Interactive Network Visualization of Gene Expression Time-Series Data

António Cruz
CISUC - Department of Informatics Engineering, University of Coimbra, 3030-290 Coimbra, Portugal
Email: antonioc@dei.uc.pt

Joel P. Arrais
CISUC - Department of Informatics Engineering, University of Coimbra, 3030-290 Coimbra, Portugal
Email: jpa@dei.uc.pt

Penousal Machado
CISUC - Department of Informatics Engineering, University of Coimbra, 3030-290 Coimbra, Portugal
Email: machado@dei.uc.pt

***Abstract*—Visualization models have shown to be remarkably important in the interpretation of datasets across many fields of study. In the field of Biology, data visualization is used to better understand processes that range from phylogenetic trees to multiple layers of molecular networks. The latter is especially challenging due to the large quantities of varying elements and complex relationships, often with no perceptible structure. Although various tools have been proposed to improve the visualization of molecular networks, many challenges still persist. In this paper, we propose a tool that uses interactive visualization models to represent the dynamic behaviors of molecular networks. The tool employs various methods to explore and organize the data, including clustering, force-directed layouts, and a timeline for navigating through time-series data. To further analyze temporal attributes, the timeline can be distorted through a force-directed layout to spatially position time points according to their similarity. Additionally, gene expression can be annotated through an integrated biological database. The visualization model was validated with the use of time-series gene expression RNA-Seq data from the HIV-1 infection.**

***Keywords*-Molecular Network, Gene Expression, Time Series Data, Protein-Protein Interaction, Data Clustering**

I. Introduction

For the past two decades, the representation of network data has been a topic of particular interest in research within the field of Biology [1]. Studies that regard regulation, gene signaling and protein interactions often describe networks that can be described as complex, as they contain of millions of relationships between diverse data elements with no apparent order. Visualization tools must keep up with the emergence of new technologies and integrate modern techniques in order to properly handle these increasingly complex datasets. In the domain of molecular biology, the current challenges include the analysis and comparison between multiple multivariate datasets and the difficulty to discover and highlight significant patterns of data from which biological interpretation can be obtained, particularly in time-series. However, many of the current biological visualization tools lack dynamic visualization models or present limited visual approaches for the exploration of temporal data.

In this paper, we propose a novel visualization tool for representing and analyzing gene expression time-series data, capable of improving the biological reasoning over large datasets. It combines the ability to present time-series data in a comprehensive manner, while dynamically sorting this data into any given number of groups through the use of a clustering algorithm and force-directed layouts. This is accomplished by analyzing and grouping nodes based on the similarity of their attributes, such as gene expression variation between time points, which identifies clusters of proteins that share similar activation patterns. Furthermore, by distorting the timeline in order to position time points according to their similarity, it is possible to identify behaviours that occur over time, such as cycles and significant changes [2]. In order to demonstrate the functionalities of the developed visualization tool, we chose to visualize a time-series gene expression RNA-Seq dataset of the HIV-1 infection on human cells. The analysis of such data may lead to increased knowledge of basic molecular mechanisms in cells and the behaviors of infections, as well as a better understanding of the underlying biology and therefore to the development and testing of new treatments [3].

II. Related Work

A vast quantity of visualization tools have emerged over the years as a response to the need for identifying patterns in large, unstructured datasets, particularly in the field of Biology [4]. These tools vary in focus and approach, implementing different visualization models and methods to help in the analysis of biological data. In this section, we focus on biological visualization tools and methods that can be used in the exploration of time-series data. In particular, it is necessary to understand the range of approaches applied in the representation of temporal attributes in biological network data, such as protein-protein interaction (PPI) networks.

Time-series gene expression data is typically represented through heatmaps or line charts. Various tools that use heatmaps in the analysis of time-series gene expression clusters also provide dendrogram visualizations to describe the clustering process, like the Hierarchical Clustering Explorer [5] and BiGGEsTS [6]. Alternatively, MLCut [7] forgoes

the heatmap and focuses on the dendrogram visualization, providing the user with sliders to dynamically adjust clustering parameters and create clusters by cutting branches at multiple levels. The tool also provides a parallel coordinates visualization that is interactively linked to the dendrogram.

