Integrated Graph Propagation and Optimization with Biological Applications

Krithika Krishnan[1], Tiange Shi[2]
[1]*Institute of Artificial Intelligence and Data Science*
[2]*Department of Biostatistics*
University at Buffalo
Buffalo, NY 14214, USA

Han Yu[3]
Department of Biostatistics and Bioinformatics
Roswell Park Comprehensive Cancer Center
Buffalo, NY 14263, USA

Rachael Hageman Blair[1,2,*]
[*]*Corresponding Author*
[1]*Institute of Artificial Intelligence and Data Science*
[2]*Department of Biostatistics*
University at Buffalo
Buffalo, NY 14224, USA
E-mail: hageman@buffalo.edu

Mathematical models that utilize network representations have proven to be valuable tools for investigating biological systems. Often dynamic models are not feasible due to their complex functional forms that rely on unknown rate parameters. Network propagation has been shown to accurately capture the sensitivity of nodes to changes in other nodes; without the need for dynamic systems and parameter estimation. Node sensitivity measures rely solely on network structure and encode a sensitivity matrix that serves as a good approximation to the Jacobian matrix. The use of a propagation-based sensitivity matrix as a Jacobian has important implications for network optimization. This work develops Integrated Graph Propagation and OptimizatioN (IGPON), which aims to identify optimal perturbation patterns that can drive networks to desired target states. IGPON embeds propagation into an objective function that aims to minimize the distance between a current observed state and a target state. Optimization is performed using Broyden's method with the propagation-based sensitivity matrix as the Jacobian. IGPON is applied to simulated random networks, DREAM4 *in silico* networks, and over-represented pathways from STAT6 knockout data and YBX1 knockdown data. Results demonstrate that IGPON is an effective way to optimize directed and undirected networks that are robust to uncertainty in the network structure.

Keywords: Network, propagation, optimization, Broyden's method, iterative methods

1. Introduction

Network analysis remains a cornerstone of systems biology that has been widely used to examine gene regulation, protein-protein interaction and metabolic systems. Mathematical representations of biological systems often depend on complex nonlinear functions that are not fully understood and lack the dynamic data to fully parameterize. These systems can be examined at steady-state, which reduces the model to a linear system. In applications, a common objective is the inference of a network structure that captures the complex biological relationship between variables. Although structure provides insights into the direct and indirect relationships in a network, it represents a premature endpoint in an analysis.

Network propagation describes the process of absorbing information into a network and propagating it through the network to update node states. Propagation can be used to initiate information flow through a graph, and thus has the potential for prediction. In the field of systems biology, this can be viewed as an *in silico* experiment within a biological network. Although propagation has been broadly used in other fields, applications in systems biology are limited. The PRIoritizatioN and Complex Elucidation (PRINCE) algorithm[1] was one of the first studies to associate network modules with disease through network propagation. The PRINCE algorithm has been used to connect nodes in a graph representing biological variables, such as proteins or genes, with disease.[1] The iterative procedure generates prioritization scores for vertices related to various diseases of interest obtained through graph propagation.

Recently, DYNamics-Agnostic Network MOdels (DYNAMO)[2] was developed to connect the ideas of propagation to the problem of characterizing perturbation patterns in a biological system. The major finding was that a sensitivity matrix derived from propagation solely on the structure of the network effectively captured the Jacobian matrix of partial derivatives for biological systems. In other words, the sensitivity matrix captures the effects of small perturbations on individual nodes in the network. In most biological applications and databases, only the network structure is known, without analytical forms of the biochemical reactions or kinetic rate parameters. Thus, in practice, the Jacobian is difficult or impossible to obtain. The performance of DYNAMO was benchmarked on a database of 120 BioModels representing different biochemical networks and model organisms. Propagation also outperformed alternative approximations based on network measures such as distance and neighborhoods.

The ability to estimate a sensitivity matrix from propagation on the structure has important implications for network optimization, which to the authors' knowledge has not been explored. Coupling network optimization with a sensitivity matrix enables the identification of optimal perturbations that will drive a system to the desired state, providing insight into Biological Engineering and identifying optimal targets for drug therapy and interventions. This work develops the first optimization framework that leverages the sensitivity matrix to identify optimal perturbation patterns to drive a network to a target steady-state. A novel approach, Integrated Graph Propagation and OptimizatioN (IGPON), is developed, which casts the problem as an unconstrained optimization that minimizes the difference between a current network state and a desired target network state.

A distinguishing feature of this method is that the optimization relies on using two primary ingredients: a parameterized network structure and target node states. Thus, IGPON bypasses the need for complex forms of biochemical reactions and derivatives. In contrast, node states are defined iteratively through the PRINCE algorithm.[1] Optimization utilizes Broyden's method,[3] a quasi-Newton method that does not require functional forms of the objective function. It leverages a network-derived sensitivity matrix to represent the Jacobian. The output of IGPON is the prediction of an optimal perturbation that will drive the network to the desired state. IGPON is applied to simulated networks, DREAM4[4] *in silico* networks and over-represented pathways from STAT6 knockdown data and YBX1 knockdown data[5]. Results demonstrate IGPON as an effective way to optimize directed and undirected networks that are also robust to noise in the sensitivity matrix that reflects potential misspecification in the structure.

