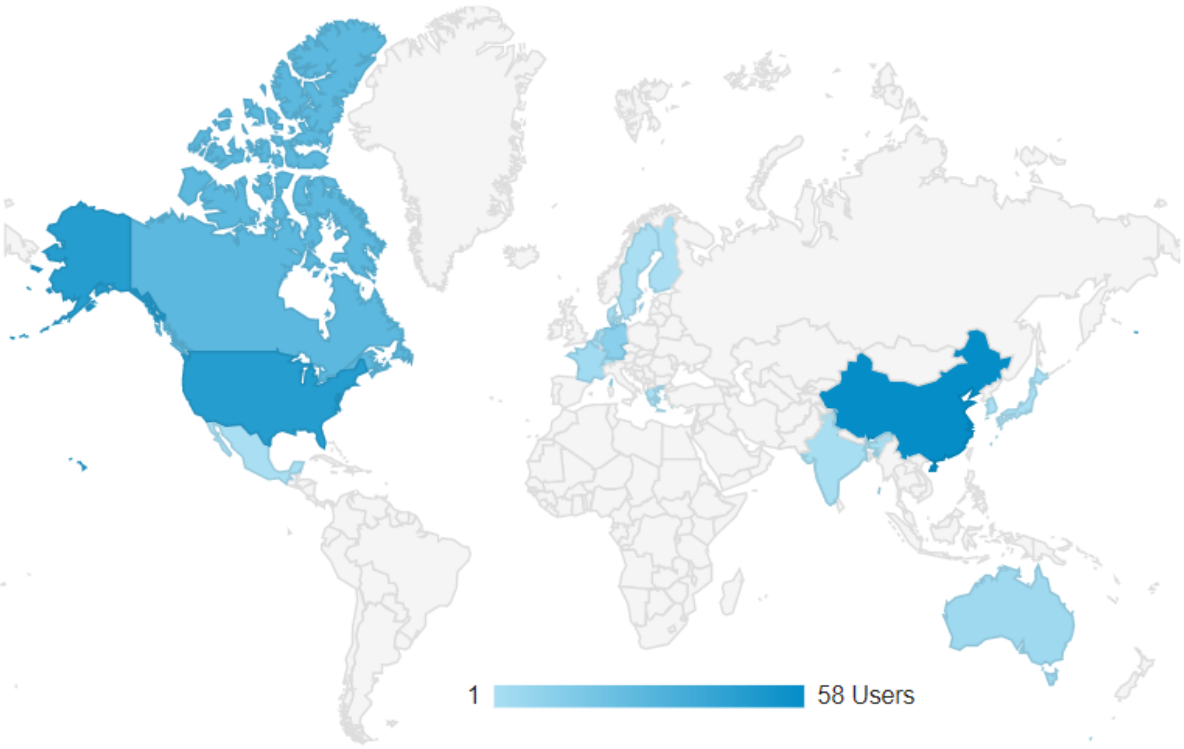

Figure 6.3: Global user distribution of SynVisio for a period of 12 months from 2019-2020.

PAG (Plant Animal Genome) 2019 [77] and also in a recent publication describing long read assemblies of two diploid Brassica species [82].

6.3 Global Usage Analysis

Although our system was initially designed based on requirements from our collaborators for use within our university it was made publicly available as an open access tool on the internet and has been used since then by researchers across the world in a wide variety of research projects and images generated by SynVisio have also been used in research publications describing new genome assemblies and annotations [58, 82].

To quantify the use of SynVisio since it was made public we analysed web traffic through Google analytics from Jan 1st 2019 to Jan 1st 2020. In this period of 12 months SynVisio had 154 unique users spanning 267 sessions with an average session duration for each session being around 2 minutes while some users spent as much as 28 minutes on the system exploring different datasets. Users were from 18 different countries across the world as shown in Figure 6.3 with a majority of the users being from China (53) followed by the United States (45) and Canada (23).

SynVisio was designed in a modular fashion as a reusable component and its source code was open-sourced through an MIT license on GitHub [5], which meant that it could be adopted and reused in other research tools and projects (without our involvement). An example of a system that has used SynVisio in this manner

is **TeaBase**, an online genome database with various tools, one of which is SynVisio, to explore the Tea plant (*Camellia sinensis*) genome as shown in Figure 6.4 [134]. Other genome databases that have used SynVisio in a similar manner are **VitisGDB** and **SilkDB 3.0** to explore the genomes of Grapevine and Silkworm respectively [54, 135].

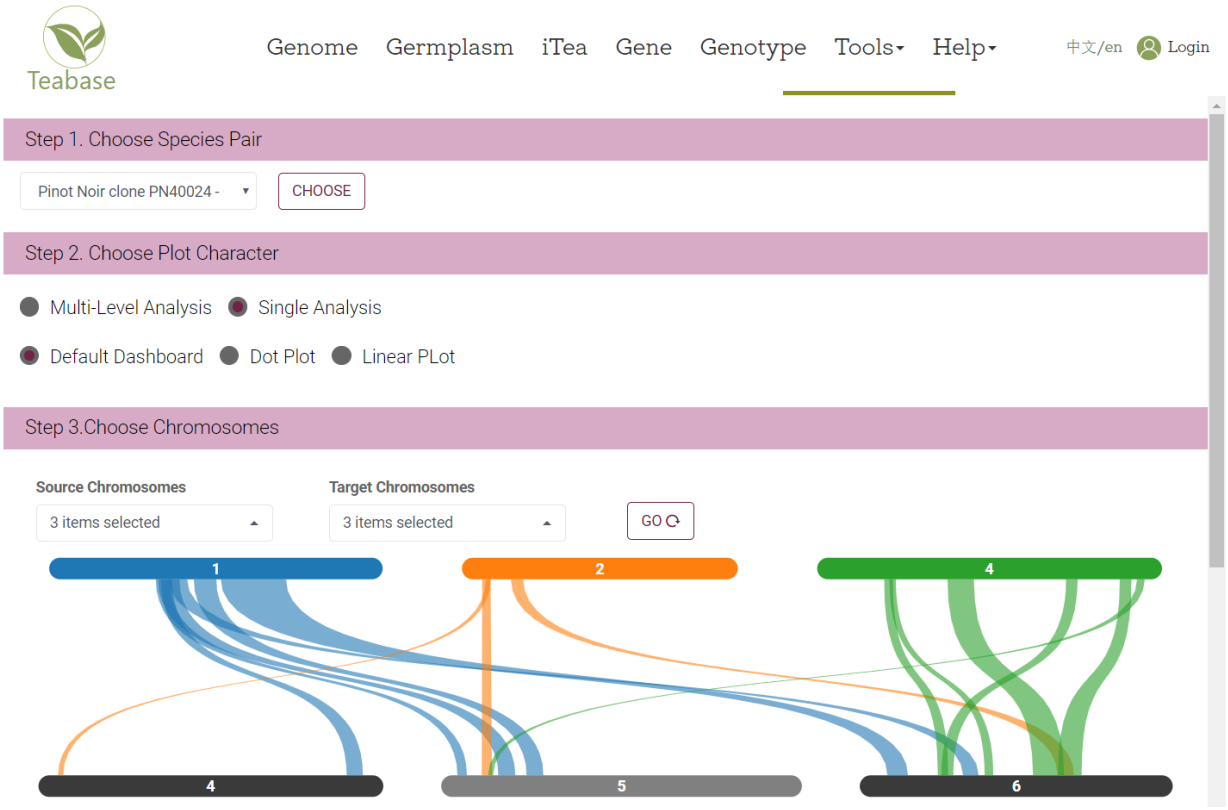

Figure 6.4: TeaBase, an online genome database for the *Tea plant* genome adapted to also include synteny exploration through the open sourced code of SynVisio.

6.4 Evaluation Summary

Although several visualization tools exist for analysing genomic conservation they are not easily accessible as mentioned by researchers we interviewed - *"There are a couple of R based tools that we use but none of them are as complete as synvisio " (R4).* SynVisio has been able to fill this critical gap - *"There isn't anything like this. Especially not where you can play around with your dataset." (R3).* When asked to rate SynVisio for its ability to visualize genomic conservation across different levels on a scale of 1 (very bad) to 5 (very good), four researchers gave the system the highest rating of 5/5 and one researcher gave it a rating of 4/5 stating that they would have liked greater control over the ordering and orientation of the chromosomes. This feedback validates the usability of the system for the initial set of visual tasks we designed the system for. Further we were also able to meet the supplementary design requirements that we had envisioned for data refinement and enhancement. Although some researchers mentioned that they did not find the filter

panel quite helpful *"I do most of my filtering ahead of time before running the tool so this filter is personally not useful for me but I can see why you have it..." (R4)* others found use for it - *"The images are quite messy and it (filter panel) is definitely helpful in cleaning it up a bit..." (R3)*, *"When looking for distant relatives, the feature with E value filter is useful." (R2).*

Making SynVisio an online tool with the ability to directly upload sequences to visualize them has improved the usability of the tool to a great extent as several researchers mentioned that this has allowed them to share their work with other collaborators easily - *" if I wanted to discuss some of the results of this with a collaborator, I just zip up the two files that you need for input. Send it off to them and they can put it in and play with it themselves" (R5).* This is also further evident by the web traffic SynVisio has received since its has been made available on the internet. Further the system design has also been adopted in several online genome databases, showing that it is a valuable tool in exploring genomes and interacting with them.

