Novel Statistical and Systems Engineering Approaches towards Gene Network Inference

A Dissertation

Presented to
the faculty of the School of Engineering and Applied Sciences,
in the department of Systems and Information Engineering
University of Virginia

In partial fulfillment
Of the requirements for the degree

Doctor of Philosophy

by

Annamalai Muthiah

May 2016

APPROVAL SHEET

The dissertation

is submitted in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

M. Annamalai

AUTHOR

The dissertation has been read and approved by the examining committee:

Dr. Jae Lee

Advisor

Dr. Stephen Patek

Dr. Gerard Learmonth

Dr. Alferdo Garcia

Dr. Susanna Keller

Accepted for the School of Engineering and Applied Science:

Craig H. Benson, Dean, School of Engineering and Applied Science

May

2016

Print Form

Novel Statistical and Systems Engineering Approaches towards Gene Network Inference

A PH.D. DISSERTATION

ANNAMALAI MUTHIAH

Systems & Information Engineering

University of Virginia

COMMITTEE
Prof. Stephen PATEK, Ph.D. (*Chair*)
Prof. Jae LEE, Ph.D. (*Advisor*)
Dr. Susanna KELLER, M.D. (Outside Member)
Prof. Alfredo GARCIA, Ph.D
Prof. Gerard LEARMONTH, Ph.D.

MAY 2016

1. Abstract

Dynamical networks such as Gene Regulatory Networks (GRNs) networks spanning the entire genome of animals and humans (network consisting of ~ 20,000 to 40,000 genes) are complex systems that possess the property of emergence and self-organization. The macroscopic property of GRNs, that is, activities of genes in the network, was simulated over time by constructing a random Boolean network and the resulting emergent properties of the network showed the dynamical network to behave stably, exhibit homeostasis, show graceful minor modification when mutated and also networks capable of complex behaviors (Kauffman, 1995). These macroscopic properties of GRN were due to the ability of complex systems such as GRN to spontaneously and freely organize themselves into an orderly dynamical network (concept known as "order for free") due to their property to self-organize.

On the other hand, large volumes of gene expression data are being generated from biological experiments and there is a great need to reverse engineer the GRN generating the data. In other words, inferring the microscopic gene interactions of the dynamical GRN from the network's macroscopic data, that is, gene expression data produced from it (the process of "network inference"). Microarrays provide a comprehensive snapshot of gene expression of the entire genome (>20,000 genes) from which the network can then be inferred. Current machine learning techniques for analyzing and constructing network from microarray data adopt a genome wide approach ('global' approach) to network inference and have a higher tendency of producing a large number of gene interactions in the network that are false positives. Instead of performing network inference among all the genes simultaneously ('globally'), the process can be simplified if a 'local' search approach is taken to network inference from gene expression data.

High-throughput gene activity data such as microarrays could either be collected as static snapshots of gene activities under different biochemical conditions or time series data of gene activities after perturbation. While inferring networks from static gene expression data, contemporary network inference algorithms performed 'global' network inference, underutilizing prior **B**iological information available about the network. Therefore, by **A**nchoring network construction around prior network **K**nowledge, the problem of network inference from large volume static data can be reduced to a 'local' network **E**xpansion problem. This principle

formed the basis of one of the novel network inference approaches I proposed and validated in my current study, Biologically Anchored Knowledge Expansion (BAKE). In particular, I applied the BAKE approach to infer gene regulatory networks from gene activity data obtained from fat cells isolated from insulin resistant mice by anchoring the construction to insulin signaling pathway genes, and thus knowledge already available about the network from the literature. Thereby, novel genes among gene activity data were organized as strong clusters around known insulin signaling pathway genes. An important advantage of the BAKE approach is that it dramatically reduces discovery of false positives in the network by discovering only those novel gene interactions tightly linked to prior knowledge of the network. When gene networks were constructed around insulin signaling pathway genes using BAKE, I discovered a novel gene, Krueppel-like factor 4 (KLF4) in the network around two key insulin signaling genes, IRS2 and TSC2, and subsequently validated interactions between the genes by additional animal experiments. I also validated the network inference ability of the BAKE approach by an in-silico (computational) experiment in which I tested whether the BAKE approach could reconstruct hidden portions of an adipogenesis network. By using a partial version of the network as prior knowledge I estimated BAKE's precision to infer edges in the network to be 44%, comparable in performance to other network inference algorithms.

While inferring networks from time series gene expression data, I also took a local approach to **N**etwork **I**nference by **A**nchoring the network around building blocks, the **M**otifs. Thereby, each motif represented a subunit of the network and was made up of three genes, and regulatory relationships between genes within each motif were inferred and the network built through the motifs. This principle formed the basis of my other novel network inference approach, Motif Anchored Network Inference (MANI). I implemented the MANI approach on time series data obtained by perturbing a 7 gene network, part of adipogenesis cascade, and validated its ability to reconstruct a small size (n=10) in-silico network made available by Dialogue on Reverse Engineering Assessment and Methods (DREAM) consortium. The precision of network inference by the MANI approach was 40%, comparable to other contemporary network inference algorithms. However, the ability of the MANI approach to generate "dynamical" features of the constructed network, such as hierarchical relationships between network genes, time sensitive activation of the network cascade, and easily interpretable network construction due to strong focus on underlying mechanisms of network regulation, distinguished it from other contemporary algorithms.

I expect these novel network inference approaches using a local as opposed to a global search approach to network inference to have wide applications in the field of gene network construction from large genomics data.

Acknowledgement

The SPC algorithm code was obtained from Dr. Eytan Domany, Department of Physics of Complex Systems, Weizmann Institute of Science. Questions regarding the execution of SPC clustering with the appropriate shell script were kindly answered by Dr. Assif Yitzhaky, a close associate of Dr. Eytan Domany. We also would like to thank Dr. Gary K. Owens at the University of Virginia for providing us with adipose tissue samples from the conditional KLF4 knockout mice. The work described in here was supported by a grant from the National Institutes of Health R01 HL081690.

Table of Contents

Chapter 1: Description of Gene Regulatory Network (GRN) as a complex dynamical network. The state of contemporary network modeling algorithms and the motivation of my modeling approaches.

1.1. Complex Systems and Dynamical Networks.

Complex systems are networks made of a number of interconnected components that interact with each other in nonlinear relationships and produce macroscopic behaviors that cannot be predicted solely from individual interactions between its components (Sayama, 2015). Complex systems science is an entire discipline devoted to developing conceptual, mathematical and computational tools to describe the macroscopic behavior of complex systems.

The major branches of complex systems are Non-Linear Dynamics (study of stability and phase space analysis of ordinary differential equations, population dynamics, chaos, bifurcation and multistability), Game Theory (complex systems that model and analyze human behavior and decision making), Collective Behavior (complex systems describing how groups of organisms self-organize and move together without a leader), Pattern Formation (self-organization processes that involve space and time such as spatial ecology, self-replication, dissipative structures etc.), Evolution and Adaptation (area covering topics such as machine learning algorithms, artificial neural networks, artificial intelligence etc.), Systems Theory (areas such as computation theory, information theory, entropy, cybernetics etc.) and Network Sciences (study of complex networks such as social networks, financial networks, adaptive networks, protein networks and **G**ene **R**egulatory **N**etworks **(GRNs)**) (Sayama, 2015). Despite its many branches, complex systems have two common properties: (i) Emergence. Macroscopic properties of complex systems are called emergent if they are more than the sum of their respective individual microscopic properties. In other words, an emergent macroscopic property of a complex system cannot be explained simply from its microscopic properties. For example, chaos (a macroscopic property) in a non-linear dynamical system could not have been predicted from the individual differential equations constituting the system and (ii) Self-organization. It is a dynamical process by which a complex system spontaneously forms nontrivial macroscopic structures and/or behaviors over time. These two properties, emergence and self-organization, are common to all major classes of the complex systems including the GRNs (Sayama, 2015). The special class of complex system that would be focused on in the present study would be the GRNs, part of the complex system category of network sciences.

Since the development and wide spread availability of computational tools for modeling complex system, researchers have begun to treat complex systems as dynamical systems, systems whose future states in time can be predicted based on their current states and interaction between their components. Cellular Automata, a computational machine developed by John von Neumann and Stanislaw Ulam, was the first major computational tool available to dynamically model complex systems (Neumann and Burks, 1966). Cellular Automata, originally designed for spatio-temporal dynamic modeling, consisted of massive number of identical finite-state machines set up as a grid structure that can update their subsequent states in time based on their current states and their neighbors' states. This ushered in a wave of dynamical modeling of complex systems in the 1970s and 1980s (Sayama, 2015). Specifically, in the field of network sciences, this led to the development of "dynamical networks", where the macroscopic behavior of a network was obtained by simulating the state of the network over time based on rules of microscopic interactions between its nodes. Before dynamical networks, networks were considered static and network analyses usually included only topological and overall structural analyses developed as part of static graph theory. Two key theoretical development in this field of dynamical networks were Artificial Neural Networks developed on the basis of biological nervous networks by Warren McCulloh and Warren Pitts (McCulloch and Pitts, 1943) and Random Boolean Networks (RBNs) initially demonstrated using GRNs by Stuart Kauffman (Kauffman, 1969). New tools and techniques are increasingly being developed to model dynamical networks and systems. Agent Based Modeling (ABM) is a recent development in the area of dynamical modeling of complex system where the individual components of the system are treated as discrete "agents" that interact with each other based on specific rules (Bonabeau, 2002). ABM has been widely used to model complex systems in areas of game theory, microeconomics, social and political processes and also in dynamical network modeling.

Dynamical networks have many real world applications because many of the real world complex systems take the shape of networks (that is, set of nodes with links connecting them). Nature has many examples of dynamical network such as the brain and GRNs (Balleza et al., 2008). The present study has chosen to study the GRN as a dynamical network. Specifically, the focus would be on techniques and methods towards inferring gene interactions within a GRN (that is, inferring microscopic interactions within a dynamical network) based on experimental gene activity/expression data collected from GRN (that is, macroscopic behavior produced from the network). Gene activity data was collected from GRN in this study either as (i) snapshots of genes activities obtained from mice under different physiological circumstances (also called "static gene expression data") or (ii) as time series gene activity data generated by perturbation of the network. This process of reverse engineering the dynamical network from its macroscopic properties is called "network inference". Constructing a dynamical GRN is important because it helps to improve understanding of key gene interactions underlying important diseases such as cancer and diabetes, which can then have therapeutic implications. The constructed dynamical GRN can be then simulated to understand its future behavior and based on the outcome of the simulation; genes in the network can be targeted so as to alter the long-term activity of the network for therapeutic benefits in disease conditions. However, in order to successfully simulate and study the macroscopic dynamical activity of GRN, the individual microscopic gene interactions of the GRN need to be constructed first. I present and discuss key techniques towards constructing a GRN from gene expression data in this dissertation.

1.2. Gene Regulatory Network (GRN) as a complex dynamical network

Gene Regulatory Networks (GRNs), the central component of cellular control, a complex network composed of thousands of genes (~ 22,000 (mouse) to 40,000 (humans)) coordinate many key cellular functions such as protein synthesis, metabolic activity, cell division and cell movement etc. (Shin and Bleris). The following description of a hypothetical four-gene network (shown in fig. 1) is a simplified explanation of how a GRN operates but helps to highlight key properties of a GRN.

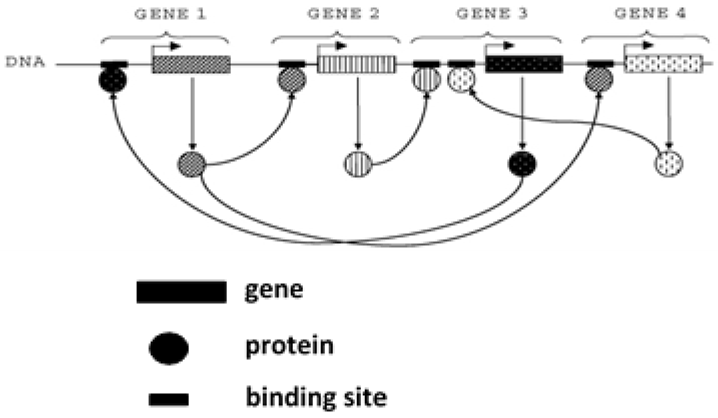


Figure 1. An example of a Gene Regulatory Network (GRN). Gene 3 regulates the activity of Gene 1, Gene 1 regulates Gene 2, Gene 2 and Gene 4 regulates Gene 3 and Gene 1 regulates Gene 4.

In the above example, assume each gene can have two states of activity – either ‘on’ or ‘off’. If a gene was active (or ‘on’, for e.g. Gene 1 in Fig. 1), it produces protein molecules that will regulate (activate or inhibit) the activity of another gene (for e.g. Gene 2 in Fig. 1) by binding to a region on the Deoxyribo Nucleic Acid (DNA) molecule called ‘binding site’ near the location of gene being regulated (Gene 2). If one gene activated another gene, it will promote protein production from the latter gene and vice versa. Gene activity regulation was analogous to an electrical circuit where genes were light bulbs and their regulatory relationships were wires connecting the light bulbs. One gene activating another gene can be seen as one light bulb turning another one on and vice versa. In the above description of GRN, an important simplification was made to assume that gene activity was binary. However, in reality, a gene shows a more graded activity.

In the immediately upcoming sections (# 1.3 and #1.4), GRN, a dynamical network, was simulated to illustrate a GRN’s various macroscopic properties using microscopic interactions between its genes but in the later section, various principles were discussed of how to infer microscopic gene interactions of a GRN from its macroscopic data (that is, gene activity data), in other words, network inference. (Kauffman, 1969) illustrated the macroscopic properties of a GRN by simulating the network using a Random Boolean Network (RBN) made of 1000s of genes. Though, in reality, gene activities were not binary, simulating GRN was made convenient if it was assumed that genes possessed binary states of activity (‘on’ or ‘off’ state) and the assumption still enabled the simulation to capture key macroscopic properties of GRN observed in nature. The microscopic rules of gene interactions within a GRN for a RBN representation are described below in section 1.3.

1.3. Mathematical representation of the dynamical GRN using RBN

Each gene is a represented by a discrete random variable g that can take two values: g=1, if gene is active or 'on' and g=0 otherwise. The complete genome (the number of nodes (N) in a typical GRN ≈ 20,000) will be represented by a set of **N** binary random variables, g_1, g_2, ... and g_N. The output of each gene, g_n, changes in time as shown by the model below.

g_n (t+1)= F_n (g_{n_1}(t), g_{n_2}(t), ..., $g_{n_{k_n}}$(t)), (Balleza et al., 2008)... (1)

where {g_{n_1}, g_{n_2}, ..., $g_{n_{k_n}}$} are the $\mathbf{K_n}$ regulators of g_n and F_n is a Boolean function (also known as logical rule), which would reflect the activator/inhibitor role of each regulator gene. The value acquired by the Boolean function for each configuration of the regulators is termed a regulatory phrase. For instance, if F(g_1, g_2) is a function of the two regulators g_1 and g_2, then the function can take the following different configurations: F(1,1)=1, F(1,0)=1, F(0,1)=1 and F(0,0)=0. The regulatory phrases for which F=1 is activatory and those for which F=0 is inhibitory. The fraction **P** of activatory phrases in the entire network, called the gene expression probability, is an important parameter that would help to identify the dynamical regime in which the network is operating (i.e. whether chaos, critical point or order). In the above example of gene function F, P is 0.75. The further P is away from 0.5 (that is closer to 0 or 1), more stable is the output from the Boolean functions and closer is the network operating at ordered regime while P closer to 0.5 indicates network is operating close to chaos. Therefore, the three key parameters of microscopic interactions of GRNs are number of genes in the network (**N**), number of regulators/neighbors for each gene (**K**, also called "connectivity") and gene expression probability (**P**).