2. Methods

2.1. *Graph propagation*

A network (graph), G, is defined by a set of nodes (vertices), V, and edges, E, that connect them. Mathematically, undirected graphs can be represented by a symmetric binary adjacency matrix with entries $g_{i,j} = 1$ when there is an edge between vertices v_i and v_j. Directed graphs are binary matrices with $g_{i,j} = 1$ if there is a directed edge between v_i and v_j. This work utilizes propagation through graphs using the PRINCE algorithm,[2] which is used to obtain influence scores for each node.

Let, $F^t \in \mathbb{R}^n$, be the updated vector of n node scores at iteration t. Let $D \in \mathbb{R}^{n\times n}$ be a diagonal matrix with entries, $d(i,i)$, that correspond to the sum of the absolute values of the i^{th} row of G. The normalized propagation weights are given as $G' = D^{-1/2}GD^{-1/2}$. The influence score at iteration t is given as $F^t := \alpha G'F^{t-1} + (1-\alpha)\cdot Y$, where α is a diffusion constant that score enforces smoothness over the network, and Y is an initial set of scores, F^0. We define the sensitivity matrix, $S \in \mathbb{R}^{n\times n}$, which captures a node's influence on other nodes in the network. The rows of the sensitivity matrix are computed by systematically setting each node to 1, and the other nodes to 0, and propagating through the network. Notably, this iterative approach to estimating the sensitivity matrix through propagation has been shown to converge to the closed form.[6] However, it has the added advantage of scalability to large networks. Whereas the closed form sensitivity calculation requires large matrix inversions, which can be infeasible or unstable.[2]

2.2. *Unconstrained optimization*

We define $F(x)$ as a system of m non-linear algebraic equations, $\{f_1(x), f_2(x), \ldots, f_m(x)\}$, in n variables, $x = \{x_1, x_2, \ldots x_n\}$. The objective is to solve the linear system: $F(x) = Ax - b = 0$, where A is the Jacobian matrix of $F(x)$. Broyden's method is an iterative quasi-Newton method for solving a nonlinear equation that can be used as an alternative to Newton's method when the Jacobian is expensive to compute, or unavailable.[3,7] In our case, quasi-Newton methods are required because both the Jacobian and the functional form of the system of nonlinear equations are not known. In contrast to a graph modeled by a well-defined system of nonlinear equations, our system is defined through graph structure and propagation. Let the initial Jacobian, $A_0 \in \mathbb{R}^{n\times n}$, be defined as the sensitivity matrix defined in Section 2.1. Let A_k be the Jacobian approximation at iteration k and let $s_k = x_{k+1} - x_k$. Then, the updated Jacobian approximation A_{k+1} must satisfy the secant equation: $A_{k+1}s_k = F(x_{k+1}) - F(x_k)$. Broyden's method generates subsequent matrices using the update formula:[3]

$$A_{k+1} = A_k + \frac{(y_k - A_k s_k)s_k^T}{{s_k}^T s_k},$$

where $y_k = F(x_{k+1}) - F(x_k)$. Broyden's method is described in Algorithm 2.2.

```
Initialize: $F : \mathbb{R}^n \rightarrow \mathbb{R}^n, x_0 \in \mathbb{R}^n, A_0 \in \mathbb{R}^{n\times n}$
for $k = 1, 2, \ldots \max$ do
    Solve $A_k s_k = -F(x_k)$ for $s_k$
    $x_{k+1} := x_k + s_k$
    $y_k := F(x_{k+1}) - F(x_k)$
    $A_{k+1} := A_k + \frac{(y_k - A_k s_k)s_k^T}{{s_k}^T s_k}$
end for
Output: $x_k$
```

2.3. *Integrated Graph Propagation and Optimization*

Integrated Graph Propagation and OptimizatioN (IGPON) is our approach to integrating propagation (Section 2.1) into optimization (Section 2.2) for the purpose of driving a graph to an optimal target state. A schematic describing IGPON is shown in Figure 1 for a simple ten node graph. The network structure, G, can be directed or undirected and contains n nodes. The structure is assumed to be known *a priori* as either inferred from data or specified by an expert or database (Figure 1A). We define the propagation function, $\Phi(\cdot)$, as the iterative PRINCE algorithm. The sensitivity matrix[2] plays the role of the initial Jacobian, A_0, and is estimated directly using graph propagation, $\Phi(G)$, as described in Section 2.1 (Figure 1B). Let $F^0 \in \mathbf{R}^{n\times 1}$ denote an observed network that we want to drive to a target state, $F^T \in \mathbf{R}^{n\times 1}$. The observed steady-state of the nodes F^0 is assumed to result from the propagation of an unobserved underlying state, x^0, through the network (Figure 1C). Our objective is to identify the underlying perturbation to this initial state, $F^0 + \Delta = x^T$, such that $\Phi\left(F^0 + \Delta\right) = \Phi\left(x^T\right) = F^T$ (Figure 1C). The unconstrained minimization problem is defined as:

$$\min_x \|\Phi(x) - F^T\|_2.$$

This objective can be embedded into an unconstrained optimization problem and solved with Broyden's method (Algorithm 1). However, in this setting, the objective function is not defined in functional form, but rather defines a set of states approximated at every iteration through propagation. Specifically, we define the state of the network through $\Phi(x) = F$. The details of IGPON are outlined in Algorithm 1.