Line charts have gained popularity over heatmaps as the quantities of expression values are more easily perceptible through position than color. Additionally, groups of genes can be represented using a single chart, calculated using the mean of every time point in each gene. This is demonstrated by STEM [8], where time-series clusters are represented as small multiples of line charts, which can then be selected in order to view a superimposition of every gene in that group. MulteeSum [9] and Pathline [10] also make use of line charts to show time-series gene expression as a matrix of small multiples, called a curvemap visualization, which displays multiple instances of expression variation for multiple genes.

In regards to the representation of PPI data, the large quantity of unstructured relationships often results in confusing networks with no clear patterns. To resolve this, network visualization tools resort interactive and dynamic layouts, such as force-directed layouts. For instance, VisANT [11] supports the exploration of large biological networks through the recursive expansion of a network by clicking on nodes. Graphia Pro, a continuation of the BioLayout Express 3D software [12], is another interactive network analysis tool that is able to cluster large networks in both 2D and 3D using a force-directed layout, while also representing changes in the data through animation.

Cytoscape is an open source generic network visualization platform [13], but it is also able to integrate networks with biological data. While its generated artifacts are static, it provides the user with various customization, navigation and layout options, including edge bundling, as well as support for hundreds of plugins. A notable plugin for the analysis of gene expression data is Cerebral [14], which uses multiple coordinated windows to show the different temporal states of a biological network. Cerebral also provides filters and adjustable clustering, where clusters are listed as line charts that can be selected to highlight the respective group of nodes in the network view.

The line chart representation of time-series has also been embedded directly in networks, where the charts are reduced in size and used as node glyphs, such as in VANTED [15]. VisBricks [16] also makes use of glyphs, representing clusters through small interactive visualization panels with various visualization options. These are displayed in a parallel coordinates layout where each axis represents a different dataset or temporal state, and edges are drawn between them to show their relationships. However, this presents a limitation as relationships can only be drawn between sequential columns.

Analyzing the evolution of networks over time and identifying key moments is still a significant challenge, particularly across large networks. Aggregation techniques can be used to abstract networks into simple graphical elements that can be compared simultaneously in large amounts. For instance, Elzen et al. [17] propose that each temporal state of a network can be reduced to a point by translating its characteristics onto coordinates on a two-dimensional plane. Similarly, Bach et al. introduced Time Curves [2], where timelines are folded onto themselves so that the distance between time points matches their similarity. These methods create network visualizations that can be used to compare multiple complex structures and overview their evolution over time.

III. Framework

The canvas of the developed tool is divided into a user interface and a data visualization (Figure 1). The data visualization consists of a network and a timeline for time-series data. The interface contains buttons and dropdowns that provide control over the amount of visible information and visualization methods. These methods allow the data to be sorted by its various attributes and dynamically organize the network into a variable number of discernible groups. Additionally, if an element of the network is selected, the user interface will display information on its attributes.

A. Data Management

The data visualization is created through user-provided dataset files: a file containing a list of the edges, required to initialize the network structure, and a file that lists numerical attributes associated with each node. The first file can be loaded on its own in order to generate a network visualization, but certain functionalities will not be available without attribute data. The names of the attributes can be defined in the file, which will then be displayed in the interface, while the values are mapped to the corresponding nodes' proprieties. While the tool was developed with the intention of representing time-series, in which this file would contain ordered lists of values, each corresponding to a time point, it is also possible to load a list in which each attribute represents an independent variable. The tool also integrates Gene Ontology (GO) databases in order to provide additional information and compare between biological elements [18]. These databases allow the application to identify, classify and sort protein datasets by utilizing their respective GO terms.