1.4. Key macroscopic properties of GRN indicating order and stability in the network

In order to replicate the complexity of gene networks, (Kauffman, 1969) constructed a random Boolean network, composed of N=100,000 genes with each gene receiving K=2 inputs and a randomly assigned Boolean function from the 16 possible Boolean functions on two inputs. The network was then simulated using different combinations of initial patterns to produce the following macroscopic properties of the network. During simulation of the dynamical network, the Boolean network could be expected to take any of the 2^{100000} or 10^{30000} possible states. Due to the magnitude of the number of network states possible, gene network operation could have been easily chaotic and lacked any orderliness. However, irrespective of the initial state of the network, as the subsequent states of the network were updated as per the Boolean rules governing each gene, trajectory of states settled and cycled among only 317 states (where 317 is the median length of a state cycle), which was only a fraction ($1/10^{29998}$) of the total number of states possible for the network. This is "order for free", order that emerged spontaneously and freely from within the complex system. (Kauffman, 1995) concluded, from further research, that the order in the system would be preserved when the number of genes in the network (N) changed but length of state cycles, however, scaled in accordance to the square root of the number of genes in the network. Therefore, since I am interested in GRN of N=20,000, I can expect the above random Boolean simulation of my network to yield a median state cycle length of 141 states.

(Kauffman, 1995) also observed that, while changing N does not alter order in the network, changing K does. If the network connectivity (K) were increased to more than 2, the length of state cycles would begin to scale exponentially, that is, it becomes chaotic. As the network increases in size, the number of states in the state cycle would grow as the square root of N. Therefore, in order to retain the order given for free in a GRN, connectivity of genes within the network should not exceed 2 ($K\leq 2$), or in other words, the connectivity within a GRN should be sparse. Therefore, tuning connectivity of genes within a complex network to be sparse can steer the network from chaos to order.

(Kauffman, 1995) explained that the macroscopic property of order the author observed to be available freely and spontaneously ("Order for free") in sparsely connected GRNs had three main qualities: network steady states with relatively short state cycles (small attractors), homeostatic stability of steady states and graceful minor modifications of network state space when network is mutated.

1) **Order in a complex network is characterized by tiny attractors in the state space.** One of the ways order is conveyed in a complex system is by the presence of attractors[*1] in the state space with short state cycles (also called "tiny" attractors", that is attractors composed of either fixed point attractors (length of state cycle=1 because the steady state is frozen to just one state) or limit cycles (composed of oscillating steady states) with relatively small number of recurrent states). In the above described GRN simulation experiment, Kauffman observed that Boolean simulation of randomly constructed GRNs with K=2 generated state cycles of size equal to the square root of the number of genes in the network, which is much smaller compared to the total number of states possible, which in turn, reflects order in the network. However, attractors should be tiny but also stable to maintain order in the network (next property).
2) **Homeostatic stability observed in attractors within gene networks.** In the same GRN simulation experiment described above, Kauffman also observed that small perturbations in the steady states of genes returned the gene network to the same attractor in the state space because states with similar initial patterns drained into the same basin of attraction (collection of trajectories in the state space that flows into an attractor is called 'basin of attraction'). Thus gene networks could be expected to have stable steady state attractors and not be chaotic to perturbations. This stability of attractors to resist small fluctuations, also called Homeostatic stability, was also a consequence of order freely available in the network.
3) **Small mutations in gene networks caused only graceful alteration in network behavior in the state space.** Kauffman observed in the above simulation experiment that minor mutational variations in gene connections (alterations in regulatory Boolean rules or connections between genes) did not adversely affect stability in the network (since the attractors in the state space had strong homeostatic stability) but led to graceful transition to slightly altered attractors and basins of attraction based on the mutation effected on the network. Kauffman noted that this property of GRNs, to not be adversely affected by minor modifications to their configurations, provided an opportunity for them to evolve and adapt to their environment and was also a product of order available in the network.

[*1] Attractors are steady states where other states in the state space settle into in the long run. Start a network with any initial patterns and after updating through a sequence of states, it will settle into a state cycle called an Attractor.

Kauffman observed that the above three qualities of macroscopic order in a GRN, (i) presence of short ordered attractors, (ii) presence of stable attractors and (iii) ability to gracefully alter their steady state behavior in face of small mutations, were a result of "order for free" available in complex networks when $K \leq 2$. When $K>2$ in the network, Kauffman found that order can still be effected in the network by tuning another parameter, gene expression probability (P). P of a Boolean function refers to fraction of the input combinations to a Boolean function for which the function would indicate a response of 1. Each Boolean function in the GRN has a value of P and if the average value of P for the entire GRNs was 0.5, the network would be chaotic because genes in the network would be equally likely to be either active or inactive and would be devoid of any sense of connection or pattern between genes. If P was closer to 0 or 1, then you can expect a more stable response from genes and order in the network. Therefore, even if K were to be greater than 2 in a GRN (densely connected GRN), they can be steered to order if the value of P, the probability of expression, for the network was closer to 0 or 1.

Another important feature Kauffman observed in GRN is the combination of stability and flexibility that makes the network capable of complex behavior. A complex GRN, which is expected to be stable, would have fixed point attractors but also limit cycle attractors within the network that produces oscillations in gene activity in some parts of the network. Fixed point parts of the network can withstand perturbations, which gives the network its homeostatic stability but the limit cycle parts of the network, the more dynamic portion of the network, gives complexity to the network's response to perturbations. (Kauffman, 1995) estimated the value for P for such a stable yet complex GRN to lie modestly between chaos (0.5) and order (0 or 1), that is, network operating at criticality. Therefore, by tuning the parameter P, nature selected GRNs to locate between chaos and perfect order so the system can be both stable and flexible giving it a survival advantage.

The above macroscopic properties of stability and complexity of GRN were only discovered after simulating the future states of the network based on individual microscopic interactions between genes in the network. True to the definition of a complex system, it would have been hard to deduce the system's macroscopic properties of stability and complexity simply from its microscopic properties.

Large amounts of high-resolution and high-volume gene activity data such as high volume gene expression data called microarrays, next generation sequencing data etc. are being collected from biological experiments that gives many dimensions of macroscopic information about GRNs. This also creates a great need to decipher microscopic interactions of GRNs from such high volume experimental data. In order to infer microscopic interactions of the dynamical network from macroscopic data, it would be useful to recognize the pattern of macroscopic data from the network and how it translates to microscopic interactions between its genes. The next section illustrates some examples of such correlation between macroscopic gene network data and microscopic gene interactions.

1.5. Properties used to infer gene interactions from gene expression data

Fig.2 below shows the activity (expression level or # of mRNA molecules) of gene B being regulated by the activity of gene A in a simple two gene network.

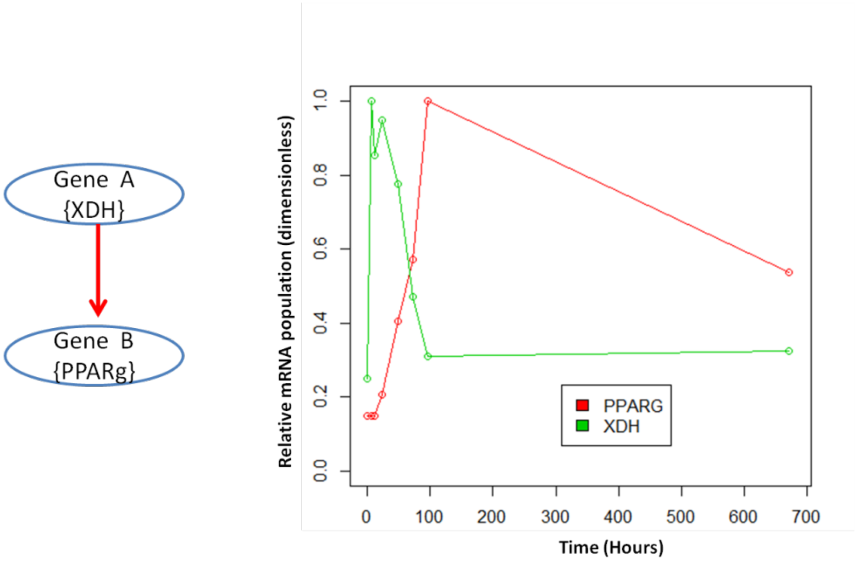


Figure 2. Gene expression profiles of a simple two gene system where Gene A (XDH) regulates the expression of Gene B (PPARg). The corresponding time course data of gene expression (or relative population levels of mRNA)

of genes XDH and PPARg shows that if gene A regulates gene B, B would follow similar pattern of expression as A although with a small temporal delay. Relative population levels of mRNA of the genes are dimensionless because they have been normalized with respect to their highest concentrations (set to 1). Data was obtained from an adipogenesis gene expression profiling study (Cheung et al., 2007).

1. Temporal lag. In the gene regulatory pair shown in fig.2 (gene A regulating gene B), the gene being regulated has a small temporal lag compared to the regulating gene. Therefore, the presence of differential lag between the expression profiles of genes can be used to arrange genes according to their order of regulation (or hierarchy) while solving for regulatory relationship between genes in a network.
2. Correlation. Notwithstanding the temporal lag, time series expression profiles of genes belonging to a regulatory pair, are usually strongly correlated. This property can be used to isolate clusters of genes belonging to a GRN from a larger pool of genes.

The above two properties play an important role in my novel network inference approaches developed in this work to construct gene networks from gene expression data.

1.6. Dynamical network of GRN using a continuous mathematical model.

Let us assume a three gene network for simplicity.

$X(t)=[a(t)\ b(t)\ c(t)]$ ... (2)

Where a(t), b(t) & c(t) are the state variables describing the expression states of the three genes a, b and c in the set.

State space equations describing the states of the three gene vector modeled using a continuous model (eqn. 3)

$$\frac{d(X(t))}{dt} = F(X(t), U(t), E) \quad ...(3)$$

Where,

U(t) - time variant external input (scalar)

e_i - Noise in the equation belongs to vector $E \sim N(\mathbf{0}, \tau)$. τ is $\tau_{constant}\,.\,\mathbf{I}$

E- Vector of e_is

$$
\begin{aligned}
F(X(t), U(t), E) = \; [& f_a(a(t),b(t),c(t),U(t),e_1) \\
& f_b(a(t),b(t),c(t),U(t),e_2) \\
& f_c(a(t),b(t),c(t),U(t),e_3)] \qquad \ldots(4)
\end{aligned}
$$

f_is (f_a, f_b and f_c) are the individual regulatory relationship describing the future state of each gene based on their current states and external input, could either be a linear/non-linear function.

If f_i is linear and x_i (state variable representing gene expression of a single gene) was sampled uniformly, F in equation 4 could be simplified as

$F(X,U, E) = A.X+B.U+E$ (Yeung et al., 2002) ...(5)

Where A is interconnectivity matrix containing regulatory coefficients specifying how one gene regulates another and vector B describes the influence of external input on the various genes.

If f_i is non-linear, it could take one of the following forms

(i) Exponential model (Gupta et al., 2005): $f_i = \dfrac{\prod_{j=1}^{N} \chi_j^{\alpha_j}}{K}$ ...(6)

where α_i captures the regulatory effect of gene j on gene i ($\alpha_i > 0$, if positive regulation and $\alpha_i < 0$ if negative regulation (inhibition)), K is scalar and χ_j is scaled activity of gene j.

(ii) Sigmoidal model (Vu and Vohradsky, 2009): $f_i = \dfrac{k_i}{1+\exp(-(\sum_{j=1}^{N} w_{ij}\chi_j + b_i U))}$ ...(7)

Where w_{ij} is the regulatory coefficient capturing the regulatory effect of gene j on gene i. b_i captures the effect of external input on gene i and k_i represents maximum rate of expression of gene i and χ_j is scaled activity of gene j.

(iii) Hill model (Yeung et al., 2002, Marbach et al., 2010): Hill kinetics would be a rigorous thermodynamic approach to describe gene regulation. Hill model can capture multiple regulators acting on a gene either independently/additively (equation #8) or synergistically/cooperatively (equation #9).

$$
f_i = \frac{K + \sum_{j \in A} \alpha_j x_j^{n_{ij}}}{M + \sum_{j \in A} \alpha_j x_j^{n_{ij}} + \sum_{k \in B} \beta_k x_k^{n_{ik}}} \qquad \ldots(8)
$$

$$
f_i = \frac{K + \sum_{j \in A} \alpha_j x_j^{n_{ij}} + \sum_{l,m,n \in C} \rho_n \gamma_n x_l^{n_{il}} x_m^{n_{im}}}{M + \sum_{j \in A} \alpha_j x_j^{n_{ij}} + \sum_{k \in B} \beta_k x_k^{n_{ik}} + \sum_{l,m,n \in C} \rho_n \gamma_n x_l^{n_{il}} x_m^{n_{im}}} \qquad \ldots(9)
$$

Where,
Genes in sets A and B are positive and negative regulators of the gene i respectively,
K, maximum rate of synthesis of gene i,
M, the hill disassociation constant of gene i,
α_j , relative activation of gene i by gene j ε A,
β_k , relative activation of gene i by gene k ε B,
γ_{n} , relative activation of gene i synergestically by pairs of genes (l, m) ε C,
ρ_{n} , co-operativity factor between the pairs of genes (l, m) ε C,
$\chi_j, \chi_k, \chi_l, \chi_m$, scaled activities of genes.
$n_{ij}, n_{ik}, n_{il}, n_{im}$, magnitude of regulation of gene i by genes j, k, l and m.

1.7. How do you collect high volume gene activity/expression data?

A cellular process is accompanied by biochemical changes caused by a network of genes. To understand the underlying active gene network causing such biochemical changes inside a cell, microarrays are collected to understand the activities of genes. Micro array data is a popular tool that simultaneously captures the states/activities of all the genes in a cell (~20,000-40,000 genes). Micro array data could either be collected over different time points in the life of a cell as time series data (eg., in the case of a cell undergoing time dependent biochemical changes or capturing changes to the expression states of genes over time after perturbing the network) or static data (eg., steady state snapshots of expression states of genes from different biochemical states of the cell). Static data can help to extract genes responsible for the differential response between any two biochemical steady states, which can be traced to the biochemical pathway containing those genes while time series data can help to tease out the order of regulation between genes using the temporal lag property discussed before (Bar-Joseph, 2004). Innovative and creative ways to infer GRNs (microscopic gene interactions) from high volume gene expression data (macroscopic information about the network) is needed.

1.8. How do you infer gene networks from gene expression data? State of contemporary algorithms for gene network inference.

Depending on whether it is a static or time series gene expression data, GRN is constructed between the genes by applying the appropriate network inference algorithms on the microarray data. The process of constructing GRN as end product consistent with the observed gene expression data is called ‘network inference’ or ‘reverse engineering’.

Some of the commonly used tools on static microarray data are clustering algorithms (k-means, hierarchical clustering etc.), Bayesian modeling approaches that models gene interactions by fitting gene expression data (from multiple conditions) to various hierarchical combinations of genes (Friedman, 2004, Friedman et al., 2000), Boolean networks (Espinosa-Soto et al., 2004) and Singular Value Decomposition (SVD) method of comparing two groups of gene expression data for underlying trends among groups of genes. The end product of such algorithms are either large clusters of genes or the computation becomes tedious as the number of genes increase because of the many combinations of networks possible. This approach is not especially useful to biologists because analyzing all the genes at once results in huge clusters or networks that are of little practical value to biologists because many of these networks are not centered around the biologists' gene(s) of interest and also produces high number of false positives that are difficult to distinguish from true biological interactions of interest.