7 Discussion

In this chapter, we discuss the the insights gained from building each of the unique features of our system and the design choices that went into their development. We also look at how some of these features can be extended to support additional genomic analysis tasks. We then explore some of the limitations of our current system based on the feedback gathered from our user evaluation study and possible improvements that can be made in the future.

7.1 Design Implications

- **Input Files and Formats** - Genomic conservation can be detected through a wide range of tools, which means that it can be represented in a wide variety of file formats depending on the type and the level of information about conservation of gene order. Although some tools like Mizbee have relied on users to supply input in a standardized format, this is not a viable solution as this often means users will have to rely on a custom script to transform their analysis files into the required format. Visualization tools like SynVisio and Mizbee are part of a larger ecosystem of genomic analysis tools; therefore, they need to offer at least partial connectivity between such tools which means outputs of most analysis systems should be directly supported in visualizations tools without the need for intermediate processing. In an effort to address this developers should consider building systems that offer support for heterogeneous data. For example, SynVisio currently supports inputs from several popular tools such as collinearity files generated by MCScanX or Orthologous files generated by Dagchainer with also partial support for MUMmer output files.

- **Web Accessibility** - Most existing genomic visualization tools are desktop applications or packages in languages such as R, Python, or Perl; however, there has been a gradual shift towards the web in the recent years. Although desktop applications are efficient at utilizing system resources, they are limited in their accessibility as they are not often supported in all operating systems and require extensive customization from developers as they are platform dependent. Web applications, on the other hand, are platform independent and can be built once and used everywhere thus requiring less development effort. Even though web apps are limited in their processing capability, they can rely on remote servers for intensive processing, and some applications like SynVisio also rely on web workers to process data in parallel threads for more efficient data processing. This easy accessibility and low-cost maintenance of web apps coupled with support for collaborative work mean that web applications should be the first

choice for developers of visualization tools in the future.

- **Multi Layout and Multi Scale Views** - Genomic data can be analyzed at multiple resolutions and the visual representations vary at each level. At the genome level, visual representations are chosen to emphasize approximate positions of the conserved regions and their chromosomal identifiers. This can be useful in scenarios such as during genome assemblies when errors or breaks in chromosomes can be easily identified. However, when looking at the chromosome level, orientation of the conserved region is also highlighted and finally at the individual gene level emphasis is placed exclusively on the order of collinear genes and their exact function and location in the genome. Visual representations can also vary based on the task at hand. For example, dot plots are a popular choice for summarizing large scale datasets as they offer a compact representation of both the position and orientation of conserved regions. However, their orthogonal representation is difficult to understand, making them less effective for browsing and locating selective conserved regions in comparison to parallel plots. Other such examples are stacked parallel plots which are good at tracing collinear regions across several genomes. Thus in designing genomic visualization systems, developers should rely on a combination of visual representations or provide users the choice to switch between different representations based on the task at hand.

- **Visual Navigation and Linked Views** - When exploring genomic data, visualization systems can provide users several ways to traverse the different representations at each genomic scale. However, the most intuitive way to explore such a dataset would be to start at the genome level and drill down all the way into the individual gene level. Therefore, visualizations at each level should be provided with interaction techniques to filter and zoom into a particular part of the dataset which can then be viewed in a different visualization at the next inner level. This form of tiered navigation combined with the support for revisitation at any point can help researchers in easily going back and forth between the levels and exploring a large number of scenarios without losing context. Also, a major part of analyzing genomic conservation involves comparison, and providing linked multi-views that are different representations of the same information can help users in contextualizing the conservation and better understanding it.

- **Linear vs Non Linear Representations** - Genomic data is usually linear and so an obvious way to represent such data would be a linear representation such as a dot plot or a parallel plot; however, circos style plots which are a form of non-linear representation are still quite popular in genomic visualizations as they are aesthetically pleasing and offer a compact picture. However, based on observations from our user evaluation, several users found circos plots challenging to navigate for an in-depth analysis. This is primarily due to the non linear curves in these plots that can make it difficult to identify connections between distinct groups which in this case are chromosomes ordered in a circular layout. While linear representations like parallel plots can be stacked on top of each other to represent conservation between

multiple genomes, circular layouts can only handle a limited number of chromosomes in the central layer before they become difficult to comprehend due to close proximity between the chromosomes. Part of the compact nature of the circos style plot arises from its ability to stack multiple layers in concentric rings to represent several tracks, however, the radial nature of this design means that tracks in the outer layers are always larger than the tracks in the inner layer. This can lead to visual bias where patterns in the outer layers are more prominent than patterns in the inner layers.

- **Adaptable Genome Scales** - Genomic data can be incredibly diverse in size and so systems visualizing such information need to automatically adapt to different scales of data instead of relying on a standard scale. Some plant genomes like wheat are extremely large (17 gigabases) and quite dispersed, meaning that genes are quite small and so when visualizing this information, SynVisio provides users with two additional levels to magnify the dataset. Similarly, when visualizing extremely small genomes such as viral genomes (30 kilobases) the system automatically loads up the data in the smallest possible resolution directly at the gene level as shown in Figure 7.1 which compares similarity between two coronavirus strains that led to global outbreaks. This disparity in genome sizes can also be an issue when comparing multiple genomes with a large difference in their sizes. In such scenarios, the visualization system should provide users an option to have different scales for each of the genomes instead of relying on a single normalized scale among the genomes.

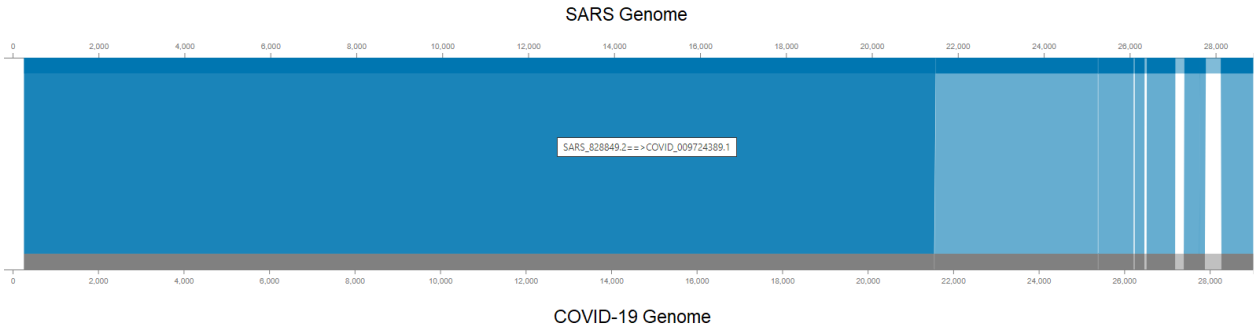

Figure 7.1: Collinearity between the genomes of the SARS Virus (2003 outbreak) and the COVID-19 Virus (2019-2020 pandemic). The first of the two replicase genes (ORFs 1a and 1b) that are translated into polyproteins, is highlighted in a darker shade.

7.2 Limitations and Future Work

Although SynVisio was designed to handle a wide variety of scenarios, there are still certain issues with our system that can be improved through additional changes in the future.

- The first and major limitation of our system lies in the dependence on an external tool (MCScanX, DAGChainer or Mummer) to detect conserved regions. This was mentioned as a bottleneck by several

of our users in analyzing their datasets. But the complexity involved in detecting similarity between two given sequences and running a collinearity detection software cannot be achieved through the existing web system as it is computation intensive. A possible way in which this can be solved in the future is by setting up a dedicated remote server that can accept sequences uploaded by users to perform synteny detection on the cloud and then send the results back to the web system to be visualized. This would also give users greater flexibility in changing the different parameters involved in detecting conserved regions such as the E-value to look at more distant matches.

- SynVisio relies on a custom algorithm in determining the visual scale of the genome. Genome scales are calculated based on the available screen width and the size of the genome in base pairs and every chromosome is normalized accordingly. But users are given the option to override the normalization and have independent scales for each genome when comparing multiple genomes. In a situation where two genomes are stacked parallel to each other such overriding will cause both the genomes to stretch to the available width. However in certain situations this can cause users to lose context of the exact size of the genomes. To address this confusion in future, we can provide information in the form of tracks or scales shown parallel to the chromosomes indicating the size of the genome in kilobases or megabases.

- SynVisio automatically sorts chromosomes alphanumerically in each genome to determine their layout. These chromosomes are then presented from left to right and oriented in the same direction. However, in some cases, this layout can cause the ribbons connecting conserved regions to excessively cross each other, making it difficult to understand the relation between the two genomes. In such scenarios, it would be helpful if users are provided an option to declutter the layout by reordering the chromosomes or reversing the orientation of each chromosome. This can be achieved in the future by developing a dedicated layout editor that lets users manually drag chromosomes around and reverse them if needed to create a more organized layout.

- Visualizing syteny in stacked parallel plots is an excellent way to trace conserved regions across multiple genomes. However, researchers are often also interested in understanding the evolutionary relationship between a given set of genomes along with an overview of the conserved regions between them. Evolutionary relationships among genomes are commonly represented as phylogenetic trees evolving from shared ancestors. Combining such a representation along with the existing parallel plots between genomes would be a significant improvement to SynVisio and offer researchers an easy way to analyze novel datasets such as the pan-genomes of different species.

