PANI: An Interactive Data-driven Tool for Target Prioritization in Signaling Networks

Huey-Eng Chua[§] Sourav S Bhowmick[§] Lisa Tucker-Kellogg[‡]
Yingqi Wang[§] C F Dewey, Jr[†] Hanry Yu[¶]

[§]School of Computer Engineering, Nanyang Technological University, Singapore
[‡]Mechanobiology Institute, National University of Singapore, Singapore
[¶]Department of Physiology, National University of Singapore, Singapore
[†]Division of Biological Engineering, Massachusetts Institute of Technology, USA
chua0530|assourav|wang0393@ntu.edu.sg, LisaTK|nmiyuh@nus.edu.sg, cfdewey@mit.edu

ABSTRACT

Biological network analysis often aims at the *target identification problem*, which is to predict which molecule to inhibit (or activate) for a disease treatment to achieve optimum efficacy and safety. A related goal, arising from the increasing availability of high-throughput screening (HTS), is to suggest many molecules as potential targets. The *target prioritization problem* is to predict a subset of molecules in a given disease-associated network which is likely to include successful drug targets. Sensitivity analysis prioritizes targets in a dynamic network model according to principled criteria, but fails to penalize off-target effects, and does not scale for large networks. In this demonstration, we present PANI (**P**utative T**A**rget **N**odes Pr**I**oritization), a novel interactive system that addresses these limitations. It prunes and ranks the possible target nodes by exploiting concentration-time profiles and network structure (topological) information and visually display them in the context of the signaling network. Through the interactive user interface, we demonstrate various innovative features of PANI that enhance users' understanding of the prioritized nodes.

Categories and Subject Descriptors

J.3 [**Life and Medical Sciences**]: Biology and genetics.

General Terms

Design, Human Factors, Performance

Keywords

PANI, target prioritization, putative target

1. INTRODUCTION

High-throughput screening (HTS) is frequently used in drug discovery and biological fields, such as genetics. The efficiency of HTS in performing parallelized experiments has led to an increase in customized designs for high-throughput experiments. Designing HTS experiments involve decision on the treatments and controls for the input samples and also the type of measurement required (*e.g.*, which genes to measure and which behaviour to quantify). These decisions are particularly challenging when the signaling network under investigation contains many different molecules. For instance, in the sea urchin endomesoderm gene regulatory network [8], there are 622 nodes, each of them representing either a gene, protein or mRNA in different embryonic territories. Measuring the activity of every node generates data of very high dimensions which make subsequent analysis challenging. *Target prioritization tools* address this issue by identifying a subset of nodes, that are relevant to the biological problem being investigated, for further studies.

In this demonstration, we present a novel data-driven graphical target prioritization system, called PANI (**P**utative T**A**rget **N**odes Pr**I**oritization), which uses network information and simple empirical scores to prioritize and rank biologically relevant target molecules in signaling networks [3]. Given a signaling network associated with a particular disease $H = (V_H, E_H)$ and an *output node* $v_o \in V_H$, it identifies those nodes which when perturbed are able achieve desirable efficacy and safety in terms of regulation of v_o. An *output node* is a protein that is either involved in some biological processes which may be deregulated, resulting in manifestation of a disease, or be of interest due to its potential role in the disease (*e.g.*, ERK in the MAPK-PI3K network [4]).

PANI is a generic system that takes a two-phase approach to identify and rank target molecules. First, it performs target pruning to obtain a set of nodes, denoted as T, for which a path exists from $t \in T$ to v_o. The pruning step reduces the target search space and hence computational cost. Then, PANI calculates the *putative target score* of each node t, which is a weighted rank aggregation of the *profile shape similarity distance* (PSSD) [3], the *target downstream effect* (TDE) [3] and the *bridging centrality* (BC) [5] of t. PSSD identifies the most relevant upstream regulators of v_o and measures the similarity between the concentration-time series profiles (plots of a node's concentration against time) of nodes using a customized distance measure; TDE assesses the potential impact on the network when a node is perturbed based on the probability of perturbing a *downstream node* w and the likelihood of w causing off-target effect. Node w is

IHI'12, January 28–30, 2012, Miami, Florida, USA.

