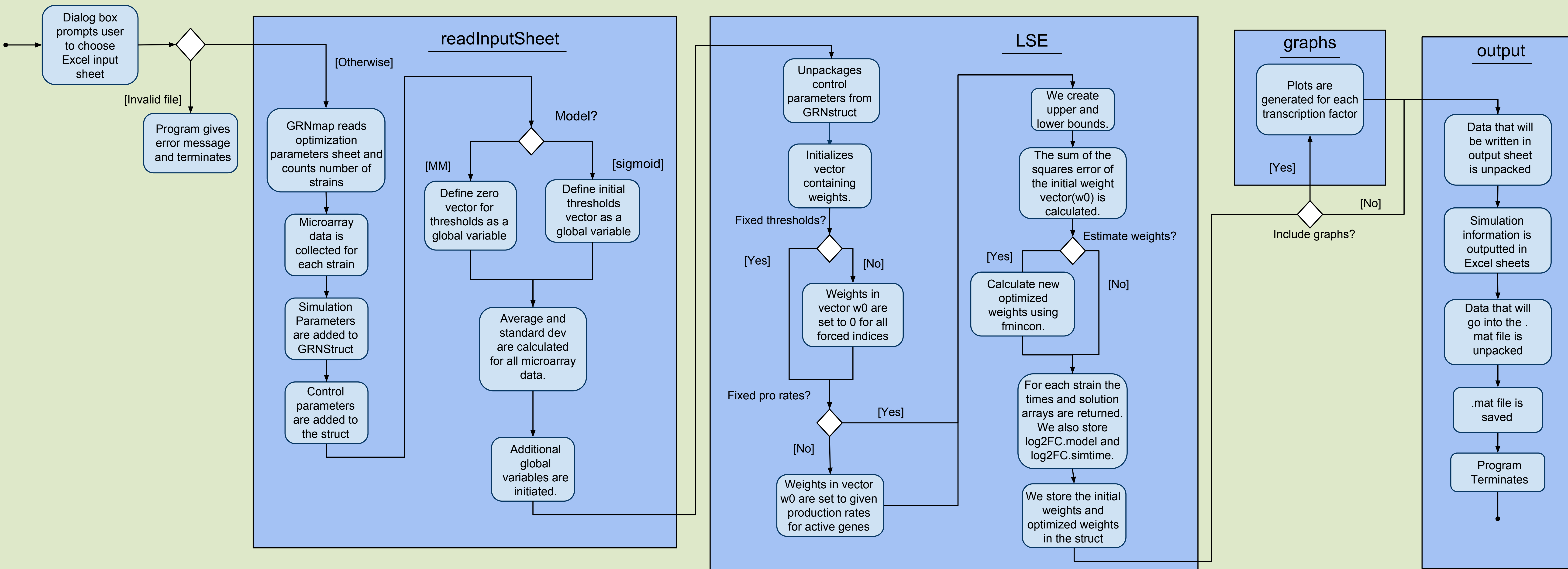


Software Refactoring and Usability Enhancement for GRNmap, a Gene Regulatory Network Modeling Application

Juan S. Carrillo¹, Trixie Anne Roque¹, Kam D. Dahlquist², and Ben G. Fitzpatrick³

¹Department of Electrical Engineering and Computer Science, ²Department of Biology, ³Department of Mathematics, Loyola Marymount University, 1 LMU Drive, Los Angeles, CA 90045 USA

Activity Diagram for GRNmap



Mathematical Model

- Our group had previously created a deterministic model in MATLAB which modeled the dynamics of how a gene regulatory network (GRN) of *Saccharomyces cerevisiae*, budding yeast, responds to the environmental stress of cold shock.
- The GRN consisted of 21 nodes which represent the genes and the transcription factors they encode. The edges of the network represent the regulatory relationship, either activation or repression, which depends on the sign of the weight term in the model (Figure 1).