- Although SynVisio lets users explore genomic data from the genome level all the way down to the individual genome level, it cannot show the actual nucleotide or the protein sequence alignment within every gene. This limitation arises due to the large size of FASTA files which cannot be quickly loaded into the Web application. However, presenting this information can help researchers in understanding

the extent of similarity in the gene alignment and gain additional information about the function of the gene and the protein it codes for. For most annotated genes, several online databases exist that curate this information and present it in a easily accessible manner such as genomeDB and NCBI. In future, we would like to link every gene in the gene level view directly to their entries in preexisting databases, such that clicking on any gene would automatically open up the FASTA entry along with information about that gene in a new tab of the browser.

8 Conclusion

Comparative genomic research plays a vital role in studying genome evolution and ancestral genome reconstruction. However, despite the availability of high-resolution genomic data, research in this field is being restricted due to the lack of proper analysis tools. While some analysis tasks can be automated to deal with the high volume of data, other tasks still require manual interpretation such as synteny analysis. Visualizing data in such scenarios can help researchers in their analysis by offloading part of the cognitive load required in processing information onto humans' inherent capacity for visual perception. Visualizing synteny blocks can aid researchers in understanding the location, size, and orientation of conserved genomic regions. Although some tools do exist for synteny analysis, they are limited in their usability and offer very little interactivity needed to explore complex datasets. Our primary contribution in this research work is SynVisio, a synteny analysis tool that offers genomic researchers different ways to visualize and explore genomic data. Researchers can access the tool through a public web-based interface and directly upload their synteny analysis files. Information can be analyzed in a primary analysis mode through pairwise comparative visualizations such as linear parallel plots and dot plots. Alternatively, researchers also have access to a multi genome analysis mode where syntenic blocks can be visualized through hive plots or stacked parallel plots to trace genomic conservation across several genomes at once.

Our second contribution was in adding interactive support to our system to help researchers in refining and enhancing their datasets. All visualizations are accompanied by a filter panel to modify the generated visualizations in real time. Syntenic blocks can be filtered based on the level of similarity (score or number of genes in a block) or the probability of the match (E value) depending on the underlying genomic question. Researchers can also augment certain visualizations such as parallel plots and dot plots with tracks representing additional information such as gene density or SNP count. These tracks are in the form of heat-maps, line charts, scatter plots or histograms. The tracks along with all visualizations can also be exported in publication-ready formats.

Our third contribution is providing support for revisitation. Synteny analysis is an exploratory task that requires researchers to investigate conservation at multiple genomic resolutions. Such exploratory analysis requires users to switch between multiple visualizations under different filter parameters. This switching can, however, cause them to lose context of their position in the dataset. SynVisio avoids this by providing users the option to snapshot the state of the system at any point in their exploratory analysis for easy and quick revisitation. This along with other features such as searching for genes in syntenic blocks, can be useful to researchers, particularly in exploring large datasets.

SynVisio has been developed as a modular component that can be reused in existing online genomic analysis tools, and the source code for the system has been open-sourced to facilitate the rapid dissemination of our work into other scenarios. Several researchers are currently using our system across the world either directly via the web interface or through the integration of our system into their existing tools. We also plan on adding additional features to the system in the coming year to offer support for other kinds of genomic analysis tasks.

References

[1] Keith L Adams and Jonathan F Wendel. Polyploidy and genome evolution in plants. *Current Opinion in Plant Biology*, 8(2):135–141, 2005.

[2] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, 1990.

[3] Jeong-Ho Baek, Junah Kim, Chang-Kug Kim, Seong-Han Sohn, Dongsu Choi, Milind B Ratnaparkhe, Do-Wan Kim, and Tae-Ho Lee. Multisyn: A webtool for multiple synteny detection and visualization of user's sequence of interest compared to public plant species. *Evolutionary Bioinformatics*, 12:EBO–S40009, 2016.

[4] Kiran Bandi. `https://kiranbandi.github.io/10wheatgenomes`. Accessed: Sept. 2019.

[5] Venkat Bandi. `https://github.com/kiranbandi/synvisio`. Accessed: Sept. 2019.

[6] Louis Bavoil, Steven P Callahan, Patricia J Crossno, Juliana Freire, Carlos E Scheidegger, Cláudio T Silva, and Huy T Vo. Vistrails: Enabling interactive multiple-view visualizations. In *VIS 05. IEEE Visualization, 2005.*, pages 135–142. IEEE, 2005.

[7] James K Bonfield, Kathryn F Smith, and Rodger Staden. A new DNA sequence assembly program. *Nucleic Acids Research*, 23(24):4992–4999, 1995.

[8] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. D3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, December 2011.

[9] Cynthia Brewer. `www.colorbrewer2.org`. Accessed: Sept. 2019.

[10] Floréal Cabanettes and Christophe Klopp. D-genies: dot plot large genomes in an interactive, efficient and simple way. *PeerJ*, 6:e4958–e4958, Jun 2018. 29888139[pmid].

[11] Guangqin Cai, Qingyong Yang, Bin Yi, Chuchuan Fan, David Edwards, Jacqueline Batley, and Yongming Zhou. A complex recombination pattern in the genome of allotetraploid brassica napus as revealed by a high-density genetic map. *PLoS One*, 9(10), 2014.

[12] Feng Cheng, Jian Wu, and Xiaowu Wang. Genome triplication drove the diversification of brassica plants. *Horticulture Research*, 1:14024, 2014.

[13] Suzanne Clancy and William Brown. Translation: DNA to mrna to protein. *Nature Education*, 1(1):101, 2008.

[14] Francis S Collins, Eric D Green, Alan E Guttmacher, and Mark S Guyer. A vision for the future of genomics research. *Nature*, 422(6934):835–847, 2003.

[15] International Human Genome Sequencing Consortium et al. Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931, 2004.

[16] International Wheat Genome Sequencing Consortium. `https://www.wheatgenome.org/News/media-resources/fact-sheets-infographics/wheat-a-key-crop-for-food-security`. Accessed: Sept. 2019.

[17] Patricia Costigan-Eaves and Michael Macdonald-Ross. William playfair (1759-1823). *Statistical Science*, pages 318–326, 1990.

[18] Jonathan Crabtree, Samuel V Angiuoli, Jennifer R Wortman, and Owen R White. Sybil: methods and software for multiple genome comparison and visualization. In *Gene Function Analysis*, pages 93–108. Springer, 2007.

[19] Aaron CE Darling, Bob Mau, Frederick R Blattner, and Nicole T Perna. Mauve: multiple alignment of conserved genomic sequence with rearrangements. *Genome Research*, 14(7):1394–1403, 2004.

[20] Alan Dix and Geoffrey Ellis. Starting simple: adding value to static visualisation through simple interaction. In *Proceedings of the working conference on Advanced visual interfaces*, pages 124–134, 1998.

[21] Maureen J Donlin. Using the generic genome browser (gbrowse). *Current Protocols in Bioinformatics*, 28(1):9–9, 2009.

[22] Guénola Drillon, Alessandra Carbone, and Gilles Fischer. Synchro: a fast and easy tool to reconstruct and visualize synteny blocks along eukaryotic chromosomes. *PLoS One*, 9(3):e92621, 2014.

[23] Ensembl. `https://uswest.ensembl.org/info/genome/genebuild/chromosomes_scaffolds_contigs.html`. Accessed: Jan. 2020.

[24] Brent Ewing, LaDeana Hillier, Michael C Wendl, and Phil Green. Base-calling of automated sequencer traces using phred. i. accuracy assessment. *Genome Research*, 8(3):175–185, 1998.

[25] Facebook. `https://www.facebook.com`. Accessed: Sept. 2019.

[26] Mi Feng, Cheng Deng, Evan M Peck, and Lane Harrison. Hindsight: Encouraging exploration through direct encoding of personal interaction history. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):351–360, 2016.

[27] L Fishman, JH Willis, CA Wu, and YW Lee. Comparative linkage maps suggest that fission, not polyploidy, underlies near-doubling of chromosome number within monkeyflowers (mimulus; phrymaceae). *Heredity*, 112(5):562–568, 2014.

[28] Juliana Freire, David Koop, Emanuele Santos, and Cláudio T Silva. Provenance for computational tasks: A survey. *Computing in Science & Engineering*, 10(3):11–21, 2008.

[29] Michael Gleicher. Considerations for visualizing comparison. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):413–423, 2017.

[30] David Gordon, Chris Abajian, and Phil Green. Consed: a graphical tool for sequence finishing. *Genome Research*, 8(3):195–202, 1998.

[31] David Gotz and Michelle X Zhou. Characterizing users' visual analytic activity for insight provenance. *Information Visualization*, 8(1):42–55, 2009.

[32] Neha Gujaria-Verma, Sally L Vail, Noelia Carrasquilla-Garcia, R Varma Penmetsa, Douglas R Cook, Andrew D Farmer, Albert Vandenberg, and Kirstin E Bett. Genetic mapping of legume orthologs reveals high conservation of synteny between lentil species and the sequenced genomes of medicago and chickpea. *Frontiers in plant science*, 5:676, 2014.

[33] Brian J Haas, Arthur L Delcher, Jennifer R Wortman, and Steven L Salzberg. Dagchainer: a tool for mining segmental genome duplications and synteny. *Bioinformatics*, 20(18):3643–3646, 2004.