Network $(H = (V_H, E_H))$	$\|V_H\|$	**Execution Time**			$\frac{\|\tau_{\text{MPSA}}\|}{\|\tau_{\text{PANI}}\|}$	$\frac{\|\tau_{\text{SOBOL}}\|}{\|\tau_{\text{PANI}}\|}$
		τ_{PANI}	τ_{MPSA}	τ_{SOBOL}		
MAPK-PI3K network [4]	36	~6sec	~18min	~3hrs	180	1800
MLC phosphorylation network [11]	105	~11sec	~2hr	~21hrs	654.55	6872.73
Endomesoderm network [8]	622	~251sec	-	-	-	-

Table 1: Execution times.

downstream of v if there exists a path from v to w; BC identifies nodes that are located at a connecting bridge between modular subregions in a network [5] and is the product of two ranks, namely, the inverses of betweenness centrality [2] and bridging coefficient [5]. The putative target score is then used to prioritize the nodes.

In order to facilitate the understanding of the prioritized nodes in the context of the signaling network, we provide several innovative interactive features. First, we superimpose the results onto the graphical representation of the network for users to visualize the location of the prioritized nodes within the network. A filtering mechanism allows users to view the top-k prioritized nodes, improving result visualization for larger networks. Second, we provide an interactive display of the signaling network to facilitate study. For instance, *Cluster Mode* provides visualization of the strongly connected components (SCCs) in different colours, giving users a sense of the modular layout of the network. Third, we provide users greater ease in assessing the properties of prioritized nodes by consolidating and categorizing the information in a single platform. For instance, we display their concentration-time series profiles, reactions that they are involved in and relevant web links to external online databases (*e.g.*, *UniProt* [1]) containing additional information. In summary, to the best of our knowledge, PANI is the first target prioritization system to be demonstrated in a conference venue.

2. RELATED SYSTEMS AND NOVELTY

Global sensitivity analysis (GSA)-based techniques, such as multi-parametric sensitivity analysis (MPSA) [14] and SOBOL [13], are frequently proposed for target prioritization. They prioritize nodes using the sensitivity values, which measure the effect of a parameter perturbation (*e.g.*, a kinetic rate constant change) on the output node. Although these GSA-based methods can identify sensitive parameters, they have several limitations. First, they require simulating the network behavior for a combinatorial number of different parameter combinations, making them computationally expensive, especially for larger networks (see Table 1). For instance, these techniques fail to complete the analysis on the endomesoderm network [8] containing 622 nodes on a modern server machine due to memory problem. Second, these methods generally identify parameters resulting in maximum output node perturbation without considering off-target effects. Third, these methods may miss "insensitive" nodes that may be important drug targets, since they only consider one property (sensitivity) in their ranking. For instance, MPSA [14] and SOBOL [13] ignore `Akt` as a target node although active `Akt` can inhibit activation of `ERK` in differentiated myotubes via `Raf-Akt` interaction [12].

PANI is designed to address the aforementioned limitations. The key differences between PANI and GSA-based approaches are as follows. PANI ranks the nodes by computing an *aggregate score* that is based on certain structural *and* kinetic properties of the network, instead of using sensitivity and focussing *solely* on the kinetic aspect of the network. Furthermore, unlike PANI, the GSA-based approaches do not use any pruning technique to filter out "irrelevant" nodes and reduce unnecessary computational cost. Indeed, PANI has been demonstrated to prioritize a majority of known drug targets in the MAPK-PI3K network [3]. Lastly, the PANI system provides interactive features such as *Cluster Mode* and superimposition of prioritization results on the signaling network for users to learn more about the characteristics of the network and the role of the prioritized nodes in the network.

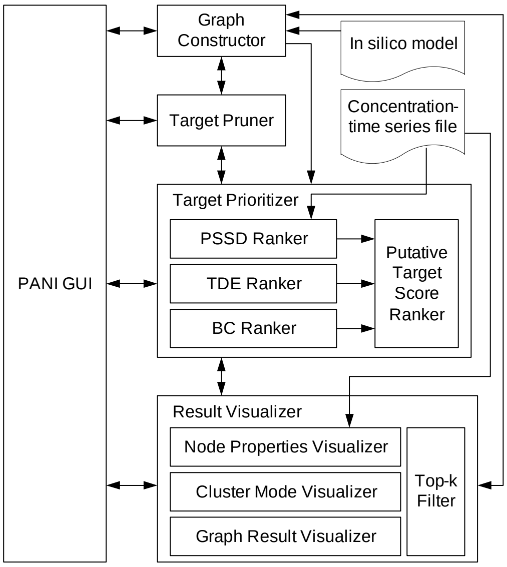

Figure 1: Architecture of PANI.

3. SYSTEM OVERVIEW

The PANI system is implemented in Java JDK 1.6 using open-source libraries JFreeChart, libSBML, JUNG and FastDTW. The reader may refer to details in [3]. Figure 1 shows the architecture of PANI which consists of the following modules.