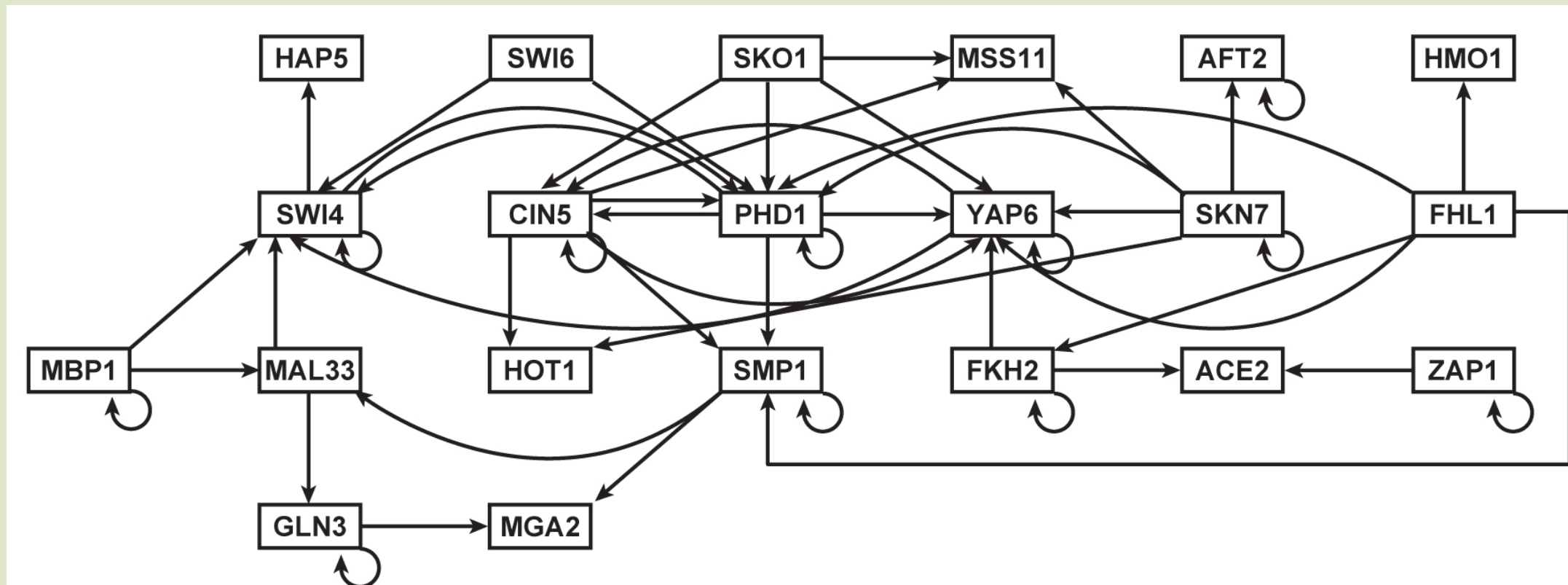


Figure 1. Gene regulatory network

- The rate of change in expression of each gene (x_i) in the network is modeled by a differential equation (Equation 1)

$$\frac{dx_i}{dt} = p_i(\vec{x}) - \lambda_i x_i \quad \text{Equation 1.}$$

where $p(x)$ is the production rate of the gene and λ_i is the degradation rate constant and x_i is the expression profile of the gene.

- We model the production term, $p(x)$, using two different models, the sigmoid model (Equation 2) and the Michaelis-Menten model (Equation 3).

$$p_i(\vec{x}) = \frac{P_i}{1 + \exp\left(\sum_j -w_{ij}x_j + b_i\right)} \quad \text{Equation 2.}$$

$$p_i(\vec{x}) = P_i \cdot \left(\sum_j \left[\frac{|w_{ij}x_j|}{\sum_k |w_{ik}x_k|} \right] \cdot \left[\frac{w_{ij}x_j}{1 + w_{ij}x_j} \right] I(w_{ij} > 0) \right) \quad \text{Equation 3.}$$

In the Sigmoid model, P_i is the production rate constant of a particular gene i , w_{ij} is the production weight of transcription factor j , and b_i is the expression threshold. For the Michaelis-Menten model, P_i is the production rate of the gene, the first bracketed term is the relative weight of a gene j , the second bracketed term represents the Michaelis-Menten reaction rate, and the third term models the effects of repression.