B. User Interface

The interface displays additional information about selected nodes and groups, along with various options. The search bar highlights nodes whose names equal or contain the text input, while the buttons allow the user to load or replace the data files, toggle the initial force-directed layout and edge visibility, group nodes into clusters and change the node selection distance. The node selection distance defines

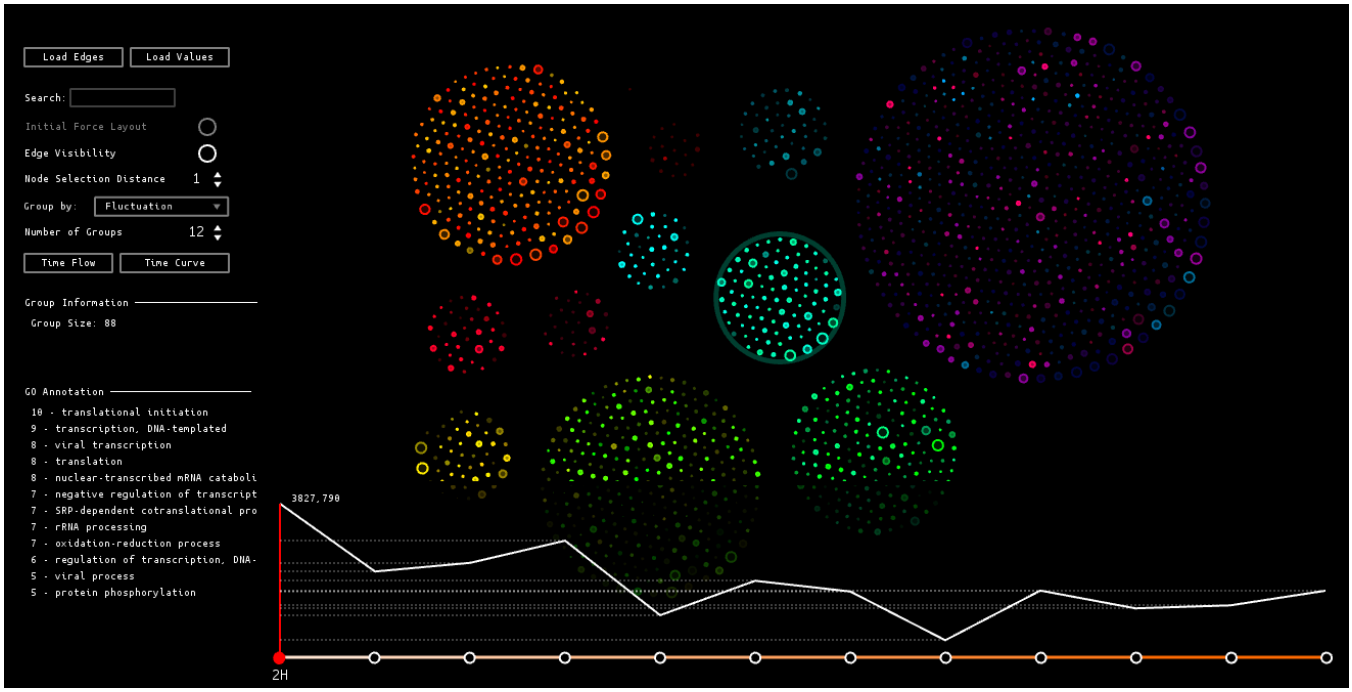

Figure 1. The developed visualization tool is able to create a network structure and dynamically organize it according to a selected attribute. A colored border marks the currently selected cluster, while the left bottom corner displays functional annotations for the selected genes. The chart on the bottom shows the average variation of the time-series data in that cluster, and the slider below controls the current time point.

how closely related a node should be to a selected node in order to be highlighted, which can be used to identify a node's level of influence in the network.

When handling biological data, the list of processes associated to any selected node is queried from the GO database and displayed at the bottom of the user interface. When a cluster is selected, this list will display all the biological processes found in that group, along with the quantity of occurrences, as shown in Figure 1. This provides the user with the ability to compare the groups of proteins based on external database data.

IV. Data Visualization

The data visualization utilizes a dynamic network model and an interactive timeline to represent and analyze relational data over time. While the developed visualization models are generic and, therefore, applicable to any dataset that comply with a specific structure, the tool integrates data-specific methods for the analysis of biological data and time-series. The data can be grouped using a clustering algorithm, while force-directed layouts are used to visually sort the network's nodes and bend the timeline into a time curve.

A. Network Model

The main visualization model is a two-dimensional network graph. However, edges are hidden by default. This is because the large amount of edges in complex networks often results in cluttered and confusing visualizations. As such, in this model, position is given more importance in portraying less-evident relationships, while edges can be enabled or disabled in the interface. When enabled, the edges between any highlighted nodes are drawn through an edge bundling algorithm. This algorithm converts each edge into a path with several intermediate points and iteratively converges points of neighboring edges with similar directions, transforming them into organic, fluid structures [19].