The PANI GUI Module: The GUI of PANI (Figure 2a) coordinates various modules of PANI and consists of five panels. In order to perform the target prioritization, the user has to select the input files containing the network model and the concentration-time series profiles using the toolbar (Figure 2a, Panel 1). After the files are read using the libSBML library, the *Graph Constructor* constructs the graphical representations of the model (model graph) using the JUNG library. The *Result Visualizer* displays the list of nodes in the network in the *Information Panel* (Figure 2a, Panel 2), the model graph in the *Satellite Graph View Panel* (Panel 3), and supports interaction with the graph through the *Interactive Graph View Panel* (Panel 4). Specifically, the *Satellite Graph View Panel* displays the overall layout of the graph while the *Interactive Graph View Panel* supports interaction with the graph, such as zooming in or out at a specific part of the graph. After the user has selected the output node from the *Information Panel*, the prioritization process can be initiated from the toolbar. The prioritization result is displayed in both Panel 2 as a ranked list of nodes and Panel 4 as nodes of varying size (larger nodes

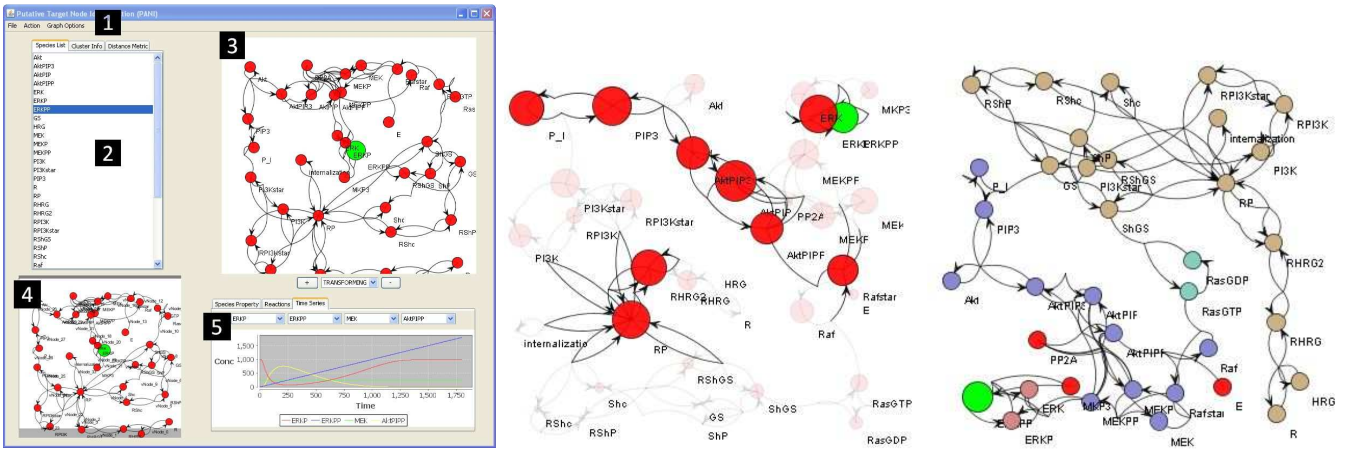

Figure 2: (a) Graphical user interface of PANI, (b) Top-10 prioritization results superimposed on the `MAPK-PI3K` network [4] with double phosphorylated `Erk` as output node, and (c) *Cluster Mode* view of the network.

have higher ranks) (Figure 2b). The user can also view additional node properties (discussed later) of selected nodes in the *Node Properties Visualizer* (Panel 5).

The Graph Constructor Module: The *Graph Constructor* creates graphical representations of the user-selected signaling model which are subsequently used for visualization and analysis. The signaling model can be represented as a directed hypergraph [7] where the nodes and hyperedges symbolize molecules (*e.g.*, proteins) and interactions, respectively. Analysis of directed hypergraphs is generally more complex than graphs and many graph algorithms cannot be used directly on hypergraphs [7]. Hence, they are often transformed into an equivalent bipartite or substrate digraph for analysis [7]. We use the bipartite digraph representation since it retains the original information of the hypergraph. Signaling networks generally contain strongly connected components (SCC) due to the existence of feedback loops that are common in complex regulatory control [9]. The bipartite digraph can be further simplified into a directed acyclic graph (DAG) representation by collapsing these SCCs into meta nodes. The *Graph Constructor* creates three different graphical representation of the biological signaling model, namely, the directed hypergraph, the bipartite digraph and the DAG. The directed hypergraph is used for the display of the model in the *Satellite Graph View* and *Interactive Graph View Panels*; the bipartite digraph is for processing of node properties such as TDE and BC using graph algorithms in the *Target Prioritizer* module; and the DAG is for analyzing reachability information of node pairs in the *Target Pruner* module.

