Sequence Tube Maps for Structural Variants

A. van den Brandt[1,2,*], J .P. J. Geelen[1,*], and H. van de Wetering[1]

[1]Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands
[2]Bioinformatics Group, Wageningen University & Research, The Netherlands *Authors have contributed equally to this work

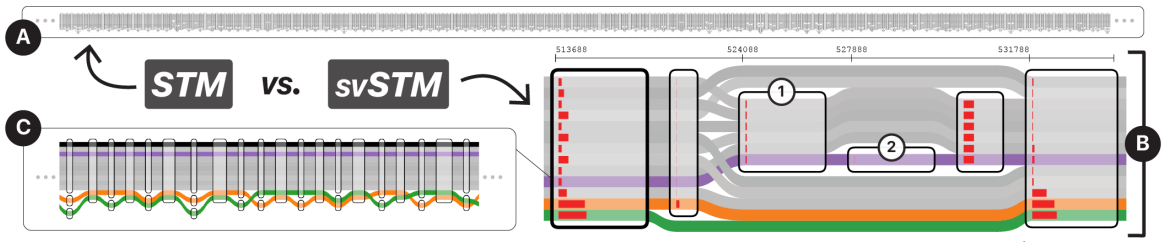

Figure 1: *(A)* STM *vs.* SVSTM *on yeast data. (B)* SVSTM *shows known copy number variants of ENA genes [SSK*15]: the purple tube has 3xENA visible in ① and ②, and grey tubes 2xENA ①; green and orange are wild strains with 1xENA. (C) Details view showing wild SNPs.*

Abstract

Discovery of large sequence changes—structural variants—is crucial yet challenging in genomics. Pangenome graphs aid in their detection, representing DNA of multiple species as a unified structure. Sequence Tube Maps (STM) visualize these graphs as metro maps but quickly become cluttered. Based on STM, we present SVSTM*, which addresses this challenge by reducing visual elements and enabling adjustable detail levels. Overview+detail views highlight large variants, summarize small ones, and offer details on demand. We show* SVSTM*'s utility through use cases and evaluations with domain experts and developers.*

CCS Concepts

• ***Human-centered computing*** → *Visual analytics;* • ***Applied computing*** → *Genomics;*

1. Introduction

Graphs have gained popularity in genomics because they offer effective ways to manage and analyze the ever growing volume of DNA sequencing data. This data reveals that members of a species share much of the same DNA sequence (genome), but also differ. Nowadays, thousands of genomes are available, from small bacteria (e.g. 5 Mb for *E. coli*) to large animals and plants (e.g. 3 Gb for human). One important application of graphs in genomics is representing *pangenomes*, models that combine genomes to capture the full genetic diversity within and across species [Com18]. This diversity is missed by methods using a single reference genome. A common type of pangenome graph is a variation graph. This directed graph has nodes representing stretches of DNA sequence, composed of nucleotide characters (A, C, T, and G). Edges connect nodes in forward or reverse reading direction; nodes connected in reverse direction represent sequence inversions (Fig. 2C). Original sequences are stored as walks through this bi-edged graph [PNGH17]. Edges form the path of each individual genome (Fig. 2A and B), and reveal conserved and variable DNA regions.

Visualization supports exploring and interpreting pangenome graphs but presents significant challenges. Numerous entities–such as sequences, variants, and metadata–vary in size and diversity and must be shown within limited screen space while preserving genomic context and avoiding visual or cognitive overload. An established method is Sequence Tube Map (STM) [BNH*19] (Fig. 2C). While its strength lies in highlighting all variations clearly, as the size and complexity of pangenome graphs increase, STM quickly leads to scalability and readability challenges. To address these, STM offers zooming and logarithmic downscaling of node size. However, these interactions have limited effect when graphs contain many nodes with short sequences, a common scenario in real-world datasets (Fig. 1A). It provides only minimal space savings and no improvements in readability. Moreover, these issues hinder the discovery of *structural variants* (SV)—variations of typically >50 characters, including insertions (INS), deletions (DEL), duplications (DUP), and inversions (INV), see examples in Fig. 2. SVs are often sought after in pangenome graphs because they can strongly impact traits and functional diversity. Yet, despite their importance, SVs generally remain challenging to detect [HHM*20].