In the case of time series data, gene interactions are deduced using one of the three methods for measuring relationship between genes: Mutual Information (MI) or pair-wise correlation or conditional probability among interacting molecules (Villaverde and Banga, Margolin et al., 2006, Zoppoli et al., 2010, Luo et al., 2008, Reshef et al., 2011, Butte and Kohane, 2000). For example, mutual information between two genes is obtained by evaluating how much information of one gene is contained in the other after estimating the conditional entropy across their different states of function (Villaverde and Banga). MI can capture non-linear and complex relationships between genes based on the ratio of their joint and marginal probability functions. The concept of MI has been implemented in different algorithms such as Algorithm for the Reconstruction of Accurate Cellular Networks (ARACNE), which minimizes indirect and redundant edge identification using Data Processing Inequality (Margolin et al., 2006), time delayed ARACNE for time dependent expression data (Zoppoli et al., 2010), and three-way MI for complex regulatory gene interactions (Luo et al., 2008, Reshef et al., 2011, Butte and Kohane, 2000, Meyer et al., 2007, Altay and Emmert-Streib, 2011). Partial correlation coefficients have also been used to infer gene networks, especially to minimize redundant edges in the network (de la Fuente et al., 2004, Veiga et al., 2007). Conditional probability measure to infer gene networks was implemented in Dynamic Bayesian Network models (Lebre et al., 2012). Some other widely-used network inference approaches are regression based algorithms for a known set of regulator genes and target genes (Huynh-Thu et al., 2010, Segal et al., 2003), shrinkage techniques (van Someren et al., 2006), and Network Component Analysis (Feizi et al., 2013). Approaches such as developing a community-based consensus of various network inference algorithms (‘wisdom of crowds’ approach) that can harness the combined strengths of various algorithms overcoming traditional biases of each are also becoming increasingly popular (Hase et al., 2013, Marbach et al., 2012). There are also a few important dynamic and differential equation models used for constructing GRN. I discuss in greater detail a few of these techniques.

1. Framework of Metabolic Control Analysis (MCA) for gene network construction (de la Fuente et al., 2002). Regulatory strengths, quantifying the direct interactions between genes, are calculated by estimating Co-control coefficients between pairs of genes. Co-control coefficients are ratios between control coefficients determined from microarray experiments and show how two system variables (expression/activity levels of genes) respond to a common perturbation capturing global system properties.
2. Solving the linear and non-linear forms of interaction matrix F (equation #3) using Singular Value Decomposition (SVD) and choosing parsimonious network (Yeung et al., 2002, Gupta et al., 2005). In the case of linear model for F (equation #5), the expression data matrix (X) was factorized using SVD and the general and specific solution of matrices A and B in equation #5 was obtained and the most parsimonious solution was chosen by linear programming. In the case of non-linear model for F, they were linearized and the same protocol as that of a linear model was followed. The only obstacle for me to apply this approach was that it necessitates uniform sampling of gene expression data.
3. Dynamic Bayesian Network (DBN, (Abegaz and Wit)) and Probabilistic Boolean Network (PBN) modeling approaches (Martin et al., 2007). In a Dynamic Bayesian Model, which can model random variables in continuous state space, gene expression data was separated into random variables representing expression states of genes at each time point. The random variable at any time point was then modeled to be related to random variable only from the previous time point as in a Markov chain. Their relationship was expressed by Vector Auto Regressive process of order 1 (VAR(1)) and the values of parameters of VAR(1) reflected

the interconnectivity among genes, which was estimated by maximizing the likelihood of the Markov chain. In a Probabilistic Boolean Network (PBN) model, which can only model random variables in discretized state space, discretizing the gene expression data was achieved by using k-means and support vector regression. Boolean Infer function helped to infer the optimal Boolean rule associated with a gene by separating the activators and inhibitors of a gene and optimizing the rule for interaction among genes to match the data.

4. Perturbation based gene network discovery. This approach treated gene network to be solved as a system with multiple input, perturbed the system multiple times through those inputs and based on the steady state outputs of the system from such perturbations, the complete GRN was constructed. Two notable examples of this approach: (i) a study (Yuan et al., 2011) that conducted multiple perturbations on a gene network (number of perturbations was same as the number of genes to be reconstructed, that is every gene in the network was perturbed once), observed the outputs from each perturbation, solved for the transfer functions in the network which captured the network connectivity. This approach is quite effective in solving small network GRNs. (ii) Another study (Kholodenko et al., 2002), where similar perturbation approach was applied, Kholodenko et al. decomposed the whole network into smaller theoretical modules and applied perturbation inputs affecting each module exclusively and based on the responses from other modules, deciphered the connectivity among all the genes. However, by this approach, for a simulated four gene network, the author had to generate 8 perturbations effects to reconstruct the network, which may quickly become too many while performing experiments on a large network to be reconstructed (iii) Another type of perturbation study is to perturb different parts of the network by knocking out (completely eliminating) or knocking down (reducing) the expression levels or activities of different genes in the network and collecting a set of steady state data after perturbation. Then the perturbation steady state data from various perturbations was used to construct a first-order predictive model (equations 3 and 5), which was then solved for gene interconnectivity by multiple linear regression applying the concept of parsimonity (Gardner et al., 2003). For example, in a system of N genes, parsimonity was applied by assuming a maximum of only K inputs to each gene. Thus, by assuming K non-zero regulatory inputs

(where $K < N$), multiple linear regression was used to estimate the model coefficients within the interconnectivity matrix A.

1.9. What is missing in these algorithms and why the local network inference approaches developed by the author are necessary?

While constructing gene networks from large volume gene expression data (consisting of ~20,000 to 40,000 genes) using the above algorithms/approaches, the tendency of majority of network inference techniques was to construct network between all the genes in the data. However, since there are thousands of genes active at the same time in a cell, such global genome-wide network inference strategies often result in construction of intricate "hair-ball" type networks. Constructing such large networks with many edges increases the probability of discovering edges that are either false positives or not directly relevant to the mechanism of interest to the investigator. It would be an "improvement" if the network inference was centered on the investigator's mechanism of interest by directly incorporating it into the inference process. Therefore, a local network inference around the mechanism of interest will provide more features/insights about the network useful to the investigator instead of an unsupervised global inference from the data.

1.10. How do I incorporate the investigator's mechanism of interest for local network inference while using static gene expression data?

The traditional techniques used for gene network inference from static gene expression data, such as clustering algorithms, Bayesian modeling and SVD, tend to produce large global networks for reasons described above (also shown below in fig. 3 bottom left). Instead an alternate approach to network inference would be to gradually expand gene networks around the investigator's prior knowledge of the network. By anchoring the network expansion around genes that are part of the prior knowledge, you only extract genes from the large data directly relevant to the investigator's target mechanism but also address the issue of false positive discovery prevalent in network inferences conducted using large volume gene expression data (shown below in fig. 3 bottom right).

I implement the above principle using Biologically Anchored Knowledge Expansion (BAKE), a novel six step network inference approach for large volume static gene expression data, that **A**nchors network **E**xpansion to investigator's prior **B**iological **K**nowledge of the network by extracting only relevant novel genes from large volume global genomic data sets to prior genes in the network and forms static clusters/network around them. Detailed implementation and validation of BAKE's approach to network inference is available in chapter 2.

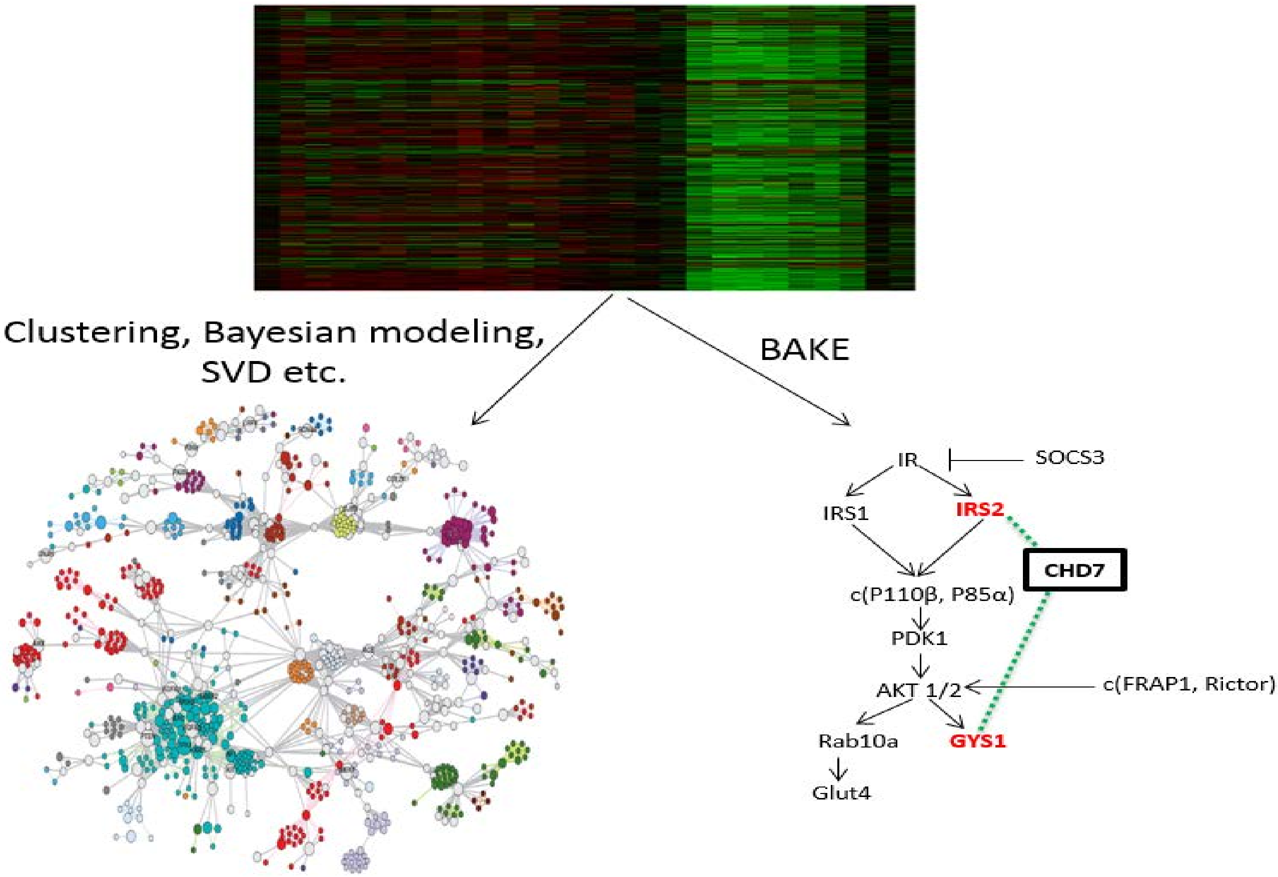


Figure 3. Contrast between contemporary gene network inference techniques and BAKE approach on static genomics data. **Top panel.** Heat map of large volume gene expression data collected as snapshots over different biochemical conditions. **Bottom panel (left).** Contemporary network inference techniques that produce a global hairball network. The investigator has to navigate the network to focus on the portion of network of interest to him. **Bottom panel (right)**. Alternately, by directly incorporating the investigator's mechanism of interest and anchoring the network inference around it, new molecules from large data could be extracted if they were only directly relevant to the mechanism of interest. In the example shown, gene CHD7 was extracted from large data because of its strong association to two genes in the investigator's mechanism of interest, IRS2 and GYS1. This type of local network inference approach (as opposed to global approach shown on the left) formed the basis of my Biologically Anchored Knowledge Expansion (BAKE) approach.

1.11. How do I implement local network inference approach to time series gene expression data?

Just as was discussed in the previous section for static gene expression data, global approach to network inference using time series also won't be beneficial. I motivate the need for a local approach to network inference while using time series data too using the example of time series gene expression data shown in fig. 4 top panel. The 7 gene network, belonging to adipogenesis network (Rosen and Spiegelman, 2000), was perturbed and time series data of gene activity was collected at different time intervals. Many of the contemporary algorithms, which can perform network inference from time series data (described in section 1.5), would infer interconnectivity between these 7 genes by solving for the interconnectivity matrix. Assuming a linear model as shown by equation #5 (from section 1.3), 7 x 7 parameters are to be estimated in matrix A and 7 x 1 parameters are to be estimated in vector B for a total of 56 parameters. Since the data points of gene expression values were not sampled at even time intervals, estimating the values of these 56 parameters could be accomplished by one of the following ways: a) optimization technique. However, optimizing such large number of parameters could be a computational burden. b) Estimating a parsimonious gene network among the 7 genes by any one of the techniques discussed before. However, the above approach by contemporary algorithms to estimate relationships between all the genes in the network simultaneously (or in other words 'globally') would result in a hairball network as shown fig. 4 bottom panel (left) and will not fully capture its "dynamical" features.

The constructed network can be made more "dynamical" if network inference was pursued locally instead of globally. The network can be constructed locally by focusing on constructing the individual motifs, which are the building blocks of the network, and then gradually building the network through the motifs. Each motif represented a subunit of the network and was made up of three genes, and regulatory relationships between genes within each motif were inferred and the network was built through these motifs. This idea formed the basis of my second novel network inference approach presented in this work, Motif Anchored Network Inference (MANI), a local approach to gene network inference using time course gene expression data (the first approach was 'BAKE', a local approach for static gene expression data). When MANI was applied to construct network from time course gene expression data in fig. 4 top panel, it produced a more “dynamical” network as can be seen by comparing the quality of network constructed by using a global approach (fig. 4 bottom panel, left) and by using MANI, a local approach (fig. 4 bottom panel, right). The exact implementation of MANI, how it constructed the adipogenesis cascade in fig. 4 bottom panel (right) from the time series data and its validation are described in chapter 3.