In regards to representation, when the network's structure if first loaded, nodes are drawn as white circles with random positions. The nodes are then sorted by the Yifan Hu force-directed layout, which iteratively recalculates the nodes' positions in accordance to their relationships. Loading attribute data will map the values to size of their respective nodes. Additionally, when representing time-series data, the brightness value of each node's color will also be mapped to the variation of the temporal values. As such, nodes will increase and decrease in brightness along with their attributes over time. Nodes can also be clustered into groups based on either their current positions on the network, or their similarity to other nodes based on their attributes.

The mouse can be used for interaction and navigation within the network. Hovering the mouse over a node displays its name and highlights other connected nodes. If the visualization is clustered, then the user can highlight a cluster by hovering the mouse over its outer edge (Figure 1). Regarding navigation, the position of the visualization can be shifted

by dragging the right mouse button, while the mouse wheel is used to zoom in and out of the position where the mouse is located.

B. Timeline & Time Curve

The interactive timeline consists of a slider that can be dragged to switch between points in time, which updates the size and brightness of nodes according to their attributes at each point. These visual proprieties are mapped to the transition between any two time points, which results in a smooth animation when moving across the timeline. The "Time Flow" button located in the interface will initiate an automatic and cyclical movement of the slider, allowing the user to interact with the visualization while visualizing changes in the values over time. When a node is hovered or selected, a line graph detailing that node's list of attributes will be shown on top of the timeline. If a cluster is hovered, the graph will display an average of each of the attributes of the nodes in that cluster, as shown in Figure 1.

Pressing the "Time Curve" button will apply a force-based layout on the timeline, applying forces between the time-points according to their similarities. This distorts the timeline into a curve, where color shows time progression and the relative distance between each time point represents their similarity (Figure 3). The time curve is capable of representing behaviors such as regressions and cycles (when the curve travels between two distinct sets of nodes back and forth) or significant changes (when there is a large gap between sets of nodes). Upon turning off the time curve, the nodes will return to their default positions on the initial timeline.

C. Clustering Algorithm

Clustering is the unsupervised classification of patterns into groups, known as clusters. In a network, these clusters consist of sets of nodes that present more similarities to the other nodes in same cluster than to those in others. However, the ideal number of clusters depends on the attributes used to determine the similarity between each node and on the type of patterns and information that the user is seeking. Taking this into account, our objective was to give the user the ability to control the number of clusters and have the visualization react dynamically. In order to achieve this without needing to recalculate the clusters each time the number of groups is changed, we implemented a hierarchical clustering algorithm. This type of algorithm results in a hierarchical tree that defines the similarity levels at which groupings change [20].

The user may choose to cluster the nodes based on either their position (Figure 2) or their attributes (Figure 3), depending on the data available. When comparing time-series, nodes are sorted based on the variations of the values over time, meaning that nodes in the same group would present similar increases and decreases at the same points in time.