- GRNmap takes as input DNA microarray data which is provided as \log_2 ratios of expression for each gene in the network. Written in MATLAB, the GRNmap software loads an Excel spreadsheet as input. The software makes heavy use of two MATLAB functions: ODE45 and FMINCON. We used ODE45 to solve the model's differential equation (Equation 1) and we used FMINCON to estimate the parameters of the model using a penalized least squares fit criterion.

- The model estimates the production rates, weights, and expression thresholds. The model then performs a forward simulation using those parameters so that model-generated expression data can be compared to the experimental data input to the model.

- GRNmap outputs an Excel spreadsheet with the optimized parameters and resulting simulated gene expression profiles, a MAT file containing the calculated values, and plots corresponding to each gene in the network showing gene expression over time.

Running GRNmap

GRNmap takes in its parameters directly from the spreadsheet. The spreadsheet (Figure 2) is expected to have the following information:

- Production rates – Initial guess
- Degradation rates – Provided by user from data
- Expression Thresholds – Initial guess
- Microarray data - \log_2 fold change of expression
- Standard deviation for data
- Adjacency matrix to describe the graph for the GRN
- Initial guess for the network weights
- Simulation times
- Optimization parameters, including which model to use, whether or not to perform a forward simulation, whether or not to set certain parameters, and whether or not to include plots for the genes

	A	B	C	D	E	F
1	optimization_parameter	value	value	value	value	value
2	alpha	0.01				
3	kk_max	1				
4	MaxIter	1.00E+06				
5	TolFun	1.00E-05				
6	MaxFunEval	1.00E+06				
7	TolX	1.00E-05				
8	Sigmoid	1				
9	estimate	1.00E+00				
10	igraph	1.00E+00				
11	fix_P	0				
12	fix_b	1				
13	time	15	30	60		
14	Strain	wt	dcin5	dglin3	dhmo1	dzap1
15	Sheet	3	4	5	6	7
16	Deletion	0	3	5	8	21
17						

