DOI: 10.1111/cgf.70147

Eurographics Conference on Visualization (EuroVis) 2025
W. Aigner, N. Andrienko, and B. Wang
(Guest Editors)

COMPUTER GRAPHICS *forum*

Multipla: Multiscale Pangenomic Locus Analysis

A. van den Brandt[1,2,*], E. Ståhlbom[3,4,*],
F.J.M. van Workum[2], H. van de Wetering[1], C. Lundström[3,4,5], S. Smit[2], and A. Vilanova[1]

[1]Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands
[2]Bioinformatics Group, Wageningen University & Research, The Netherlands
[3]Department of Science and Technology, Linköping University, Sweden
[4]Sectra AB, Sweden [5]Center for Medical Image Science and Visualization, Linköping University, Sweden
*Authors have contributed equally to this work

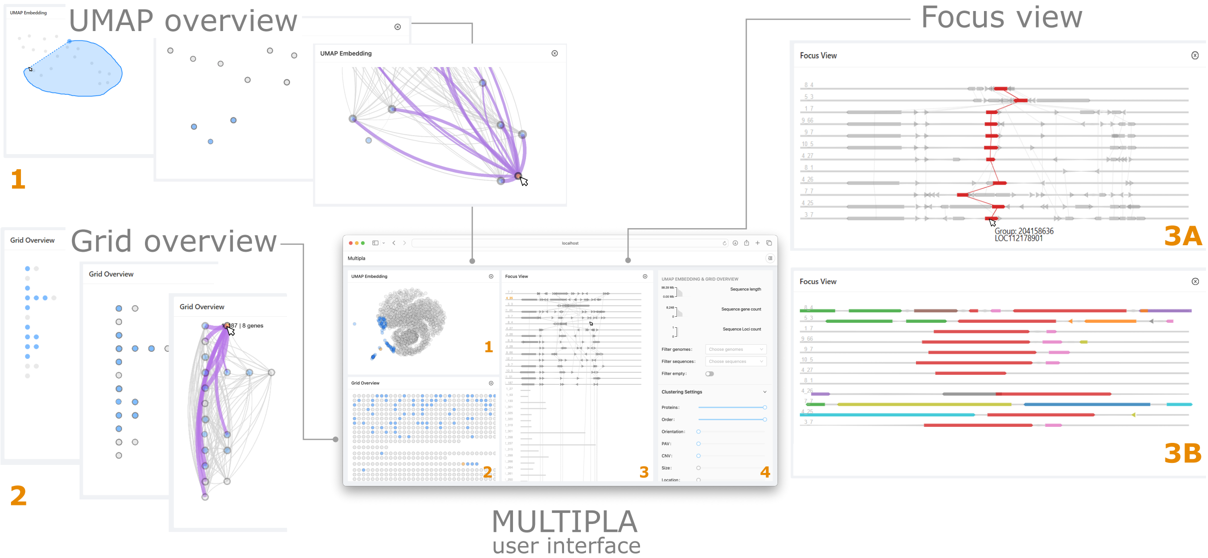


Figure 1: *The conceptual design of* MULTIPLA*, annotated to show the main components. The UMAP Overview (1) and the Grid Overview (2) show overviews of which sequences are present in the data with three semantic zoom levels, allowing lasso selection. The Focus View (3) displays the sequences selected in the overviews in two different levels of detail shown in 3A and 3B. The Control Panel (4) provides summary metrics and can be used to configure the views. The figure consists of different views on rose data.*

Abstract

Comparing gene organization across genomic sequences reveals insights into evolutionary and functional diversity among different organisms and varieties. Performing this task across many sequences, such as from a pangenome, is challenging because of the scale, the density of information, and the inherent variation. Often, analyses are centered on a genomic region of interest—a locus that might be associated with a trait or contain genes within the same family or biological pathway. Within these regions, researchers examine the conservation of gene order and orientation across organisms and assess sequence similarity, along with other gene content features such as gene size, to find biological variations or potential errors in the data. Automated methods in comparative genomics struggle to identify meaningful patterns due to varying and often unknown features of interest, leaving manual, time-intensive, and scalability-challenged visualization as the primary alternative. To address these challenges, we present a multiscale design for studying gene organization within pangenomes, developed in close collaboration with domain experts. Our tool, MULTIPLA*, enables users to explore organization at multiple levels of detail in a decluttered manner through layout abstractions, semantic zooming, and layouts with flexible distance definitions and feature selections, combining the advantages of manual and automated methods used in practice. We evaluate the design of* MULTIPLA *through two pangenomic use cases and conclude with lessons learned from designing multiscale views for pangenomic locus analysis.*

CCS Concepts

• ***Human-centered computing*** → *Visual analytics; Visualization design and evaluation methods;* • ***Applied computing*** → *Genomics;*

1 Introduction

Studying the way genes are organized on genomes is an important topic in comparative genomics. Gene organization—referring at a minimum to their order—influences how genes work together in biological processes and contribute to an organism's observable and biochemical characteristics (i.e., phenotype). Comparing gene organizations across genomes can reveal variations, indicating either variation in biological functionality or errors in the genome assembly or annotation, both of which are valuable to uncover for understanding diversity and improving data quality. It can provide insights into evolutionary relationships or explain the genetics underlying differences in function and characteristics, as genomes change over time under evolutionary pressures. For example, genes, or entire regions, can change position, become duplicated, or get lost, all of which can severely impact an organism by causing disease, affecting morphology, inducing resistance, etcetera.

The field of comparative genomics increasingly makes use of pangenomes to study gene organization to extend the analysis scope. Pangenomes combine the genetic content of multiple related organisms to capture their diversity [M*18]. Thus they enable all-versus-all analysis of the gene organization as opposed to traditional approaches that use a single genome as a reference for comparison. Such reference-based approaches are biased towards the contents of the chosen reference, often ignoring genomic regions that are missing in or too different from the reference. Moreover, reference-free pangenomes enable the analysis of organization both across genomes and within genomes, thus providing a more complete picture. This is especially relevant for diploid and polyploid organisms containing two or more (likely non-identical) copies of the genome respectively [VdPMM17].

However, interpreting pangenomic information is challenging due to the complex nature of the data and the need to integrate multiple sources of information. Distinguishing common variation from relevant deviations is difficult with automated methods alone as the importance of features is not known in advance and may vary between organisms and genomic contexts. Therefore, integrating automated analyses with human domain expertise is necessary to explore and interpret gene organization patterns effectively.

Researchers rely on visualization tools for exploring gene organization across (pan)genomes. However, our interactions with domain experts uncovered that existing tools make assumptions that limit their flexibility and effectiveness for pangenome exploration. These include pre-determining which features are most important for studying gene organization or which order to use when displaying sequences. There is also a lack of methods for effective interaction with the complex and inherently multiscale pangenome data. Thus, there is an important gap for visualization research to fill.

In this paper, we present MULTIPLA, an interactive visual interface for exploring gene organizations at multiple scales in a pangenomic region of interest or locus. Developed in close collaboration with domain experts, MULTIPLA combines interactive clustering with flexible encoding and rearrangement options. It supports multiple levels of detail, across several linked views, enabling researchers to explore genes in their genomic context and facilitating comparisons. The components of the developed visualization design also bring several distinct research contributions: (1) the introduction of interactively customizable distance metrics for clustering, (2) a force-directed layout achieving reference-free alignment, and (3) a semantic zoom approach for multiscale analysis combining multiple comparison strategies.

With MULTIPLA, we also contribute to the transformation of established visualization techniques into a domain-specific design for (pan)genomics, demonstrating how these techniques can be adapted to support user needs. We evaluate the effectiveness of our approaches with two case studies involving pangenomes of roses and peppers, and share lessons learned from our design process.

2 Biological Background

The **genome** of an organism is the complete set of genetic material, found in each cell. It consists of one or more large DNA molecules, called chromosomes. Chromosomes are chains of four types of nucleotides or bases: Adenine, Thymine, Guanine, and Cytosine. Because sequencing technologies cannot read entire chromosomes front to back, shorter reads have to be assembled into larger **sequences**, representing stretches of nucleotides in the DNA. In genomics, this **assembly** of sequences represents the genome of an organism (Fig. 2). So-called "phased" assemblies separate maternally and paternally inherited (non-identical) copies, enabling the analysis of subgenomes. Depending on the assembly quality, sequences may correspond to complete chromosomes or merely parts of them. Furthermore, assemblies vary widely in completeness.