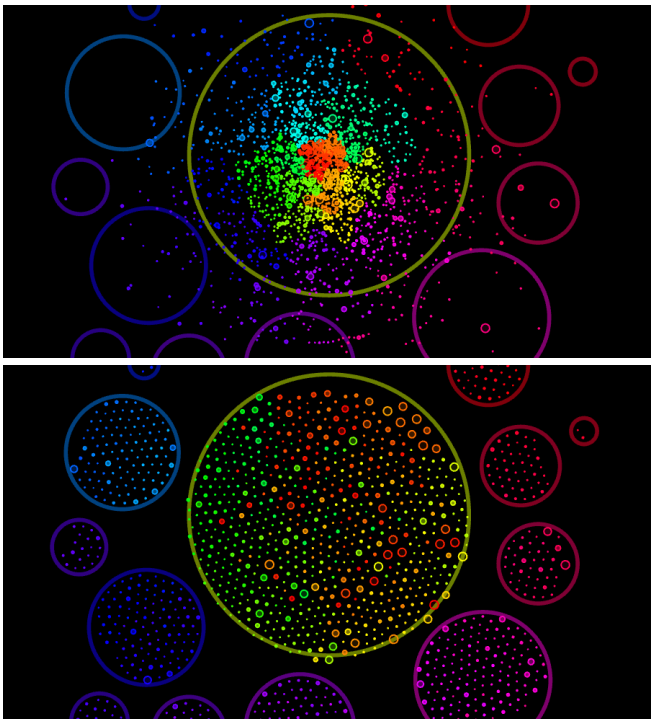

Figure 2. Screenshots of the network clustered by position, showing the attraction points for each cluster as circles (normally not visible during the clustering process). Initially the nodes are positioned using the Yifan Hu layout (top) and then they are attracted to their respective clusters (bottom).

The implemented algorithm is the generic algorithm based on Michael Anderberg's approach [21] described by Daniel Müllner [22]. It utilizes a bottom-up strategy that successively groups the closest clusters until only a single cluster remains. This is achieved by iteratively finding the lowest value in the matrix, corresponding to the two most similar clusters, and combining them into a single cluster. In each iteration, the distance matrix is updated with the distance between the new cluster and every remaining cluster. During this, the algorithm builds a hierarchical tree that will allow the user to switch between any number of clusters without having to recalculate the similarity matrix.

Each node is assigned a hue color value based on their positions in the similarity matrix. This means that similar nodes will also be closer chromatically, allowing for the comparison of nodes both within and between clusters. Nodes without attributes are placed in a separate cluster and are colored white. The black background was chosen in order to emphasize the variation in these colors.

D. Force-Directed Layouts

The visualization utilizes two dynamic layouts: a layout proposed by Yifan Hu [23], which initializes the network by positioning nodes based on their relationships, and a generic

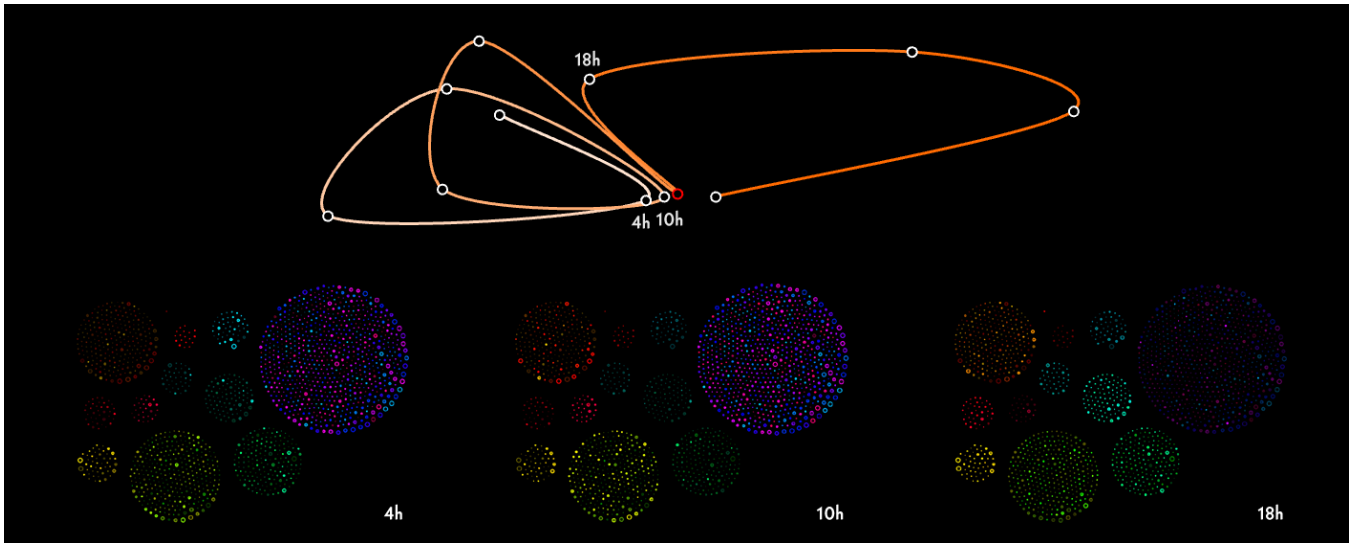

Figure 3. Screenshot of the timeline transformed into a time curve (top) where temporal progression is shown with the color orange. The time curve is shown alongside with screenshots of the network visualization (bottom) clustered based on temporal variance, shown at different time points. While the time points at which the largest clusters show peaks of gene expression were placed next to each other (at 4 and 10 hours), the time point corresponding to 18 hours shows fewer matching genes with expression peaks between the other two time points, and so it has been placed further away.