[34] Kousuke Hanada, Cheng Zou, Melissa D Lehti-Shiu, Kazuo Shinozaki, and Shin-Han Shiu. Importance of lineage-specific expansion of plant tandem duplicates in the adaptive response to environmental stimuli. *Plant Physiology*, 148(2):993–1003, 2008.

[35] Ross C Hardison. Comparative genomics. *PLoS Biology*, 1(2), 2003.

[36] Leland Hartwell, Michael L Goldberg, Janice A Fischer, Leroy E Hood, and Charles F Aquadro. *Genetics: from genes to genomes*. McGraw-Hill New York, 2008.

[37] Asher Haug-Baltzell, Sean A Stephens, Sean Davey, Carlos E Scheidegger, and Eric Lyons. Synmap2 and synmap3d: web-based whole-genome synteny browsers. *Bioinformatics*, 33(14):2197–2198, 2017.

[38] Jeffrey Heer, Jock Mackinlay, Chris Stolte, and Maneesh Agrawala. Graphical histories for visualization: Supporting analysis, communication, and evaluation. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1189–1196, 2008.

[39] Karsten Hokamp, Aoife McLysaght, and Kenneth H Wolfe. The 2r hypothesis and the human genome sequence. In *Genome Evolution*, pages 95–110. Springer, 2003.

[40] Martin Hunt, Chris Newbold, Matthew Berriman, and Thomas D Otto. A comprehensive evaluation of assembly scaffolding tools. *Genome Biology*, 15(3):R42, 2014.

[41] JW Ijdo, Antonio Baldini, DC Ward, ST Reeders, and RA Wells. Origin of human chromosome 2: an ancestral telomere-telomere fusion. *Proceedings of the National Academy of Sciences*, 88(20):9051–9055, 1991.

[42] Roy A Jensen. Orthologs and paralogs-we need to get it right. *Genome Biology*, 2(8):interactions1002–1, 2001.

[43] Donna Karolchik, Robert Baertsch, Mark Diekhans, Terrence S Furey, Angie Hinrichs, YT Lu, Krishna M Roskin, Matthias Schwartz, Charles W Sugnet, Daryl J Thomas, et al. The ucsc genome browser database. *Nucleic Acids Research*, 31(1):51–54, 2003.

[44] W James Kent, Charles W Sugnet, Terrence S Furey, Krishna M Roskin, Tom H Pringle, Alan M Zahler, and David Haussler. The human genome browser at ucsc. *Genome Research*, 12(6):996–1006, 2002.

[45] Martin Krzywinski, Inanc Birol, Steven JM Jones, and Marco A Marra. Hive plots—rational approach to visualizing networks. *Briefings in Bioinformatics*, 13(5):627–644, 2011.

[46] Martin Krzywinski, Jacqueline Schein, Inanc Birol, Joseph Connors, Randy Gascoyne, Doug Horsman, Steven J Jones, and Marco A Marra. Circos: an information aesthetic for comparative genomics. *Genome Research*, 19(9):1639–1645, 2009.

[47] Stefan Kurtz, Adam Phillippy, Arthur L Delcher, Michael Smoot, Martin Shumway, Corina Antonescu, and Steven L Salzberg. Versatile and open software for comparing large genomes. *Genome Biology*, 5(2):R12, 2004.

[48] Jill H Larkin and Herbert A Simon. Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11(1):65–100, 1987.

[49] Anders Larsson. Aliview: a fast and lightweight alignment viewer and editor for large datasets. *Bioinformatics*, 30(22):3276–3278, 2014.

[50] Ed Lee, Nomi Harris, Mark Gibson, Raymond Chetty, and Suzanna Lewis. Apollo: a community resource for genome annotation editing. *Bioinformatics*, 25(14):1836–1837, 2009.

[51] Jongin Lee, Woon-young Hong, Minah Cho, Mikang Sim, Daehwan Lee, Younhee Ko, and Jaebum Kim. Synteny portal: a web-based application portal for synteny block analysis. *Nucleic Acids Research*, 44(W1):W35–W40, 2016.

[52] David J Lipman and William R Pearson. Rapid and sensitive protein similarity searches. *Science*, 227(4693):1435–1441, 1985.

[53] Shengyi Liu, Yumei Liu, Xinhua Yang, Chaobo Tong, David Edwards, Isobel AP Parkin, Meixia Zhao, Jianxin Ma, Jingyin Yu, Shunmou Huang, et al. The brassica oleracea genome reveals the asymmetrical evolution of polyploid genomes. *Nature Communications*, 5:3930, 2014.

[54] Fang Lu, Zhaoyuan Wei, Yongjiang Luo, Hailong Guo, Guoqing Zhang, Qingyou Xia, and Yi Wang. Silkdb 3.0: visualizing and exploring multiple levels of data for silkworm. *Nucleic Acids Research*, 48(D1):D749–D755, 2020.

[55] Nicolas Luc, Jean-Loup Risler, Anne Bergeron, and Mathieu Raffinot. Gene teams: a new formalization of gene clusters for comparative genomics. *Computational biology and chemistry*, 27(1):59–67, 2003.

[56] Andreas Madlung. Polyploidy and its effect on evolutionary success: old questions revisited with new tools. *Heredity*, 110(2):99–104, 2013.

[57] Andrew CR Martin. Viewing multiple sequence alignments with the javascript sequence alignment viewer (jsav). *F1000Research*, 3, 2014.

[58] Thomas C Mathers. Improved genome assembly and annotation of the soybean aphid (aphis glycines matsumura). *G3: Genes, Genomes, Genetics*, 2020.

[59] Miriah Meyer, Tamara Munzner, and Hanspeter Pfister. Mizbee: A multiscale synteny browser. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):897–904, November 2009.

[60] T. Munzner. A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928, Nov 2009.

[61] U Nagaharu and N Nagaharu. Genome analysis in brassica with special reference to the experimental formation of b. napus and peculiar mode of fertilization. 1935.

[62] Maria Nattestad. `http://omgenomics.com/circa/`. Accessed: Jan. 2020.

[63] Saul B Needleman and Christian D Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453, 1970.

[64] National Human Genome Research Institute NHGRI. `https://www.genome.gov/human-genome-project/Completion-faq`. Accessed: Jan. 2020.

[65] National Human Genome Research Institute NHGRI. `https://www.genome.gov/about-genomics/fact-sheets/a-brief-guide-to-genomics`. Accessed: Jan. 2020.

[66] Cydney B Nielsen, Michael Cantor, Inna Dubchak, David Gordon, and Ting Wang. Visualizing genomes: techniques and challenges. *Nature Methods*, 7(3s):S5, 2010.

[67] Cydney B Nielsen, Shaun D Jackman, Inanç Birol, and Steven JM Jones. Abyss-explorer: visualizing genome sequence assemblies. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):881–888, 2009.

[68] Jorge Dionisio Nunez Siri. Accusyn: Using simulated annealing to declutter genome visualizations. Master's thesis, Computer Science, University of Saskatchewan, University of Saskatchewan, November 2019. https://harvest.usask.ca/handle/10388/12368.

[69] Sabrina Nusrat, Theresa Harbig, and Nils Gehlenborg. Tasks, techniques, and tools for genomic data visualization. In *Computer Graphics Forum*, volume 38, pages 781–805. Wiley Online Library, 2019.

[70] Stephen J O'Brien, Marilyn Menotti-Raymond, William J Murphy, William G Nash, Johannes Wienberg, Roscoe Stanyon, Neal G Copeland, Nancy A Jenkins, James E Womack, and Jennifer A Marshall Graves. The promise of comparative genomics in mammals. *Science*, 286(5439):458–481, 1999.

[71] Trevor O'Brien, Anna Ritz, Benjamin Raphael, and David Laidlaw. Gremlin: an interactive visualization model for analyzing genomic rearrangements. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):918–926, 2010.

[72] World Health Organization et al. Genomics and world health: Report of the advisory committee on health research. 2002.

[73] Sarah P Otto. The evolutionary consequences of polyploidy. *Cell*, 131(3):452–462, 2007.

[74] Georgia Panopoulou and Albert J Poustka. Timing and mechanism of ancient vertebrate genome duplications–the adventure of a hypothesis. *TRENDS in Genetics*, 21(10):559–567, 2005.

[75] I AP Parkin, AG Sharpe, and DJ Lydiate. Patterns of genome duplication within the brassica napus genome. *Genome*, 46(2):291–303, 2003.

[76] IAP Parkin, AG Sharpe, DJ Keith, and DJ Lydiate. Identification of the a and c genomes of amphidiploid brassica napus (oilseed rape). *Genome*, 38(6):1122–1131, 1995.

[77] Isobel Parkin. `https://pag.confex.com/pag/xxvii/recordingredirect.cgi/oid/Recording3903/paper33268_1.pd`. Accessed: Sept. 2019.

[78] Eberhard Passarge, Bernhard Horsthemke, and Rosann A Farber. Incorrect use of the term synteny. *Nat Genet*, 23(4):387, 1999.

[79] PL Pearson. The uniqueness of the human karyotype. *Chromosome identification techniques and application in biology and medicine*, page 145, 1973.