A core concept in genomics data analysis is the **gene**. Genes are

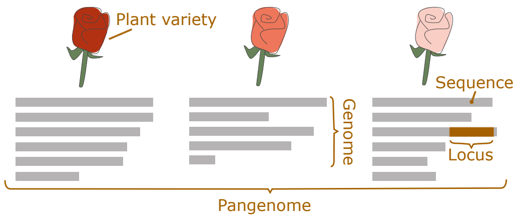


Figure 2: *Schematic of the basic concepts of pangenomes. A pangenome consists of the genomes of a number of plant varieties. Each genome is an assembly of several sequences. A* ***locus*** *is a section of a sequence.*

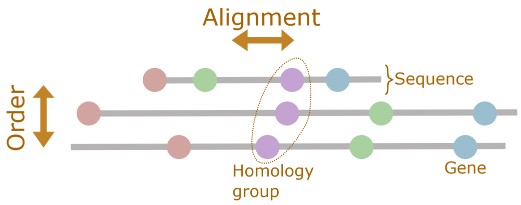


Figure 3: *Basic concepts of pangenome visualization. A sequence has a length and contains genes. Genes belong to homology groups. The concept of* ***alignment*** *refers to the x-position of genes and sequences, and the concept of* ***order*** *refers to the order of the rows.*

segments of the DNA with a function, like encoding a molecule. A protein-coding gene produces one or more messenger RNAs (mRNAs), each of which gets translated into a protein. In genomics, genes and mRNAs are computationally predicted, *ab initio* or based on evidence, and represented as annotations on a genome assembly. Sequences are often visualized as stacked horizontal tracks, with glyphs representing the presence of genes, see Fig. 3. For additional background we refer to the survey by Nusrat et al. [NHG19].

Comparative analysis of these genes relies strongly on the concepts of **homology** and **synteny** [Tek16]. Homologous genes, conceptually "the same" gene across species, are evolutionarily related and often identified through similarity of (protein) sequence. Synteny, the conservation of gene order and orientation among genes, is used as additional evidence to disentangle evolutionary relationships. To further inform the analysis, various other gene features are used, such as size, location, presence-absence variation (PAV), and copy number variation (CNV), as well as sequence context metrics, such as the sequence length or nucleotide frequency. Together, these features provide a rich source to identify and characterize the majority pattern of organization and variations thereof. It is often unclear in advance which feature will define a variation, making an exploratory approach essential.

Pangenomes enable the analysis of gene organization across multiple genomes from different (related) organism varieties, see Fig. 2. A **locus** is a section of a sequence. Analysis typically starts with a query locus on one of the genomes containing one or more genes of interest. The pangenome is used to identify matching loci across the other genomes based on sequence similarity and/or homology relationships between genes. In addition to genes that are physically co-localized within a locus, such as gene clusters or families, researchers also explore non-clustered sets of genes such as those involved in a biological pathway.

3 Characterization of Pangenome Locus Analysis

Through two informal in-person meetings (two hours each) with three genomics researchers—including two co-author collaborators, front-line analysts [SMM12], with whom we met weekly (one hour) for a year—we identified key user questions and tasks related to exploring gene organizations. As a starting point for exploration, we assume that the user has queried a locus or set of genes from the pangenome. We identified five main tasks (labeled T1– T5):

T1 *Identify and compare gene content across sequences.* Before diving into the organization patterns, users need an overview of the gene content across sequences, addressing questions such as: How many sequences, with how many genes, are there in the pangenome? Which of those have loci matching the query, and how many genes do those loci contain? How many homology groups are shared between the sequences?

T2 *Select sequences for gene organization analysis.* After having gained an overview of the gene content across sequences, users want to select sequences for analysis of their gene organization.

T3 *Explore and identify feature patterns.* Users want to assess how the organizations of the genes across sequences may differ from the perspectives of various (gene) features such as (protein) sequence similarity from homology, order, orientation, PAV, CNV, size, and position. What is the majority organization pattern for different (combinations of) features? What features contribute to deviations? Are neighboring genes relative to a target conserved?

T4 *Browse sequence context information.* To gain further insights into the biological meaning and likelihood of the gene organization patterns, sequence context information is taken into account. For example: what is the length of a sequence? What is the overall count of genes on this sequence (beyond the selected locus)?

T5 *Corroborate patterns.* To understand whether a gene organization deviation could be an annotation issue or a true biological variation, users want to cross-correlate it with additional (sequence) data attributes that can be mapped to that location, e.g., mRNA abundance, gene size, or phenotype information.

4 Related Work

In this section, we review literature related to synteny, pangenome, and gene visualization. We also include force-directed and multiscale layouts as they are relevant to our proposed solution.

4.1 Synteny Visualization

Many tools for visualizing gene organization calculate and show macro-level synteny, allowing researchers to identify genomic regions of conserved gene order and arrangement. So-called syntenic blocks (of genes) are calculated across the genomes or chromosomes and are typically displayed in linear (e.g., Apollo [LSH*02], Cinteny [SM07]) or circular (e.g., Circos [KSB*09], MizBee [MMP09]) layouts for comparison. Synteny relations between these blocks, such as size, orientation, and position on the genome, are then depicted using a combination of color, connection links, and glyph encodings. The above tools are all designed to support pairwise comparisons to a reference genome. Tools that allow beyond pairwise, pangenomic, comparisons use flexible layouts (e.g., GENESPACE [LSS*22], NGenomeSyn [HYJ*23]). These tools mainly aid in macro-level synteny exploration. Some tools allow inspections at the micro-level, i.e., showing genes in addition to blocks, but with limited context *(T4)*, feature details *(T2)*, and flexibility to explore patterns *(T2, T3)*.

Tools to view micro-level synteny typically depict genes as colored arrows along genomes (e.g., BactoGeNIE, [ARJ*15], clustermap.js [GC21], and more [BL24, HAA*24, HCLZ18]). Genome browsers (e.g., JBrowse [DSX*23]) are limited in the number of genomes and genes they can display. While these tools provide detailed representations, they lack overview *(T1)*, are constrained to specific features (i.e., based solely on protein similarity), or generally lack the interactivity and customization required for effective exploratory analysis *(T2, T3)*.

There are also tools for visualizing the context at both macro- and micro-levels. Genome Context Viewer [CF18] visualizes synteny using two track-views. The user can search new areas for gene context and tune the parameters for gene matching and genome alignment, but the tool lacks continuous zoom functionality, essential for navigation and contextual interpretation *(T4, T5)*. SynVisio [BG20] supports analysis from chromosome to micro-level and the user can zoom and interact with the visualization, but it differs in design objectives by focusing on conserved regions rather than gene organization *(T1, T3, T5)*. Plotsr [GS22] encodes synteny with ribbons but assumes a predetermined order of the genomes *(T3)*. JCVI [TKZ*24] also uses ribbons for synteny and is customizable but requires coding to do so and it is not interactive, reducing the effectiveness of the exploration process *(T2, T3)*.

4.2 Pangenome and Gene Set Visualization

Set operations and visualizations are often used to compare the gene content of genomes [AMA*16]. They are frequently used for pangenomes to understand what is the core set of genes (intersection) and what is variable per genome. Most visualizations support genomic researchers in exploring subset divisions of the entire gene collection. Except for Pan-Tetris [HBN15], which uses a matrix representation to explore and curate genes in homology groups, these tools mainly show the sets in abstract overviews with chord diagrams (e.g., PanViz [PNWUM17], GenomeRing [HJBN12]), or as parallel sets (e.g., GenoSets [CKG12]), providing effective overviews. Furthermore, tools often have detailed views for set attributes or metadata and interactivity. Since these tools are designed for a different analysis purpose, views and encodings for exploring organization and context are not present, and tasks *T1–T5* are not supported. Our design incorporates encodings used in synteny and gene set visualization that are adapted to allow for micro-level exploration of gene organization.

4.3 Force-Directed Layouts

Force-directed layouts aim at dynamically aligning elements based on relationships or constraints *(T3)*. Using a physical simulation of springs to find an optimal graph layout has a long history [Ead84, FR91, FLM95]. It has been paired with simulated annealing, where the temperature of the system decreases with time, decreasing the movement with it [DH96]. Schreiber et al. [SDMW09] apply this algorithm to the visualization of biological networks, including constraints that must be met. Force-directed layouts are common in biological network visualization [EBK*24], and a relevant variant is attribute-driven faceting, where an attribute restricts the node positions for example to specific areas or lines [NMSL19]. Dang et al. [DPF16] and Pham et al. [PND20] both use forces to layout comparison of timelines, fixing the x-position and letting forces act on the y-position to group connected rows.

4.4 Multiscale Visualization

Multiscale visualizations allow users to present, navigate, and relate large data across multiple abstraction levels. They are widely applied across data types and domains, combining different view configurations and interaction strategies (see Cakmak et al. [CJS*22] for a comprehensive review). There are diverse view configurations, e.g., Miao et al. [MKK*19] or Lekschas et al. [LBB*19]: overview+detail, single view focus+context, or insets. Interactions such as semantic zooming [PF93] are important for effective navigation between scales. One of the main challenges in the application of semantic zooming is the specific tailored visual design and interaction per domain and data types. For graphs (e.g., [PFH*18]), a single focus+context or inset-based approach is common, and semantic zooming is applied to reveal details in graph topological substructures (e.g., clusters or edges) or attributes (e.g., specific data properties of nodes or edges). 1D data, specifically genomic sequences (e.g., [MMP09, CF18]), primarily use juxtaposed overview+detail views, with linked selection and highlighting enabling users to drill down into specific sequence attributes. We expand this approach by combining it with semantic zoom. We apply semantic zoom to the exploration of pangenomes, specifically, gene organization analysis. This analysis depends on dynamic attribute exploration across scales, sequences, *and* genomes. The semantic zoom highlights gene organization and reveals increasing information on gene features on zoom. To our knowledge, this is not yet well-represented in existing multiscale (pan)genomics tools.