force-directed layout, capable of creating visual clusters of nodes based on variable grouping parameters. Unlike the Yifan Hu layout which is only applied to the loaded nodes at the beginning, the latter layout was created to control various other visual aspects of the visualization.

The Yifan Hu layout repositions nodes over continuous iterations in order to find placements that adequately reflect the relationships between the nodes. This is combined with a graph coarsening technique which reduces complexity [23]. It was chosen based on its balance of graph quality and performance speed on large graphs, when compared to other algorithms [24].

The generic force-directed layout utilizes attraction and repulsion forces to dynamically adapt the visualization in regards to the user's selections. It is utilized by three different methods to control the relative position of elements according to their relationships: the distribution of clusters, the distribution of nodes throughout each cluster, and the creation of a time curve.

To create visual clusters of nodes, the attraction forces pull each group of nodes closer towards common points which we designate as attraction points. An attraction point is created for every current cluster and positioned based on the initial position of its nodes, the size of the cluster and its relationships with other clusters. The starting position of an attraction point is calculated based on the average position of every node on the initial Yifan Hu layout for each cluster. The size of the cluster is determined by the sum of sizes of every node that it contains. The strength of the attraction forces between the clusters is determined by the sum of edges that the nodes in each cluster share between other clusters. This positions the clusters in such a way that they reflect the average relationship of the nodes between the groups through proximity. After positioning the attraction points, forces pull the nodes towards their respective cluster's attraction point, while repulsion forces prevent nodes from overlapping, as shown in Figure 2. When the selected number of clusters is zero, the nodes take up the default position calculated through the Yifan Hu layout.

The same attraction and repulsion forces are applied over the points in the timeline in order to distort it into a time curve. However, the strength of these forces is determined through a similarity matrix. The similarity score between each time point is based on how many nodes share the same behavior of temporal variation between them. For instance, if a large number of nodes have peaks happening at the same time points (where the value increased but is then followed by a decrease), those time points will have a high similarity score. The attraction and repulsion forces are applied simultaneously to every node based on these scores, where similarity is directly proportional to the attraction strength, and inversely proportional to the repulsion strength. As such, repulsion forces assure that two time points only get as close as they are similar. This means that as the time curve stabilizes, the distance between time points should represent their similarity between each other in regards to the variation of values over time.

V. Results

To demonstrate the developed application and visualization models, we utilized a time-series gene expression RNA-Seq dataset from the HIV-1 infection, which measured

expression across 24 hours with intervals of 2 hours, and respective PPI network.

When visualizing the PPI network without time-series data, we were able to analyze the topological proprieties of the network through the Yifan Hu layout and position-based clustering (Figure 2). This allowed for the identification of the proteins with the highest degrees that are central to the protein-protein interaction network, which appear located in the central cluster. However, this only highlights the superficial structure of the network. In order to analyze the behaviors resulting from the HIV-1 infection, the gene expression time-series dataset was loaded into the visualization and the proteins were clustered by temporal variance. By moving through the timeline, it was possible to observe that proteins within the same clusters presented similar gene expression profiles, meaning increases and decreases at the same points in time. This was portrayed as visible increases and decreases in brightness and size for whole groups, which is shown in Figure 3. Proteins that exhibit similar behaviors may contribute to the condition being tested, in this case the HIV-1 infection. Analyzing these behaviors is necessary to build knowledge on the basic molecular mechanisms in cells, used in the development and testing of new treatments.