[80] George H Perry, Nathaniel J Dominy, Katrina G Claw, Arthur S Lee, Heike Fiegler, Richard Redon, John Werner, Fernando A Villanea, Joanna L Mountain, Rajeev Misra, et al. Diet and the evolution of human amylase gene copy number variation. *Nature Genetics*, 39(10):1256–1260, 2007.

[81] Alexander Pertsemlidis and John W Fondon. Having a blast with bioinformatics (and avoiding blastphemy). *Genome Biology*, 2(10):reviews2002–1, 2001.

[82] Sampath Perumal, Chu Shin Koh, Lingling Jin, Miles Buchwaldt, Erin Higgins, Chunfang Zheng, David Sankoff, Stephen J Robinson, Sateesh Kagale, Zahra-Katy Navabi, et al. High contiguity long read assembly of brassica nigra allows localization of active centromeres and provides insights into the ancestral brassica genome. *BioRxiv*, 2020.

[83] William A Pike, John Stasko, Remco Chang, and Theresa A O'connell. The science of interaction. *Information Visualization*, 8(4):263–274, 2009.

[84] Harald Piringer, Christian Tominski, Philipp Muigg, and Wolfgang Berger. A multi-threading architecture to support interactive visual exploration. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1113–1120, 2009.

[85] CoGe Web Platform. `https://genomevolution.org/coge`. Accessed: Jan. 2019.

[86] Hartmut Prautzsch, Wolfgang Boehm, and Marco Paluszny. *Bezier and B-Spline Techniques*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.

[87] Leslie A Pray. Semi-conservative DNA replication: Meselson and stahl. *Nature Education*, 1(1):98, 2008.

[88] 10+ Genome Project. `http://www.10wheatgenomes.com/`. Accessed: Sept. 2019.

[89] Sebastian Proost, Jan Fostier, Dieter De Witte, Bart Dhoedt, Piet Demeester, Yves Van de Peer, and Klaas Vandepoele. i-adhore 3.0—fast and sensitive detection of genomic homology in extremely large data sets. *Nucleic Acids Research*, 40(2):e11–e11, 2011.

[90] React. `https://reactjs.org/`. Accessed: Sept. 2019.

[91] James H Renwick. The mapping of human chromosomes. *Annual review of genetics*, 5(1):81–120, 1971.

[92] Kashi V Revanna, Chi-Chen Chiu, Ezekiel Bierschank, and Qunfeng Dong. Gsv: a web-based genome synteny viewer for customized data. *BMC Bioinformatics*, 12(1):316, 2011.

[93] Kashi V Revanna, Daniel Munro, Alvin Gao, Chi-Chen Chiu, Anil Pathak, and Qunfeng Dong. A web-based multi-genome synteny viewer for customized data. *BMC Bioinformatics*, 13(1):190, 2012.

[94] MA Reza, S Swarup, and RM Kini. Structure of two genes encoding parallel prothrombin activators in tropidechis carinatus snake: gene duplication and recruitment of factor x gene to the venom gland. *Journal of Thrombosis and Haemostasis*, 5(1):117–126, 2007.

[95] Rigomar Rieger, Arnd Michaelis, and Melvin M Green. *Glossary of genetics: classical and molecular*. Springer Science & Business Media, 2012.

[96] J. C. Roberts. State of the art: Coordinated multiple views in exploratory visualization. In *Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV 2007)*, pages 61–71, July 2007.

[97] George Robertson, Mary Czerwinski, Kevin Larson, Daniel C. Robbins, David Thiel, and Maarten van Dantzich. Data mountain: Using spatial memory for document management. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*, UIST '98, pages 153–162, New York, NY, USA, 1998. ACM.

[98] Christian Rödelsperger and Christoph Dieterich. Cyntenator: progressive gene order alignment of 17 vertebrate genomes. *PloS One*, 5(1):e8861, 2010.

[99] Louis Rosenfeld and Peter Morville. *Information architecture for the world wide web*. O'Reilly Media, Inc., 2002.

[100] Gerald M Rubin, Mark D Yandell, Jennifer R Wortman, George L Gabor, Catherine R Nelson, Iswar K Hariharan, Mark E Fortini, Peter W Li, Rolf Apweiler, Wolfgang Fleischmann, et al. Comparative genomics of the eukaryotes. *Science*, 287(5461):2204–2215, 2000.

[101] Taro L Saito, Jun Yoshimura, Shin Sasaki, Budrul Ahsan, Atsushi Sasaki, Reginaldo Kuroshu, and Shinichi Morishita. Utgb toolkit for personalized genome browsers. *Bioinformatics*, 25(15):1856–1861, 2009.

[102] Lacey-Anne Sanderson, Carolyn T Caron, Reynold Tan, Yichao Shen, Ruobin Liu, and Kirstin E Bett. Knowpulse: a web-resource focused on diversity data for pulse crop improvement. *Frontiers in Plant Science*, 10, 2019.

[103] Michael C Schatz, Adam M Phillippy, Ben Shneiderman, and Steven L Salzberg. Hawkeye: an interactive visual analytics tool for genome assemblies. *Genome Biology*, 8(3):R34, 2007.

[104] Kamran Sedig and Paul Parsons. Interaction design for complex cognitive activities with visual representations: A pattern-based approach. *AIS Transactions on Human-Computer Interaction*, 5(2):84–133, 2013.

[105] Kamran Sedig, Sonja Rowhani, and Hai-Ning Liang. Designing interfaces that support formation of cognitive maps of transitional processes: an empirical study. *Interacting with computers*, 17(4):419–452, 2005.

[106] Cathal Seoighe. Turning the clock back on ancient genome duplication. *Current Opinion in Genetics & Development*, 13(6):636–643, 2003.

[107] Cathal Seoighe and Chris Gehring. Genome duplication led to highly selective expansion of the arabidopsis thaliana proteome. *Trends in Genetics*, 20(10):461–464, 2004.

[108] Fereidoon Shahidi. *Canola and rapeseed: production, chemistry, nutrition, and processing technology*. Springer Science & Business Media, 1990.

[109] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *In IEEE Symposium on Visual Languages*, pages 336–343, 1996.

[110] Amit U Sinha and Jaroslaw Meller. Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. *BMC Bioinformatics*, 8(1):82, 2007.

[111] Mitchell E Skinner, Andrew V Uzilov, Lincoln D Stein, Christopher J Mungall, and Ian H Holmes. Jbrowse: a next-generation genome browser. *Genome Research*, 19(9):1630–1638, 2009.

[112] Amy Skopik and Carl Gutwin. Finding things in fisheyes: Memorability in distorted spaces. In *in Conference on Graphics Interface GI'03*. Citeseer, 2003.

[113] Temple F Smith, Michael S Waterman, et al. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147(1):195–197, 1981.

[114] Carol Soderlund, Matthew Bomhoff, and William M Nelson. Symap v3. 4: a turnkey synteny system with application to plant genomes. *Nucleic Acids Research*, 39(10):e68–e68, 2011.

[115] Douglas E Soltis and Pamela S Soltis. The role of phylogenetics in comparative genetics. *Plant Physiology*, 132(4):1790–1800, 2003.

[116] Erik L.L. Sonnhammer and Richard Durbin. A dot-matrix program with dynamic threshold control suited for genomic DNA and protein sequence analysis. *Gene*, 167(1):GC1 – GC10, 1995.

[117] James Stalker, Brian Gibbins, Patrick Meidl, James Smith, William Spooner, Hans-Rudolf Hotz, and Antony V Cox. The ensembl web site: mechanics of a genome browser. *Genome Research*, 14(5):951–955, 2004.

[118] Lincoln D Stein, Christopher Mungall, ShengQiang Shu, Michael Caudy, Marco Mangone, Allen Day, Elizabeth Nickerson, Jason E Stajich, Todd W Harris, Adrian Arva, et al. The generic genome browser: a building block for a model organism system database. *Genome Research*, 12(10):1599–1610, 2002.

[119] Keith Stenning and Jon Oberlander. A cognitive theory of graphical and linguistic reasoning: Logic and implementation. *Cognitive Science*, 19(1):97–140, 1995.

[120] C. Stolte, D. Tang, and P. Hanrahan. Multiscale visualization using data cubes. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):176–187, April 2003.

[121] Helga Thorvaldsdóttir, James T Robinson, and Jill P Mesirov. Integrative genomics viewer (igv): high-performance genomics data visualization and exploration. *Briefings in Bioinformatics*, 14(2):178–192, 2013.

[122] Christian Tominski. Interaction for visualization. *Synthesis Lectures on Visualization*, 3(1):1–107, 2015.

[123] Edward R Tufte, Nora Hillman Goeler, and Richard Benson. *Envisioning Information*, volume 126. Graphics press Cheshire, CT, 1990.

[124] David Wayne Ussery, Trudy M Wassenaar, and Stefano Borini. *Computing for comparative microbial genomics: bioinformatics for microbiologists*, volume 8. Springer Science & Business Media, 2009.

[125] Ajit Varki and Tasha K Altheide. Comparing the human and chimpanzee genomes: searching for needles in a haystack. *Genome Research*, 15(12):1746–1758, 2005.

[126] Rajeev K Varshney, Chi Song, Rachit K Saxena, Sarwar Azam, Sheng Yu, Andrew G Sharpe, Steven Cannon, Jongmin Baek, Benjamin D Rosen, Bunyamin Tar'an, et al. Draft genome sequence of chickpea (cicer arietinum) provides a resource for trait improvement. *Nature Biotechnology*, 31(3):240, 2013.