5 Flexible Multiscale Design

Here we describe the core design concepts to support our goal of gene organization exploration. The concepts are guided by data characteristics (Section 2), tasks (Section 3), our review of related work (Section 4), and discussions with our front-line analysts. Below we describe the concepts and our design philosophy.

5.1 Multiscale Design through Hierarchical Aggregation

From our task and related work analysis, we learned that our design needs to *present overview and details (C1)*, i.e., the "overview first, zoom and filter, then details-on-demand" mantra [Shn96]. To achieve this, the design should provide viewing a high-level (pan)genome-wide content and organization, while enabling detailed inspection of specific sequences of interest through interactive selection, navigation, and filtering *(T1, T2, T3)*.

To ensure seamless navigation between levels and support contextual exploration, our design incorporates *semantic zooming (C2)*, integrating aggregated and detailed information within a single view, minimizing clutter, and allowing users to drill down to deeper levels of detail through smooth interaction *(T1, T3, T4)*.

A multiscale visualization through hierarchical aggregation offers a way to achieve these objectives. This approach provides overviews while also conveying information about the underlying data, preventing information overload by simplifying and aggregating the data. We adopt the hierarchical aggregation model by Elmqvist and Fekete [EF10] to design multiscale views for exploring gene organization. The power of the model is to create scalable and less cluttered visualizations, through the use of aggregation. MULTIPLA employs aggregation on multiple levels, following the model's design guidelines. This helps keep the number of visual entities down. The entities provide visual summaries of aggregated data, and visual simplicity is maintained. Importantly, in genomics data, which is inherently multiscale, we can directly utilize the existing tree structure (e.g., genes, mRNAs, exons) as the basis for hierarchical aggregation. The aggregation is described in Section 6.

5.2 Flexibility versus Automation

To understand the contributions of features to organizations and reveal patterns from different feature combinations *(T3)*, the design needs to facilitate interactive *feature-based reordering (C3)* of the sequences carrying the genes (Fig. 3). Furthermore, the design should not only support vertical reordering of sequences but also provide flexible alignment options (Fig. 3). These options should enable users to *flexibly align sequences (C4)* according to their analysis needs, ensuring the most relevant information is visible in a single view. This is particularly important because genes may be located far apart across sequences and gene order is of higher relevance to organization analysis than gene position *(T3, T5)*.

Together, these concepts highlight the need to integrate manual and automated methods to balance efficiency and user control for effective exploration. Our target data is error-prone, and the feature importances are not known in advance. This requires that visualizations and layouts are steered by users' domain knowledge. At the same time, the size of data—specifically the number of sequences and genes to compare—calls for automated layout schemes to provide a starting point for exploration and ensure scalability. The key idea is to enable an interplay between automation and flexibility, recognizing that automated layouts serve as a reasonable data-driven approximation for domain experts to identify patterns and incorporate their knowledge.

We apply this combination of automation and flexibility in all views of our design: in the overviews, we use a projection of features to reveal high-level **structures**, while in the Focus View design, we enable customizable **ordering** and **alignment** automation to uncover lower-level feature organization patterns (Fig. 3).

6 MULTIPLA

This section introduces the views implemented in MULTIPLA, an interactive design for exploring gene organization through **MULTI**scale **P**angenomic **L**ocus **A**nalysis. The design consists of three views (UMAP, Grid Overview, and Focus View) and a collapsible control panel shown in Fig. 1. The three views are linked such that the sequence selection is consistent in all views. Moreover, on hovering over a sequence, the corresponding sequence glyph or label is highlighted in the other views, helping the user switch between the views. Tooltips offer on-demand details for lower-level features, alternative identifiers, or names. A demo is available here: http://vapp1.win.tue.nl/multipla. Below we describe the design details for each view.

6.1 UMAP Overview

The UMAP Overview (Fig. 1.1) functions as an overview of the data and an interface for selecting sequences to analyze in the Focus View *(T1, T2)*. Each sequence is represented by a glyph shaped like a dot, indicating selection with a change in color. Lasso selection allows the user to rapidly select sequences for the Focus View, with CTRL-key interaction to add individuals. The view supports panning and semantic zooming with three levels of detail. In the first, most zoomed-out level, the glyph is a simple dot. At the second level, the border width of the dot encodes the count of genes on that sequence (in the locus). At the third level, the sequences that share a homology group are connected with ribbons (edges), which are highlighted on hover. The width of the edge encodes the number of shared homology groups.

The position of each glyph is determined via a UMAP [MHM20] projection of sequence feature vectors. The UMAP could be based on any combination of features of the sequences. We selected protein distance (obtained from the homology calculations) as a starting point based on collaborator recommendations. We selected UMAP over other dimensionality reduction layouts such as t-SNE because it tends to better capture global structures [CVW23], which

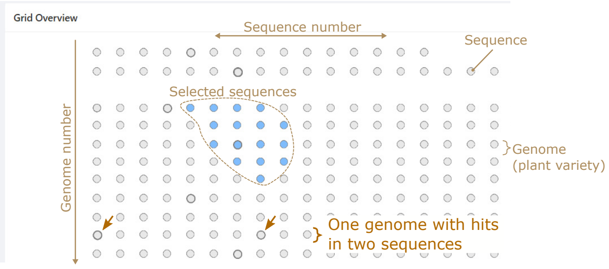


Figure 4: *Organization of the Grid Overview. Each sequence is represented by a dot, and the dots are arranged in rows based on which genome (plant variety) they belong to. Sequences are sorted within the rows based on their sequence number. Selection is indicated with blue and sequences with query hits have a thicker outline.*

are important to overview and selection tasks *(T1, T2)*. By clustering similar sequences together, the user can detect structures in the data. Through the semantic zoom, they can further investigate the sequences before making a selection and moving to the Focus View.

6.2 Grid Overview

The Grid Overview (Fig. 1.2) presents identical data to the UMAP Overview, using the same glyph with another layout scheme. It also features the same interactive functionalities as the UMAP Overview. The Grid Overview visualizes the sequences grouped in rows, where the groups are based on some sequence characteristic, creating a structured view of the data, Fig. 4. The default grouping characteristic is which genome the sequence belongs to. Units within the group are ordered by their sequence number, which is often analogous to their chromosome number.

The Grid Overview complements the UMAP by enabling another perspective of the data, but it aims to support the same tasks *(T1, T2)*. The grid structures the sequence glyphs as a discrete segregated layout [MMP09], allowing the user to select and spot characteristics within one genome (e.g. for phased sequences), whereas the UMAP would resemble an interleaved segregated layout, helping to spot differences between sequences regardless of their sequence labels, which is useful for error-prone annotation data.

6.3 Focus View

In the Focus View (Fig. 1.3), the user can inspect selected sequences at a more zoomed-in level compared to the overviews. Here the organization and different features of each sequence are visible, with flexible ordering and alignment for dynamic rearrangement, and semantic zooming to prevent clutter and overload.

6.3.1 Sequence ordering

The order of sequences (rows) of the Focus View plays a crucial role in determining which organization patterns can be spotted. In the Focus View, this order is determined by a customizable distance. Initially, the sequence order is determined by clustering based on protein similarity. However, a key feature of MULTIPLA is that the clustering algorithm includes flexible distance definitions for changing the order to explore different feature patterns *(T3)*.

There is no single feature or distance that completely expresses

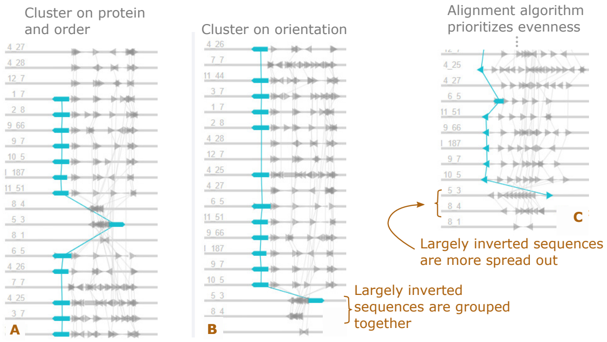

Cluster on protein and order
Cluster on orientation
Alignment algorithm prioritizes evenness
A
B
C
Largely inverted sequences are grouped together
Largely inverted sequences are more spread out

Figure 5: *The same data ordered based on different cluster algorithm settings: protein and order (A) and orientation (B). It also displays the result of rerunning the spring simulation with higher weighting for evenness (gravity and repulsion) (C).*

a gene organization. Depending on the analysis task, some features may be more important than others, and their contributions can be explored through flexible use of the distances. In the Control Panel (Fig. 1.4), users can adjust the weights for the clustering parameters. As a result, the clustering updates, leading to a change in the sequence order. This supports rapid iteration to find insightful orderings and to consider different aspects of the data. For example, a user suspecting that CNV is important for their analysis can prioritize this feature by increasing its weight and reducing others.