Furthermore, by bending the timeline into a time curve, as shown in Figure 3, it was possible to further explore patterns of behaviors happening over time. The resulting time curve shows four non-sequential time points (4, 10, 16 and 24 hours) placed close together, and selecting each of these time points shows the same clusters of nodes portraying peaks of gene expression. The state of network of two of these time points is shown in Figure 3, where it is possible to observe in both that the large purple cluster and the two bottom left clusters appear to have a large quantity of brightly colored nodes, indicating peak expression. Additionally, other time points also approach each other, such as 2 and 8 hours, which also share three clusters with peaks of expression. While this requires a more in-depth analysis, these periodic, reoccurring behaviors may be interpreted as waves of expression changes, previously observed in the HIV-1 infection [25], [26].

VI. Conclusion and Future Work

In this paper, we have presented a novel visualization tool that allows the creation of networks from user datasets and compare between its attributes. The developed models provide an interactive environment where data can be easily explored through comprehensible artifacts, which represent multiple variables simultaneously and highlight patterns. This was demonstrated through representation of the sequence of events that describes the HIV-1 infection on human proteins. We have shown that the tool was able to visually sort this user-provided dataset through clustering and force-directed layouts, annotate the genes with data from an internal database. Additionally, the tool provides multiple methods that facilitated the analysis of temporal variation, which were capable of graphically representing known behaviors of the HIV-1 infection over time.

Our future goals include tackling the data dimensionality problem, as current tools still either exhibit performance issues when dealing with datasets of considerable sizes, or cannot represent them comprehensively. Through modern visualization techniques that can reduce visual density, it may be possible to reduce noise while highlighting significant groups or patterns while giving the user more control over the visualization. The developed visualization models were created with flexibility in mind due to the multivariate characteristics of biological datasets, and the ability to be able to visualize and compare these attributes must be improved as we address more complex networks. Refining our models and interface will require periodical user tests with researchers in the field, which will also result in the evaluation of the visualization tool against various other examples of sequence data.

Acknowledgment

The first author is funded by the Fundação para a Ciência e Tecnologia (FCT), Portugal under the grant SFRH/BD/124538/2016.

References

[1] X. Ma and L. Gao, "Biological network analysis: insights into structure and functions," *Briefings in functional genomics*, vol. 11, no. 6, pp. 434–442, 2012.

[2] B. Bach, C. Shi, N. Heulot, T. Madhyastha, T. Grabowski, and P. Dragicevic, "Time curves: Folding time to visualize patterns of temporal evolution in data," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 1, pp. 559–568, 2016.

[3] S. Jain, J. Arrais, N. J. Venkatachari, V. Ayyavoo, and Z. Bar-Joseph, "Reconstructing the temporal progression of hiv-1 immune response pathways," *Bioinformatics*, vol. 32, no. 12, pp. i253–i261, 2016.

[4] A. Cruz, J. P. Arrais, and P. Machado, "Interactive and coordinated visualization approaches for biological data analysis," *Briefings in Bioinformatics*, p. bby019, 2018. [Online]. Available: + http://dx.doi.org/10.1093/bib/bby019

[5] J. Seo and B. Shneiderman, "A rank-by-feature framework for interactive exploration of multidimensional data," *Information Visualization*, vol. 4, no. 2, pp. 96–113, Jul. 2005. [Online]. Available: http://dx.doi.org/10.1057/palgrave.ivs.9500091

[6] J. P. Gonçalves, S. C. Madeira, and A. L. Oliveira, "Biggests: integrated environment for biclustering analysis of time series gene expression data," *BMC research notes*, vol. 2, no. 1, p. 124, 2009. [Online]. Available: https://doi.org/10.1186/1756-0500-2-124

[7] A. Vogogias, J. Kennedy, D. Archambault, V. A. Smith, and H. Currant, "Mlcut: Exploring multi-level cuts in dendrograms for biological data," in *Proceedings of the Conferece on Computer Graphics & Visual Computing*, ser. CGVC '16. Goslar Germany, Germany: Eurographics Association, 2016, pp. 1–8. [Online]. Available: https://doi.org/10.2312/cgvc.20161288

[8] J. Ernst and Z. Bar-Joseph, "Stem: a tool for the analysis of short time series gene expression data," *BMC bioinformatics*, vol. 7, no. 1, p. 191, 2006. [Online]. Available: https://doi.org/10.1186/1471-2105-7-191

[9] M. Meyer, T. Munzner, A. DePace, and H. Pfister, "Multeesum: a tool for comparative spatial and temporal gene expression data," *IEEE transactions on visualization and computer graphics*, vol. 16, no. 6, pp. 908–917, 2010.