[127] Daniel Veltri, Martha Malapi Wight, and Jo Anne Crouch. Simplesynteny: a web-based tool for visualization of microsynteny across multiple species. *Nucleic Acids Research*, 44(W1):W41–W45, 2016.

[128] Yupeng Wang, Haibao Tang, Jeremy D DeBarry, Xu Tan, Jingping Li, Xiyin Wang, Tae-ho Lee, Huizhe Jin, Barry Marler, Hui Guo, et al. Mcscanx: a toolkit for detection and evolutionary analysis of gene synteny and collinearity. *Nucleic Acids Research*, 40(7):e49–e49, 2012.

[129] Michelle Q Wang Baldonado, Allison Woodruff, and Allan Kuchinsky. Guidelines for using multiple views in information visualization. In *Proceedings of the working conference on Advanced visual interfaces*, pages 110–119, 2000.

[130] Michelle Q. Wang Baldonado, Allison Woodruff, and Allan Kuchinsky. Guidelines for using multiple views in information visualization. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '00, pages 110–119, New York, NY, USA, 2000. ACM.

[131] Colin Ware. *Information visualization: perception for design*. Elsevier, 2012.

[132] Andrew M Waterhouse, James B Procter, David MA Martin, Michèle Clamp, and Geoffrey J Barton. Jalview version 2—a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25(9):1189–1191, 2009.

[133] Liping Wei, Yueyi Liu, Inna Dubchak, John Shon, and John Park. Comparative genomics approaches to study organism similarities and differences. *Journal of biomedical informatics*, 35(2):142–150, 2002.

[134] Yang Dong Wei Chen. `http://teabase.ynau.edu.cn/synvisio.html`. Accessed: Sept. 2019.

[135] Yang Dong Wei Chen. `http://vitisgdb.ynau.edu.cn/`. Accessed: Sept. 2019.

[136] Max Wertheimer. Untersuchungen zur lehre von der gestalt. ii. *Psychological Research*, 4(1):301–350, 1923.

[137] World Wide Web Consortium. Web Workers, 2013.

[138] Yiqing Xu, Changwei Bi, Guoxin Wu, Suyun Wei, Xiaogang Dai, Tongming Yin, and Ning Ye. Vgsc: a web-based vector graph toolkit of genome synteny and collinearity. *BioMed research international*, 2016, 2016.

[139] Guy Yachdav, Sebastian Wilzbach, Benedikt Rauscher, Robert Sheridan, Ian Sillitoe, James Procter, Suzanna E Lewis, Burkhard Rost, and Tatyana Goldberg. Msaviewer: interactive javascript visualization of multiple sequence alignments. *Bioinformatics*, 32(22):3501–3503, 2016.

[140] Ji Soo Yi, Youn ah Kang, and John Stasko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224–1231, 2007.

[141] Xinghuo Zeng, Matthew J Nesbitt, Jian Pei, Ke Wang, Ismael A Vergara, and Nansheng Chen. Orthocluster: a new tool for mining synteny blocks and applications in comparative genomics. In *Proceedings of the 11th international conference on Extending database technology: Advances in database technology*, pages 656–667, 2008.

[142] Tao Zhao and M Eric Schranz. Network-based microsynteny analysis identifies major differences and genomic outliers in mammalian and angiosperm genomes. *Proceedings of the National Academy of Sciences*, 116(6):2165–2174, 2019.

[143] Hong Zhou, Panpan Xu, Xiaoru Yuan, and Huamin Qu. Edge bundling in information visualization. *Tsinghua Science and Technology*, 18(2):145–156, 2013.

Appendix A

Exploring Conservation in Wheat

To demonstrate SynVisio lets walk through the process of exploring conservation in Wheat. The data has been preloaded and can be accessed at `https://synvisio.usask.ca/#/Dashboard/ta_cs`.

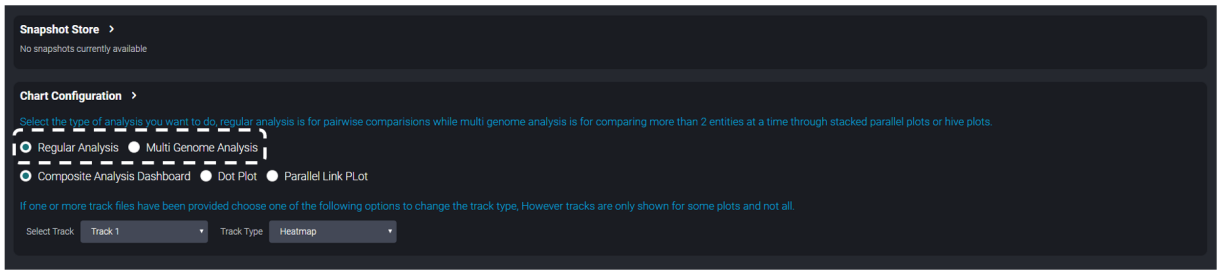

Figure A.1: Select analysis mode

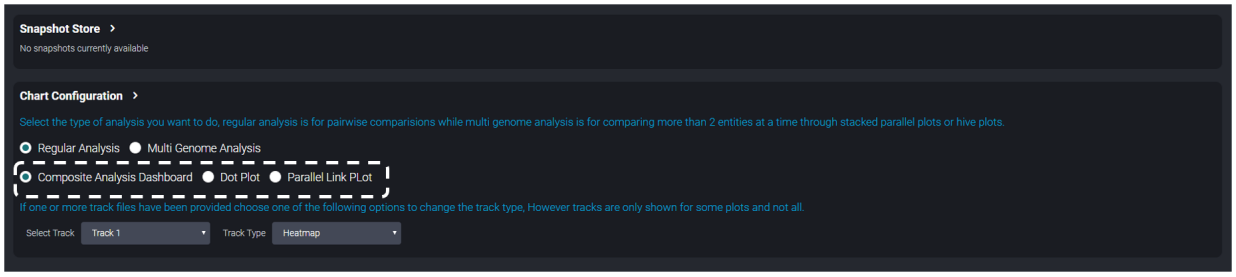

Figure A.2: Select default dashboard or an individual plot type

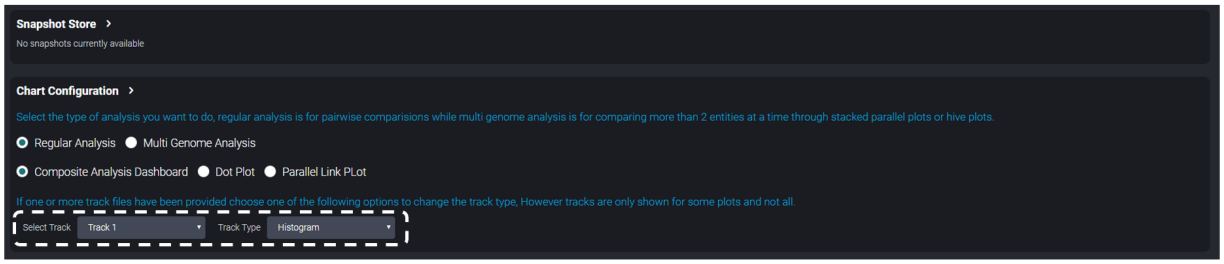

Figure A.3: Select track type for supplementary datasets

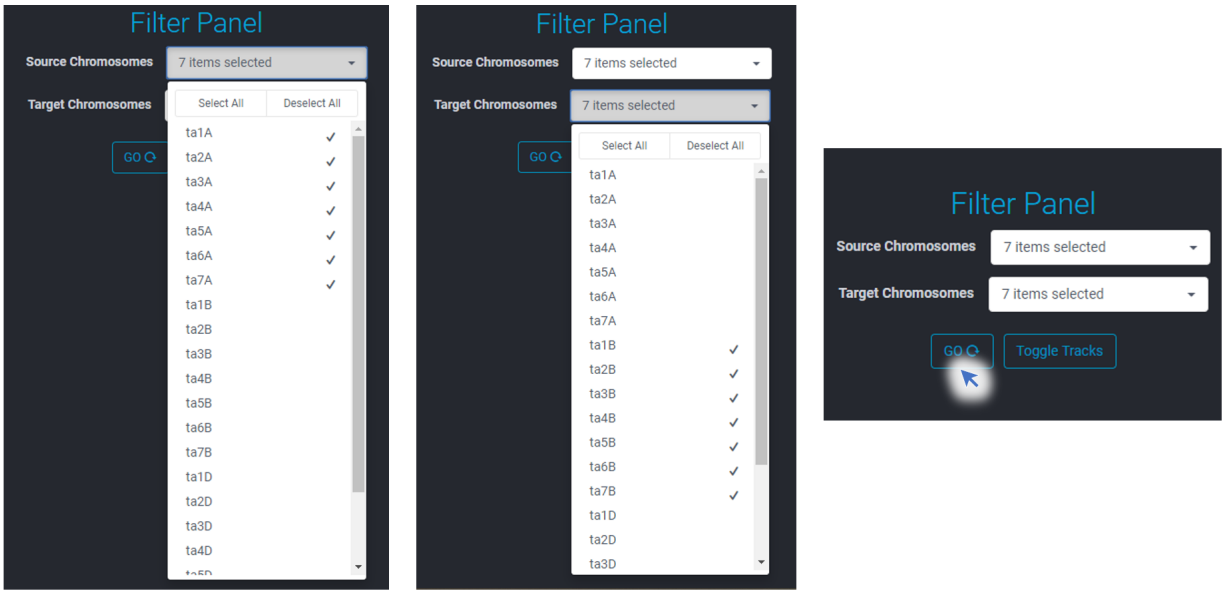

Figure A.4: Select source and target chromosomes which in this case belong to two sub genomes of wheat (A and B donors)

