BIOINFORMATICS APPLICATIONS NOTE Vol. 26 no. 14 2010, pages 1790–1791
doi:10.1093/bioinformatics/btq274

Systems biology

Advance Access publication May 27, 2010

MODEVO: exploring modularity and evolution of protein interaction networks

Michał Woźniak, Jerzy Tiuryn and Janusz Dutkowski*,†

Institute of Informatics, University of Warsaw, Banacha 2, 02-097 Warsaw, Poland

Associate Editor: Olga Troyanskaya

ABSTRACT

Summary: Interrogating protein complexes and pathways in an evolutionary context provides insights into the formation of the basic functional components of the cell. We developed two independent Cytoscape plugins that can be cooperatively used to map evolving protein interaction networks at the module level. The *APCluster* plugin implements a recent affinity propagation (AP) algorithm for graph clustering and can be applied to decompose networks into coherent modules. The *NetworkEvolution* plugin provides the capability to visualize selected modules in consecutive evolutionary stages.

Availability: The plugins, input data and usage scenarios are freely available from the project web site: http://bioputer.mimuw.edu.pl/modevo. The plugins are also available from the Cytoscape plugin repository.

Contact: janusz@ucsd.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on November 1, 2009; revised on April 25, 2010; accepted on May 21, 2010

1 INTRODUCTION

Molecular systems are composed of many loosely coupled components that together orchestrate higher cellular functions. The emergence of this inherent modularity is only beginning to be elucidated. For example, up-close investigation of selected protein complexes have shown how some of them have been formed through a series of duplication and rewiring events. Such studies have previously focused on complexes of known 3D structure (Pereira-Leal *et al.*, 2006). More recently systematic analysis based on large-scale protein–protein interaction (PPI) data have been performed (Pereira-Leal *et al.*, 2007; Yosef *et al.*, 2009). For the purpose of analyzing conserved modules in PPI networks it seems natural to employ network alignment techniques (Sharan and Ideker, 2006), in particular those which attempt to infer the network snapshots at previous evolutionary stages (Dutkowski and Tiuryn, 2007). In such scenario, one may first identify complexes and pathways highly conserved in multiple species and then infer duplication, deletion and rewiring events that gave rise to the observed modules. Ideally, the entire process should be automated and interactive, allowing the user to easily identify the conserved network regions and seamlessly move between networks of different species to observe the evolutionary changes. For this task we have implemented new software tools and integrated them with the popular Cytoscape platform (Shannon *et al.*, 2003) *via* its plugin interface. Our first plugin allows for interactive comparison of networks across evolutionary stages, which is a unique feature not available in other graph visualization tools (Suderman and Hallett, 2007). The second plugin implements a recent and already widely used affinity propagation (AP) algorithm (Frey and Dueck, 2007), thus extending the set of existing clustering and module detection methods available within Cytoscape.

2 METHODS

Below we describe the main features of the new Cytoscape plugins and suggest possible application scenarios. User manuals, data files and examples are provided on the project web site.

2.1 Network evolution plugin

The *NetworkEvolution* plugin is a tool for interactive comparative analysis of networks across different species. Our primary goal here is to allow the user to walk step-by-step through the process of evolution of a selected network module. This process possibly includes protein duplications, losses and speciations, as well as the emergence or deletion of network links. Parent–child and sibling–sibling relationships between homologous proteins are also taken into account and color-coded to allow visual interpretation. Standard input data is provided by the procedure described in Dutkowski and Tiuryn (2007). The data includes a species tree, protein family trees, experimentally identified interactions and reconstructed ancestral networks.

We now briefly describe a single usage scenario in which a selected module is visualized across many species to display its evolutionary divergence (see also Fig. 1A). After loading the phylogenetic trees and interaction data, we select and display the network of one of the species, hereafter referred to as the source network. In the next step, from the selected interactome we choose a subnetwork (i.e. network module), either manually or by means of automatic graph clustering. Using the phylogenetic tree interface, we identify the interactomes to which the selected module should be mapped, referred to as the target networks. The mapping can be done in both directions: upwards to the predecessor network of the source species or downwards onto the networks of its descendants. The software identifies the corresponding proteins and interactions in the target networks considering the possible speciations, duplications or losses in the protein space and emergence or deletion of interactions. The computed target networks are laid out accordingly to the source network allowing straightforward visual network comparison and spotting of evolutionary changes. Additionally, proteins from one lineage are represented by the same color and direct protein ancestors are provided for each protein through the tool-tip interface. The source and the target interactomes are presented as standard Cytoscape networks enabling the user to take advantage of a wide variety of

*To whom correspondence should be addressed.
†Present address: Departments of Medicine and Bioengineering, University of California, San Diego, 9500 Gilman Drive, La Jolla, CA 92093, USA

