MetagenomeScope

Web-Based Hierarchical Visualization of Metagenome Assembly Graphs

Marcus Fedarko1,2, Jay Ghurye1,2, Todd Treangen2, and Mihai Pop1,2
1Department of Computer Science, University of Maryland, College Park
2Center for Bioinformatics and Computational Biology, University of Maryland, College Park
mfedarko@umd.edu, jayg@cs.umd.edu, treangen@umd.edu, mpop@umiacs.umd.edu

Motivation

Complexities such as repetitive sequences and sequencing errors can create branches and cycles in assembly graphs. These graphs thus often require manual examination after their generation to resolve ambiguous connections and correct assembly errors [6]. This has brought about the need for tools that can visualize assembly graphs effectively, displaying relevant biological metadata and graph structural information alike in a readily accessible manner. Furthermore, there is a documented dearth of hierarchical visualization tools that allow the user to navigate “from the large structure down to the base level [of the assembly graph]” [4].

To fulfill this need we present MetagenomeScope, an interactive web-based tool for the visualization of assembly graphs. MetagenomeScope contains a number of features intended to aid bioinformaticians in exploratory analysis of these graphs at both coarse and fine levels of complexity.

Implementation and Availability

MetagenomeScope is composed of two software components—a preprocessing script (written in Python and C++) and a client-side viewer interface. The preprocessing script takes an assembly graph file as input and performs layout on its connected components using Graphviz’ [1] dot fit and (for portions of the SPQR decomposition mode) sgraph layout tools, generating a SQLite3 database file that can be loaded in the viewer interface. The viewer interface uses Cytovue.js [2] to visualize these graphs accordingly. Although MetagenomeScope was designed to be useful in a metagenomic context, the tool can also be used to visualize single-genome assembly graphs.

MetagenomeScope is licensed under the GNU Public License; version 3.0. Its code is publicly available on GitHub at https://github.com/marbl/MetagenomeScope.

Related Work

In view of previous efforts to visualize assembly graphs, there is a need for tools that can visualize assembly graphs effectively, displaying relevant biological metadata and graph structural information alike in a readily accessible manner. Furthermore, there is a documented dearth of hierarchical visualization tools that allow the user to navigate “from the large structure down to the base level [of the assembly graph]” [4].

Future Work

• Parallelizing layout operations across connected/biconnected components
• Using tree maps to indicate relative sizes of connected components
• Further utilizing SPQR tree decompositions to purposefully decompose regions of the graph spanning large horizontal distances in a linearized layout

Acknowledgements

The authors were supported in part by the NIH, grant R01-AI-100947, the NSF, grant IIS-1117247, and the Navy Research Laboratories' companion program NA6999015382, all to MP. T.O. is supported by the Blue Gill Travel Fellowship. This poster has been made using the “a0poster-portrait” LaTeX template, available online at https://latex-template.com/TeX/ports/a0poster-portrait

References


Figure 1: Seventeenth largest connected component of a human metagenome assembly graph (accession ID SRS049959), visualized in MetagenomeScope. Gray pentagons represent contigs; fragments of DNA paired together by an assembler, edges correspond to overlaps between contigs.

Figure 2: Screenshots of a SPQR tree visualized in MetagenomeScope’s “explicit” SPQR tree decomposition mode, in various states of collapsedness. Fig. 4a shows the tree as it is displayed upon first being drawn, collapsed to its root metanode. Each additional uncollapsing operation reveals further information in the tree, indicating paths through the underlying biconnected component structure in increasing levels of detail. The graph shown here was taken from the “explicit” SPQR tree decomposition mode visualization of Fig. 6a after its root metanode was uncollapsed. Each additional uncollapsing operation reveals further information in the tree, including paths through the underlying bi-connected component structure in increasing levels of detail. The graph shown here was taken from the graph shown in Fig. 6a after its root metanode was uncollapsed.

Figure 3: Screenshots of MetagenomeScope’s standard mode, focused on another region of the SRS049959 graph. From a biological perspective, the intersecting paths through this region of the graph might indicate variation between two otherwise similar DNA samples contained in this metagenome.

Figure 4: Screenshots of a SPQR tree visualized in MetagenomeScope’s “standard mode” of the SRS049959 graphs. As demonstrated in Fig. 4a, structural pattern collapsing can significantly reduce the visual complexity of the display, thus simplifying manual analysis.

Figure 5: Region of a biofilm assembly graph visualized in MetagenomeScope. Contigs contained within a selected scaffold are colored darker than other contigs to indicate their selection status.

Feature

Hierarchical layout

Figure 3: Screenshots of MetagenomeScope’s standard mode, focused on another region of the SRS049959 graph. From a biological perspective, the intersecting paths through this region of the graph might indicate variation between two otherwise similar DNA samples contained in this metagenome.

Figure 4: Screenshots of a SPQR tree visualized in MetagenomeScope’s “explicit” SPQR tree decomposition mode, in various states of collapsedness. Fig. 4a shows the tree as it is displayed upon first being drawn, collapsed to its root metanode. Each additional uncollapsing operation reveals further information in the tree, indicating paths through the underlying bi-connected component structure in increasing levels of detail. The graph shown here was taken from the “explicit” SPQR tree decomposition mode visualization of Fig. 6a after its root metanode was uncollapsed. Each additional uncollapsing operation reveals further information in the tree, including paths through the underlying bi-connected component structure in increasing levels of detail. The graph shown here was taken from the graph shown in Fig. 6a after its root metanode was uncollapsed.

Figure 5: Region of a biofilm assembly graph visualized in MetagenomeScope. Contigs contained within a selected scaffold are colored darker than other contigs to indicate their selection status.

SPQR tree mode

(a) Fully collapsed SPQR tree.
(b) Partially collapsed SPQR tree.
(c) Fully uncollapsed SPQR tree.

Feature

Hierarchical layout

Figure 3: Screenshots of MetagenomeScope’s standard mode, focused on another region of the SRS049959 graph. From a biological perspective, the intersecting paths through this region of the graph might indicate variation between two otherwise similar DNA samples contained in this metagenome.

Figure 4: Screenshots of a SPQR tree visualized in MetagenomeScope’s “explicit” SPQR tree decomposition mode, in various states of collapsedness. Fig. 4a shows the tree as it is displayed upon first being drawn, collapsed to its root metanode. Each additional uncollapsing operation reveals further information in the tree, indicating paths through the underlying bi-connected component structure in increasing levels of detail. The graph shown here was taken from the “explicit” SPQR tree decomposition mode visualization of Fig. 6a after its root metanode was uncollapsed. Each additional uncollapsing operation reveals further information in the tree, including paths through the underlying bi-connected component structure in increasing levels of detail. The graph shown here was taken from the graph shown in Fig. 6a after its root metanode was uncollapsed.

Figure 5: Region of a biofilm assembly graph visualized in MetagenomeScope. Contigs contained within a selected scaffold are colored darker than other contigs to indicate their selection status.

SPQR tree mode

(a) Fully collapsed SPQR tree.
(b) Partially collapsed SPQR tree.
(c) Fully uncollapsed SPQR tree.