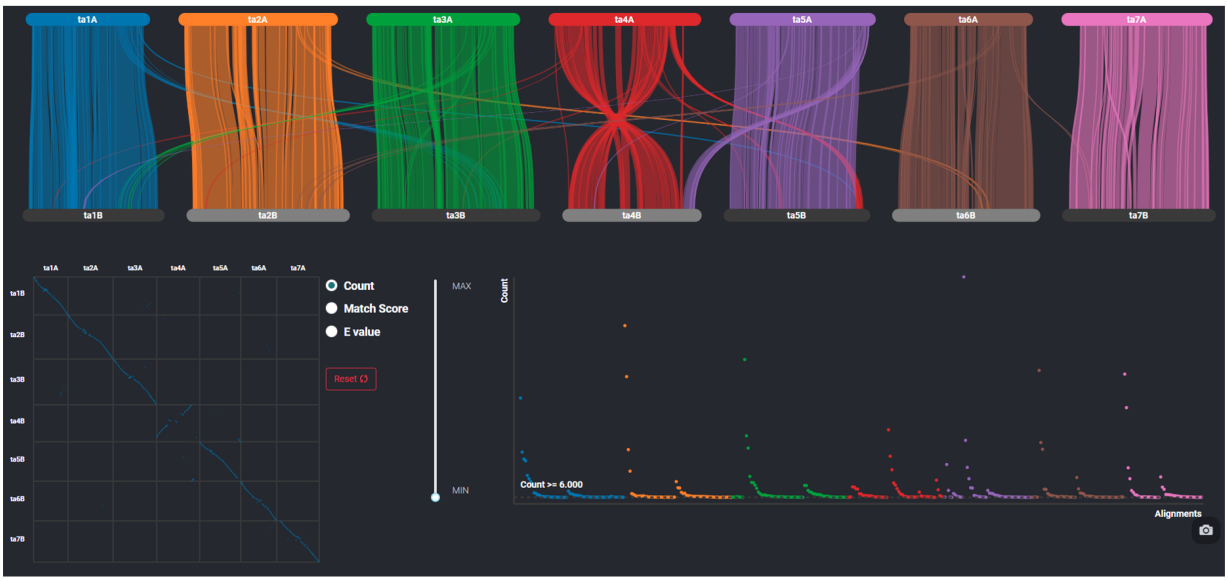

Figure A.5: Composite analysis dashboard showing conservation between two sub genomes of wheat (A and B donors)