Beyond STM, other visualization tools support exploring genomic graphs and SVs, but not without limitations. Some tools use node-link diagrams and show SVs as bubbles or squiggly lines [WSZH15, NJBJ09], but these often become hairballs with many (sub)genomes. Others use linear overviews with colored

© 2025 The Author(s).
Proceedings published by Eurographics - The European Association for Computer Graphics.
This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

DOI: 10.2312/evs.20251091

delivered by EUROGRAPHICS DIGITAL LIBRARY www.eg.org diglib.eg.org

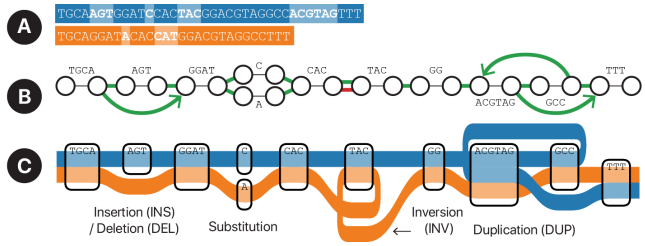

Figure 2: *(A) Two example sequences; (B) A bi-edged graph, forward (green) and reverse (red) edges; (C) A depiction of SVs—INS/DEL, INV, and DUP—and substitution in STM [BNH*19].*

blocks and connecting lines, but the lines cause clutter, obscuring SVs [GHN*22]. Moreover, the fixed vertical ordering hinders spotting groups of (structural) variants [GSN*18]. Sankey-based diagrams highlight groups but are limited to small variations and regions [ZLD24]. Linked circular whole-genome and detail views aim to improve SV detection, yet clutter from small variants remains a challenge [YSS*19]. Population-wide read alignment aids SV detection [LHX*23], but scales poorly to many genomes. Effective and scalable visualization of SVs in large graphs is lacking.

To tackle this challenge, we enhance the visibility of SVs in STM, which also improves scalability and readability. We developed a design called SVSTM, which uses a modified graph layout algorithm that summarizes similar nodes at user-adjustable detail levels. Furthermore, we transformed the STM into an interactive overview+detail display: the overview highlights SVs while indicating areas with smaller variants through summary encodings, and the detail view shows the original STM for a selected node. We also added coordinated highlighting and selection. We evaluated our design with five domain experts, demonstrated use cases on two datasets, and discussed feasibility with STM developers.

2. Original Sequence Tube Map

Sequence Tube Maps [BNH*19] are designed to depict variation graphs, which require encodings beyond standard node-link diagrams to accurately show sequence continuation and direction. These graphs represent multiple sequences, where each sequence is a path through the graph's nodes, traveled in forward or backward direction. To address this, STMs use a metro map-inspired encoding, where each sequence appears as a colored tube passing through nodes (stations) which can be visited by multiple tubes (Fig. 2C).

A Sequence Tube Map $M = (V,T)$ consists of a set of nodes V and a set of tubes T. Each node $v \in V$ represents a sequence of characters: *sequence*(v) with characters from $\{A,C,T,G\}$ representing the nucleotides. A tube $t \in T$ is a list of nodes v, such that v or $v^{-1} \in V$. The sequence of node v^{-1} is the reverse of node v. A tube's sequence is the concatenation of sequences in its node list.

We identified 8 tasks supported in STM using a genomics taxonomy [NHG19] with their support levels (• high / medium / • low):

- • *T1 Lookup variance between (selected) sequences in a region*: well supported for small regions or junction points, as tubes highlight differences and similarities by position.
- • *T2 Explore pangenome variation graph structure*: effective for a limited part of the pangenome; not all nodes fit on the screen.
- • *T3 Locate small variants*: good in small regions; nodes stand out.
- • *T4 Locate SVs*: poorly supported because SVs are less frequent and require extensive scrolling to discover.
- *T5 Identify variation type*: works well as tubes form distinguishable shapes like loops unless many appear in a small region.
- *T6 Identify nucleotide*: possible via letters, but not the main goal.
- *T7 Identify and trace a sequence*: intuitive via the metro map metaphor but becomes cluttered with many tubes and nodes.
- • *T8 Explore read mapping*: well supported, as tubes enable clear tracing of reads (small subsequences) mapped to the graph.