The Target Pruner Module: The *Target Pruner* handles the pruning of irrelevant candidate nodes. The relevancy of the candidate nodes is determined based on a reachability rule which is the existence of a path from the candidate node to the user-selected output node. The *Target Pruner* computes the reachability information using the DAG. It first indexes the DAG using depth-first traversal, then uses the index to determine the reachability of each candidate node to the output node. Nodes that can reach the output node are retained for subsequent analysis.

The Target Prioritizer Module: The *Target Prioritizer* computes the *putative target score* for prioritizing the nodes using several submodules.

The PSSD *Ranker Module:* This module computes the PSSD and uses it to rank the nodes with reference to the output node. The PSSD of node u with respect to the output node v_o, denoted as $\Phi_{(u,v_o)}$, is the minimum dynamic time warping distance (DTW) [6] of two pairs of concentration-time series profiles, namely, (ζ_u, ζ_{v_o}) and (ζ'_u, ζ_{v_o}). Hence, $\Phi_{(u,v_o)}{=}Min(\text{DTW}(\zeta_u, \zeta_{v_o}), \text{DTW}(\zeta'_u, \zeta_{v_o}))$ where ζ_u and ζ'_u are the original and inverted concentration-time profiles of node u, respectively [3]. We use the FastDTW library to compute the DTW value.

The TDE *Ranker Module:* This module computes the TDE and uses it to rank the nodes. The TDE of node u, denoted as Υ_u, is the sum of the effect of each of its downstream node w, which in turn is measured by the product of w's degree and the probability of perturbing w. Hence, $\Upsilon_u = \sum_{w \in W}(\rho_{u,w} \times \theta_w)$ where $\rho_{u,w}$ is the probability of perturbing $w \in W$ when target node u is perturbed and θ_w is the degree of w [3].

The BC *Ranker Module:* This module computes the BC (bridging centrality) and uses it to rank the nodes. The BC of node u, denoted as Λ_u, is the product of two ranks, namely, the inverses of betweenness centrality [2] and bridging coefficient [5]. Hence, $\Lambda_u = \Psi_{\frac{1}{\Gamma:u}} \times \Psi_{\frac{1}{\Omega:u}}$ where $\Psi_{\frac{1}{\Gamma:u}}$ and $\Psi_{\frac{1}{\Omega:u}}$ are the inverse rank of betweenness centrality and bridging coefficient, respectively [5].

The Putative Target Score Ranker Module: This module computes the putative target score as a weighted sum of the ranks of the PSSD, the TDE and the inverse of BC obtained from the aforementioned submodules. Specifically, the putative target score of node u is $score_u = \sum_{c \in C}(\omega_c \times \Psi_{c:u})$ where C is the set of node properties $\{\Phi_{v_o}, \Upsilon, \frac{1}{\Lambda}\}$; $\Psi_{c:u}$ is the rank of u based on property $c \in C$; ω_c is the weight of property $c \in C$ and $\sum_{c \in C} \omega_c = 1$. The computed putative target score is then used to prioritize the nodes and the results are formatted for display by the *Result Visualizer*.

The Result Visualizer Module: The *Result Visualizer* handles the visualization aspect of PANI, such as the display of the graph model, using several submodules.

The Cluster Mode Visualizer Module: This module handles the modular layout display of the graph model which is provided as a viewing option in the toolbar. It assesses the SCCs information from the *Graph Constructor* and then assigns nodes in the same SCC to the same color (Figure 2c).

The Graph Result Visualizer Module: This module superimposes the prioritization result on the graph model by assessing the prioritization results from the *Target Prioritizer* and then assigning node sizes based on their prioritized ranks. Figure 2b shows the superimposed prioritization results specific for double phosphorylated ERK on the MAPK-PI3K signaling network [4]. Nodes having larger radius are prioritized over those with smaller radius.

The Top-k Filter Module: This module facilitates the display of the top-k results. The k value can be specified using the toolbar and the module increases the transparency of the nodes outside this top-k set to de-emphasize them in the *Interactive Graph View Panel* (Figure 2b), allowing the user to view the results with increased clarity.

The Node Properties Visualizer Module: This module handles the visualization of the node properties such as the concentration-time series profiles of user-selected nodes, the roles of the nodes in reactions in the signaling network and the web links for external databases containing additional node information. The concentration-time series profiles are plotted using the JFreeChart library. The role of the nodes can be obtained from their edges which are stored in the *Graph Constructor*. For instance, an incoming edge to a node indicates that the node is a product of the reaction represented by that edge. We make use of the annotations in the signaling network model (*e.g.*, *UniProt* ID) to interface with external databases (*e.g.*, *UniProt* [1]).