2.4. *Applications to simulations and biological pathways*

Simulation: IGPON was applied to both simulated random graphs and data from the DREAM4 *in silico* challenge.[4] Random graphs were generated with 50 and 150 nodes us-

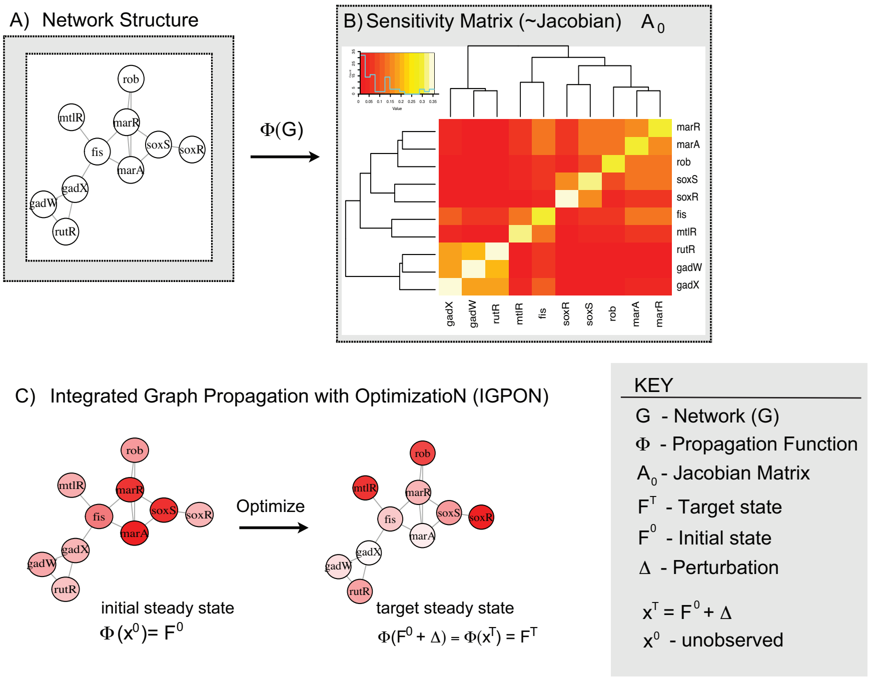

Fig. 1. A schematic of the integrated graph propagation and optimization with biological applications (IGPON) method. **(A)** The structure of the network (graph), G, is assumed to be given. **(B)** The sensitivity matrix derived through graph propagation, $\Phi(G)$, on the network structure, serves as the initial Jacobian, A_0. **(C)** IGPON drives an observed initial steady-state of the network, F^0, to a target steady-state, F^T, through the identification of an optimal perturbation, Δ, such that $\Phi(F^0 + \Delta) = F^T$.

Algorithm 1 Integrated Graph Propagation and Optimization (IGPON)

```
Initialize: $A_0 \in \mathrm{I\!R}^{n \times n}, x_0 = F^0 \in \mathrm{I\!R}^n, F^T \in \mathrm{I\!R}^n$
for $k = 1, 2, \dots \max$ do
    Solve $A_k s_k = -\Phi(x_k)$ for $s_k$
    $x_{k+1} := x_k + s_k$
    Propagate $F_{k+1} = \Phi(x_{k+1})$
    $y_k := F_{k+1} - F_k$
    $A_{k+1} := A_k + \frac{(y_k - A_k s_k) s_k^T}{{s_k}^T s_k}$
end for
Output: $\hat{x}^T = x_k$, $\hat{F}^T = F_k$
```

ing the Barabasi-Albert model[8] implemented in the `igraph` package.[9] The probability of an edge between two arbitrary vertices was set at $p = 0.10$. The DREAM4 data are derived from biological networks and are used as a benchmark in the community.[4] DREAM networks that are 10 nodes and 98 nodes were considered. The 98 node graph was derived from the DREAM4 100 node graph after the removal of two unconnected nodes.

The experimental setup was identical for simulated random graphs and the DREAM4 networks. For each graph, the values of target variable were drawn from a uniform distribution

$x^T \sim \mathbf{U}[0, 1]$. This variable was propagated through the graph to obtain the target state, $\Phi(x^T) = F^T$. The values x^T and F^T are what we are seeking to estimate using IGPON (Figure 1). Initial estimates of the sensitivity matrix, A_0, were obtained as described in Section 2.1. A random initialization was generated, $x_0 \sim \mathbf{U}[0, 1]$, and propagated to obtain $\Phi(x_0) = F_0$. IGPON was applied until convergence $\|F^T - \hat{F}_k\|_2 < 10^{-6}$ and $\|x^T - \hat{x}_k\|_2 < 10^{-6}$. Convergence of individual nodes, x_i, was also assessed using relative error: $F_{err}(i) = \frac{|F^T(i) - F_k(i)|}{F^T(i)}$. In order to examine how robust IGPON is to misspecification in the network structure, we systematically added white noise (10% – 50%) to the initial sensitivity matrix. A total of 100 graphs were generated for each experimental condition.