We can compute distances $D_*(A,B)$ between sequences A and B based on gene set content (PAV and CNV) and their distinct gene features (order, orientation, protein distance, location, and size). We present the combined distance $D(A,B)$ as the weighted sum of these distances, all normalized to an interval $[0,1]$, with weighted factors w_*. The distance $D_*(A,B)$ with $*$ one of $\{r,p\}$ is given as the sum of distances d_* of the mRNA sets corresponding to the gene sets $g(A)$ and $g(B)$ restricted to the different homology groups h. The features were obtained through discussion with collaborators and review of related work [MMP09, ARJ*15, GC21, ZES19]. We implemented five distances as a proof of concept: PAV, CNV, order, protein distance, and orientation. We shortly present them here; formulas and details per feature are available in the Supplement.

PAV and CNV The genes' PAV based distance D_v is based on the Jaccard dissimilarity J of the sets of genes $g(A)$ and $g(B)$ as they appear in A and B, respectively. The CNV based distance D_c is the same but using *multisets* (sets allowing duplicates) of genes.
Order Distance D_o reflects gene order, using the Levenshtein [Lev65] distance on the sequence of genes on A and B.
Orientation Distance $d_r = J(signs(a), signs(b))$ also uses Jaccard dissimilarity of two multisets where $signs(x)$ contains the orientations of mRNAs in a set x.
Protein The distance d_p is based on protein sequence similarity scores s between mRNAs, obtained from PanTools [SAdRSS18].

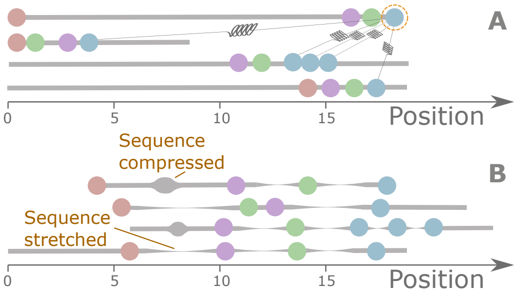


Figure 6: *Spring system concept. Each node is connected to all other nodes of the same type with springs, as the circled node in (A). Springs only operate in the x-direction but are drawn in 2D for illustrative purposes. Neighboring nodes are also connected. When the simulation ends the nodes are realigned so that nodes of the same type stack up (B), as the purple nodes. Stretching and compressions are marked by thinner and bumped lines respectively.*

The distances D_* resulting from these five user-adjustable features are used for clustering and ordering of the sequences. Users can interactively adjust their weights, w_*, depending on their importance at each point during the exploration, allowing for flexibility and accounting for changing perspectives that a fixed weighting scheme cannot offer. The use of weighted distances has been used successfully in existing tools for gene sets [GC21, BSK*21]. However, they only use protein similarity and order, and little to no interactive adjustment of the distance components is possible. We use hierarchical clustering with Ward linkage by default, as it has been shown to be adequate in other work [GC21], but we also provide a choice among single, average, and complete linkage [DHS01]. Fig. 5.A and B show example sortings based on different features.

6.3.2 Design considerations for alignment

A fundamental issue in sequence comparison is alignment. Genomics data in particular is sparse, consisting of long featureless stretches, with densely populated intervals in between, where exact positions have no direct meaning. Different orders of genes and copy number variations add to the alignment complexity. The prerequisites constrain the design space as follows.

No reference sequence Genomics analysis typically employs a reference sequence, to which all other sequences are aligned. Pangenomes avoid using a single reference to reduce biasing the analysis, but as a consequence, this complicates how positions can be aligned or compared.
No joint coordinate system The data structure defines position as the distance from the sequence start, but this is not useful for comparing the gene organization between sequences, as shown in Fig. 6.A. Neighboring genes might appear with vastly different separations in two sequences, meaning that all the subsequent genes are offset. Therefore, a joint coordinate system preserving relative positions is not appropriate.
No homology anchoring In initial designs, we explored anchoring based on one homology group. However, this is not possible for sequences where the homology group is absent and becomes complicated when the group has several instances in a sequence.
Preserved gene order Domain experts indicated that preserving gene order within each sequence is a hard constraint, order should not be altered. In contrast, distances between genes are less relevant. It matters if there is a relatively large or small distance, but the exact distance is of no relevance.
Repeated rearrangement Our collaborators described their work

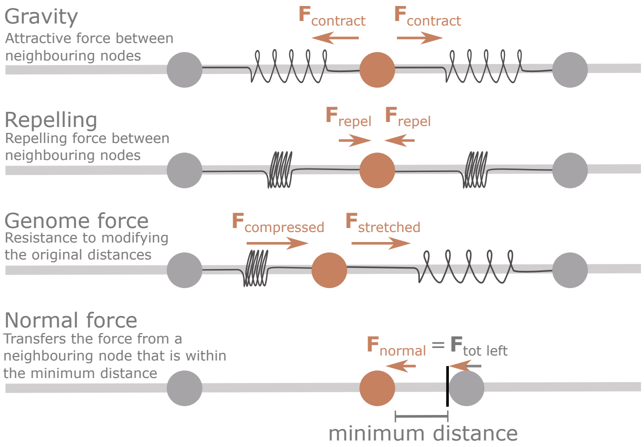

Gravity
Attractive force between neighbouring nodes
$F_{contract}$
$F_{contract}$
Repelling
Repelling force between neighbouring nodes
F_{repel}
F_{repel}
Genome force
Resistance to modifying the original distances
$F_{compressed}$
$F_{stretched}$
Normal force
Transfers the force from a neighbouring node that is within the minimum distance
$F_{normal} = F_{tot\ left}$
minimum distance

Figure 7: *The types of forces on a node applied by the nodes on the same sequence. The gravity force pulling neighboring nodes together, the repelling force pushing them apart. The genome force works to maintain the sequence length, and the normal force applies when a neighboring node is within the minimum distance.*

not as finding a single optimal arrangement, but rather as repeated rearrangements adapting to how the analysis evolves.

6.3.3 Spring simulation system

We designed a layout algorithm adapted to the alignment design constraints above. The algorithm preserves and aligns genes of the same homology group. By compressing the scale in-between genes separately for each sequence, comparison of gene order in sequences of vastly different lengths is made possible. Distance and position are prioritized lower according to the **preserved gene order** constraint, however, compression and stretching of empty intervals is indicated by a bulge or thinning in the connecting line, as illustrated in Fig. 6.B. The effect is more subtle in MULTIPLA, to avoid interference with visual pattern detection.

With respect to the repeated rearrangement process, we sought to create an automation that captured the trade-offs of the manual process well enough to satisfy the domain expert. We modeled the manual process by creating a system of springs and nodes, similar to force-directed layouts but operating only in one dimension and with a specific set of restrictions. Each factor affecting the alignment is represented in the system by one or two types of springs, allowing the system to consider all factors simultaneously. The user can manually change the settings for the springs, to tune the impact of each factor on the resulting alignment.

The spring simulation architecture is as follows. Firstly, genes are transformed into nodes, where overlapping genes from the same sequence form one node. Each node is connected to its immediate neighbors in the sequence, and to all nodes with the same homology group. The connections apply forces on the node, resulting in a position shift when the node is updated. Homology group connections pull related nodes together, as illustrated in Fig. 6. The forces are further specified in the Supplement. The algorithm iterates over states of the spring system, updating one node for each iteration. The global heat decreases with time and limits the step size, as recommended by Davidson and Harel [DH96], and each node has a local heat to decrease oscillations [FLM95]. For a random selection of nodes, a step is calculated based on the force and the heat. The proposed step is constrained to fulfill minimum distance and order conditions. The node with the largest step is then updated.

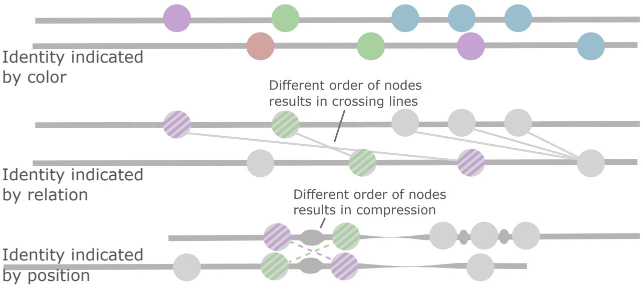

Identity indicated by color
Different order of nodes results in crossing lines
Identity indicated by relation
Different order of nodes results in compression
Identity indicated by position

Figure 8: *Different strategies for comparing sequences. Strategy 1 (top) colors each element according to its type. Strategy 2 (middle) places connecting lines between elements of the same element type. Strategy 3 (bottom) does not encode the element type directly. Elements of the same type are grouped together by shifting and stretching the axes. Reversed node order results in compression.*

Different termination criteria can be applied, and in our case, it is either the heat, time elapsed, or the size of the constrained step. Termination by time elapsed ensures termination and limits the runtime. Termination can occur while the system still moves a lot and no minimum has been found. However, finding a minimum is not required since the purpose is to re-layout the visualization.