The STM design has strengths and limitations. It uses the strongest visual channel (position) to highlight differences (*T1, T3*) and tube continuity to trace sequences (*T7, T8*). The graph is maximally linearized, ensuring sequences with the same variants stay together (*T1*). However, graphs with many sequences and variants cause readability and scalability issues (*T2, T3, T4*). The loops of INV and DUP are spacious, difficult to distinguish (*T5*), and violate the graph aesthetics rule to minimize edge lengths [DB00]. Interactions are limited; and users have little control over graph layout (*T4, T7*).

3. Design Improvements in SVSTM

From expert input and STM analysis, we derive tasks, goals, and designs for SVSTM. See supplemental note for the code and video.

We aim to support three additional tasks (*T9–T11*):

- *T9 Summarize variance between sequences in the graph.* Users need an overview of the graph to find significant variations.
- *T10 Filter small variants.* Allow excluding small variants (SNPs and indels). Most users focus on larger features such as insertions, while SNPs often cause visual clutter.
- *T11 Explore variants of interest.* Users want to query and visually inspect certain large variations, e.g. INS $\geq$ 500 nucleotides.

Furthermore, we define three design goals (*G1–G3*):

- *G1 Enhance visibility of SVs.* The design should prioritize large structures and selection, which supports *T4, T9, T10*, and *T11*.
- *G2 Improve horizontal scalability.* Increasing the sequences and number of variants shown in the view is essential for effective real-world data exploration, specifically *T2, T3, T4*, and *T9*.
- *G3 Improve vertical scalability.* Larger pangenomes means more sequences must shown clearly, involving tasks *T1, T5*, and *T7*.

3.1. Candidate Designs

We considered various strategies to fulfill our design goals (Fig. 3). G1 is the primary goal, aided by G2, while G3 is lower priority. The naive approach is to **downscale visual elements (1)**—nodes and tubes— so more sequence, including SVs, fit in view. Although it also addresses G2 and G3, downscaling complicates tracing sequences, and thus finding SVs. In real-world graphs, the effect is limited due to many nodes and tubes. Alternatively, a **space-filling**

Approach	Advantage	Disadvantage	Goals Supported G1	G2	G3
#1. Downscale visual elements	Direct increase scalability Can be interactive	Limited effect Readability decreases	✗	✓	✓
#2. Space-filling	Optimal screen usage	Readability decreases	◆	✓	✗
#3. Reduce number of nodes	Direct increase scalability Can be interactive	Limited effect Readability decreases	✓	✓	◆

Figure 3: *A summary of candidate designs and (dis)advantages.*

© 2025 The Author(s).
Proceedings published by Eurographics - The European Association for Computer Graphics.

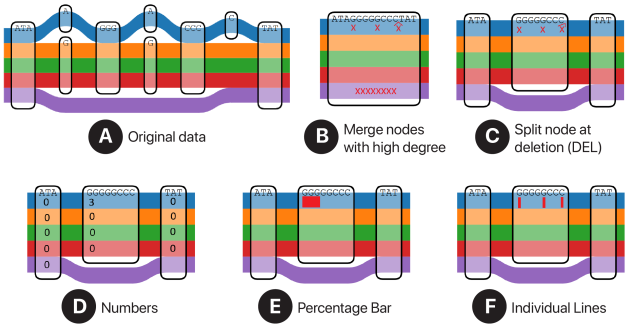

Figure 4: *Node reduction and deviation visualization in* SVSTM. *After node merging (degree* ≥ 3*), one node represents all sequences (B). Two tubes are incorrect: one tube (blue) has 3 deviations and one (purple) has 8; Cutting at the start and end of the DEL (purple), deviations are reduced while adding only 2 nodes (C). Alternatives for showing deviations: numbers (D), percentage bar (E), lines (F).*

layout (2) can arrange 1D sequence data in a 2D grid while preserving local proximity [NHG19]. While space-efficient, it disrupts left-to-right genome navigation and parallel tube arrangements—key for STM—and negatively impacts G3, as fewer tubes can be shown. Third, **reducing the number of nodes (3)** frees visual space, but with information loss. Omitting small nodes supports G1 and G2, and slightly improves G3 by reducing tube movement. As it preserves STM encoding and genome navigation, it is preferred. Information loss can be mitigated by overlaid encoding and interactions.