Biological Pathways: Gene expression data was utilized from the knockTF database for two different sets of experimental conditions.[5] Knockout data for transcription factor signal transducer and activator of transcription 6 (STAT6) was extracted from the database.[10] The knockout was reported to significantly alter pathways related to IL4/interleukin-4- and IL3/interleukin-3-mediated signaling, and apoptotic activity. The gene expression data contained wild-type controls ($N = 27$) and STAT6 knockout ($N = 27$).[10] The mean gene expression data used in this study was taken from the Gene Expression Omnibus[10,11] accession GSE17851, and our focus was the downstream IL-17 signaling pathway in KEGG,[12] which was reported as significant in the pathway enrichment analysis. Data related to the pro-oncogenic transcription factor YBX1 was also extracted from the database. Briefly, YBX1 is an RNA-binding protein involved in many important signaling pathways and associated with the occurrence and development of numerous cancers. Our focus was restricted to the Hedgehog (HH) pathway and P53 pathway from the KEGG database,[12] which were two downstream pathways over-represented in pathway enrichment analysis reported in the database. The HH signaling pathway is shown to be closely related to the development of tumor cells.[13] The P53 signaling pathway plays an important role in tumor suppression.[14] The data included several different breast cancer cell lines with both normal cell types ($N = 24$) and YBX1 knockdown ($N = 24$).[15]

KEGG identifiers from these pathways were mapped to the data and KEGGgraph[16] was used to construct the graphs in the R programming environment. The nodes that were unconnected were eliminated. The HH pathway contained 52 genes and 162 edges, the P53 signaling pathway contained 62 edges and 75 edges and the IL-17 pathway contained 53 genes and 147 edges. These subgraphs were used in connection with the IGPON algorithm. Both directed and undirected versions of the graph were utilized. The undirected graphs were derived using igraph[9] conversion tools.

Each of the subgraphs was parameterized with the gene expression data to create two graphs with the same structure, one for the treated (knockout/ knockdown), and one for the controls. The objective was to use the IGPON algorithm to drive the states of the graph, F^0, to the states of the target graph, F^T. Without loss of generality, we assume the target graph states correspond to the knockout or knockdown data, and the initially observed graph is parameterized by the controls. Note that the selection of initial and target was arbitrary and either set of states could play the role of the target. Sets of minimum driver node set (MDS)[17] were also estimated from the graph structures as one of the following; critical (if that node must always be controlled to control the system), redundant (never required for control), or intermittent (if it is a driver node in some control configurations, but not in others).

3. Results

IGPON was tested on simulations of random graphs, DREAM4 networks[4] and using data from a knockout database.[5] In each simulation, the objective was to use IGPON to drive the network to a target state. The number of iterations for the optimization varied according to graph size, noise and complexity, but the number of iterations needed for the network propagation required for the objective function was kept constant at 500, which was sufficient for all cases considered. Overall, the results were found to be rather insensitive to the parameter α, which controls the relative importance of prior information in the graph, which supports previous findings.[1]

In the simulations of scale-free graphs and the DREAM4 networks, IGPON was able to drive all simulations to their target states (Figure 2). Note that since, $\hat{F}^T = \Phi(\hat{x}^T) = \Phi(F^0 + \hat{\Delta})$, we expect these error profiles to be correlated, which indeed they are for all simulations. IGPON was also observed to be robust to up to 50% noise in the initial Jacobian (Figure 2). With no noise applied to the Jacobian, the graphs converge within only a few iterations (Figure 2 A-D) in Figure 2. On the other hand, as the percentage of white noise is increased from 10%, 25% to 50%, the iterations needed to bring the graph to the target state naturally increases. In addition to noise levels, convergence is also clearly a function of graph size (Figure 2). For example, nearly three times the number of iterations are needed to push a larger graph, such as the simulated $N = 150$ nodes, to its target state when the noise level was increased from 25% (Figure 2H) to 50% (Figure 2L).

Individual node convergence profiles were also examined. Figure 3 shows the relative difference between the target for a node $F^T(i)$ and its estimated state $\hat{F}^T(i)$ for our simulation with 50 nodes. The random initialization is relatively close to the target state by nature of the parameters used for the uniform distribution (Figure 3A). However, as IGPON proceeds, the nodes move further away from their targets (Figure 3B). Some nodes more actively move around and take longer to settle (Figure 3B-D). In fact, many nodes begin to converge to their target (Figure 3C) before again moving further away from the target (Figure 3D), and finally converging (Figure 3B-D). This demonstrates the push and pull of node state values gained through the propagation that are ultimately required to drive the graph to the target. There does not appear to be any clear association between the node trends and graph properties such as degree, and clustering coefficients (data not shown). Similar patterns and trends were observed for graphs of various sizes in the simulations.