6.3.4 Design considerations for identity coding

To compare the organization of sequences, the analyst must understand which genes are present and in what order they appear. There are multiple ways to encode this information, as shown in Fig. 8. The homology group of each gene can be explicitly encoded, for example by color, so the user can compare the sequences manually (**Strategy 1**). Alternatively, the order of genes relative to another sequence can be directly encoded, for example with ribbons (**Strategy 2**). In the absence of explicit encodings, the user can try to deduce the most likely homology group of genes by their aligned position (**Strategy 3**), see Section 6.3.2.

The above strategies have benefits and drawbacks, making them suitable for different situations. Strategy 1 allows the most versatile comparisons for the user but does not scale well. Strategy 2 highlights order changes and scales better than Strategy 1, but is prone to clutter and is mainly useful for pairwise comparisons. Strategy 3 scales better than the other two but does not allow the user to see exactly which genes have the same homology group or capture order variations. The application of the strategies in MULTIPLA is described in the Semantic Zoom section below.

6.3.5 Zooming and interactions

The challenges of genomics data are largely due to the multiscale nature of the data. MULTIPLA uses semantic zooming *(C2)*, allowing analysts to seamlessly move from overview to detail *(C1)*, leveraging the strengths of each identity coding strategy above.

Semantic Zoom The Focus View features semantic zooming in three layers. The spring system layout algorithm functions as a first layer by reducing unnecessary whitespace while maintaining a glanceable arrangement. The identity coding of genes constitutes the second layer of semantic zooming, with two seamless levels. **Strategy 1** is used to encode identity at the most zoomed-in level. When the number of homology groups exceeds a certain threshold, **Strategy 2** is used instead. At this zoom level, genes are all colored gray (see Strategy 2 Fig. 8), since nominal color schemes have a limited number of values. Changes of order are visualized by lines connecting genes of the same homology group. Connections are only displayed for homology groups that appear in different orders between sequences. This helps minimize clutter and highlights areas with disarray *(T3, T4)*. The spring system also indicates node identity, according to **Strategy 3**.

The third layer of semantic zooming is the glyph representation of genes. To maximize information transfer and minimize clutter, genes have two levels of semantic zoom. The gene is represented by a bar with an arrow indicating its orientation, according to common practice in the domain. The length of the gene is encoded by the bar length, and presence of multiple RNAs is indicated by dots below the bar. When the length of the gene in the current viewport goes below a certain threshold, it is represented by a triangle of a fixed size, oriented according to the gene's orientation.

Hover To complement the above-described strategies for encoding identity, we implemented a highlight on hover. As a gene is highlighted, all genes of the same type are marked with higher opacity and the gene's color. All other elements are given a lower opacity during hover. In addition, there is a tooltip with the gene name and homology group unique index, and hovering over a sequence provides an indication of the genome position at that hover location.

Brushing To reduce clutter, the user can use brushing to select which homology groups' links should be highlighted. The interaction is similar to how brushing works for parallel coordinates.

6.4 Control Panel

The Control Panel (Fig. 1.4) allows the user to overview the data and change the visualization settings. Sliders are used to change the weights of each feature in the clustering algorithm, controlling the order of sequences. The spring simulation is tuned with four sliders: for the genome force, group force, minimum distance, and a merged measure for the gravity and repulsive forces.

7 Evaluation

We evaluate MULTIPLA's design through two use-case-based evaluations. For each use case, the setup involved two experimenters—one with visualization expertise and one with bioinformatics knowledge—along with one participant. The participants were pangenomics experts at the PhD or postdoctoral level, were not involved in the design process, and engaged in thinking aloud. We here report the insights obtained for each use case.

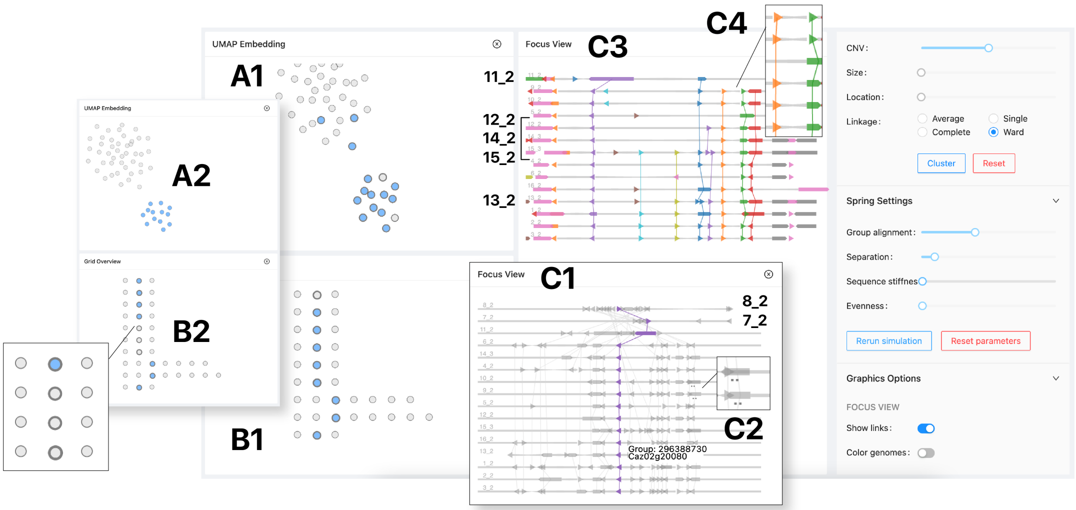
7.1 Datasets and Format

We use PanTools [JvWSA*22] to obtain gene features and genome sequence information for predetermined loci. We illustrate our work with two real-world datasets of 16 pepper and 12 rose genome assemblies. Each dataset consists of 50–2000 sequences, with 15–20 loci, containing 200–220 genes/mRNAs, and 15–60 homology groups. The gene organization in each locus is queried from the graph pangenome and can be accessed as a JSON file, similar to Clinker [GC21]. Our data is hierarchical and follows the AGAT and Sequence Ontology conventions to describe parent-child relationships, such as genes (level 1) containing mRNAs (level 2), and mRNAs containing exons (level 3). It includes: genomes, sequences, loci, genes, mRNAs, exons, non-coding RNAs, repeats, functional domains, homology groups, and homology links.

7.2 Capsicum *Pun1* Locus

Pungency is a trait in (chili) peppers (*Capsicum* species), which is determined by capsaicinoid production. It is controlled by a locus, containing one or more *Pun1*-like genes [SJKL*05, SMS*07]. The study participant wanted to investigate variation (in gene sizes, order, and copy number) in this locus in a pangenome of 16 assemblies of domesticated and wild *Capsicum* species. All but two assemblies are at the chromosome level. Using the *Pun1* locus (171,671 bp) on chromosome 2 (chr2) of the CM334 assembly [K*14], they found matches on 16 out of 57 sequences *(T1)*.

The exploration started with getting an overview of gene content *(T1)* and sequence selection *(T2)*. The participant used the zoomed-in view in the Grid to confirm hits on chr2 (sequence 2) but also found some on sequence 3 (Fig. 9.B1, B2, thick border). Selecting the small cluster from the UMAP shows that sequence 2 of genomes 11, 12, and 13 are not included (Fig. 9.A2, B2). They used the Grid Overview again to add these sequences with matching homologs to the selection. In the Focus View, genomes 7 and 8 stood out because of density and crossing lines, and highlighting a few genes through mouse over showed that all genes in these sequences were in opposite orientations compared to the other genomes, complicating analysis (Fig. 9.C1). Both were removed from the selection. As shown in Fig. 9.C2, another notable observation is that some genes encode two mRNAs *(T1)*. As most genomes are from domesticated *C. annuum* species, except 11, 12, 14, 15 (and the excluded 7 and 8), the participant wanted to investigate if these could be grouped. Grouping on a single distance (proteins, order, CNV) did not place all four genomes together. The user decided to settle for a mixture of these three distances that gave a group of genomes 12, 14, and 15, and genome 11 apart, likely because it misses some genes in the selected locus (Fig. 9.C3). Finally, interested in CNV in *Pun1*-like genes *(T3)*, the participant tested various spring layout settings to align genes of interest in the Focus View *(T4)*. Increased Group alignment and Separation helped to get a clear view. In the Focus View, groups ending in 730-737 (known to be *Pun1*-like genes) were brush-selected to trace them during zooming and panning. The subtle stretching in the intergenic regions (Fig. 9.C4) helped to get a sense of the distance between genes *(T4)*. The color-coded and connected homology groups helped to trace patterns in gene order and size. Genes in group 730 (purple) were present in all genomes, but the gene in sequence 11_2 deviates in size (Fig. 9.C3), likely due to an annotation error, requiring inspection with expression data *(T5)*. The other groups show PAV and tandem duplicates (e.g., two blue genes on 13_2). In conclusion, the ex-



CNV:
Size:
Location:
Linkage: ○ Average ○ Single ○ Complete ● Ward

Cluster | Reset

Spring Settings

Group alignment:
Separation:
Sequence stiffnes
Evenness:

Rerun simulation | Reset parameters

Graphics Options

FOCUS VIEW

Show links: ●
Color genomes: ○

Figure 9: *Exploration of the* Capsicum Pun1 *locus using* MULTIPLA. *(A1–2) UMAP reveals two groups. (B1–2) Grid Overview allows sequence selection. Focus View shows locus organization: (C1) Initial view shows crossing links and reverted sequences; (C2) Observation of two mRNAs; (C3) Size deviation of group 730 and PAV of others, visible via spring layout and brushing; (C4) sequence stretching.*

ploration showed that there is significant copy number variation in *Pun1*-like genes, while gene order is conserved.