3.2. Node Reduction

In our node reduction approach, we try to concatenate nodes and only show SVs (INS and DELs), i.e., sequences with lengths larger than a threshold σ (default 50). The approach consists of 2 steps:
1. Find and merge majority nodes Sequences often have highly conserved stretches disrupted by many small variations in some sequences (Fig. 4A). To create a large concatenation with few errors, we merge all nodes through which a majority of the tubes pass. For merging, we compute a topological ordering with Kahn's algorithm [Kah62] to preserve the node order in the original tubes, resulting in one main node (Fig. 4B).
2. Introduce cuts at SVs In practice, the character sequences of the tubes deviate from the main node. To reduce these deviations, we enrich the main node with SVs larger than threshold σ. By cutting up the main node at the start and end of an SV, the total number of deviations can be reduced without introducing many new nodes (Fig. 4C). Tubes can now follow or bypass newly inserted nodes to handle INS and DEL, respectively. Adding SVs in this way gives support for location (*T4*) and overview (*T9*) of SVs.

3.3. Visualization of Deviations

Reducing nodes forces some tubes to follow imperfect matches. To support tasks *T2*, *T4*, and *T9*, which require overview rather than details, our goal is to convey both the existence and magnitude of these deviations. We considered three alternatives for their display (Fig. 4D–F): a number (D), a bar representing the misrepresented fraction (E), and lines marking individual errors by position (F). We evaluated the alternatives based on ease of comparison, effectiveness and pre-attentiveness, and integration in STM. Numbers were least favored, as users must scan many values to spot outliers, whereas bars and lines facilitate comparison. Bars risk being misinterpreted as positional encodings, while individual lines provide position but suffer from overplotting. All are equally suited for integration. Ultimately, the bar encoding (E) was chosen for its ability to highlight significant deviations while maintaining clarity.

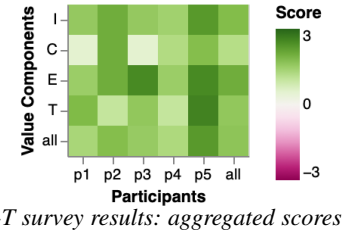

Figure 5: *ICE-T survey results: aggregated scores per component per participant and for the overall value for all participants ("all").*

3.4. Overview, Details, and Interactions

In SVSTM users can drill down (*T11*) by clicking on a node shared by multiple tubes, which opens a second, linked detail view (Fig. 6) that renders only the tubes and nodes within the clicked node. Interacting in one view updates the other accordingly, allowing simultaneous multi-scale exploration. In the overview, users can set a length threshold σ for nodes, aiding tasks (*T10, T11*). For linked highlighting, we implemented tube selection with distinguishable colors [HB03] and set the default STM to grayscale to better support tracing (*T7*). In STM, many tubes likely contain small variations. SVSTM omits these, making tubes appear similar, though their deviation bars may show differences. Similar tubes can be grouped into a single slightly wider tube (*G3*), with member deviations shown as a 1D scatter scatter plot (supplemental Fig. S1).

4. ICE-T Evaluation

SVSTM was evaluated using the ICE-T method [WAM*19], scoring **I**nsights, **C**onfidence, **E**ssence, and **T**ime via survey questions.

Participants Following ICE-T's guideline, we recruited *five* postdocs and PhD researchers with the required pangenomics expertise. All are familiar with STM but had not necessarily used it before.
Procedure Participants received a brief demo of STM and explored an Arabidopsis pangenome region for 10–15 minutes with example tasks. They repeated the process with SVSTM for a different region. To capture qualitative insights, participants were encouraged to think aloud. They completed the survey, which also initiated the discussion. To evaluate SVSTM's improvements over STM, we adapted the question prompt for direct comparison and the terminology for clarity in genomics. Questions were rated on a 7-point Likert scale, with zero as neutral and positive scores as improvements. Each session lasted one hour and was recorded. The protocol, setup, and survey questions are available in the supplemental note, and the data is available at doi:10.17605/osf.io/9rj2e.
Analysis and Insights We averaged component scores and computed an overall value score per participant; see Fig. 5 (and supplementary Table S1). All components were rated positively. The overall average score for SVSTM is 1.6, reflecting a positive assessment of the improvements. On average, the strongest compo-

© 2025 The Author(s).
Proceedings published by Eurographics - The European Association for Computer Graphics.