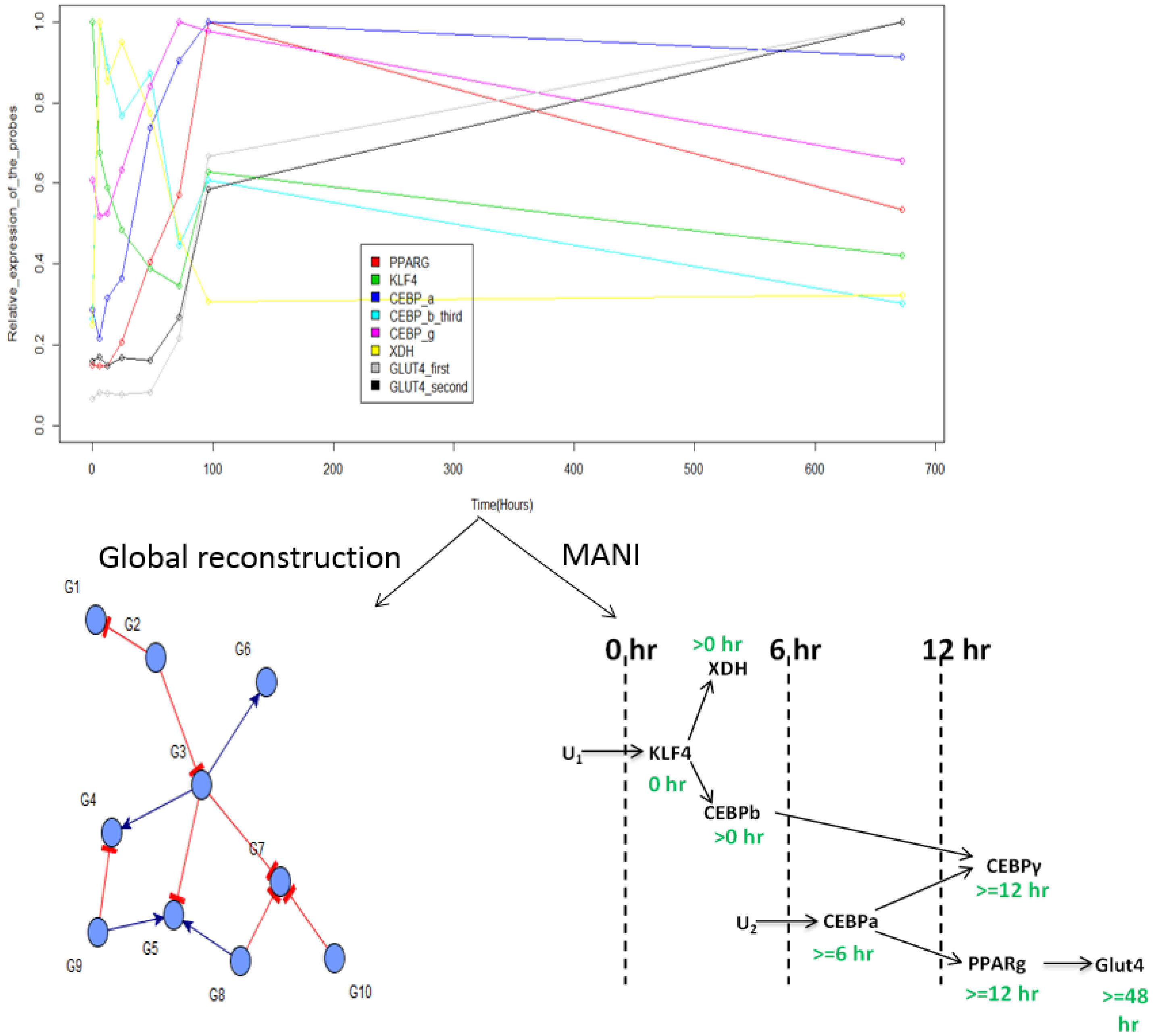


Figure 4. Benefits of local approach to network inference on time series gene expression data. Top panel. Time series data (uneven sampling intervals - 0, 6 hrs, 12 hrs, 24 hrs, Day2, Day3, Day4, Day 28) of expression

levels of 7 genes forming a regulatory cascade involved in the process of adipogenesis (Cheung et al., 2007). **Bottom panel (left).** Constructed network produced as end product by contemporary algorithms that use a global approach to network inference. Note that it is a generic network (as can be seen from the generic nature of gene names in the network, G1, G2 etc.) used as an example of what can be expected from contemporary algorithms and not specific to the adipogenesis network. **Bottom panel (right).** Network inference from the MANI approach. Network inference was performed locally through individual motifs that make up the network and the final product resulting from such an approach to network inference is as shown. MANI approach adds significant “dynamical” quality to the final constructed network in terms of (i) hierarchical relationship between genes and (ii) time sensitive activation of the network, that is, how this hierarchical regulatory network switches itself on at various time intervals. For example, gene CEBPa is only triggered more than 6 hours into the cascade, which it follows up by driving the expression of genes, PPARg and GLUT4, downstream.

CHAPTER 2: BAKE approach description and its validation

2.1. Problem definition and goals

In order to study the molecular network involved in insulin resistance caused by high fat diet, mice genetically predisposed to insulin resistance were fed two different diets (high fat and low fat diet) and biological tissue (adipose tissue) was extracted from them at two time points, 8 and 16 weeks, to generate four groups of experimental mice: Group A (mice fed with low fat diet and adipose tissue extracted from them at 8 weeks, n=8 mice), Group B (mice fed with low fat diet and adipose tissue extracted from them at 16 weeks, n=5 mice), Group C (mice fed with high fat diet and adipose tissue extracted from them at 8 weeks, n=5 mice) and Group D (mice fed with high fat diet and adipose tissue extracted from them at 16 weeks, n=9 mice). The genomic material extracted from the tissues were measured for the activity of genes (~ 40,000 genes) across the four biochemical conditions, A, B, C and D [activity levels of fraction of those genes are shown across the four conditions as heat map in step1 of fig.5].

Therefore, the problem to solve is to use the above static snapshots of gene expression data across the four biochemical conditions with varying degrees of insulin resistance and construct the gene network activated by high fat diet that is responsible for developing insulin resistance in mice. As discussed in the previous chapter, there are a few different clustering and dimension reduction techniques that would be appropriate for constructing global gene networks from the large volume static gene expression data. Since my goal is to construct the network ‘locally’, I will expand the gene network around insulin signaling pathway genes (prior knowledge already available about the network from literature). My task in the present chapter is to decide how to ‘locally’ expand the network around this prior knowledge, the insulin signaling pathway, using the above static gene expression data collected from four different biochemical conditions. Or in other words, as can be seen in fig.5 top panel, the question is to how to organize this new global molecular data (large data of gene activity) locally around the prior knowledge of the network to get an expanded network.

2.2. Implementation of BAKE approach

I first outline the guiding principles of BAKE before I propose the exact steps of how BAKE was implemented. In order to integrate the data with the prior network, it is important to extract a select pool of novel genes (L_0), from the global pool of genes (~40,000 genes), that have differential activity across the biochemical conditions of interest (among the four biochemical conditions available, the biochemical conditions of interest is chosen by the investigator and how it was chosen will be discussed later). Genes within the pool, L_0, can now be the source of genes suitable for constructing network around the genes beloning to the prior network. Once the global pool of genes is reduced to L_0, it is important to reduce it further only to those genes (L_1) relevant to the genes present in the prior network. Having restricted the pool of genes to only those genes that are associated with the prior network (L_1), the next step is to construct networks around genes in the prior network using genes from L_1. These broad principles were aimed to convert a global computational mining of candidate genetic networks to a local restricted gene network search, efficiently discovering novel gene network interactions tightly linked to prior network.

I translated the above broad principles of BAKE approach to the following five sequential analysis steps: 1) Biological context-dependent discovery of pool of novel genes (selecting $L_{0,}$ step 1 in fig.5), 2) Developing prior gene network relevant to the mechanism of interest (step 2 in fig.5), 3) Identification of novel genes associated with prior gene network (selecting L_1 from L_0), 4) Constructing network around genes in the prior network using novel genes from L_1 leading to network expansion (steps 3 and 4 in fig.1) and 5) Checking literature to validate novel network expansions around prior network.

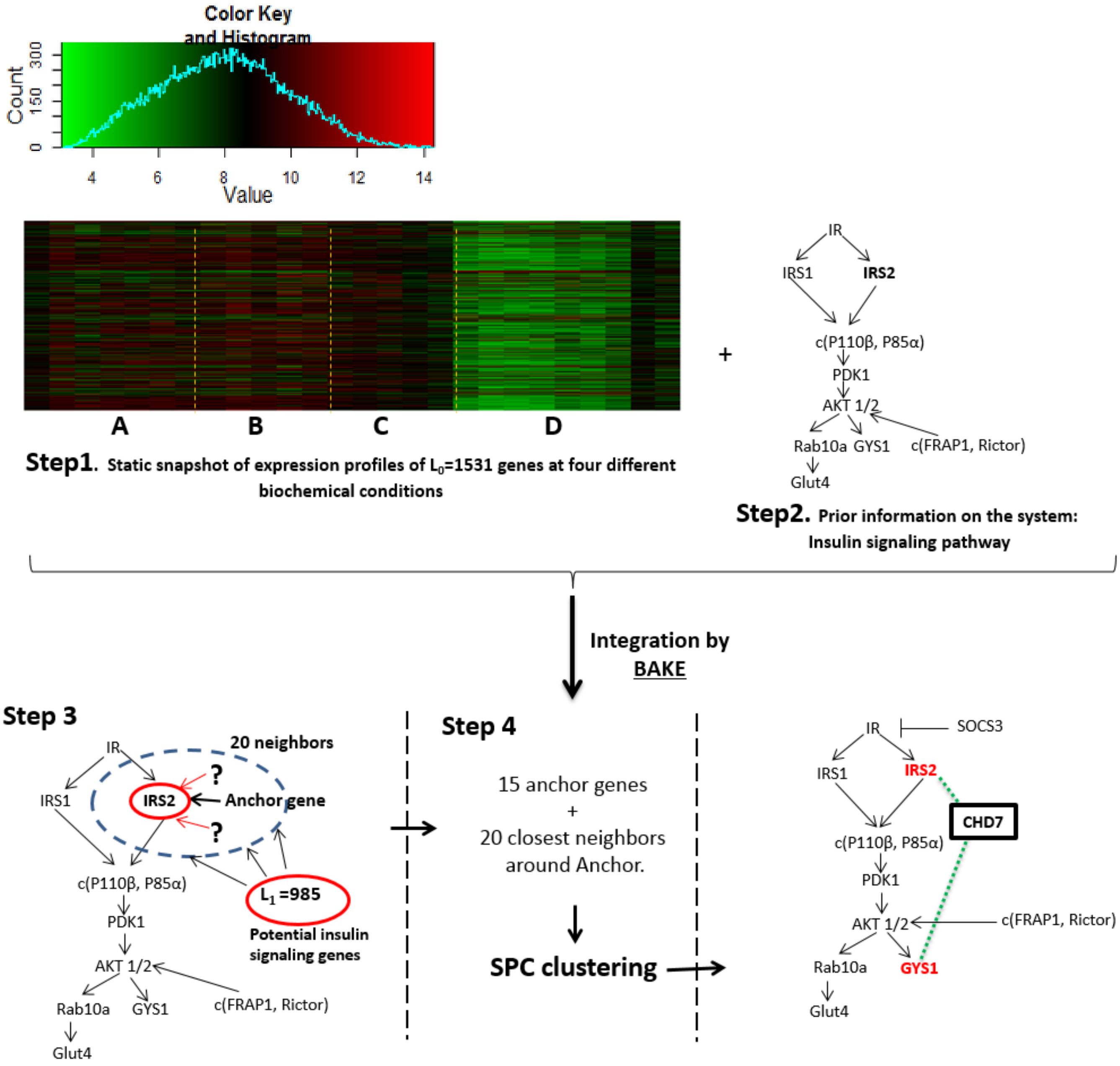


Figure 5. BAKE approach applied for network expansion around insulin signaling pathway. Step 1. Selection of L_0. The heat map shows the activity levels of genes in L_0 across the four biochemical conditions (A: mice fed with low fat diet and adipose tissue extracted at 8 weeks, B: mice fed with low fat diet and adipose tissue extracted at 16 weeks, C: mice fed with high fat diet and adipose tissue extracted at 8 weeks and D: mice fed with high fat diet and adipose tissue extracted at 16 weeks). Genes that showed differential activity between the conditions B and D were chosen as the pool of novel genes (L_0=1317 genes) useful for constructing network around genes in the prior

network. The color key for the three colors used in heat map (colors of green, black and red) in terms of levels of gene activity is also shown. **Step 2.** Prior gene network, genes belonging to Insulin signaling pathway. Though there are 42 genes in this network, I show a select few in the pathway. **Step3.** Network expansion around the anchor gene, IRS2. **Step 4.** Cluster formation between novel genes and anchor genes through SPC clustering to identify strong clusters based on the correlation strength of expression profiles of genes. In the example shown, CHD7 was a novel gene that formed strong clusters with the pathway genes, IRS2 and GYS1. The strong association between the novel gene and the pathway genes was used to expand the network around the two pathway genes (shown by the green dotted edges).

2.3. Detailed Implementation of BAKE approach

Step 1. Biological context-dependent discovery of pool of novel genes, L_0. The context for novel gene discovery was to identify gene network involved in propagating insulin resistance in adipose tissue of mice as a result of high fat diet. Among the four experimental diet conditions, the contrast in the levels of gene activity in adipocytes was most pronounced at 16 weeks between the two diets (fig.1, step1) and therefore, genes that were differentially active between the two biochemical conditions, B (mice fed with low fat diet and adipose tissue extracted from them at 16 weeks, n=5) and D (mice fed with high fat diet and adipose tissue extracted from them at 16 weeks, n=9), were chosen as the pool of novel genes, L_0, suitable for constructing network around genes in the prior network. The contrast in gene activity between these two biochemical states was conducted using significance analysis algorithms for small sample size microarrays, LIMMA and SAM (28, 29), and it yielded L_0=1317 differentially expressed genes (selected at a statistical criterion of 1 % False Discovery Rate (FDR)).

Step 2. Developing prior gene network relevant to the mechanism of interest. Prior gene network relevant to insulin resistance is Insulin signaling pathway and I identified 42 known genes within the pathway from the KEGG (Kyoto Encyclopedia of Genes and Genomes) database and literature (Taniguchi et al., 2006) that could be treated as genes belonging to the prior network (L_{path}, list of genes in supplementary table **Table S1**). I can use the activity of these genes present in prior network to extract genes relevant to prior network (L_1) from L_0 by virtue of their degrees of correlation across the various biochemical conditions.

Step3: Identification of novel genes associated with prior network. In this approach, the pool of novel genes (L_0) was reduced to genes specific to Insulin signaling network (L_1) by correlating the activity levels of genes (by spearman rank correlation) in L_0 to those of the insulin signaling pathway genes (L_{path}) across the four biochemical states. In order to distinguish genes within L_0 that are specific to Insulin signaling pathway and those that are not, a correlation threshold needs to be estimated for selecting network specific genes from L_0. This is obtained by comparing the correlation of activity levels of L_0 genes to network (L_{path}) and non-network genes.

A list of non-network genes ($L_{path.random}$) was generated, analogous to network genes (L_{path}), by excluding genes showing differential activity across any pair of biochemical conditions (among all the different biochemical conditions available) and excluding those genes from the complete set of genes representing the entire genome (~ 45,000 genes) and randomly sampling identical number of genes (42 genes) as the pathway genes (genes within L_{path}) from the remainder pool of genes (L_{random} ~ 35,000, L_{random} is expected to consist of a pool of genes which did not show any increase/decrease in activity in any condition and can be considered non-network genes).

The degrees of correlation between genes in L_0 and L_{path} and $L_{path.random}$ respectively were compared (comparison was performed 100 times by repeated sampling from L_{random}) by their fifth highest percentile values in the respective population using Response Operator Characteristics (ROC) curve for different threshold values of correlation coefficients. The rationale for comparing the 5th percentile value of correlation coefficients from the top between the two groups is because genes in L_1 were considered 'network specific genes' only if they were significantly correlated to at least 5% of the network genes. The correlation values between L_0 genes and the network genes and L_0 genes and the non-network genes was compared using a ROC (Response Operator Characteristic) curve (fig.6 left) by comparing the number of true positives (genes selected from L_1 if their fifth highest percentile value of correlation coefficient to network genes met the correlation threshold) and false positives (genes selected from L_1 if their fifth highest percentile value of correlation coefficient to non-network genes met the correlation threshold) for different correlation thresholds and the optimal correlation threshold to separate network specific and non-network genes within L_0 was determined to be 0.72. The Spearman correlation threshold $\rho \geq 0.72$ was derived as the optimal cutoff by maximizing the Youden's J index (=sensitivity+specificity-1, fig.6 right) based on the previously developed ROC (Response Operator Characteristic) curve comparing the correlation coefficients of L_0 genes to network genes and non-network genes. Genes from L_0 that exceeded a threshold of 0.72

in their degree of correlation to network genes (L_{path}) were considered /network specific genes (L_1=985 genes).