Figure 2. Optimization Parameters Sheet

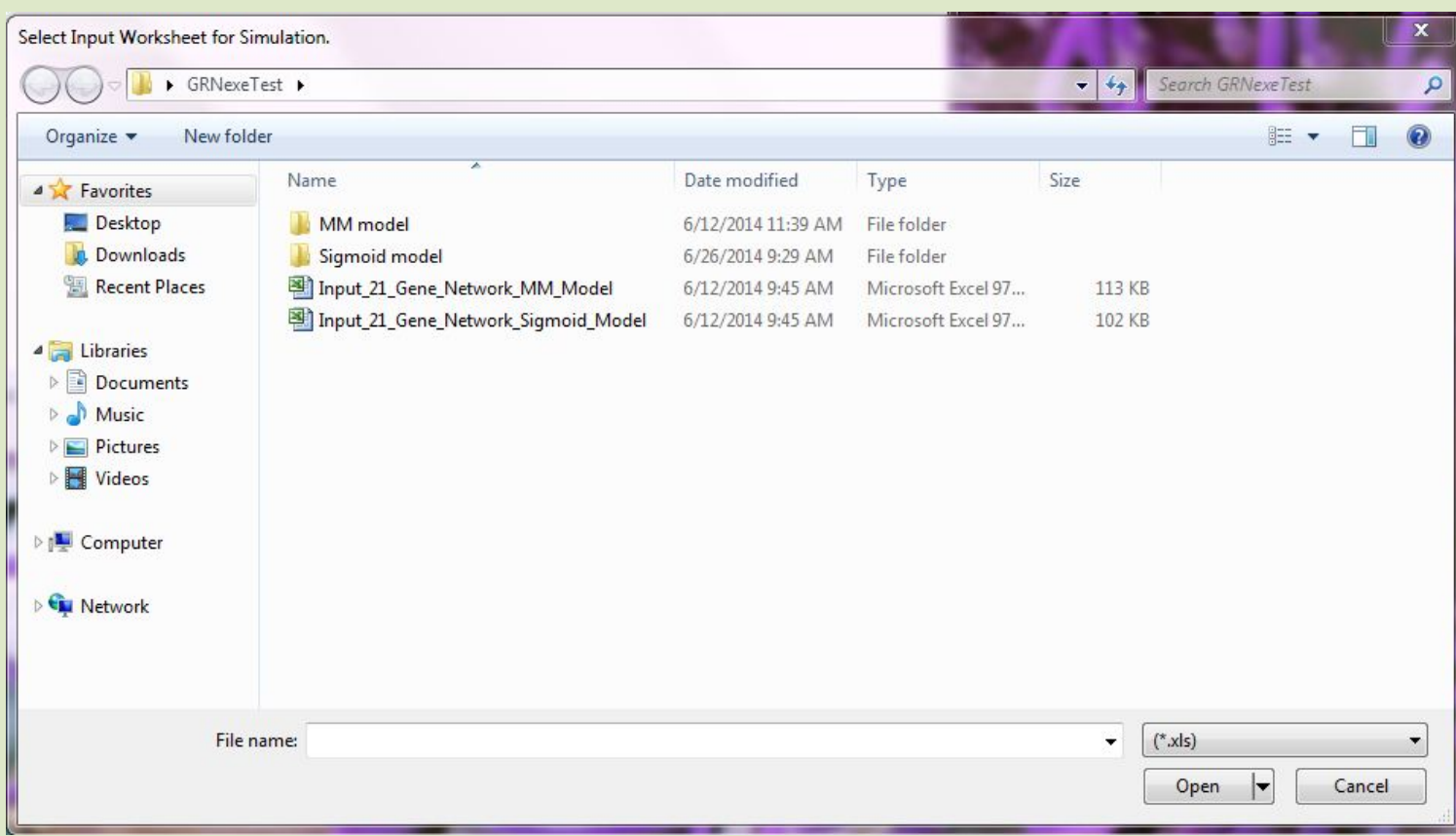


Figure 3. Dialog Box for choosing spreadsheet

- When GRNmap is run a dialog box will open to allow the user to select the spreadsheet(Figure 3). After GRNmap is finished it will output an estimation spreadsheet and .mat file. The estimation workbook contains a sheet with the optimized network weights. GRNmap will also output MATLAB figures containing the plots of the gene expression over time for each gene.