nents are **essence** and **insight**. Three users explicitly stated that the compression mechanism combined with red deviation bars was a valuable improvement. Participant P4 emphasized: *"The main reason for me to agree or strongly agree [with survey statements] is because the way that you allow the compression of STM and then also on that compressed node you also show this percentage of variation."* P1 and P5 noted the original STM struggles with scalability: *"[the tool] solves one of two major problems with Sequence Tube Maps"* (P1), that being the scalability; *"too many SNPs"* and *"SVs are difficult to identify"* (P5). Confidence had the lowest mean scores (0.5), mainly due to potentially misleading error bars that could also encode misalignment at a nucleotide position. As P5 noted: *"... the error bar [was somewhat misleading], but once I got used to it it was fine."* P1 and P3 also noted that if the nodes' nucleotides remain visible (i.e., without STM's downscaling), the sequence may be misinterpreted even with error bars. They also said that deviations from the majority are not always relevant.

5. Use Cases and Qualitative Evaluation

We briefly show use cases of yeast (n=14) and *Arabidopsis* (n=32) pangenomes with two domain experts; see supplement for details.
Yeast Literature [SSK*15] suggests copy number variants in gene ENA. To explore this SV visually while omitting small variants (*T10*), the user set $\sigma \approx$ gene length (1000 bases). Figure 1B shows the overview: the purple tube (*S. cerevisiae S288C*) has three copies (in ① and ②), while orange and green tubes (wild species) have just one. Other grey tubes have one or two copies. The wild strains showed more deviations, prompting detailed exploration (Fig. 1C).
Arabidopsis The user searched for SVs on chromosome 1 within a 10,000-base region. Varying σ's in the overview (Fig. 6A), three INS were found (*T9*). Exploring deviating node ① in detail (*T11*) revealed that a smaller INS is causing tube deviations (Fig. 6B).

We conducted a feedback session (online, 1 hour) with an STM developer. They were overall enthusiastic about the presented design improvements. The STM developer also noted that the error bar representation might be misleading. They considered it a minor concern as long as sufficient documentation was provided, stating they were *"excited to bikeshed it."* They were working on a similar compression idea to filter out small nodes but without showing deviations or providing an interactive σ, and found that offering both approaches could be beneficial. Furthermore, they noted that *"the selection feature and sort of minimap view [overview+detail] are both features that would make sense for existing users and that we would want to merge."* Overall, the session highlighted SVSTM's feasibility, utility, and potential for integration into STM.

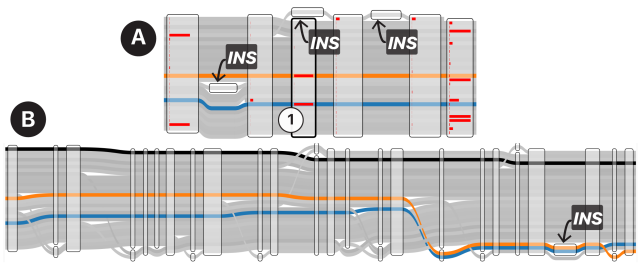

Figure 6: SVSTM *(σ=1000) showing Arabidopsis: (A) Overview with 3 INSs; (B) Deviations in node ① orange and blue by an INS.*

6. Discussion and Conclusion

Our goal was to enhance Sequence Tube Maps with added scalability and readability for SV discovery while preserving its intuitive encoding scheme. Compared to STM, SVSTM enables the display of a large pangenomic region in limited space. By summarizing nodes, displaying deviations, and allowing detailed inspection, users can explore pangenomes more effectively. SVSTM was positively received. User tests suggest that SVSTM enhances insight and conveys the dataset's essence. The yeast use case successfully revealed a known SV, and STM developers showed strong interest in integrating SVSTM. Surprisingly, interactive selection was not familiar to some participants, but through exploration and discussion, they recognized its value and the overall design coherence.

Evaluations showed SVSTM's potential, but also areas for improvement and future research. Many users had questions after a walk-through, indicating learnability challenges. The original STM also posed difficulties for some users (2/5), who struggled to interpret the tube length between nodes as non-meaningful, unlike traditional alignments. Furthermore, users and developers had differing opinions on the deviation bar encoding, suggesting other encodings might be worth exploring. Two users suggested allowing interactive selection of a reference. Additionally, integrating (gene) annotation tracks could aid in biological interpretation (2/5 users). The current approach focuses on insertions and deletions, making inversions and duplications less immediately visible. Since the importance of such SVs varies among species—rare among humans but more common among plants [Yan20, JSM24]—it would be good to include them in future work. The current algorithm can be easily extended by introducing more cuts for this purpose.