IGPON was also used to drive expression profiles to targets in the HH, IL-17, and p53 pathways. In both directed and undirected representations, convergence was achieved across all noise levels (Table 1). As noise levels increased, more iterations were required to achieve convergence. It is also clear that the directed graphs achieve faster convergence across the board. Upon further investigation, there are substantial differences in the MDS node characterizations[17] in the directed and undirected representations. In the IL-17 directed pathway, 18 of the nodes were identified as critical, 10 were intermittent, and the remaining were redundant. In the undirected representation of the IL-17 pathway, only two nodes were identified

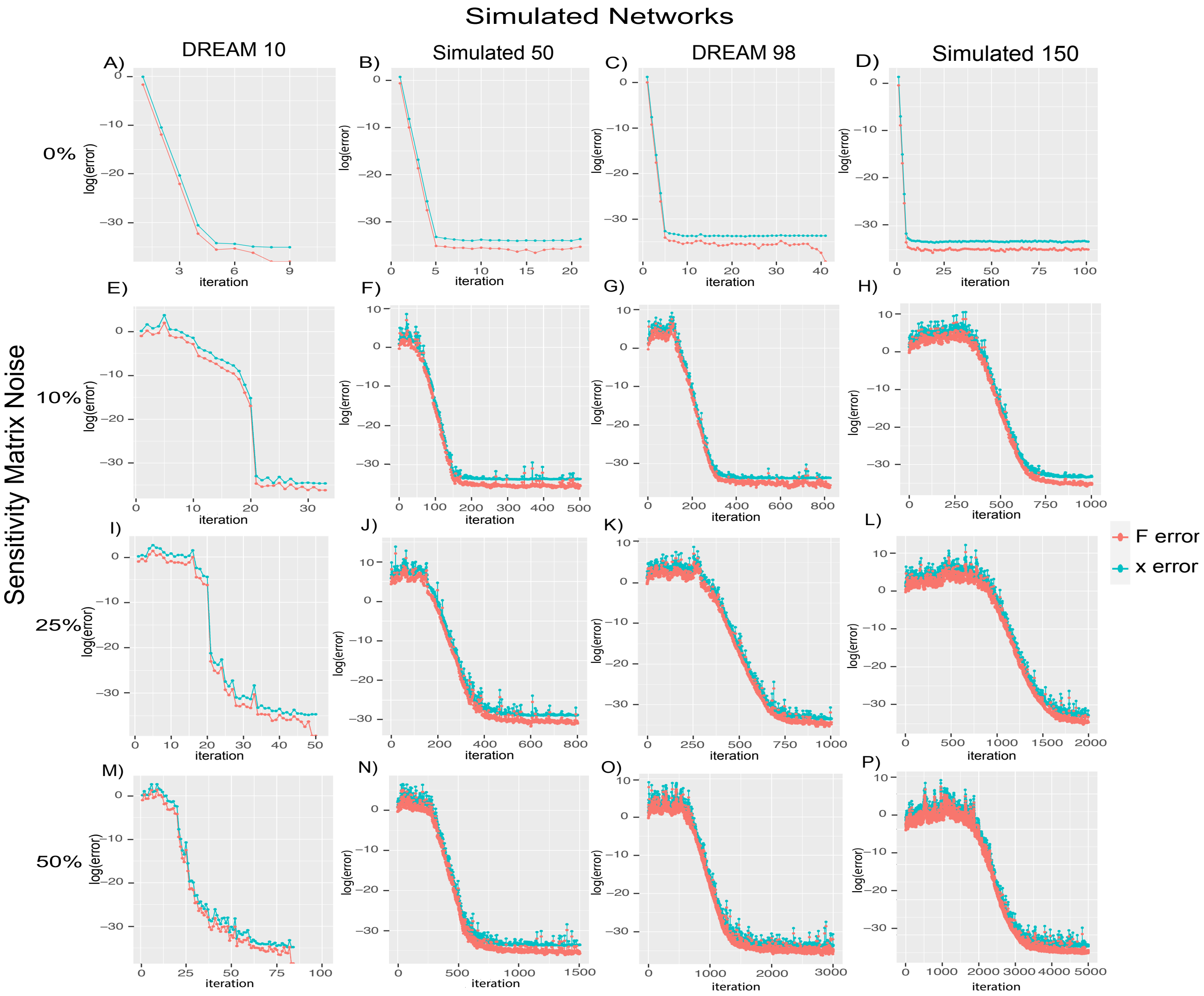

Fig. 2. Convergence profiles of the log(error) for F (coral) and x (blue). Simulated graphs are ordered according to size (columns): columns 1 ($N = 10$), column 2 ($N = 50$), column 3 ($N = 98$), and column 4 ($N = 150$). The rows represent the noise level added to the sensitivity matrix in the optimization. **(A-D)** No noise is added **(E-H)** 10%, **(I-L)** 25% and **(M-P)** 50%.