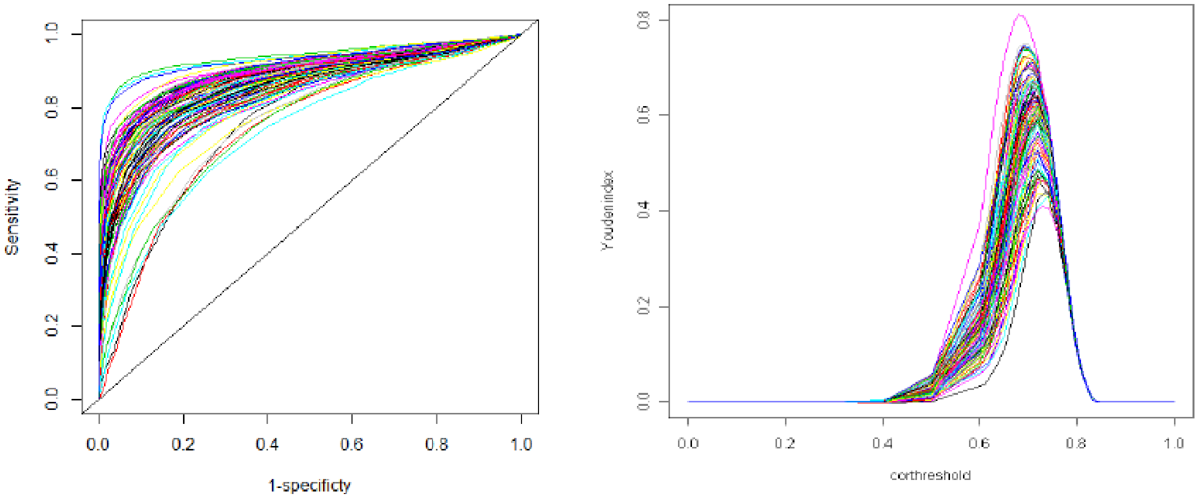


Figure 6. Determining the correlation threshold to select network specific genes (L_1) from L_0. Left. ROC curve comparing the correlation coefficients between genes in L_0 and network genes and L_0 and non-network genes for different correlation thresholds. **Right.** Youden's J Index plot to determining the optimal correlation threshold to distinguish network and non-network genes.

In order to map the network specific genes in L_1 as expanded networks around insulin signaling pathway genes, the genes within insulin signaling pathway were checked for differential activity in the same condition of contrast that was used to extract L_0 (B and D) so that the network expansion can be anchored around only the active genes in the network called the **anchor genes**.

Anchor genes for network expansion. Anchor genes (L_{anchor}) is a list of genes among the network genes (L_{path}) that has shown significant differential activity in the same pair of biochemical conditions that was used to derive L_0. They shall be the centers in the network around which gene networks shall be expanded. In the present study, among the network genes (L_{path}) forming the insulin signaling pathway, the number of genes showing differential activity between the two biochemical conditions B and D (selected using p-value<=0.01) was 15. Therefore, these 15 genes were selected as anchor genes (L_{anchor}, Table S2) for network expansion. These anchor genes, which showed significant expression changes during progression of insulin resistance as a result of diet, were used as anchors to identify other novel network genes from L_1 that showed similar expression changes and high associations with them to form networks around them.

Step4: Expansion of network around known network genes. Among the many anchor genes available to expand the network around, the specific anchor gene around which the gene network would be expanded is chosen based on the investigator's gene of interest. For example, in this case, we first attempted to expand gene network from IRS2 (insulin receptor substrate 2), a well-characterized early mediator of insulin signaling (step 3 in fig.5). In order to expand the network around the anchor gene IRS2, I identified the top 20 genes among L_1 genes that were most highly correlated with IRS2 across all the four biochemical conditions as candidate novel genes to form a network around IRS2. We explored potential network relationships between the 20 novel candidate genes and all the anchor genes (L_{anchor}=15 genes) in the network by Super Paramagnetic Clustering (SPC), a temperature annealing-based clustering technique that allows gradual selection of gene clusters based on their degrees of association (i.e., correlation of expression patterns) in an unsupervised manner. We added the other anchor genes as well to the focal anchor gene IRS2 to be clustered with the novel genes so that relationships can be created between the novel genes not just around IRS2 but with genes across the whole pathway. In order to form unsupervised clusters between the 20 novel genes introduced into the network and the 15 anchor genes in the pathway by SPC, a symmetric distance matrix ([X]) consisting of correlation distances between the genes was generated. The correlation distance between each pair of gene was estimated by 1- (spearman's correlation coefficient between the expression profiles of the genes across all four conditions). This distance matrix was then subjected to Super Paramagnetic Clustering (SPC) algorithm (Blatt et al., 1996).

Motivation for using SPC algorithm. Briefly, Super Paramagnetic Clustering (SPC) is a clustering algorithm developed by Dr. Eytan Domany's lab that takes advantage of stochastic property of Potts model of spin-spin clustering to cluster objects based on their distance under the influence of a theoretical temperature gradient. In this study, SPC algorithm was used to cluster genes based on the correlation distance between their expression profiles in the presence of a temperature gradient simulated from T=0 to 1 in 0.01 temperature steps. As the temperature of simulation was increased, only edges between genes that are strongly correlated in terms of their expression profiles were favored (probability of edge formation between two genes= 1-exp($-J_{ij}$/T), where Jij=$\frac{e^{\frac{-(\text{correlation distance})}{2*a^2}}}{K}$, K=number of nearest neighbors, a is parameter of simulation) and therefore, the concept of temperature gradient in SPC algorithm was useful for identifying strong gene associations within clusters as only such associations within clusters survived at higher temperatures. This concept is visually captured using an example of output of SPC algorithm (Fig.7).

Additional parameters of the simulation. In the present study, genes were assigned spins, s=2 (either positive spin or negative spin) and number of neighbors, K=10. The number of nearest neighbor (K, chosen by the user) assigned to each gene helped to control the size of clusters. Higher the number of nearest neighbors, larger the size of clusters. K=10 was optimal because it helped to produce clusters of size ~ 2 to 7 genes appropriate for network expansion around the various network genes. The executable C program file for SPC was kindly donated to our work by Dr. Eytan Domany, Weizmann Institute of Science, and was run in Linux environment to generate the SPC output files.

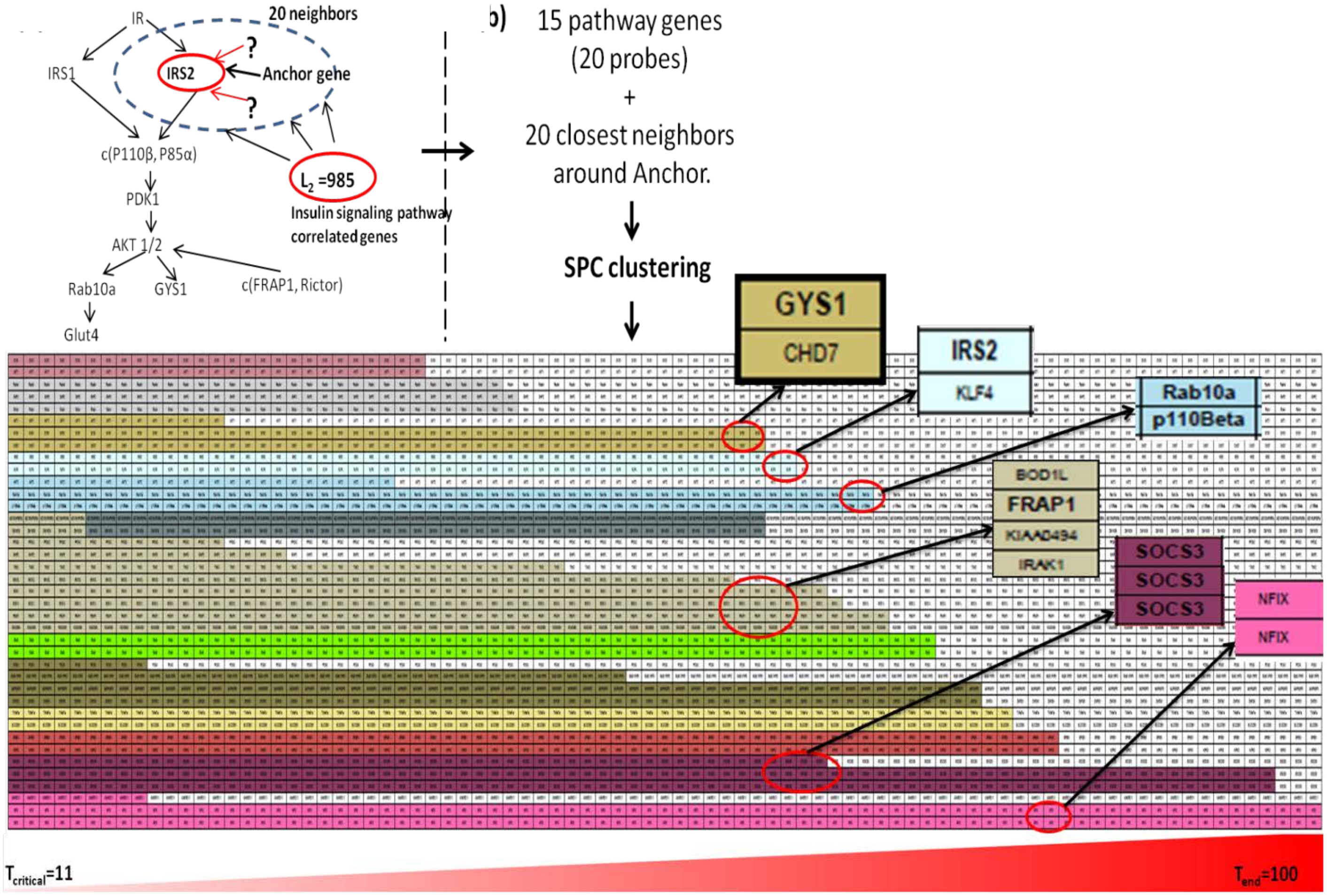


Figure 7. Visual representation of SPC clustering output and its interpretation. The clustering reaction between novel genes and the anchor genes, the differentially expressed prior network genes (insulin signaling pathway genes), was conducted using SPC clustering, a temperature annealing-based clustering technique using a theoretical temperature gradient. Temperature is shown as a red wedge in the bottom to indicate a rising temperature gradient from T=0 to 1 in 0.01 steps. The visual representation of the output of SPC algorithm shows that as temperature rises in the gradient (from left to right), only the stronger clusters survive. Longer a cluster survives the temperature gradient, reflects on how strong the gene associations are within the cluster. The purpose of clustering was to identify strong clusters between network and novel genes. Within the clusters that are shown in insets, the network genes (insulin signaling pathway genes) such as GYS1, IRS2 etc. are shown in bold. Clusters between novel and network genes can be used to expand the network around the network genes. The cluster CHD7-GYS1 was used to expand the network in fig.6 step4. Cluster KLF4-IRS2 will be discussed soon.

Since many clusters are produced by SPC, only certain clusters are useful for network expansion around prior genes. To gradually expand network relationships between network genes and novel genes, we selected clusters that contained one novel gene together with one or more anchor genes for network expansion, e.g. IRS2-KLF4 and GYS1-CHD7 clusters (clusters highlighted in fig.7). After selecting gene associations between novel and network genes from clusters, the network was expanded using the selected gene associations by creating edges between novel and network genes (as shown in step4 of fig.5). An example how the network was expanded using the gene associations is shown in step4 of fig.5. Since CHD7 formed cluster with GYS1 and was discovered as a novel gene neighbor of IRS2, approximate location for the novel gene in the network (CHD7) was determined using the location of the two anchor genes and thus the network was expanded using the novel gene. Similar steps of network expansion can be conducted around other anchor genes in the network and network can be further expanded. Having expanded the network by gene associations identified by BAKE clusters, it is important to validate these network expansions. Conducting biological experiments to verify these predicted gene associations would be the most rigorous way to validate gene network expansions. However, since biological experiments are expensive to conduct, in-silico confirmations prior to expensive validation techniques can add confidence to the predicted gene associations and can better guide of how to conduct the biological experiments.

Step 5. *In-silico* confirmation of novel network genes. For the novel network genes being bioinformatically discovered above by BAKE clusters, we closely examined their biological relevance to the network being expanded in-silico based on 1) the literature and public genome databases and 2) reverse BAKE application. The following is an example of using reverse BAKE application for in-silico confirmation of gene association originally discovered by BAKE itself. Among the different clusters produced by BAKE for expansion, the IRS2-KLF4 gene association was of special interest to me because a tight association between the expression levels of novel gene KLF4 and the network gene, IRS2 was previously observed (Modica et al., 2009). Since KLF4 is a well-known master gene regulator with many gene targets with key biological functions such as regulating the progression of cancer (Rowland and Peeper, 2006) , I was interested in expanding the network around KLF4 by treating KLF4 as it were a prior network anchor gene using the same strategy of network expansion of BAKE (Steps 3 and 4 of BAKE in fig.6) so as to check if BAKE can discover IRS2 if network were to be expanded KLF4 just the way KLF4 was discovered when network around IRS2 was expanded. This application of BAKE is called reverse BAKE application since it can add credibility to the tightly linked gene associations. This approach can also be viewed as serial BAKE application because we are expanding the network by BAKE around KLF4, which in turn was discovered by expanding the network around IRS2. This reverse BAKE application (or serial network expansion by BAKE application) is schematically represented by fig.8 below.