Future work could explore incorporating more metro map aspects to further improve readability, e.g., emphasizing important nodes (stations) where many edges (lines) cross, evenly spreading nodes, and balancing edge angles [SRMOW11]. Layout strategies from genome-wide multiple sequence graphs could also be explored [SZW24]. Beyond layout, node recomputation needs optimization for even larger regions. This work focused on visual SV detection in a pangenome graph. Other visualizations include circular overviews [HJBN12], also well suited for SV detection, and linear block views [DSCR21] or matrices [HBN15], which pose some limitations for this task but do support integrating metadata.

For evaluation, we recruited, despite their scarcity, five domain experts per ICE-T's minimum. More studies with diverse users and datasets are needed to validate real-world utility. Usability may be further assessed using the SUS [B*96], and a comparative study could quantify benefits, e.g., space savings or task accuracy.

In conclusion, we presented SVSTM as an extension of STM, designed to support structural variant exploration. The design enables better navigation of complex pangenome graphs, leveraging summarization and deviation encodings to detect SVs at scale. We anticipate that these improvements will further enhance Sequence Tube Maps' effectiveness for the genomics community.

Acknowlegdements

We thank the participants in the user evaluations for their feedback and Dirk-Jan van Workum for preparing the datasets.

© 2025 The Author(s).
Proceedings published by Eurographics - The European Association for Computer Graphics.

References

[B*96] BROOKE J., ET AL.: Sus-a quick and dirty usability scale. *Usability evaluation in industry 189*, 194 (1996), 4–7. 4

[BNH*19] BEYER W., NOVAK A. M., HICKEY G., CHAN J., TAN V., PATEN B., ZERBINO D. R.: Sequence tube maps: making graph genomes intuitive to commuters. *Bioinformatics 35*, 24 (2019), 5318–5320. doi:10.1093/bioinformatics/btz597. 1, 2

[Com18] COMPUTATIONAL PAN-GENOMICS CONSORTIUM: Computational pan-genomics: status, promises and challenges. *Briefings in Bioinformatics 19*, 1 (2018), 118–135. doi:10.1093/bib/bbw089. 1

[DB00] DI BATTISTA G.: Graph drawing: the aesthetics-complexity trade-off. In *Operations Research Proceedings 1999* (Berlin, Heidelberg, 2000), Inderfurth K., Schwödiauer G., Domschke W., Juhnke F., Kleinschmidt P., Wäscher G., (Eds.), Springer Berlin Heidelberg, pp. 92–94. 2

[DSCR21] DURANT , SABOT F., CONTE M., ROUARD M.: Panache: a web browser-based viewer for linearized pangenomes. *Bioinformatics 37*, 23 (2021), 4556–4558. doi:10.1093/bioinformatics/btab688. 4

[GHN*22] GUARRACINO A., HEUMOS S., NAHNSEN S., PRINS P., GARRISON E.: ODGI: understanding pangenome graphs. *Bioinformatics 38*, 13 (2022), 3319–3326. doi:10.1093/bioinformatics/btac308. 2

[GSN*18] GARRISON E., SIRÉN J., NOVAK A. M., HICKEY G., EIZENGA J. M., DAWSON E. T., JONES W., GARG S., MARKELLO C., LIN M. F., ET AL.: Variation graph toolkit improves read mapping by representing genetic variation in the reference. *Nature biotechnology 36*, 9 (2018), 875–879. doi:10.1038/nbt.4227. 2

[HB03] HARROWER M., BREWER C. A.: ColorBrewer.org: An online tool for selecting colour schemes for maps. *Cartographic Journal 40*, 1 (2003), 27–37. doi:10.1179/000870403235002042. 3

[HBN15] HENNIG A., BERNHARDT J., NIESELT K.: Pan-Tetris: an interactive visualisation for Pan-genomes. *BMC Bioinformatics 16*, S11 (2015), S3. doi:10.1186/1471-2105-16-S11-S3. 4

[HHM*20] HICKEY G., HELLER D., MONLONG J., SIBBESEN J. A., SIRÉN J., EIZENGA J., DAWSON E. T., GARRISON E., NOVAK A. M., PATEN B.: Genotyping structural variants in pangenome graphs using the vg toolkit. *Genome Biology 21* (2020), 1–17. doi:https://doi.org/10.1186/s13059-020-1941-7. 1