[10] M. Meyer, B. Wong, M. Styczynski, T. Munzner, and H. Pfister, "Pathline: A tool for comparative functional genomics," *Computer Graphics Forum*, vol. 29, no. 3, pp. 1043–1052, 2010. [Online]. Available: http://dx.doi.org/10.1111/j.1467-8659.2009.01710.x

[11] B. R. Granger, Y.-C. Chang, Y. Wang, C. DeLisi, D. Segrè, and Z. Hu, "Visualization of metabolic interaction networks in microbial communities using visant 5.0," *PLoS Comput Biol*, vol. 12, no. 4, p. e1004875, 2016.

[12] T. C. Freeman, L. Goldovsky, M. Brosch, S. Van Dongen, P. Mazière, R. J. Grocock, S. Freilich, J. Thornton, and A. J. Enright, "Construction, visualisation, and clustering of transcription networks from microarray expression data," *PLoS Comput Biol*, vol. 3, no. 10, p. e206, 2007.

[13] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker, "Cytoscape: a software environment for integrated models of biomolecular interaction networks," *Genome research*, vol. 13, no. 11, pp. 2498–2504, 2003.

[14] A. Barsky, T. Munzner, J. Gardy, and R. Kincaid, "Cerebral: Visualizing multiple experimental conditions on a graph with biological context," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1253–1260, Nov. 2008. [Online]. Available: http://dx.doi.org/10.1109/TVCG.2008.117

[15] H. Rohn, A. Junker, A. Hartmann, E. Grafahrend-Belau, H. Treutler, M. Klapperstück, T. Czauderna, C. Klukas, and F. Schreiber, "Vanted v2: a framework for systems biology applications," *BMC systems biology*, vol. 6, no. 1, p. 139, 2012. [Online]. Available: https://doi.org/10.1186/1752-0509-6-139

[16] A. Lex, H.-J. Schulz, M. Streit, C. Partl, and D. Schmalstieg, "Visbricks: multiform visualization of large, inhomogeneous data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2291–2300, 2011.

[17] S. van den Elzen, D. Holten, J. Blaas, and J. J. van Wijk, "Reducing snapshots to points: A visual analytics approach to dynamic network exploration," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 1, pp. 1–10, 2016.

[18] G. O. Consortium *et al.*, "The gene ontology project in 2008," *Nucleic acids research*, vol. 36, no. suppl 1, pp. D440–D444, 2008.

[19] E. Polisciuc, P. Cruz, H. Amaro, C. Maçãs, and P. Machado, "Flow map of products transported among warehouses and supermarkets," in *7th International Conference on Information Visualization Theory and Applications*, vol. 2, 2016.

[20] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[21] M. R. Anderberg, "Cluster analysis for applications. monographs and textbooks on probability and mathematical statistics," 1973.

[22] D. Müllner, "Modern hierarchical, agglomerative clustering algorithms," *arXiv preprint arXiv:1109.2378*, 2011.

[23] Y. Hu, "Efficient, high-quality force-directed graph drawing," *Mathematica Journal*, vol. 10, no. 1, pp. 37–71, 2005.

[24] M. Jacomy, T. Venturini, S. Heymann, and M. Bastian, "Forceatlas2, a continuous graph layout algorithm for handy network visualization designed for the gephi software," *PloS one*, vol. 9, no. 6, p. e98679, 2014.

[25] P. Mohammadi, S. Desfarges, I. Bartha, B. Joos, N. Zangger, M. Muoz, H. F. Gnthard, N. Beerenwinkel, A. Telenti, and A. Ciuffi, "24 hours in the life of hiv-1 in a t cell line," *PLOS Pathogens*, vol. 9, no. 1, pp. 1–11, 01 2013. [Online]. Available: https://doi.org/10.1371/journal.ppat.1003161

[26] S. Jain, J. Arrais, N. J. Venkatachari, V. Ayyavoo, and Z. Bar-Joseph, "Reconstructing the temporal progression of hiv-1 immune response pathways," *Bioinformatics*, vol. 32, no. 12, pp. i253–i261, 2016. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/btw254