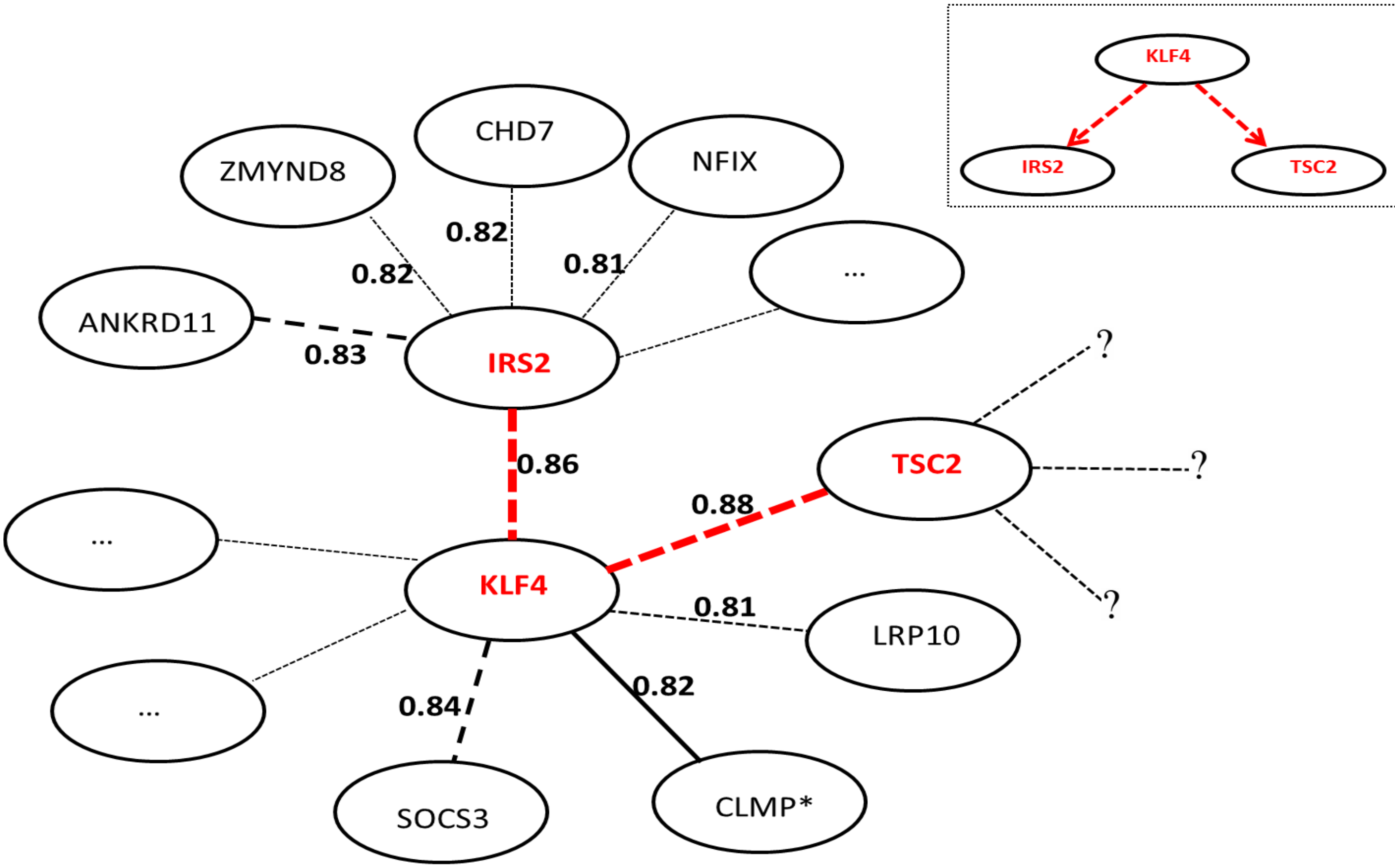


Figure 8. Reverse BAKE application process. The degrees of correlation between genes are shown on the edges. KLF4 was identified as strongly associated gene to IRS2 and TSC2 was in turn identified as a strongest neighbor of KLF4 upon network expansion around KLF4. Only select neighbors of IRS2 and KLF4 were highlighted based on their degrees of association. Based on the two strongest edges (shown in red) in the serially expanded network, I speculated that KLF4 could potentially regulate the genes, IRS2 and TSC2, based on previously reported regulatory roles taken by KLF4. This hypothesis of regulation of IRS2 and TSC2 by KLF4 is shown in the inset.

From this reverse BAKE application, I re-identifiied IRS2 as a well correlation neighbor of KLF4 (degree of correlation (ρ=0.86)), which confirmed the strong association between IRS2 and KLF4. Among the novel genes that were identified as closely associated neighbors of KLF4 upon network expansion were tuberous sclerosis protein 2 (TSC2, ρ=0.88), suppressor of cytokine signaling 3 (SOCS3, ρ=0.84), coxsackie- and adenovirus receptor-like membrane protein (CLMP, ρ=0.82) and low density lipoprotein receptor-related protein 10 (LRP10, ρ=0.81), all of which are known to play a role in insulin signaling and in adipocytes differentiation (Gabrielsson et al., 2005). Since KLF4 gene has served in regulatory roles in cellular growth and cancer, I decided to check if any of the neighbors of KLF4 identified by BAKE were potential KLF4 regulatory targets and could be validated by literature. I was able to find direct literature validation for the association between KLF4 and one of the neighbors of KLF4, CLMP. CLMP has been reported as a KLF4 target gene in mouse TM4 Sertoli cells (Sze et al., 2008). Among other KLF4 neighbors, I decided to pursue two of its most correlated neighbors, viz. IRS2 and TSC2, for further advanced experimental validation due to their high degree of correlation. In fact, snapshots of expression levels of these three genes across various biochemical conditions (Fig. 9 below) also seem to suggest strong associations between them. Having verified through multiple in-silico evidences that IRS2 and TSC2 could be good regulatory targets of KLF4 to be experimentally validated, biological experiment was then conducted to prove the same.

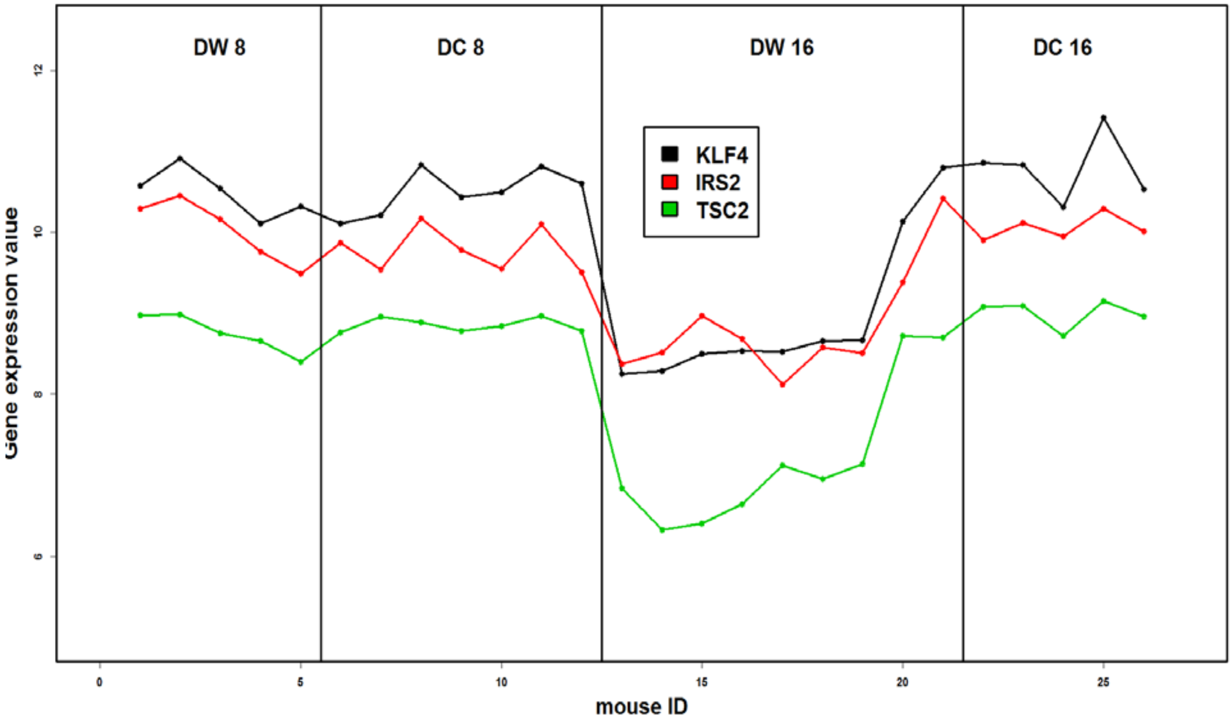


Figure 9. Static snapshots of expression levels of genes, IRS2, TSC2 and KLF4, across the four biochemical conditions, DW 8. DC8, DW16 and DC 16 (same as A, B, C and D respectively from fig. 5)

2.4. Experimental validation of BAKE inference about KLF4 regulating the expression of genes IRS2 and TSC2.

In order to test the above regulatory hypothesis, we used the property of co-expression of genes (property #2 in section# 1.2, that is, if the regulator gene undergoes change in expression, it is reflected in the expression levels of the regulated genes) to demonstrate regulation of IRS2 and TSC2 by KLF4. We used a knockdown mouse model of KLF4 (KLF4 +/-, mice with KLF4 activity reduced by half) in comparison to wild type (KLF4 +/+, in other words mice with regular activity of KLF4 gene) to experimentally verify if the activity of genes IRS2 and TSC2 were affected by the altered expression of KLF4. We found that expression of genes IRS2 and TSC2 were, in fact, significantly down-regulated with 3.5 folds (p=0.04) and 2.8 folds (p=0.057), respectively, in adipose tissues of KLF4 knockdown mice compared to KLF4 wild-type mice, experimentally confirming the regulation (**Fig. 10**). Thus we biologically validated strong gene associations discovered by BAKE that had significant literature evidences.

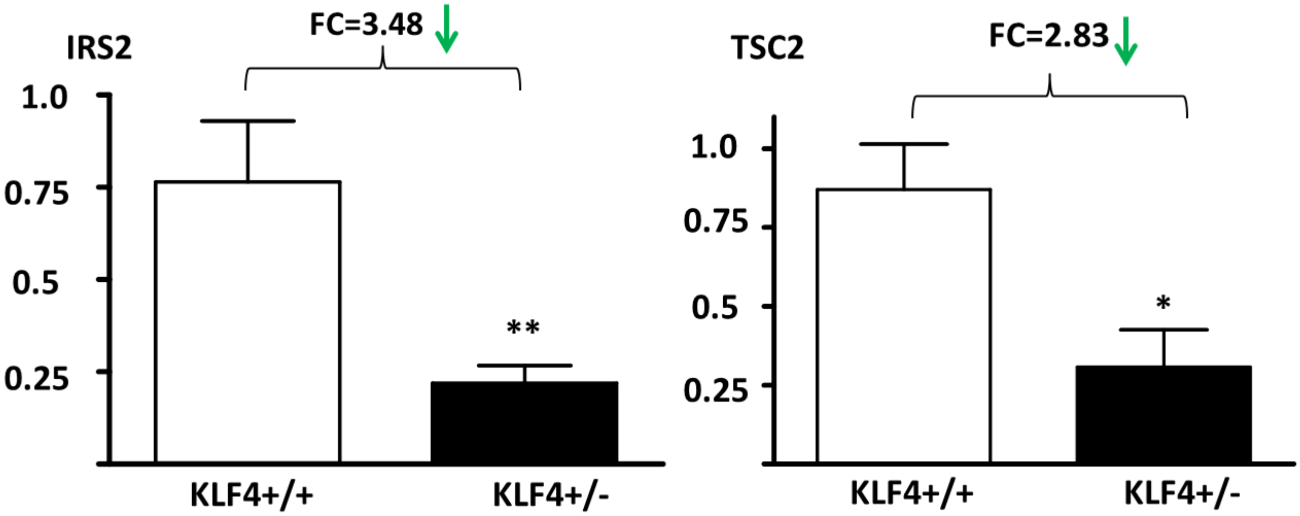


Figure 10. Developing and verifying a BAKE hypothesis using a biological experiment. Gene expression levels (represented as mean ± standard error (SE), n=4) of IRS2 (KLF4 +/+ = 0.77 ± 0.16, KLF4 +/- = 0.22 ± 0.05) and TSC2 (KLF4 +/+ = 0.87 ± 0.14, KLF4 +/-= 0.31 ± 0.12) across two KLF4 genotypes indicating down regulation of genes, IRS2 and TSC2 (by magnitude shown by Fold Change (FC)), due to decreased expression of KLF4 confirming the hypothesis that KLF4 regulates the expression of the two genes. * refers to p-value<0.06 and ** refers to p-value<0.05. MTAP1B gene was used as negative control.

A more objective method to validate the BAKE approach was to see how well BAKE can reconstruct a complete network by using only a partial set of genes from the network as prior network genes and expanding the network around them. The advantage of setting up the problem this way is you can measure the performance of the BAKE approach objectively because you would know the accuracy of the genes retrieved by BAKE. While deciding on the network to be constructed by BAKE using the gene expression data we extracted from insulin resistant mice, adipogenesis gene network seemed the most appropriate network since the gene expression data was extracted from the adipose tissues of insulin resistant mice. Therefore, the reconstruction task was set up as reconstructing the full adipogenesis network using select genes from the network that will serve as anchor genes for expansion so as to retrieve the hidden parts of the network. This network inference exercise is explained in the next section.

2.5. Computational validation of BAKE approach

Reconstructing adipogenesis network by BAKE approach involves first obtaining the full network with all its genes, separating part of the network from its full network and using those genes as anchor to expand the network (Step 2 of BAKE). The 5 step BAKE approach procedure was slightly modified and applied in the following way for validation. As an application of step 1 of BAKE approach (Biological context-dependent discovery of pool of novel genes, L_0), the pool of novel genes (L_0=1317 genes, derived from contrast of biochemical conditions, states B and D) that was used to expand the network around anchor genes in insulin signaling pathway was the same pool that was used to expand the network around adipogenesis genes. For step 2 of BAKE approach, Developing prior gene network relevant to the mechanism of interest, the full adipogenesis network genes was generated, L_{path} (84 genes) from literature. In step 3 of BAKE approach, Identification of novel genes associated with prior network, BAKE approach is modified as described below. List of pathway genes was first modified so as to generate the partial list of genes from the full list of network genes which will then serve as anchor genes to extract other network genes. Among the 84 network genes, 50 of them ($L_{pathsig}$, represented by 73 probes) showed significant differential activity (at 1% FDR). 20 genes were randomly sampled from these 50 network genes with differential activity ($L_{pathsig}$) to represent the anchor genes (L_{anchor}, shown in red in fig. 11 top panel). The remaining genes (L_{hidden}, represented by 53 probes) were part of the hidden network to be recovered by BAKE as a result of knowledge expansion around the anchor genes (L_{anchor}). Therefore, these hidden networks genes were added to the pool of novel genes (L_{0_mod}= L_0+ L_{hidden}) to test the ability of BAKE to extract these hidden genes (L_{hidden}), that is the relevant network genes, from the large pool of novel genes (L_{0_mod}) during BAKE expansion around and depending on how many of these hidden genes successfully discovered by BAKE, can thus help objectively measure the performance (precision and sensitivity) and judge the merit of BAKE approach of knowledge/network expansion. This is the key modification to step 3 of BAKE approach so as to facilitate objective validation of the approach by measuring BAKE's ability to extract relevant network genes (L_{hidden}) from L_{0_mod}. The next part of step 3 is to extract network specific genes, L_1, from $L_{0\_mod.}$