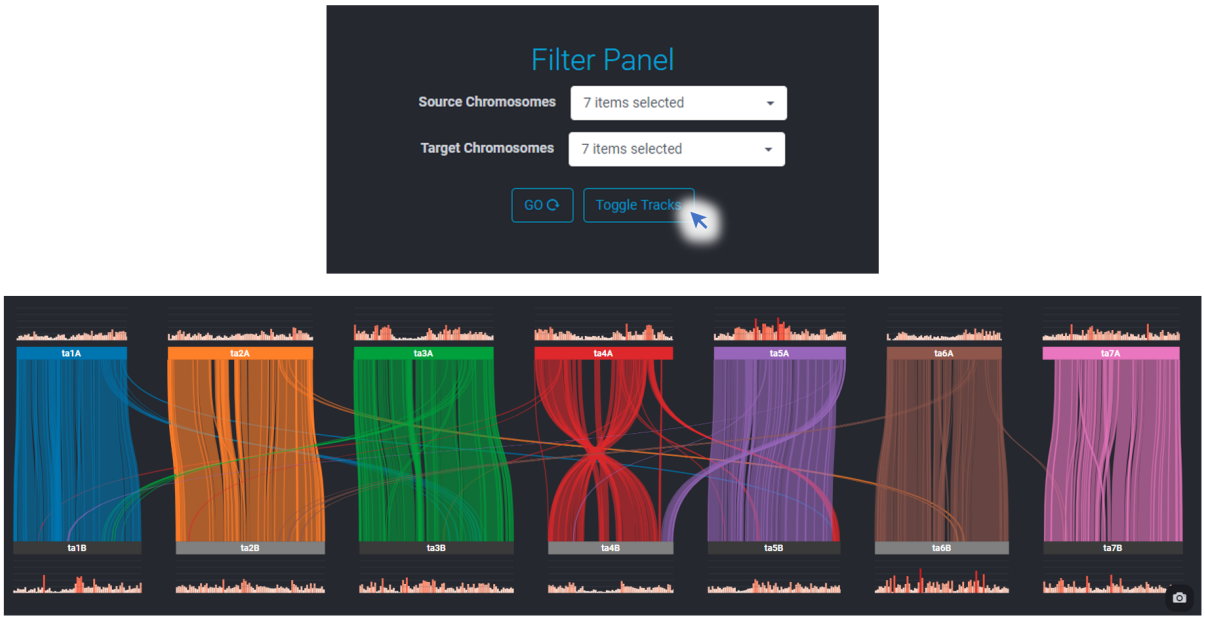

Figure A.6: Toggle track visibility

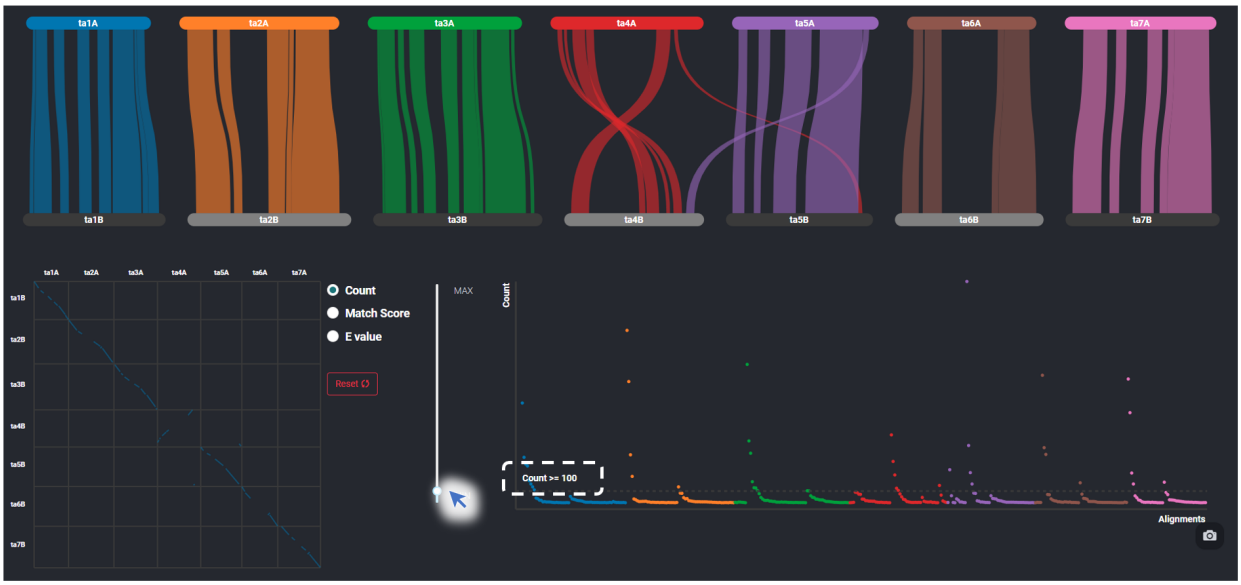

Figure A.7: Filter conserved regions by gene count

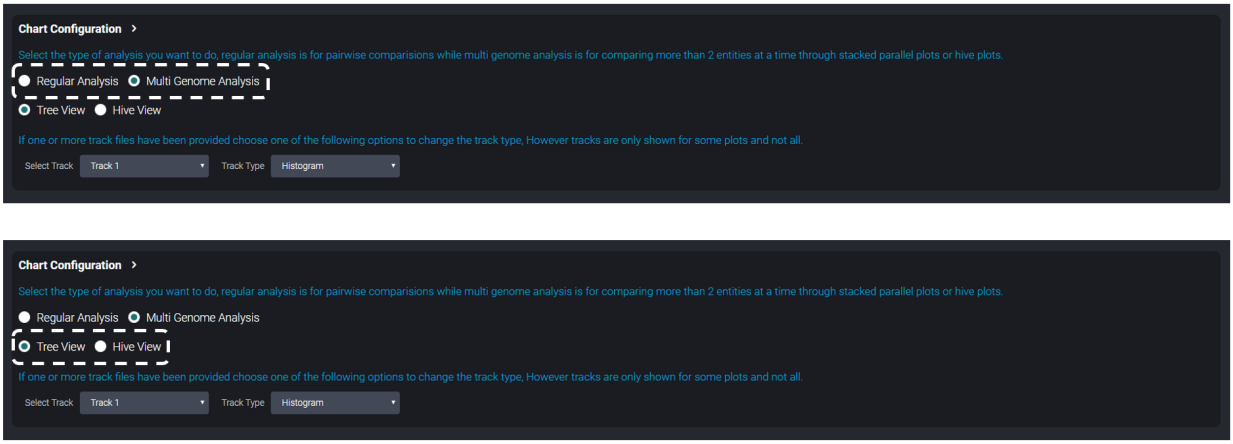

Figure A.8: Select multi genome analysis and tree view

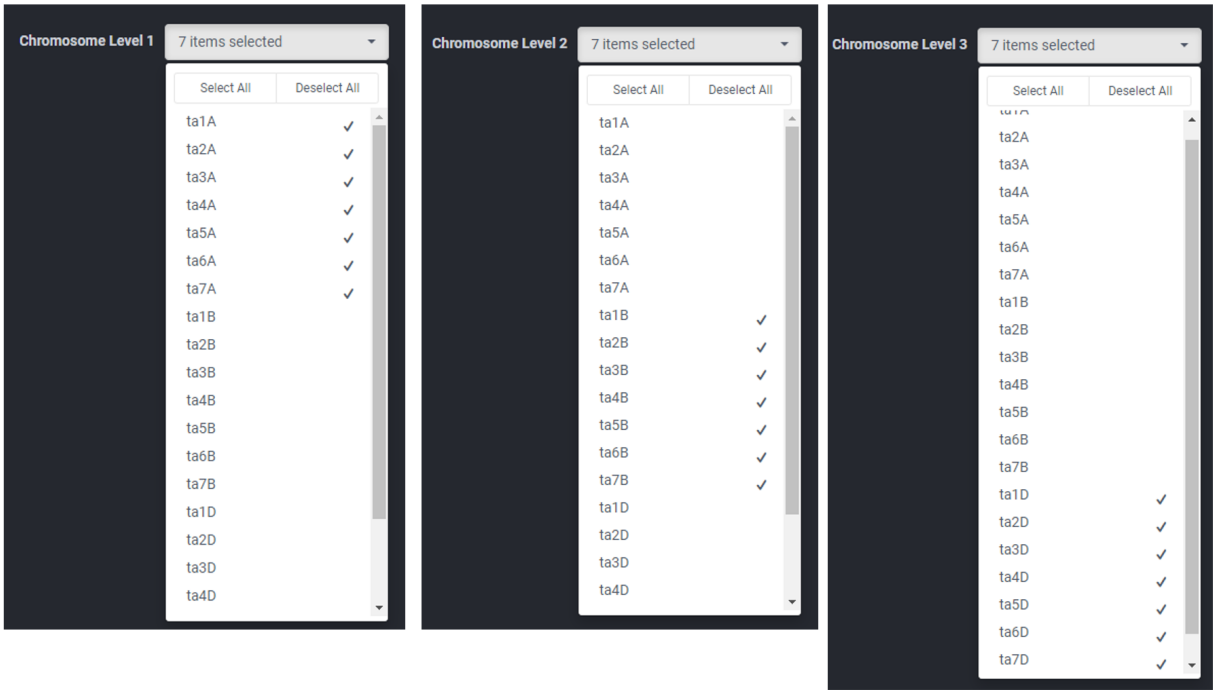

Figure A.9: Select chromosomes in each of the sub genomes of wheat.

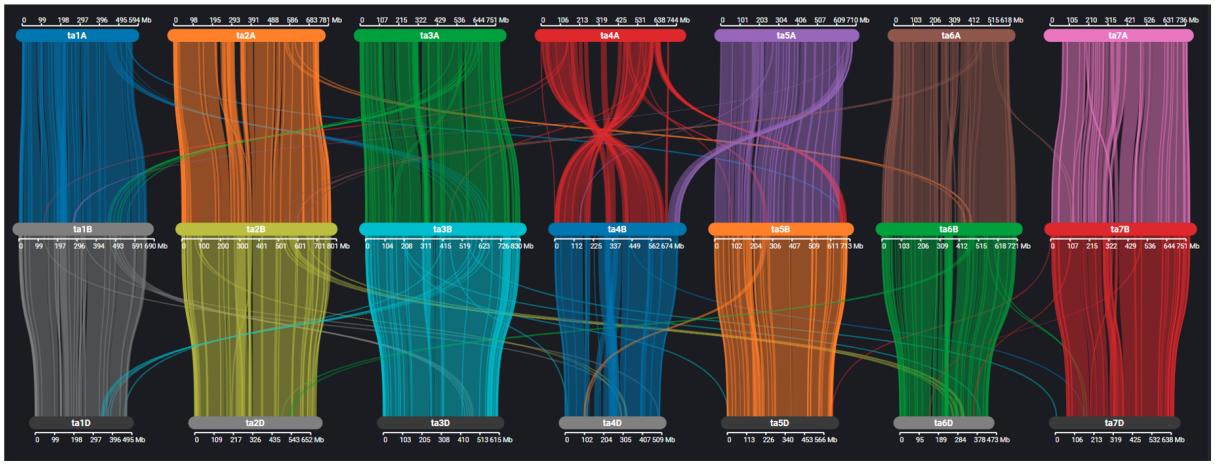

Figure A.10: Tree view for multi genome analysis showing conservation between the three sub genomes of wheat (A, B, and D donors)

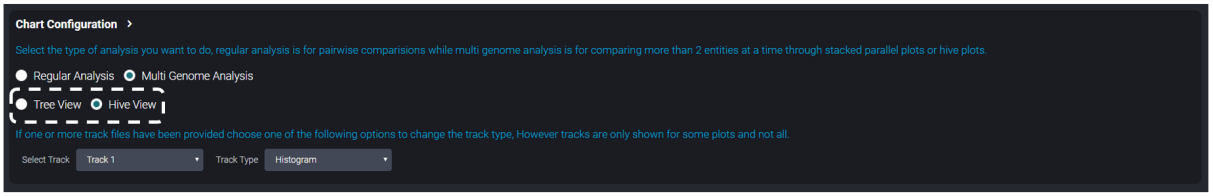

Figure A.11: Select multi genome analysis and hive view, then turn on normalized scales and chromosome labels

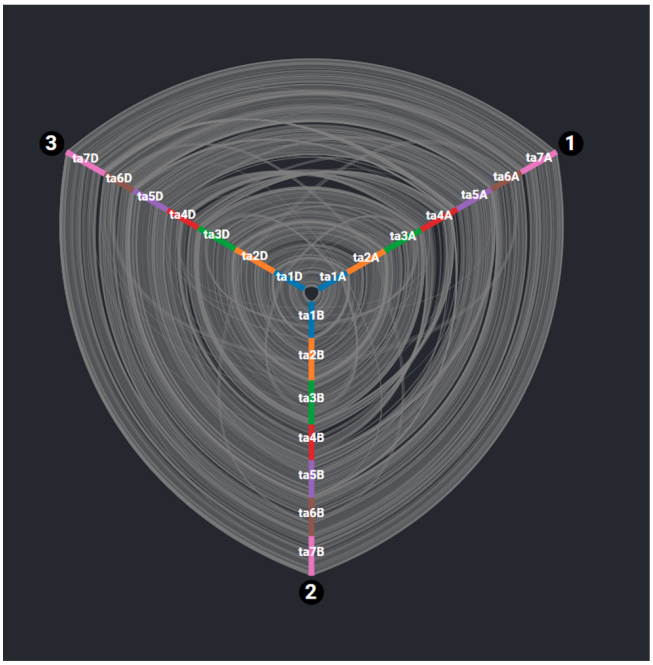

Figure A.12: Hive view showing conservation between the three sub genomes of wheat (A, B, and D donors)

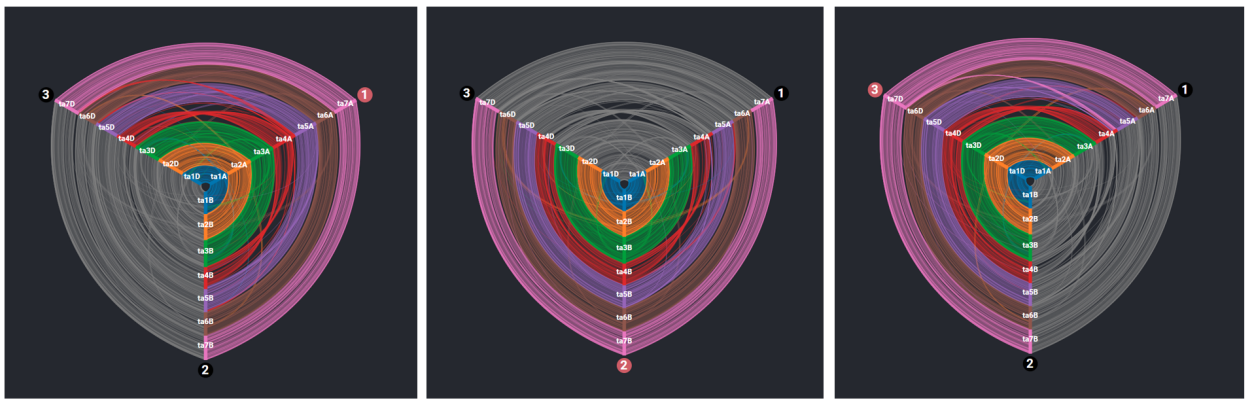

Figure A.13: Highlight conserved regions emerging from each sub genome by clicking on the corresponding marker for that genome.