as critical, 31 were intermittent, and the remaining were redundant. This trend was observed for the other two pathways as well. In the HH pathway, in the directed representation 10 of the nodes were identified as critical (1 in the undirected) and 12 were intermittent (19 in the undirected). In the P53 pathway, in the directed representation 10 of the nodes were identified as critical (3 in the undirected) and 6 were intermittent (13 in the undirected). Taken together, there is a migration of nodes from critical to intermittent classifications when moving from directed to undirected representations. This may also influence the slower convergence observed in the undirected representations. These observations regarding the diffuse structure and weaker control in the undirected graphs are further supported by an examination of the

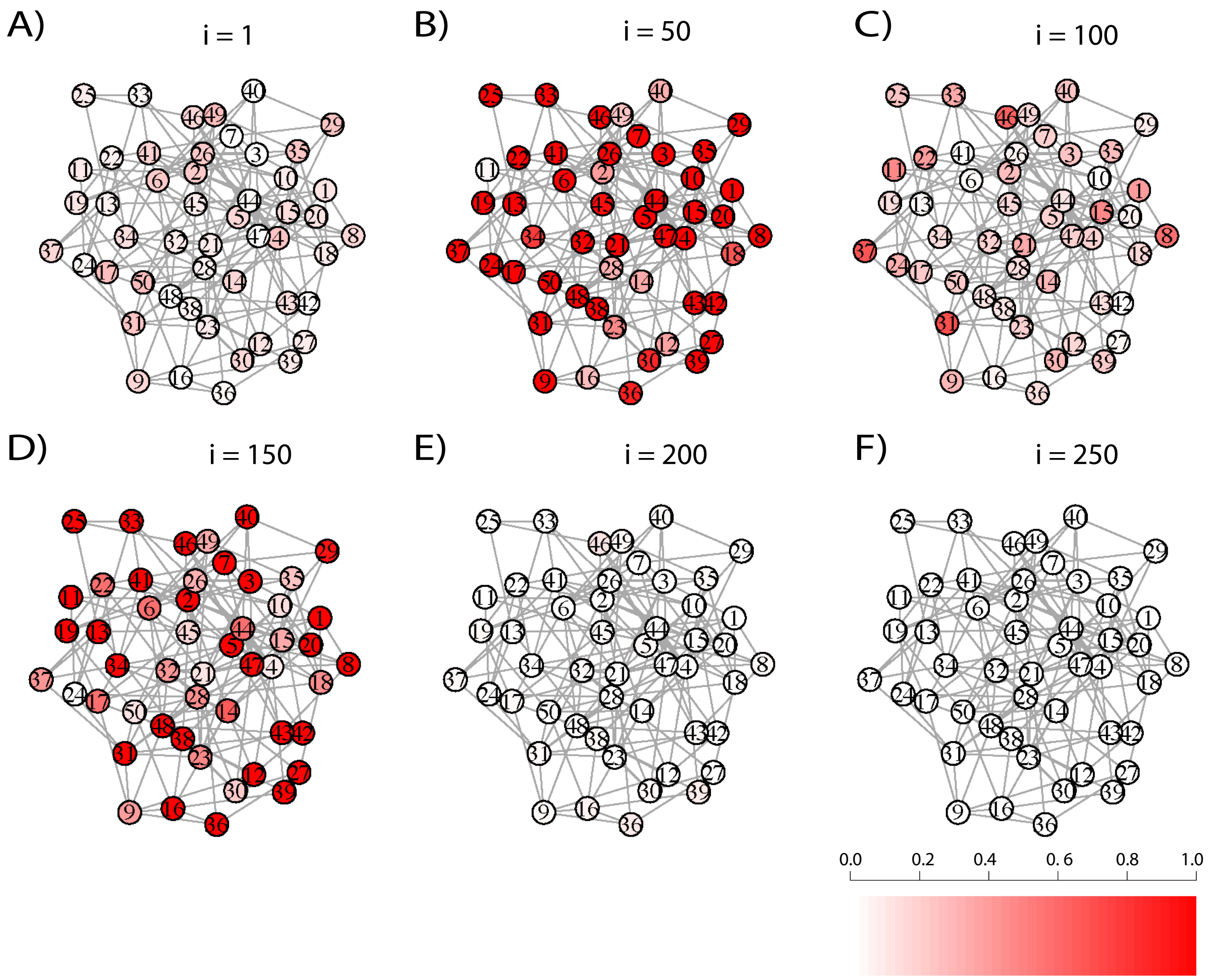

Fig. 3. Node convergence profiles for the simulated 50 network with 25% noise added to the Jacobian at select IGPON iterations k. The coloring of a node i corresponds to the relative error, $\frac{|F^T - F_k(i)|}{F^T}$ at iteration **(A)** $k = 1$, **(B)** $k = 50$, **(C)** $k = 100$, **(D)** $k = 150$, **(E)** $k = 200$ and **(F)** $k = 250$.

sensitivity matrices. Overall, sensitivity matrices for the undirected graphs were found to be of lower magnitude and exhibit weaker co-regulation patterns. In contrast, the sensitivity matrices for the directed graphs had a larger range of magnitudes and patterns of co-regulation. Sensitivity matrices for the IL-17 pathway directed and undirected representations are shown in Figure 4. The HH and p53 exhibited similar trends (data not shown).