7.3 Rose *Myb114* Locus

Roses are highly valued for their beautiful flowers and attractive scent. Anthocyanin is the primary pigment that determines flower color. The biosynthesis of anthocyanin is regulated by a complex network of genes, including *Myb114*, a transcription factor that influences the expression of other genes [YMA*17]. In *Rosa chinensis*, it was found that a 148-bp insertion in the regulatory region of the *Myb114* gene upregulated the expression of the gene, causing red petal coloration [LZY*22]. This sparked the participant to investigate the variation in the *Myb114* locus across a *Rosa* pangenome with 12 assemblies (4 of which are phased) from 7 different species. The data set spans 2011 sequences of which 19 contain a locus similar to the query. The participant was interested in PAV between and within genomes, gene order, and size variation.

After hiding the empty sequences, the Grid and UMAP Overviews showed all sequences on which a locus was identified. Since the query locus was on genome 3 (*R. chinensis*), the participant first explored highly similar assemblies from the same rose variety *(T1, T2)* using the lasso selection in the Grid Overview (Fig. 10.A1). In the Focus View, they could identify the *Myb114* homology group with the tooltip and they used the brush interaction to turn on the links as a visual reference point (Fig. 10.A2). They then included the other sequences. Because of significant length differences both in sequences and loci, the participant tuned the spring layout to obtain a better alignment of groups, which showed the conservation more clearly *(T3, T4)*. By increasing group alignment and brushing, the user observed that *Myb114* homologs were not present in all genomes. Notably, the gene was found only on one chromosomal copy (4_25) in the phased tetraploid assembly of *R. hybrida* (genome 4) [ZYL*24] (Fig. 10.B), indicating significant intragenomic variation. Comparing the sequences from genome 4 to those from 3 (*R. chinensis*) (Fig. 10.C), the semantic zoom *(T1, T4)* in the UMAP Overview (Fig. 10.D) shows that haplotype A (4_25) is more similar to sequence 3_7 than haplotype B (4_26) in this locus, as expected [ZYL*24]. This use case demonstrated the usefulness of MULTIPLA's views for exploring inter and intragenomic variation.

8 Discussion

The properties of our data present challenges such as vast differences in scale and a large spread of the data characteristics of interest depending on the organism. Reflecting on our design process, we see that these challenges are met by design choices with minimal assumptions about the data structure while centering the tasks. Particularly, MULTIPLA differs from previous work by visualizing data from *multiple genomes* at *multiple levels*. Our case studies showed that multiscale views through semantic zooming have strong potential. The novel visualization entails, however, a learning curve. Users expressed varied initial understanding and needed time to adapt to this interaction. In contrast, UMAP, which has previously been applied to genomic analysis, was intuitive for users. Another lesson learned through use case exploration is that additional space-filling layouts, such as matrices, could complement the existing unit visualizations and UMAP embeddings, offering perspectives beyond the groupings in the data.

Interactivity and flexibility are key aspects for effective visual exploration of synteny, which is reflected in feedback from the user study: "[The interactivity] really helped me to analyze this locus. Freedom of selection, zooming, and dynamic changing [of alignment and order] led to a different understanding." MULTIPLA features several interaction techniques for synteny exploration. The two overviews support additional perspectives on the data as compared to exclusively track-based visualizations such as the R-package gggenomes [HAA*24]. MULTIPLA extends the convention of using color and links for visualizing synteny by utilizing color, links, *and position* on different zoom levels to create semantic zooming. The brushing and on-hover highlighting of the links is

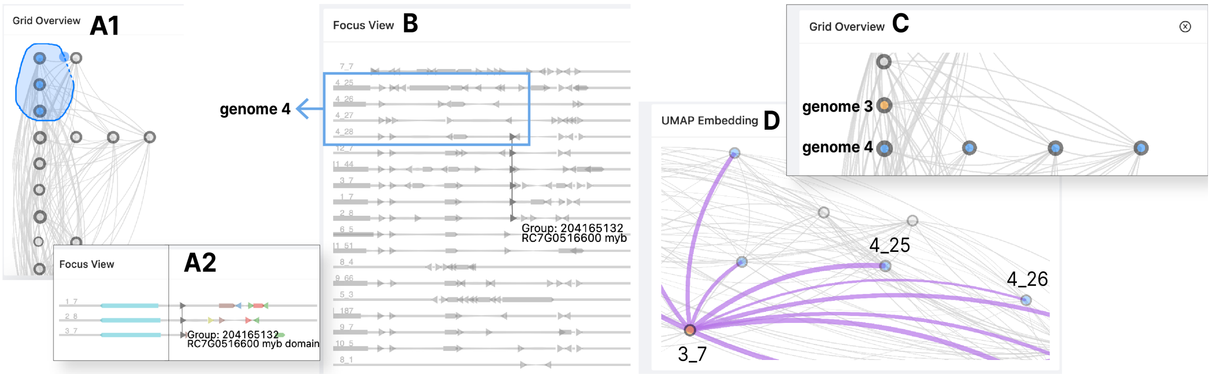


Figure 10: *Exploration of the rose* Myb114 *locus using* MULTIPLA. *(A1) Lasso selection in the Grid Overview. (A2) Focus View brushing to highlight links. (B) Spring layout reveals* Myb114 *presence on only one chromosomal copy (4_25) in the phased tetraploid genome. (C,D) Semantic Zoom in UMAP and Grid Overviews for analyzing inter- and intragenomic variation.*

inspired by interaction patterns of parallel coordinates [JF16] and makes it feasible to render links also to non-adjacent sequences which causes excessive clutter in current visualizations.

For data with patterns at many scales such as genome data, special attention must be given to navigating across those scales. Our evaluation shows that MULTIPLA supports analysis on macro- and micro-level well by providing an overview first and details on demand [Shn96] through continuous semantic zoom. MULTIPLA's zooming functionality is similar to that in SynVisio [BG20], but the focus on genes and homology groups, i.e., sparse hierarchical features, requires more complicated semantic zooming and different alignment algorithms. Our system addresses the challenges of this datatype through customizable alignment and sequence ordering. While JCVI [TKZ*24] allows much customizability through coding, MULTIPLA supports tuning of key parameters in the interface, making it accessible to less code-savvy users and suitable for rapid iteration as shown in the evaluation. This is key since MULTIPLA explicitly targets exploration of different orderings, and thus requires quickly reordering and realigning sequences based on customized features. Our evaluation demonstrates the usefulness of this flexibility. It may also apply to other data, e.g., event sequences.

In the evaluation, the force-directed layout proved to fulfill the intended purpose. Users tweaked the forces to achieve alignment and separation to assist the exploration and create an organized overview. Our solution is of the type attribute-driven faceting with additional restrictions. Previous work used forces to determine the ordering of rows with a common x-axis [DPF16, PND20]. In MULTIPLA, row order is instead handled through clustering, and the spring system is employed in the x-direction. Thus we introduce the novel concept of individual, non-linear x-axes for each sequence. This prioritizes alignment over distance visualization, motivated by a corresponding prioritization in the data analysis tasks. Though this could lead to misinterpretation of the data in some scenarios, whether axis manipulations are misleading or not is task dependent [LK24]. Moreover, we observed examples of data hunches [LAML23] when our collaborators knew the data to be flawed and performed manual rescaling to compare sequences of different sizes. In the capsicum case, the spring system performed this rescaling and the user noted it through the bulges in the segment. Thus, the spring system constitutes a novel visualization technique for sequence data, centering comparison of element order while maintaining a notion of distance.