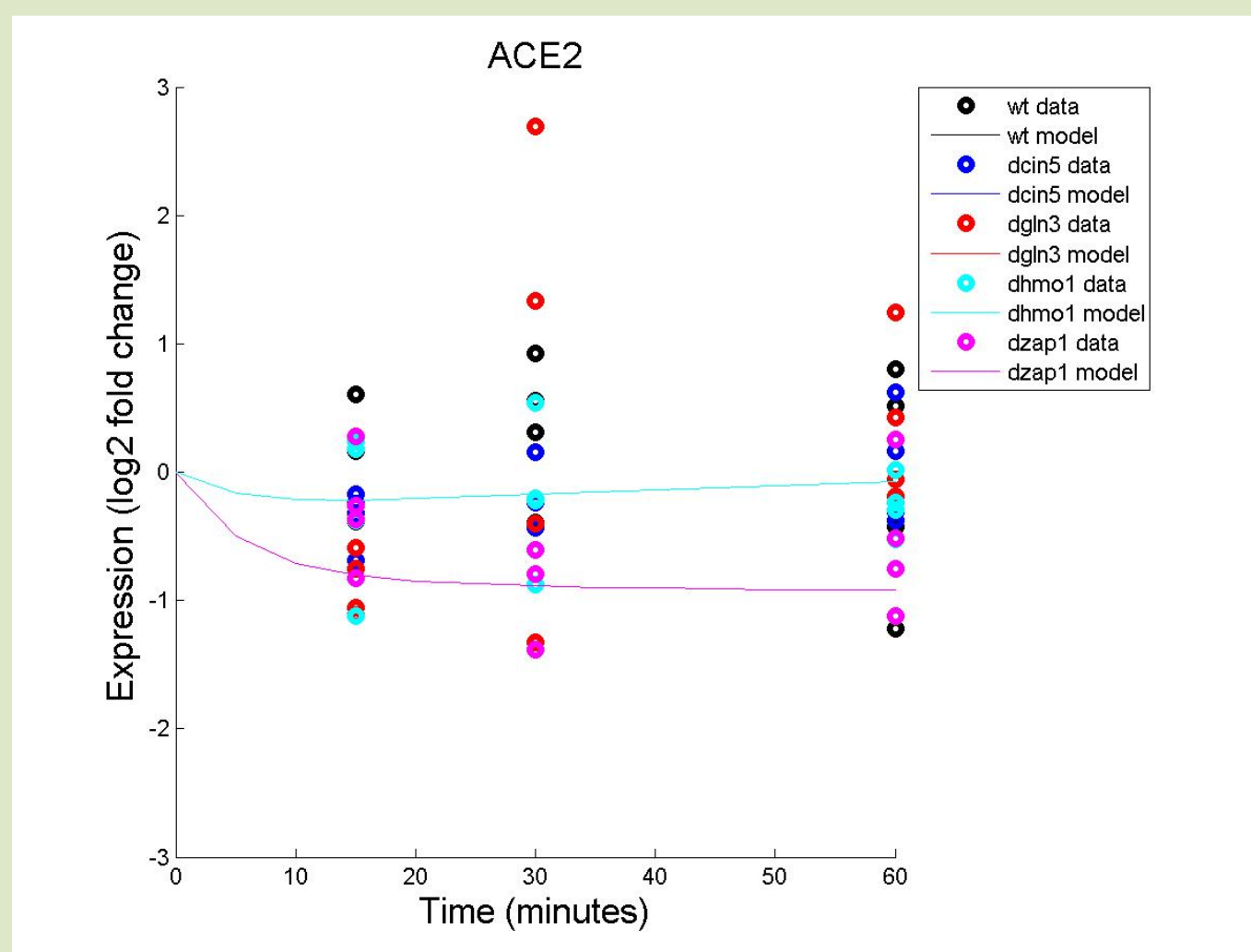


Figure 4. Sample of plot generated by GRNmap

- To see GRNmap applied to some problems of network selection, please see the poster "Comparing the Dynamics of the Cold Shock Gene Regulatory Network in Yeast with a Random Network" by Johnson and Williams.

Unit Testing

- In order to maintain the MATLAB package efficiently and effectively, a unit testing framework is created. Unit testing, a method for checking whether individual units of a software program function as intended, ensures that every change we make to the code does not affect its functionality.
- Unit testing allows for better means of pinpointing specific problems in the program by breaking up its operations into smaller parts and then inputting values to which we know the answer.

```
DEFINE main function
    CALL functiontests (localfunctions) to make a tests array
END

DEFINE function firstTest (testCase)
    actualOutput = evaluate function by using known inputs
    expectedOutput = assign expected results
    VERIFY actualOutput equals expectedOutput
END
```

Figure 5. Pseudocode for Unit Testing Framework

Summary

- GRNmap, a MATLAB program for gene regulatory network modeling and parameter estimation, had previously been developed by our group but the code needed to be reevaluated and improved.
- Unit testing is in the early stages of design and development.
- GRNmap is user-friendly as it allows users to change parameters for the model without using MATLAB code.
- GRNmap is accessible as it can be run on any Windows machine with the free MRC library installed.
- Visit us at <http://kdahlquist.github.io/GRNmap>

Future Work

- Making GRNmap more accessible by including radio buttons and check boxes in Excel rather than typing values in cells.
- We will integrate GRNmap with GRNsight (<http://dondi.github.io/GRNsight/>), a web application and service that is being developed to visualize the results of the GRNmap modeling.
- Implementing the unit testing framework will accelerate model improvement and scientific inquiry by streamlining the verification, validation, and assurance process.

Acknowledgments

We would like to thank Nicholas A. Rohacz, Alondra Vega, Stephanie D. Kuelbs, Nathan C. Wanner, and Erika T. Camacho for previous work on the GRNmap program. This project was supported by the Summer Undergraduate Research Program at Loyola Marymount University (J.S.C.), NSF-DMS award #0921038 (K.D.D., B.G.F., and K.S.), and the Clarence Wallen, S.J. Chair in Mathematics (B.G.F.).