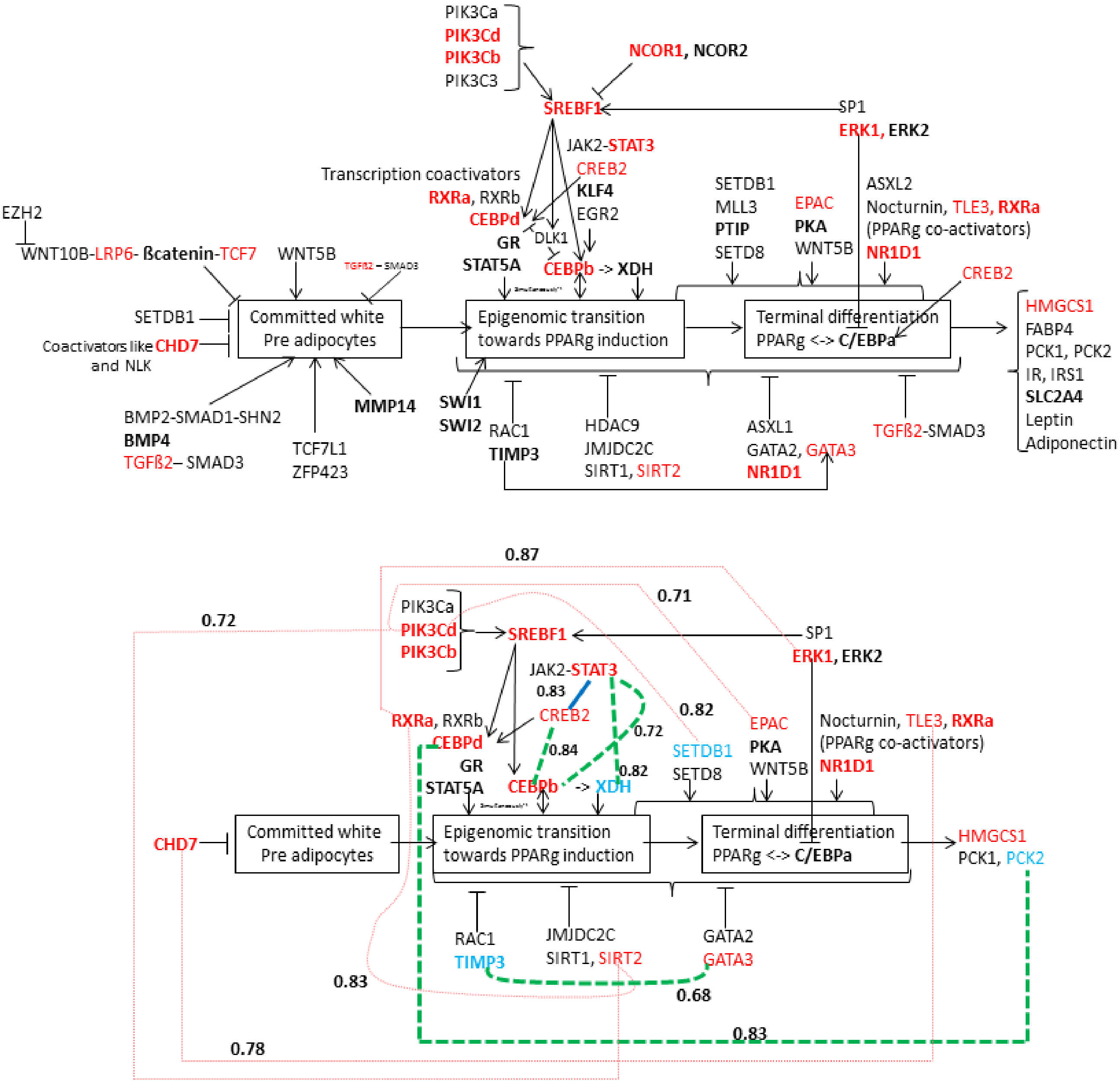


Figure 11. BAKE ability to reconstruct around a partial adipogenesis network. **Top panel.** The complete adipogenesis network. Genes with magnitude of $\log_2$(Fold Change)>=2 in the contrast B v D were highlighted in bold. Anchor genes in red were used as prior knowledge and the hidden genes (40 significant genes among the black genes) were expected to be retrieved by BAKE through knowledge/network expansion around the anchors**. Bottom panel** shows the result of reconstruction performed by BAKE on the partial network. The hidden genes retrieved by BAKE are shown in blue. Among the edges predicted by BAKE, only those with absolute degrees of correlation>=0.68 (correlation threshold determined from the ROC curve in step3 of BAKE to select pathway/network specific genes (L_1) from genome) are shown in green, blue and pink lines. Green dotted lines represent transcriptional interaction predicted by BAKE that have been confirmed by

literature (True Positives (TP)). Pink dotted lines represent transcriptional interaction predicted by BAKE currently without any literature support (False Positives (FP)). Blue solid lines represent edges predicted by BAKE that are either non transcriptional in nature or transcriptional interactions that appeared as weak predictions by BAKE (Weak True Positives (WTP)).

To increase BAKE's ability to extract network relevant genes (L_{hidden}) from L_{0_mod} using BAKE approach, we used additional gene expression data made available from some alternate biochemical conditions. The four previously biochemical conditions, A, B, C and D, were derived from mice genetically predisposed to insulin resistance. The additional gene expression data was made available from mice genetically not predisposed to insulin resistance (such mice were genetically known as ApoE mice). They were also fed two different diets (high fat and low fat diet) and adipose tissue was extracted from them at two time points (8 and 16 weeks) as well to generate three groups of experimental mice representing additional data points representing alternate biochemical conditions (Condition E- mice fed with high fat diet and adipose tissue extracted from them at 8 weeks (n=6), Condition F- mice fed with high fat diet and adipose tissue extracted from them at 16 weeks (n=8) and Condition G- mice fed with low fat diet and adipose tissue extracted from them at 16 weeks (n=5)). These new biochemical conditions when also added to the analysis gave rise to 7 biochemical conditions in total or, in other words, 45 data points, for estimating degree of correlation between gene activities in all future steps of BAKE. These additional data points improved accuracy of estimating degree of correlation between genes because representing gene activity over wider physiological conditions helped to better estimate the degree of correlation coefficients between gene activities.

The first key step is to reduce thousands of genes in the large volume of genes (L_{0_mod}) to what is relevant to the adipogenesis network (L_1). Novel genes relevant to the Adipogenesis network (L_1=919 probes or 857 genes) was selected from L_{0mod} genes if their top two percentile correlation coefficient value to adipogenesis network genes (L_{path}^{mod}) exceeded a correlation threshold of 0.68 (threshold determined from ROC curve comparing correlation of genes in L_{0_mod} to both network and non-network genes and the correlation coefficient offering optimal separation of network relevant and non-network genes within L_0 was 0.68). The remaining steps of BAKE were followed till the clusters produced by SPC were selected for network expansion around anchor genes

Summary of performance of BAKE in reconstructing adipogenesis network. Genes retrieved by knowledge expansion around anchor genes were classified as classified either as True Positives (TPs) (hidden genes that were retrieved by BAKE) or False Positives (FPs) (new genes not previously reported to be part of the adipogenesis network). Among the 37 genes that were introduced in the network by BAKE, 6 of them represented hidden genes (four of the hidden genes retrieved by BAKE in blue in **fig. 11 bottom panel,** the remaining two were not indicated because they were alternate probes/transcripts of anchor genes, TGFβ2 and CEBPβ) and the remaining were not part of the initial network. Therefore, precision (TP/ (TP+FP)) of BAKE, which indicates the rate of selection of biologically relevant genes (TP) to the network to all the genes retrieved by BAKE was 16% (6/37). This ratio estimate is conservative because some of the novel genes classified as False Positive (FP) may be relevant to the network but could not be confirmed by us either by literature or by experiment. Of the 40 hidden genes, 6 were retrieved by BAKE and hence BAKE's sensitivity (extent of ability to retrieve relevant genes in the network, 6/40, 15%) was low. However, since the estimate is conservative, sensitivity can also be expected to be a little higher.

Although BAKE’s ability to identify biologically relevant genes by knowledge expansion may not have been high, the algorithm's ability to identify the connections of the novel genes introduced in the network with respect to the network genes (associations predicted in the network by BAKE through gene clusters) correctly was better. Please note that, among the predicted associations, only those with absolute degree of correlation >=0.68 are shown in **fig 11 bottom panel**. Therefore, though the total tallies of gene associations are discussed below, not all of them are shown in the figure. Predicted associations between genes were of three categories: between just network genes, between network and novel genes introduced into the network by BAKE and between only novel genes.

40 gene associations were predicted by BAKE between only network genes (shown in red in **fig 11 bottom panel**) which were then evaluated for their accuracy using gene regulatory information from Ingenuity Pathway Analysis (IPA) network analysis tools and classified into three categories: (i) associations that were supported by literature (True Positives (TP)=10 gene associations, shown in green lines in **fig 11 bottom panel**), (ii) associations that were not supported by literature (False Positives (FP)=22 gene associations, shown in pink lines in **fig 11**

bottom panel) and (iii) associations that were Weak True Positives (WTP=7 gene associations, treated as half TP, WTP=0.5*TP, shown in blue lines in **fig 11 bottom panel**), that is, associations between genes that were not of regulatory nature but other forms of interactions (like phosphorylation, protein binding etc.), excessively long distant and non-linear gene associations etc. Among the associations predicted by BAKE between network and novel genes, only those associations involving novel genes that were also hidden genes (shown in blue in **fig 11 top panel**) were evaluated (6 associations, TP = 4, WTP=1 and FP=1). Other gene associations that were derived from BAKE clusters were not evaluated because of the unknown relevance or function of novel genes in the network. The overall ratio of true gene associations (TPs and WTPs) to all the associations (40 edges) predicted by BAKE (precision) was 44%. Estimate of BAKE's precision of gene associations is also expected to be conservative and higher.

2.6. How does the performance of BAKE compare to other algorithms?

We evaluated the ability of BAKE approach to expand a partial adipogenesis network to a full network using gene expression data from insulin resistant mice and the approach demonstrated >44% recovery rate of true positive network relationships among all the network relationships predicted by BAKE. In an effort to compare performance of BAKE to its other contemporary network reconstruction algorithms, the closest measure of performance of algorithms was done by DREAM. Dialogue on Reverse Engineering Assessment and Methods (DREAM) is a community wide challenge for objective assessment of network inference methods for biological networks and the third edition of the challenge, DREAM 3, had measured the performance of many of the current crop of network inference algorithms that used time series and static gene expression data. More details on DREAM to follow later because DREAM data was used to validate my time series network inference approach, MANI, in the next chapter. It was not used to validate BAKE because the DREAM 3 challenge involved using deletion data and time series data and no prior knowledge was available about the network.

The DREAM3 challenge had measured the performance of many contemporary network inference algorithms over common data sets to objectively compare their performances. The amount of data used for network inference in DREAM3 challenge by the best performing algorithm (yip et al. 2010) used 4, 23 and 46 different perturbation time series data sets (each consisting of 21 time points) and additional causality information through gene knock out and knock down data to construct networks of sizes 10, 50 and 100 genes respectively at a precision of <50%. In DREAM 5 challenges for a genome wide network inference, consensus network for *E. coli* and *S. aureus* with 1700 edges was constructed with a precision of ~ 50% from 53 and 487 experimental conditions consisting of 160 and 805 microarrays (microarrays represent the # of data points for gene expression) respectively. BAKE algorithm used relatively less and much simpler data (45 microarrays from 7 experimental conditions) and reconstructed 46 edges of adipogenesis network between 50 genes to achieve similar precision (44%) compared to other contemporary algorithms in DREAM3 challenge while constructing target network of a similar size.

2.7. Weakness of BAKE

It is worthwhile to note some of the limitations and weaknesses of BAKE. Most notably, while we have tried to objectively define all analysis steps in the BAKE approach, there are still several steps that are dependent on a researcher's subjective selection and decision on network inference (since it is only approach and not a hard algorithm) such as the use of specific gene expression data for the biological conditions of interest and prior gene network, especially the latter information often varies with different degrees of uncertainty from the literature and the curation of large genetic databases. Using BAKE, a researcher's novel network gene discovery will inevitably depend on these factors. We, however, believe that it is necessary to subject our network investigation to each researcher's experimental data that have been obtained specifically for their biological conditions of interest.

Despite these limitations, we stress that utilizing these subjective data and information under systematic BAKE analysis steps enabled us to identify direct relevant novel network factors of interest, avoiding numerous false positive discoveries, which we term "*objective use of subjective information*." BAKE network predictions are based on static data and cannot resolve direction of edges in the network and complex motifs of gene regulations such as feedback loops, cis regulations, interactions such as AND, XOR and other combinatorial regulations. To further improve BAKE predictions, time series data, gene knock out/down data and an algorithm to solve those data in combination with BAKE should be used in order to achieve a greater resolution in network inferences. We suggest one such approach that can construct network from time series data in the next section. Nevertheless, BAKE still is an important contribution for its novelty in combining large volume data incorporating existing network knowledge to achieve significant knowledge expansion.

Chapter 3: Mechanism Anchored Network Inference (MANI) approach and its in-silico validation

3.1. Problem definition and goals

In order to solve for the gene network involved in adipogenesis cascade, gene activity data was collected for 7 key genes involved in the cascade (names of the 7 genes in table 1) over the course of time from a study on adipogenesis (Cheung et al., 2007) as shown below in **Fig. 12.**

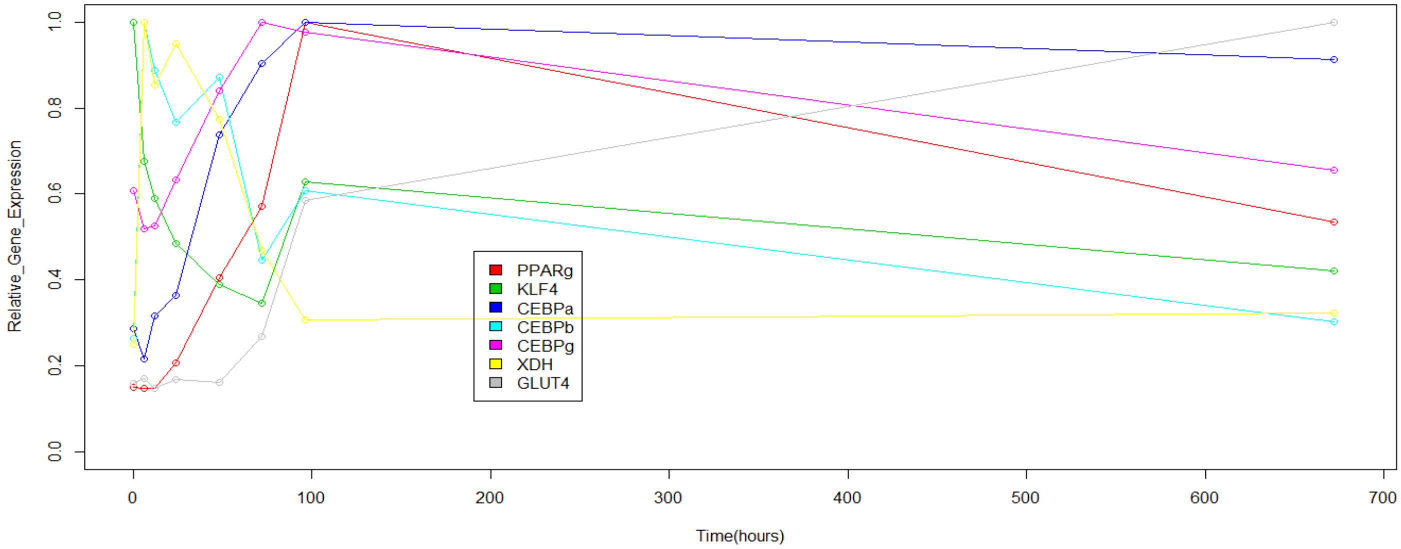


Figure 12. Time series gene expression data of the 7 genes belonging to the adipogenesis cascade. Their gene expression values have been normalized to have maximum value of 1. The expression data of genes was collected during the following time intervals (measured in hours) {0, 6, 12, 24, 48, 72, 96, 672}.

The previous network inference approach, BAKE, constructed gene network using static snapshot data of gene activity across different biochemical conditions and incorporated prior knowledge about the network during inference. However, if you are presented only a time course gene expression data, how would you construct the network between the genes? The current problem is to solve the network between the 7 genes in fig. 12 constituting the adipogenesis cascade using their gene activity data measured over time.

As motivated in the introduction (Chapter 1), I have deviated from the traditional approach to globally construct the network of the 7 genes from their time series data and instead opted to construct the network locally from the individual motifs (made of three genes) making up the network. The approach is therefore termed as Motif Anchored Network Inference (MANI) and I will show the implementation of MANI approach here by constructing network between the seven genes from the above time course expression data (Fig.12).