4. Discussion

IGPON embeds propagation into an optimization that can be used to drive an undirected/ a directed graph to a desired steady-state. To the authors knowledge, this is the first approach of this type that aims to drive a network to the desired state by optimizing node perturbations. This novel approach harnesses connectivity patterns in the graph, and information propagation through the graph to guide the optimization. We demonstrate this approach to be successful in

Table1: Convergence of Biological Pathways to Target States

Pathway	KEGG	Nodes (genes)	Graph	Number of iterations			
Name	Identifier	× Edges	Type	0% noise	10% noise	25% noise	50% noise
HH	04340	52 × 162	Directed	2	69	154	278
			Undirected	2	85	178	338
IL-17	04657	53 × 147	Directed	2	72	162	335
			Undirected	2	87	202	388
p53	04115	62 × 75	Directed	2	79	199	383
			Undirected	2	90	230	390

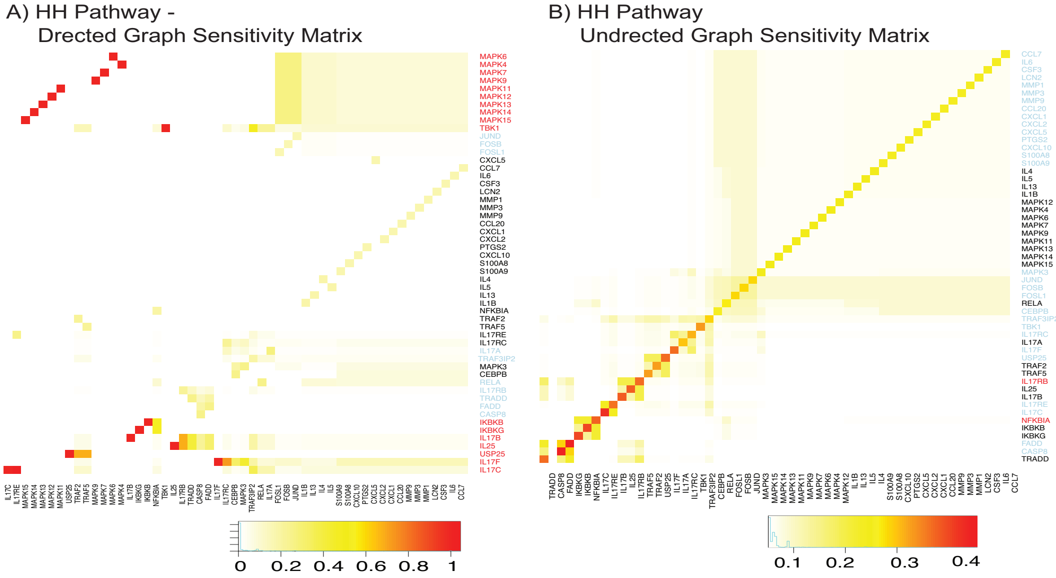

Fig. 4. Sensitivity matrices for the IL-17 pathway **(A)** directed and **(B)** undirected representations. The matrices are clustered to show patterns of co-regulation. Critical nodes (red), intermittent (blue) and redundant nods are indicated by text color.

real and simulated networks with different sizes, different architectures, and with knockdown and knockout data. IGPON is able to drive both directed and undirected graphs with up to a 0.5 signal-to-noise ratio that expresses the uncertainty in the structure of the network.

In the area of biological networks, examples of analysis of steady-state biological systems often center on flux estimation methods.[18] In these methods, the objective is flux rate estimation through the optimization of an objective function subject to physiological constraints. Flux rates are represented as the edges in the graph, which depict biochemical reaction rates or biochemical species uptake and release. IGPON can also be viewed as an optimization of a steady-state model. However, in contrast with flux analysis, the quantity of interest are the node values, not the flux rates.

This approach has many strengths. IGPON works with an assumed graph structure, but makes no parametric assumptions, and does not require parameter inference. Our experiments examine the addition of noise to the sensitivity matrix to demonstrate the robustness of our approach to structural uncertainty and misspecification in the edges. Even in severe cases, with noise levels as high as 50%, IGPON converged to the target state. This notion of misspecification is an important one because in many applications, e.g., in the biological or social sciences, the network structure may not be known exactly, or is assumed to have some structural uncertainty. A future direction of this work will be to extend this algorithm to address problems with structural uncertainty through summarizations over ensembles of graphs. There are also some limitations to our approach. The unconstrained optimization occurs over the full set of nodes in the network. However, it may not be desirable, or even feasible to fully perturb the entire network. A future direction of this work will be to couple IGPON with a feature selection method. Extensions of IGPON into a constrained optimization framework would enable feature selection and enable the use of bounds to enforce feasible values of nodes. This extension will broaden the applications of this approach to drug discovery and intervention predictions.