There are several opportunities for future work to address the current limitations. The design was developed and evaluated with partly the same set of end users, and it would be valuable to investigate how the tool is received outside of this group. Due to limited time and resources, we prioritized the most important functionality in the tool, but qualitative feedback from the use cases highlighted some relevant feature requests. For example, allowing users to click and drag elements to reorder them interactively, centering the alignment on a homology group when clicked, and enhancing zooming in overviews with a minimap to maintain context could be valuable additions. Extension of the complexity of the hierarchical data would also be of interest, e.g., showing locus boundaries, more advanced strategies for visualizing a gene encoding multiple mRNAs, and enabling zoom levels beyond gene length (e.g., intron-exon structure). Finally, improving the performance of the spring simulation could yield more stable results faster. This would increase the ability to recreate layouts since the algorithm would terminate by minima rather than by time elapsed. In general, further refining the automation algorithms and adding the suggested features has the potential to improve the tool.

9 Conclusion

We presented a design and implementation of MULTIPLA, a visual interface for exploring pangenomic gene set organizations. Exploring patterns and deviations in a set is important for uncovering annotation errors and biological variants. However, analysis of pangenomes requires visualizations that support flexible sorting and alignment functionality as well as interactivity. Our approach enables users to explore the organization and the role of different features by providing interactive clustering and encoding options facilitating comparison. Through two case studies, we found that MULTIPLA supports our listed tasks and design goals.

Acknowlegdements

We thank C. Papastolopoulou, Z. Yang, E. M. Jonkheer, M. Yang, and F. de Vries for their support. This research has partly been funded by the Swedish Foundation for Strategic Research, grant ID20-0092.

References

[AMA*16] Alsallakh B., Micallef L., Aigner W., Hauser H., Miksch S., Rodgers P.: The State-of-the-Art of Set Visualization. *Computer Graphics Forum 35*, 1 (2016), 234–260. doi:10.1111/cgf.12722. 4

[ARJ*15] Aurisano J., Reda K., Johnson A., Marai E. G., Leigh J.: BactoGeNIE: A large-scale comparative genome visualization for big displays. *BMC Bioinformatics 16*, 11 (2015), S6. doi:10.1186/1471-2105-16-S11-S6. 3, 6

[BG20] Bandi V., Gutwin C.: Interactive exploration of genomic conservation. In *Interactive Exploration of Genomic Conservation* (2020). URL: https://openreview.net/forum?id=7-C5VJWbnI. 3, 10

[BL24] Branger M., Leclercq S. O.: GenoFig: a user-friendly application for the visualization and comparison of genomic regions. *Bioinformatics 40*, 6 (2024), btae372. doi:10.1093/bioinformatics/btae372. 3

[BSK*21] Blin K., Shaw S., Kloosterman A. M., Charlop-Powers Z., Van Wezel G. P., Medema M. H., Weber T.: AntiSMASH 6.0: Improving cluster detection and comparison capabilities. *Nucleic Acids Research 49*, W1 (2021), W29–W35. doi:10.1093/nar/gkab335. 6

[CF18] Cleary A., Farmer A.: Genome context viewer: visual exploration of multiple annotated genomes using microsynteny. *Bioinformatics 34*, 9 (2018), 1562–1564. doi:10.1093/bioinformatics/btx757. 3, 4

[CJS*22] Cakmak E., Jackle D., Schreck T., Keim D. A., Fuchs J.: Multiscale Visualization: A Structured Literature Analysis. *IEEE Transactions on Visualization and Computer Graphics 28*, 12 (2022), 4918–4929. doi:10.1109/TVCG.2021.3109387. 4

[CKG12] Cain A. A., Kosara R., Gibas C. J.: GenoSets: Visual Analytic Methods for Comparative Genomics. *PLoS ONE 7*, 10 (2012), 1–9. doi:10.1371/journal.pone.0046401. 4

[CVW23] Collaris D., Van Wijk J. J.: StrategyAtlas: Strategy Analysis for Machine Learning Interpretability. *IEEE Transactions on Visualization and Computer Graphics 29*, 6 (June 2023), 2996–3008. doi:10.1109/TVCG.2022.3146806. 5

[DH96] Davidson R., Harel D.: Drawing graphs nicely using simulated annealing. *ACM Trans. Graph. 15*, 4 (1996), 301–331. doi:10.1145/234535.234538. 4, 7

[DHS01] Duda R. O., Hart P. E., Stork D. G.: *Pattern Classification*, 2 ed. Wiley, New York, 2001. doi:10.1007/s00357-007-0015-9. 6

[DPF16] Dang T. N., Pendar N., Forbes A. G.: TimeArcs: Visualizing fluctuations in dynamic networks. *Computer Graphics Forum 35*, 3 (2016), 61–69. doi:10.1111/cgf.12882. 4, 10

[DSX*23] Diesh C., Stevens G. J., Xie P., De Jesus Martinez T., Hershberg E. A., Leung A., Guo E., Dider S., Zhang J., Bridge C., Hogue G., Duncan A., Morgan M., Flores T., Bimber B. N., Haw R., Cain S., Buels R. M., Stein L. D., Holmes I. H.: JBrowse 2: a modular genome browser with views of synteny and structural variation. *Genome Biology 24*, 1 (Apr. 2023), 74. doi:10.1186/s13059-023-02914-z. 3

[Ead84] Eades P.: A heuristic for graph drawing. *Congressus Numerantium. 42* (1984), 149–160. 4

[EBK*24] Ehlers H., Brich N., Krone M., Nöllenburg M., Yu J., Natsukawa H., Yuan X., Wu H.-Y.: An introduction to and survey of biological network visualization. *Computers & Graphics* (2024), 104115. doi:10.1016/j.cag.2024.104115. 4

[EF10] Elmqvist N., Fekete J.-D.: Hierarchical Aggregation for Information Visualization: Overview, Techniques, and Design Guidelines. *IEEE Transactions on Visualization and Computer Graphics 16*, 3 (May 2010), 439–454. doi:10.1109/TVCG.2009.84. 4

[FLM95] Frick A., Ludwig A., Mehldau H.: A fast adaptive layout algorithm for undirected graphs (extended abstract and system demonstration). In *Graph Drawing* (1995), Tamassia R., Tollis I. G., (Eds.), Springer, pp. 388–403. doi:10.1007/3-540-58950-3_393. 4, 7

[FR91] Fruchterman T. M. J., Reingold E. M.: Graph drawing by force-directed placement. *Software: Practice and Experience 21*, 11 (1991), 1129–1164. doi:10.1002/spe.4380211102. 4

[GC21] Gilchrist C. L., Chooi Y. H.: Clinker & clustermap.js: Automatic generation of gene cluster comparison figures. *Bioinformatics 37*, 16 (2021), 2473–2475. doi:10.1093/bioinformatics/btab007. 3, 6, 8

[GS22] Goel M., Schneeberger K.: plotsr: visualizing structural similarities and rearrangements between multiple genomes. *Bioinformatics 38*, 10 (2022), 2922–2926. doi:10.1093/bioinformatics/btac196. 3

[HAA*24] Hackl T., Ankenbrand M., Adrichem B. v., Wilkins D., Haslinger K.: gggenomes: effective and versatile visualizations for comparative genomics, 2024. arXiv:2411.13556, doi:10.48550/arXiv.2411.13556. 3, 9

[HBN15] Hennig A., Bernhardt J., Nieselt K.: Pan-Tetris: An interactive visualisation for Pan-genomes. *BMC Bioinformatics 16*, 11 (8 2015), 1–11. doi:10.1186/1471-2105-16-S11-S3. 4

[HCLZ18] Harrison K. J., Crécy-Lagard V. d., Zallot R.: Gene graphics: a genomic neighborhood data visualization web application. *Bioinformatics 34*, 8 (2018), 1406–1408. doi:10.1093/bioinformatics/btx793. 3

[HJBN12] Herbig A., Jäger G., Battke F., Nieselt K.: GenomeRing: Alignment visualization based on SuperGenome coordinates. *Bioinformatics 28*, 12 (2012), 7–15. doi:10.1093/bioinformatics/bts217. 4

[HYJ*23] He W., Yang J., Jing Y., Xu L., Yu K., Fang X.: NGenomeSyn: an easy-to-use and flexible tool for publication-ready visualization of syntenic relationships across multiple genomes. *Bioinformatics 39*, March (2023), 10–12. doi:10.1093/bioinformatics/btad121. 3

[JF16] Johansson J., Forsell C.: Evaluation of parallel coordinates: Overview, categorization and guidelines for future research. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 579–588. doi:10.1109/TVCG.2015.2466992. 10

[JvWSA*22] Jonkheer E. M., van Workum D.-J. M., Sheikhizadeh Anari S., Brankovics B., de Haan J. R., Berke L., van der Lee T. A. J., de Ridder D., Smit S.: PanTools v3: functional annotation, classification and phylogenomics. *Bioinformatics 38*, 18 (9 2022), 4403–4405. doi:10.1093/bioinformatics/btac506. 8

[K*14] Kim S., et al.: Genome sequence of the hot pepper provides insights into the evolution of pungency in Capsicum species. *Nature Genetics 46*, 3 (Mar. 2014), 270–278. doi:10.1038/ng.2877. 8