4. DEMONSTRATION OBJECTIVES

Our demonstration will be loaded with several signaling network models (*e.g.*, the MAPK-PI3K network [4], myosin light chain (MLC) phosphorylation [11], endomesoderm network [8]) of different sizes obtained from the *Biomodels.net* database [10] and the concentration-time series profiles of the nodes in these models. The concentration-time series profiles are generated by simulating these models using *Copasi*. We shall use these models to interactively demonstrate the process of target prioritization, understand the characteristics of the network, and the roles the prioritized targets play in the network. A demonstration video is available at `http://www.youtube.com/watch?v=UjMBe9FWvvI`.

Interactive target prioritization process. The main goal of the demonstration is for the audience to experience the process of fast discovery of superior quality putative target nodes in signaling networks. Using the PANI GUI, we will demonstrate target prioritization specific to a user-selected output node and how the interactive features of PANI such as the *Top-k Filter* can be used to view the prioritization results (Figure 2b). The users can use the superimposition feature of the prioritization results on the graph model to visualize the relationship of the target nodes in the network (*e.g.*, the distance between the nodes and the position of the nodes in the network). Additionally, the user will be able to experience the fast response time of PANI to perform target prioritization for networks of different sizes. Users can also load their own models described in the SBML format. The *Copasi* tool will be provided to assist the users in generating the concentration-time series profiles.

Understanding the modular layout of the network. We shall exploit the *Cluster Mode* to display the modular layout of the network (Figure 2c). Users can visualize the structure of the network easily as the SCCs are presented in different colors. When complemented with the prioritization results, users can enhanced their understanding of the roles of the target nodes in the network. For example, nodes in a particular module may be favored as targets for a particular network, suggesting that this module may play a critical role, and suggesting further experiments on the module.

Understanding properties of prioritized nodes. We shall demonstrate the use of the interactive features provided by the *Node Properties Visualizer* to understand properties of user-selected nodes (Panel 5 in Figure 2a). Specifically, we will show how the users can view the dynamic relationship between different nodes using the concentration-time profiles comparison charts; learn about roles of the nodes in network (*e.g.*, the reactions they are involved in and the roles they play in the reactions); and find out additional properties using web links to external online databases.

5. ACKNOWLEDGMENTS

The authors are supported by grant from the Singapore-MIT Alliance Programme in Computational and Systems Biology.

6. REFERENCES

[1] R. Apweiler et al. Uniprot: the universal protein knowledgebase. *Nucleic Acids Research*, 32(suppl 1):D115–D119, 2004.

[2] U. Brandes. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25:163–177, 2001.

[3] H. Chua et al. Pani: A novel algorithm for fast discovery of putative target nodes in signaling networks. In *ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, 2011.

[4] M. Hatakeyama et al. A computational model on the modulation of mitogen-activated protein kinase (MAPK) and Akt pathways in heregulin-induced ErbB signalling. *Biochem J*, 373(Pt 2):451–463, Jul 2003.

[5] W.-C. Hwang et al. Identification of information flow-modulating drug targets: a novel bridging paradigm for drug discovery. *Clin Pharmacol Ther*, 84(5):563–572, Nov 2008.

[6] E. Keogh et al. Derivative dynamic time warping. In *In First SIAM International Conference on Data Mining (SDMŠ2001*, 2001.

[7] S. Klamt et al. Hypergraphs and cellular networks. *PLoS Comput Biol*, 5(5):e1000385, May 2009.

[8] C. Kuhn et al. Monte carlo analysis of an ode model of the sea urchin endomesoderm network. *BMC Systems Biology*, 3(1):83, 2009.

[9] Y.-K. Kwon et al. Coherent coupling of feedback loops: a design principle of cell signaling networks. *Bioinformatics*, 24(17):1926–1932, Sep 2008.

[10] N. Le Novère et al. Biomodels database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Res*, 34(Database issue):D689–D691, Jan 2006.

[11] A. Maeda et al. Ca2+-independent phospholipase a2-dependent sustained rho-kinase activation exhibits all-or-none response. *Genes to Cells*, 11:1071–1083, 2006.

[12] C. Rommel et al. Differentiation stage-specific inhibition of the raf-mek-erk pathway by akt. *Science*, 286(5445):1738–1741, Nov 1999.

[13] I. Sobolá. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Math. Comput. Simul.*, 55(1-3):271–280, 2001.

[14] Z. Zi et al. In silico identification of the key components and steps in IFN-gamma induced JAK-STAT signaling pathway. *FEBS Lett*, 579(5):1101–1108, Feb 2005.