The Jacobian of a biological system conveys the sensitivity of individual nodes (e.g., biochemical species) to changes in parameters. However, when the functional form of the system is unknown, the specification of the Jacobian is not possible. This work builds from an important result from Santolini *et al.*,[2] which shows that the sensitivity matrix obtained through systematic propagation within the network is a good approximation of the true Jacobian of the underlying system. Although the Jacobian is updated at every iteration, the updated sensitivity matrix in Broyden's method was not considered an output of interest, although also found to converge. Results suggest that both propagation through the structure and the sensitivity matrix provide good approximations to the functional form of the system and its partial derivatives, respectively. Taken together, we conclude that optimization frameworks can be effectively bridged with propagation methodologies.

Network propagation is also used in connection with Probabilistic Graphical Models (PGMs).[19] In the PGM setting, evidence is incorporated into the graph and propagated through derived clique graphs to make queries of interest regarding changes in joint, conditional, and marginal probabilities. There are fundamental differences between PGM propagation[19] and the propagation described in PRINCE.[1] PGMs require parametric assumptions and parameter learning, whereas PRINCE relies on network structure only, but cannot be interpreted probabilistically. Moreover, in PGMs exact probabilistic reasoning can only be performed in directed acyclic graphs known as Bayesian Networks. PGMs that are directed or undirected graphs with cycles are not guaranteed to converge to exact posterior probabilities, making reasoning with them challenging. On the other hand, PRINCE can work with both directed and undirected network structures, with no restriction on cycles.

In conclusion, the use of graph structure and the integrated propagation to optimize has enabled us to drive any graph from an initial steady-state to another. IGPON works directly with a network structure and does not rely on any complex parameterizations. Predicting optimal perturbations to drive biological systems to a desired state is a promising area of research in biological and genetic engineering. This approach is implemented in the `igpon` package on GitHub and will be made available on CRAN upon publication.

References

1. O. Vanunu, O. Magger, E. Ruppin, T. Shlomi and R. Sharan, Associating genes and protein complexes with disease via network propagation, *PLoS Computational Biology* **6**, p. e1000641 (January 2010).
2. M. Santolini and A.-L. Barabási, Predicting perturbation patterns from the topology of biological networks, *Proceedings of the National Academy of Sciences* **115**, E6375 (2018).
3. C. G. Broyden, A class of methods for solving nonlinear simultaneous equations, *Mathematics of Computation* **19**, 577 (1965).
4. D. Marbach, T. Schaffter, C. Mattiussi and D. Floreano, Generating realistic in silico gene networks for performance assessment of reverse engineering methods, *Journal of Computational Biology* **16**, 229 (2009).
5. C. Feng, C. Song, Y. Liu, F. Qian, Y. Gao, Z. Ning, Q. Wang, Y. Jiang, Y. Li, M. Li, J. Chen, J. Zhang and C. Li, KnockTF: a comprehensive human gene expression profile database with knockdown/knockout of transcription factors, *Nucleic Acids Research* **48**, D93 (2020).
6. D. Zhou, O. Bousquet, T. N. Lal, J. Weston and B. Schölkopf, Learning with local and global consistency, in *Advances in Neural Information Processing Systems*, 2004.
7. J. E. Dennis Jr and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* (SIAM, 1996).
8. A.-L. Barabási and R. Albert, Emergence of scaling in random networks, *science* **286**, 509 (1999).
9. G. Csardi and T. Nepusz, The igraph software package for complex network research, *InterJournal, Complex Systems* **5**, p. 1695 (2006).
10. L. L. Elo, H. Järvenpää, S. Tuomela, S. Raghav, H. Ahlfors, K. Laurila, B. Gupta, R. J. Lund, J. Tahvanainen, R. D. Hawkins, M. Oresic, H. Lähdesmäki, O. Rasool, K. V. Rao, T. Aittokallio and R. Lahesmaa, Genome-wide profiling of interleukin-4 and STAT6 transcription factor regulation of human Th2 cell programming, *Immunity* **32**, 852 (2010).
11. E. Clough and T. Barrett, The gene expression omnibus database, in *Statistical Genomics*, (Springer, 2016) pp. 93–110.
12. M. Kanehisa and S. Goto, KEGG: Kyoto encyclopedia of genes and genomes, *Nucleic Acids Research* **28**, 27 (2000).
13. M. Evangelista, H. Tian and F. J. de Sauvage, The Hedgehog signaling pathway in cancer, *Clinical Cancer Research* **12**, 5924 (2006).
14. S. L. Harris and A. J. Levine, The p53 pathway: positive and negative feedback loops, *Oncogene* **24**, 2899 (2005).
15. H. Goodarzi, X. Liu, H. C. Nguyen, S. Zhang, L. Fish and S. F. Tavazoie, Endogenous tRNA-derived fragments suppress breast cancer progression via YBX1 displacement, *Cell* **161**, 790 (2015).
16. J. D. Zhang and S. Wiemann, KEGGgraph: a graph approach to KEGG pathyways in R and bioconductor, *Bioinformatics* **25**, 1470 (2009).
17. T. Jia and A.-L. Barabási, Control capacity and a random sampling method in exploring controlability of complex networks, *Scientific reports* **3**, p. 2354 (2013).
18. J. D. Orth, I. Thiele and B. Ø. Palsson, What is flux balance analysis?, *Nature Biotechnology* **28**, 245 (2010).
19. D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques* (The MIT Press, 2009).