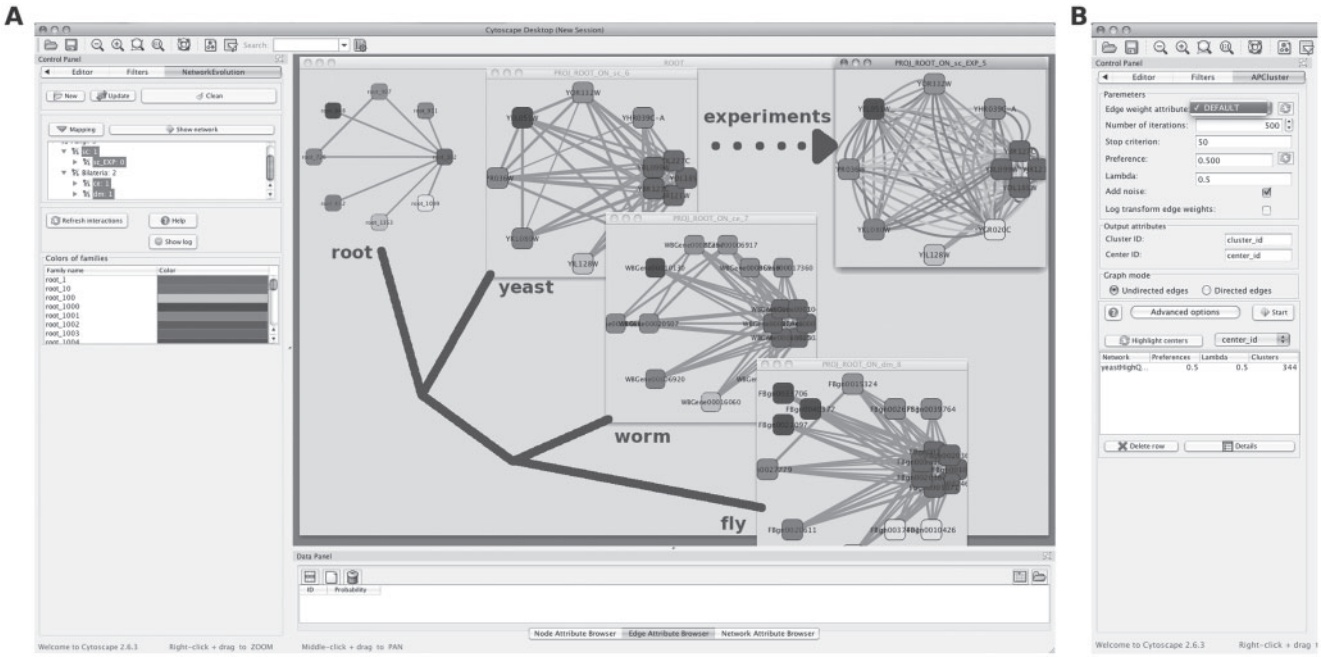

Fig. 1. The main view of the *NetworkEvolution* plugin interface (**A**) and the *APCluster* plugin control panel (**B**) inside Cytoscape. (A) One cluster from the ancestral network (root) identified using the AP algorithm and mapped onto three target species: *Saccharomyces cerevisiae*, *Caenorhabditis elegans* and *Drosophila melanogaster*. For *S.cerevisiae* a separate subnetwork corresponding to experimental data is shown. Note that the phylogenetic tree was added manually for presentation purposes. (B) APCluster interface for adjusting parameters and keeping track of previous runs of the algorithm.

tools to interactively analyse, manipulate and export the results to different data formats.

2.2 AP plugin

The *APCluster* plugin provides a Cytoscape-compliant implementation of the AP algorithm (Frey and Dueck, 2007). The algorithm identifies meaningful clusters by passing messages that encode the affinity of one data point to become an exemplar for another data point. The main benefit of our implementation is that it enables the user to run AP on any network straight from the Cytoscape GUI and immediately visualize the results. *APCluster* also makes it easy to control the parameters of the algorithm by keeping track of parameter settings and results from multiple executions. Due to its speed and ability to cluster a wide range of networks (both weighted and unweighted, as well as directed and undirected), AP presents a valuable alternative to other clustering algorithms available in Cytoscape. Among other possible applications, we found that the algorithm is particularly well suited to perform the first step of the evolutionary analysis described above, i.e. identify coherent modules in biological networks. When tested on our default datasets, AP identified modules that closely match to known protein complexes (see project web site for details), performing favorably to MCL (Enright *et al.*, 2002; implemented as part of the clusterMaker plugin) and MCODE (Bader and Hogue, 2003).

APCluster provides a dedicated tab inside the Cytoscape GUI (Fig. 1B). From there the user can configure the algorithm parameters and choose either the *matrix* or the *siblings* implementation preferable for dense or sparse graphs, respectively. The similarities between data points (edge weights) can be fetched directly from a selected edge attribute and clustering results are saved as node attributes. The last feature enables a seamless integration with the Cytoscape layout mechanism allowing the user to visualize the identified clusters.

ACKNOWLEDGEMENTS

We would like to thank Clement Chung and anonymous reviewers for valuable comments on the *APCluster* plugin.

Funding: Polish Ministry of Science (grants PBZ-MNiI-2/1/2005 and N N301 065236).

Conflict of Interest: none declared.

REFERENCES

Bader,G.D. and Hogue,C.W. (2003) An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, **4**, 2.

Dutkowski,J. and Tiuryn,J. (2007) Identification of functional modules from conserved ancestral protein-protein interactions. *Bioinformatics*, **23**, i149–i158.

Enright,A.J. *et al.* (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res.*, **30**, 1575–1584.

Frey,B.J.J. and Dueck,D. (2007) Clustering by passing messages between data points. *Science*, **15**, 972–976.

Pereira-Leal,J. *et al.* (2006) The origins and evolution of functional modules: lessons from protein complexes. *Phil. Trans. R. Soc. Lond. Ser. B Biol. Sci.*, **361**, 507–517.

Pereira-Leal,J. *et al.* (2007) Evolution of protein complexes by duplication of homomeric interactions. *Genome Biol.*, **3**, R51.

Shannon,P. *et al.* (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.*, **13**, 2498–2504.

Sharan,R. and Ideker,T. (2006) Modeling cellular machinery through biological network comparison. *Nat. Biotechnol.*, **24**, 427–433.

Suderman,M. and Hallett,M. (2007) Tools for visually exploring biological networks. *Bioinformatics*, **23**, 2651–2659.

Yosef,N. *et al.* (2009) A complex-centric view of protein network evolution. *Nucleic Acids Res.*, **37**, e88.