New features for GRNsight: a web application for visualizing models of small- to medium-scale gene regulatory networks

Kam D. Dahlquist[1], John David N. Dionisio[2], Mihir Samdarshi[1], Nicole A. Anguiano[2], Anindita Varshneya[1], Eileen J. Choe[2], Yeon-Soo Shin[2], Alexia M. Filler[2], John L. Lopez[2], Edward B. Bachoura[2], Justin Kyle T. Torres[2], Kevin B. Patterson[2], Ona O. Igbinedion[2]

[1] Department of Biology, Loyola Marymount University, 1 LMU Drive, Los Angeles, CA 90045 USA. Email: kdahlquist@lmu.edu

[2] Department of Computer Science, Loyola Marymount University, 1 LMU Drive, Los Angeles, CA 90045 USA. Email: dondi@lmu.edu

Project Website: https://dondi.github.io/GRNsight/
Source Code: https://github.com/dondi/GRNsight
License: BSD License

GRNsight is an open source web application for visualizing small- to medium-scale models of gene regulatory networks (GRNs; Dahlquist et al. 2016, https://doi.org/10.7717/peerj-cs.85). It was originally conceived of as a companion application to GRNmap (Gene Regulatory Network Modeling And Parameter estimation), an open source MATLAB software package that uses ordinary differential equations to model the dynamics of medium-scale GRNs (http://kdahlquist.github.io/GRNmap/). GRNmap uses a penalized least squares approach to estimate mRNA production rates, expression thresholds, and regulatory weights for each transcription factor in the network based on user-provided gene expression data and mRNA degradation rates. GRNsight accepts GRNmap- or user-generated Excel workbooks containing an adjacency matrix representation of the GRN, as well as SIF and GraphML files, and automatically lays out the graph of the GRN model. It is written in JavaScript, with diagrams facilitated by D3.js. Node.js and the Express framework handle server-side functions. GRNsight's diagrams are based on D3.js's force graph layout algorithm, which was then extensively customized. GRNsight uses pointed and blunt arrowheads, and colors the edges and adjusts their thicknesses based on the sign (activation or repression) and magnitude of the GRNmap weight parameter.

Since the last presentation of GRNsight at BOSC in 2016, the code has been refactored to the model-view-controller paradigm, a number of new features have been developed to enhance data visualization, and the user interface has been redesigned to accommodate the new feature set. A grid layout for nodes has been added. The viewport is now independent of the underlying bounding box of the graph, and users can resize the viewport, move the graph within it, and zoom in and out. The display of edge weight numerical values can be toggled on and off, and the user can manually set the edge weight normalization factor and gray color threshold to facilitate comparison between graphs. A color blind-friendly palette has been adopted. Nodes can now be colored, displaying a mini "heat map" where each vertical stripe represents a timepoint in the user-inputted expression data. In beta, a backend PostgreSQL database containing four public yeast datasets allows users to color nodes without having to upload their own data. The database is hosted by the Amazon Web Services Relational Database Service from which GRNsight performs dynamic queries in response to user requests. We also created pop-up gene information webpages. When a user right-clicks on a node, a webpage appears that is dynamically populated with data from the JASPAR, NCBI Gene, UniProt, Ensembl, and Saccharomyces Genome Databases. Initially implemented for budding yeast, five other species are available in beta. GRNsight is developed by undergraduates with faculty oversight. The latter two features were prototyped as part of an open source/open science pedagogy in a Biological Databases course and completed as senior capstone projects (e.g., see https://xmlpipedb.cs.lmu.edu/biodb/fall2017/index.php/Main_Page).