[KSB*09] Krzywinski M., Schein J., Birol I., Connors J., Gascoyne R., Horsman D., Jones S. J., Marra M. A.: Circos: an information aesthetic for comparative genomics. *Genome Research 19*, 9 (2009), 1639–1645. doi:10.1101/gr.092759.109. 3

[LAML23] Lin H., Akbaba D., Meyer M., Lex A.: Data hunches: Incorporating personal knowledge into visualizations. *IEEE Transactions on Visualization and Computer Graphics 29*, 1 (2023), 504–514. doi:10.1109/TVCG.2022.3209451. 10

[LBB*19] Lekschas F., Behrisch M., Bach B., Kerpedjiev P., Gehlenborg N., Pfister H.: Pattern-Driven Navigation in 2D Multiscale Visualizations with Scalable Insets. *IEEE Transactions on Visualization and Computer Graphics* (2019), 611–621. doi:10.1109/TVCG.2019.2934555. 4

[Lev65] Levenshtein V. I.: Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics. Doklady 10* (1965), 707–710. 6

[LK24] Long S., Kay M.: To cut or not to cut? a systematic exploration of y-axis truncation. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (2024), CHI '24, Association for Computing Machinery, pp. 1–12. doi:10.1145/3613904.3642102. 10

[LSH*02] Lewis S. E., Searle S. M., Harris N., Gibson M., Lyer V., Richter J., Wiel C., Bayraktaroglir L., Birney E., Crosby M. A., Kaminker J. S., Matthews B. B., Prochnik S. E., Smithy C. D., Tupy J. L., Rubin G. M., Misra S., Mungall C. J., Clamp M. E.: Apollo: a sequence annotation editor. *Genome biology 3*, 12 (2002), 1–14. doi:10.1186/gb-2002-3-12-research0082. 3

[LSS*22] Lovell J. T., Sreedasyam A., Schranz M. E., Wilson M., Carlson J. W., Harkess A., Emms D., Goodstein D. M., Schmutz J.: GENESPACE tracks regions of interest and gene copy number variation across multiple genomes. *eLife 11* (Sept. 2022), e78526. doi:10.7554/eLife.78526. 3

[LZY*22] Li M., Zhang H., Yang Y., Wang H., Xue Z., Fan Y., Sun P., Zhang H., Zhang X., Jin W.: Rosa1, a transposable element-like insertion, produces red petal coloration in rose through altering rcmyb114 transcription. *Frontiers in Plant Science 13* (2022). doi:10.3389/fpls.2022.857684. 9

[M*18] Marschall T., et al.: Computational pan-genomics: Status, promises and challenges. *Briefings in Bioinformatics 19*, 1 (2018), 118–135. doi:10.1093/bib/bbw089. 2

[MHM20] McInnes L., Healy J., Melville J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, 2020. arXiv:1802.03426 [stat]. doi:10.48550/arXiv.1802.03426. 5

[MKK*19] Miao H., Klein T., Kouřil D., Mindek P., Schatz K., Gröller M. E., Kozlíková B., Isenberg T., Viola I.: Multiscale Molecular Visualization. *Journal of Molecular Biology 431*, 6 (2019), 1049–1070. doi:10.1016/j.jmb.2018.09.004. 4

[MMP09] Meyer M., Munzner T., Pfister H.: MizBee: A multiscale synteny browsers. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 897–904. doi:10.1109/TVCG.2009.167. 3, 4, 5, 6

[NHG19] Nusrat S., Harbig T., Gehlenborg N.: Tasks, techniques, and tools for genomic data visualization. *Computer Graphics Forum 38*, 3 (2019), 781–805. doi:10.1111/cgf.13727. 3

[NMSL19] Nobre C., Meyer M., Streit M., Lex A.: The state of the art in visualizing multivariate networks. *Computer Graphics Forum 38*, 3 (2019), 807–832. doi:10.1111/cgf.13728. 4

[PF93] Perlin K., Fox D.: Pad: an alternative approach to the computer interface. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques* (Anaheim CA, 1993), ACM, pp. 57–64. doi:10.1145/166117.166125. 4

[PFH*18] Pezzotti N., Fekete J., Höllt T., Lelieveldt B. P., Eisemann E., Vilanova A.: Multiscale Visualization and Exploration of Large Bipartite Graphs. *Computer Graphics Forum 37*, 3 (2018), 549–560. doi:10.1111/cgf.13441. 4

[PND20] Pham V., Nguyen V. T. N., Dang T.: DualNetView: Dual views for visualizing the dynamics of networks. *EuroVis Workshop on Visual Analytics (EuroVA)* (2020), 5 pages. doi:10.2312/EUROVA.20201082. 4, 10

[PNWUM17] Pedersen T. L., Nookaew I., Wayne Ussery D., Månsson M.: PanViz: Interactive visualization of the structure of functionally annotated pangenomes. *Bioinformatics 33*, 7 (2017), 1081–1082. doi:10.1093/bioinformatics/btw761. 4

[SAdRSS18] Sheikhizadeh Anari S., de Ridder D., Schranz M. E., Smit S.: Efficient inference of homologs in large eukaryotic pan-proteomes. *BMC bioinformatics 19*, 1 (2018), 340. doi:10.1186/s12859-018-2362-4. 6

[SDMW09] Schreiber F., Dwyer T., Marriott K., Wybrow M.: A generic algorithm for layout of biological networks. *BMC Bioinformatics 10* (2009), 375. Publisher: BMC. doi:10.1186/1471-2105-10-375. 4

[Shn96] Shneiderman B.: The eyes have it: a task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages* (1996), pp. 336–343. ISSN: 1049-2615. doi:10.1109/VL.1996.545307. 4, 10

[SJKL*05] Stewart Jr C., Kang B.-C., Liu K., Mazourek M., Moore S. L., Yoo E. Y., Kim B.-D., Paran I., Jahn M. M.: The pun1 gene for pungency in pepper encodes a putative acyltransferase. *The Plant Journal 42*, 5 (2005), 675–688. doi:https://doi.org/10.1111/j.1365-313X.2005.02410.x. 8

[SM07] Sinha A. U., Meller J.: Cinteny: Flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. *BMC Bioinformatics 8* (2007), 1–9. doi:10.1186/1471-2105-8-82. 3

[SMM12] Sedlmair M., Meyer M., Munzner T.: Design Study Methodology: Reflections from the Trenches and the Stacks. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2431–2440. doi:10.1109/TVCG.2012.213. 3

[SMS*07] Stewart Charles J., Mazourek M., Stellari G. M., O'Connell M., Jahn M.: Genetic control of pungency in C. chinense via the Pun1 locus. *Journal of Experimental Botany 58*, 5 (2007), 979–991. doi:10.1093/jxb/erl243. 8

[Tek16] Tekaia F.: Inferring orthologs: Open questions and perspectives. *Genomics Insights 9* (2016), 17–28. doi:10.4137/GEI.S37925. 3

[TKZ*24] Tang H., Krishnakumar V., Zeng X., Xu Z., Taranto A., Lomas J. S., Zhang Y., Huang Y., Wang Y., Yim W. C., Zhang J., Zhang X.: JCVI: A versatile toolkit for comparative genomics analysis. *iMeta 3*, 4 (2024), e211. doi:10.1002/imt2.211. 4, 10

[VdPMM17] Van de Peer Y., Mizrachi E., Marchal K.: The evolutionary significance of polyploidy. *Nature Reviews Genetics 18*, 7 (2017), 411–424. doi:10.1038/nrg.2017.26. 2

[YMA*17] Yao G., Ming M., Allan A. C., Gu C., Li L., Wu X., Wang R., Chang Y., Qi K., Zhang S., Wu J.: Map-based cloning of the pear gene 114 identifies an interaction with other transcription factors to coordinately regulate fruit anthocyanin biosynthesis. *The Plant Journal 92*, 3 (2017), 437–451. doi:https://doi.org/10.1111/tpj.13666. 9

[ZES19] Zhao T., Eric Schranz M.: Network-based microsynteny analysis identifies major differences and genomic outliers in mammalian and angiosperm genomes. *Proceedings of the National Academy of Sciences of the United States of America 116*, 6 (2019), 2165–2174. doi:10.1073/pnas.1801757116. 6

[ZYL*24] Zhang Z., Yang T., Liu Y., Wu S., Sun H., Wu J., Li Y., Zheng Y., Ren H., Yang Y., Shi S., Wang W., Pan Q., Lian L., Duan S., Zhu Y., Cai Y., Zhou H., Zhang H., Tang K., Cui J., Gao D., Chen L., Jiang Y., Sun X., Zhou X., Fei Z., Ma N., Gao J.: Haplotype-resolved genome assembly and resequencing provide insights into the origin and breeding of modern rose. *Nature Plants 10*, 11 (Oct. 2024), 1659–1671. doi:10.1038/s41477-024-01820-x. 9