[HJBN12] HERBIG A., JÄGER G., BATTKE F., NIESELT K.: GenomeRing: alignment visualization based on SuperGenome coordinates. *Bioinformatics 28*, 12 (2012), i7–i15. doi:10.1093/bioinformatics/bts217. 4

[JSM24] JAYAKODI M., SHIM H., MASCHER M.: What Are We Learning from Plant Pangenomes? *Annual Review of Plant Biology* (2024). doi:10.1146/annurev-arplant-090823-015358. 4

[Kah62] KAHN A. B.: Topological sorting of large networks. *Commun. ACM 5*, 11 (1962), 558–562. doi:10.1145/368996.369025. 3

[LHX*23] LI F., HU H., XIAO Z., WANG J., LIU J., ZHAO D., FU Y., WANG Y., YUAN X., BU S., ZHOU X., ZHAO J., WANG S.: Visualization and review of reads alignment on the graphical pan-genome with VAG, 2023. doi:10.1101/2023.01.20.524849. 2

[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. 2, 3

[NJBJ09] NIELSEN C., JACKMAN S., BIROL I., JONES S.: ABySS-Explorer: Visualizing Genome Sequence Assemblies. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 881–888. doi:10.1109/TVCG.2009.116. 1

[PNGH17] PATEN B., NOVAK A. M., GARRISON E., HICKEY G.: Superbubbles, ultrabubbles and cacti. In *Research in Computational Molecular Biology* (Cham, 2017), Sahinalp S. C., (Ed.), Springer International Publishing, pp. 173–189. doi:10.1007/978-3-319-56970-3_11. 1

[SRMOW11] STOTT J., RODGERS P., MARTÍNEZ-OVANDO J. C., WALKER S. G.: Automatic metro map layout using multicriteria optimization. *IEEE Transactions on Visualization and Computer Graphics 17*, 1 (2011), 101–114. doi:10.1109/TVCG.2010.24. 4

[SSK*15] STROPE P. K., SKELLY D. A., KOZMIN S. G., MAHADEVAN G., STONE E. A., MAGWENE P. M., DIETRICH F. S., MCCUSKER J. H.: The 100-genomes strains, an *S. cerevisiae* resource that illuminates its natural phenotypic and genotypic variation and emergence as an opportunistic pathogen. *Genome Research 25*, 5 (2015), 762–774. doi:10.1101/gr.185538.114. 1, 4

[SZW24] SCHEBERA J., ZECKZER D., WIEGREFFE D.: A layout framework for genome-wide multiple sequence alignment graphs. *Frontiers in Bioinformatics 4* (2024), 1358374. doi:10.3389/fbinf.2024.1358374. 4

[WAM*19] WALL E., AGNIHOTRI M., MATZEN L., DIVIS K., HAASS M., ENDERT A., STASKO J.: A Heuristic Approach to Value-Driven Evaluation of Visualizations. *IEEE Transactions on Visualization and Computer Graphics 25*, 1 (2019), 491–500. doi:10.1109/TVCG.2018.2865146. 3

[WSZH15] WICK R. R., SCHULTZ M. B., ZOBEL J., HOLT K. E.: Bandage: interactive visualization of *de novo* genome assemblies. *Bioinformatics 31*, 20 (2015), 3350–3352. doi:10.1093/bioinformatics/btv383. 1

[Yan20] YANG L.: A practical guide for structural variation detection in the human genome. *Current protocols in human genetics 107*, 1 (2020), e103. doi:10.1002/cphg.103. 4

[YSS*19] YOKOYAMA T. T., SAKAMOTO Y., SEKI M., SUZUKI Y., KASAHARA M.: MoMI-G: modular multi-scale integrated genome graph browser. *BMC Bioinformatics 20*, 1 (2019), 548. doi:10.1186/s12859-019-3145-2. 2

[ZLD24] ZDĄBŁASZ K., LISIECKA A., DOJER N.: Sequence Flow: interactive web application for visualizing partial order alignments. *BMC Genomics 25*, 1 (2024), 973. doi:10.1186/s12864-024-10886-y. 2

© 2025 The Author(s).
Proceedings published by Eurographics - The European Association for Computer Graphics.