Table 1. Names of the 7 genes in the adipogenesis cascade

Gene symbol	Gene name
KLF4	Kruppel like factor 4,
CEBPa/ CEBPα	CCAAT/enhancer binding protein-Alpha
CEBPb/ CEBPβ	CCAAT/enhancer binding protein-Beta
CEBPg/ CEBPγ	CCAAT/enhancer binding protein-Gamma
GLUT4	Glucose Transporter type 4
XDH	Xanthine Dehydrogenase
PPARg	Peroxisome Proliferators-Activated Receptor-g

3.2 Implementation of MANI approach to solve the adipogenesis network

Since the goal of MANI approach is to construct the network through the basic building block of motifs in the network, the motifs of the network are identified through a metaphorical 'window' that covered three genes in the network at a time and the window was then gradually moved through the entire network. Therefore, to implement MANI, the sequence of steps involved identifying the location of the initial window(s), fitting the best possible mechanism of gene regulation for genes within the initial window(s), followed by migrating the window to the next location in the network and fitting the mechanism for the new genes within the new window with respect to the genes already covered in the previous windows and the last two steps are repeated until all the genes in the network are covered at least once. These broad principles of the MANI approach were implemented through the following 4 steps to construct network from the time series data of the 7 genes in fig. 12.

Step1. Selecting initial window(s).

This step describes the criteria for choosing genes for the initial window. The initial window was created by selecting pair(s) of genes with maximum correlation between their time series expression profiles and then selecting a third gene to complete the window by choosing a gene (outside the pair of genes) with maximum correlation to either of the genes forming the pair. The correlation matrix in Table 2 can be used to form the windows.

Table 2. Correlation matrix of the 7 genes belonging to the adipogenesis cascade.

	PPARG	KLF4	CEBP_a	CEBP_b	CEBP_g	XDH	GLUT4
PPARG	1.00	0.53	0.87	0.30	0.85	0.32	0.75
KLF4	0.53	1.00	0.63	0.27	0.72	0.37	0.42
CEBP_a	0.87	0.63	1.00	0.13	0.88	0.17	0.67
CEBP_b	0.30	0.27	0.13	1.00	0.07	0.88	0.05
CEBP_g	0.85	0.72	0.88	0.07	1.00	0.03	0.63
XDH	0.32	0.37	0.17	0.88	0.03	1.00	0.05
GLUT4	0.75	0.42	0.67	0.05	0.63	0.05	1.00

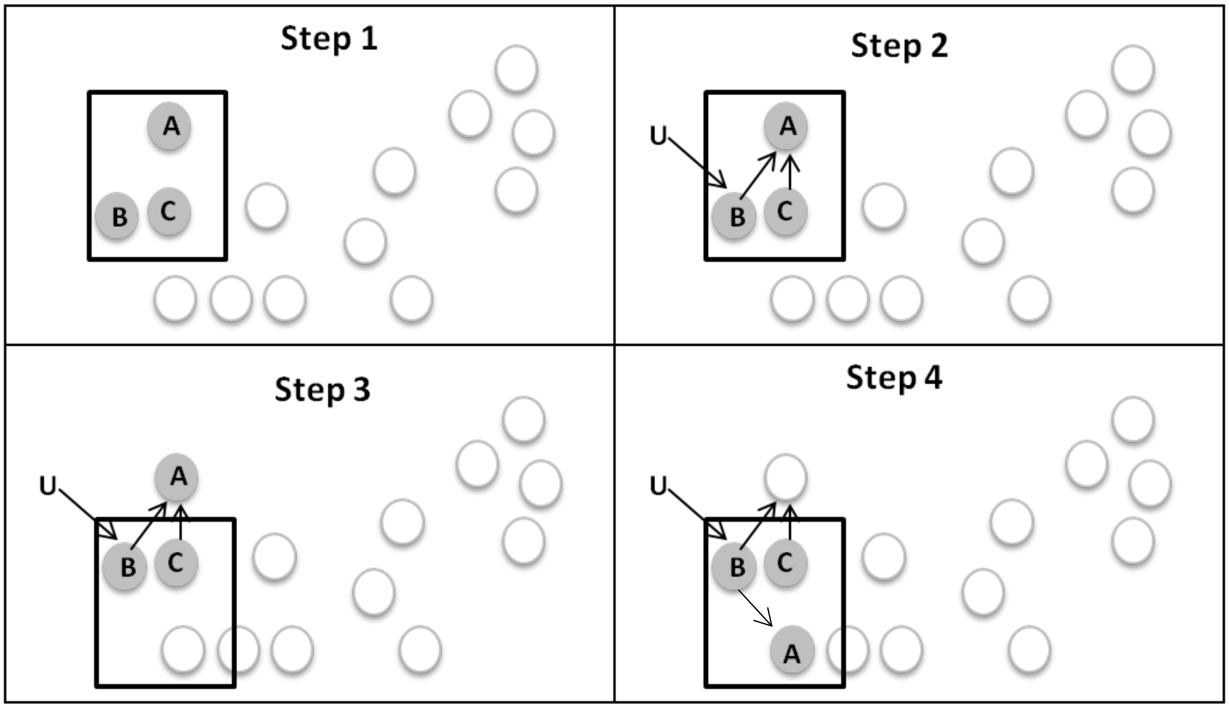


Figure 13. Implementation of MANI approach towards gene network inference. (a) Step1: Selection of initial genes (A, B and C) for the window, (b) Step2: Constructing the network among the three genes in the window by fitting different motifs based on either a parallel or sequential or parallel-sequential mechanism of regulation. U is an external input to the system (Input is optional with some motifs), (c) Step 3: One Gene In and One Gene Out. Adding a new member to and removing an old member from the window to keep the window to size 3. (d) Step 4: Fitting the motif with respect to the new gene in the window. Steps 3 and 4 were carried out in a loop until the window covered all the genes in the network at least once in its path.

In the case of adipogenesis network, selecting pair(s) of gene among the 7 genes with maximum correlation of their time series expression profiles resulted in two pairs of genes (**pair_1**={XDH-CEBPb} and **pair_2**={CEBPa-CEBPb}); both with degrees of correlation, ρ=0.88. Selecting the third gene to complete the window was performed by selecting a gene from the list outside the pair that had maximum correlation to either gene in the pair. Therefore, **window_1**={KLF4-XDH-CEBPb} was obtained by adding KLF4 to pair_1 and **window_2**={CEBPa-CEBPb-PPARg} was similarly constructed by adding PPARg to pair_2. Expression profiles of genes within the two windows are plotted in figs. 14 and 15.

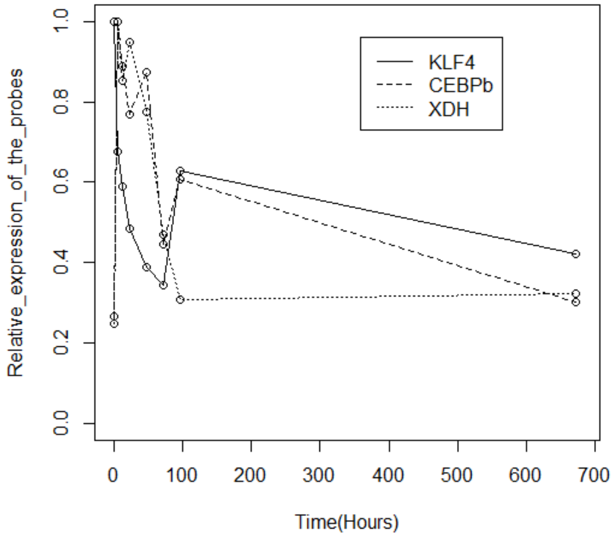


Figure 14. The normalized expression profiles of the 3 genes in window #1 created for adipogenesis network inference. Note that CEBP_b is same as CEBPb or CEBPβ.

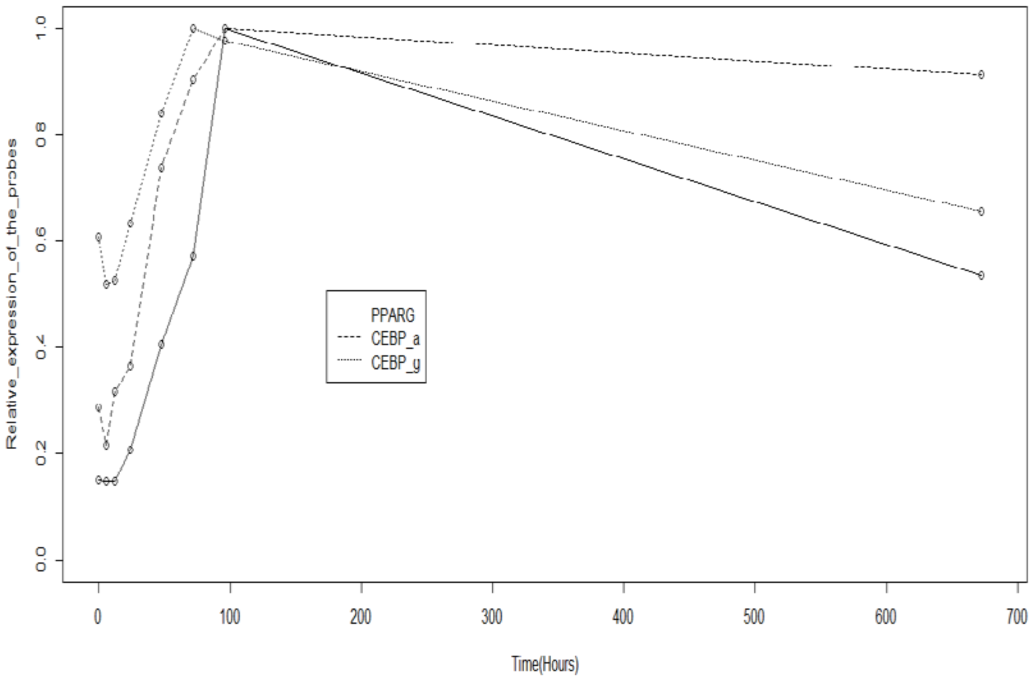


Figure 15. The normalized expression profiles of the 3genes in window #2 created for adipogenesis network inference. Note that CEBP_a is same as CEBPa or CEBPα. Similarly CEBP_g is same as CEBPg or CEBPγ.

Step 2. Fitting the best possible mechanism of gene regulation for genes selected within the initial windows.

Solving for the mechanism of regulation between genes within the window involved testing various motifs of gene regulation such as serial gene regulation, parallel gene regulation, a combination of serial and parallel regulations etc. followed by selecting the most optimal motif that fits the data of the genes. The various motifs that can be potentially tested on a window of three genes are listed below.

Preliminary motif development. Before exploring the different motifs to capture the gene relationships within the window, a preliminary step could be to check if the time course expression data of genes within the window showed differences in their duration of lags. The differences in the lag between expression profiles, as highlighted in introduction (Chapter 1), gives clues on the hierarchy of gene regulation, that is, which gene comes higher in the order of regulation. Gene with shorter lag would appear higher in the hierarchy of regulation. The duration of lag from an expression profile could be objectively estimated as the last time point after which the expression profile sees a continuous increase or decrease in gene expression for 2 or more time points. For example, applying the above rule in estimating lag, in window # 2, the duration of lag for gene CEBPa is 6 hours and gene PPARg is 12 hours. Since CEBPa has shorter lag compared to PPARg, CEBPa can be expected to be higher in the hierarchy compared to PPARg in the order of regulation.

Using this information about gene hierarchy, one could develop simpler forms of gene relationships before fitting the motifs to come later. For example, if gene G1 within a window had shorter lag compared to G2, then a gene relationship could be described between the two genes such that an external input acts on G1, which in turn regulates the activity of G2.

Motif #1.1: U -> G1 -> G2

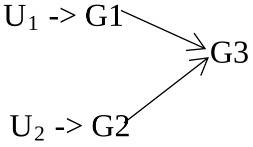
If there was no discernible difference in the lag of genes in a window, then different combinations of genes were tried in the locations of G1 and G2 and the combination that best fitted the data was chosen as the optimal relationship between the genes, which can then be used to build the motif further. If such relationships as described in motif # 1 did not produce good fit for the data, the default motif could be that all three genes in the window would have a separate input of their own.

Motif #1.2: U -> G1

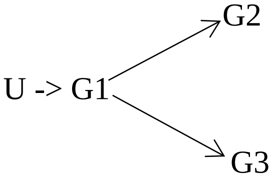
Advanced motifs. Based on the relationships developed in preliminary motif development, the motif development could be completed by fitting one of the following relationships for the three genes in the window.

Motif #2 (Serial Regulation): U -> G1 -> G2 -> G3

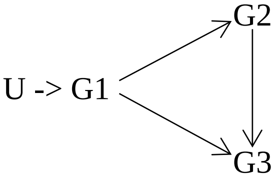
Motif #3 (Multiple Regulation):



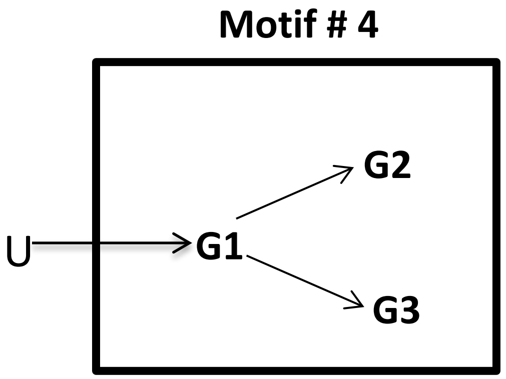
Motif #4 (Parallel Regulation):

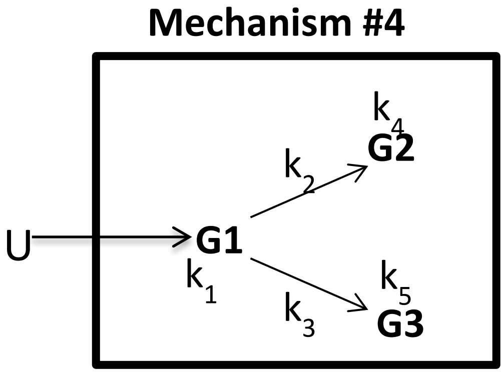


Motif #5 (Serial-Parallel Regulation):



Though the term motif has been used to describe the structure of gene relationships between the three genes in a window, the term called 'mechanism' refers to the motif representing the gene relationships in a window along with the associated kinetic parameters of the relationships. See below for example.





Estimating parameters of a mechanism and judging the best motif. Bayesian Information Criterion **(**BIC) score for each motif, derived from the aggregate Sum of Square of Errors (SSE) for all three genes in the windows, was used as the goodness of fit measure to select the most optimal motif to describe the regulatory relationships between genes in a window

BIC= $\frac{\sum_{i=1}^{g} SSE_i}{\tau}$ + p_{total}. ln (n) ...(10)

Where

$\sum_{i=1}^{g} SSE_i$ refers to sum of Sum of Square of Error (SSE) of fit for all the genes within the window (g refers to the number of genes within the window).

τ refers to the standard deviation in error distribution.

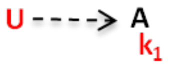
p_{total} refers to the total number of parameters in the model

n refers to the number of data points that was used to fit the model.

The derivation of the above relationship for BIC is shown in supplementary material using the example of a generic window consisting of three genes. The various parameters of regulation were estimated using SIMBIOLOGY toolbox within MATLAB R2012b (using SBIONLINFIT command in SIMBIOLOGY toolbox).

Applying the principles discussed above, I will now find the motif that best describes the regulatory relationships between genes within windows #1 and #2.

Window # 1. As can be seen in fig. 14, there is no significant difference in lag between genes in window #1. In the absence of apparent difference in lag, in order to determine the first gene in the hierarchy, I tested all three genes within the window in that position. Mathematical description of a gene at top of the hierarchy is as shown below.



$$\frac{dA}{dt} = U - k_1.A \qquad ...(11)$$

Where
A= KLF4, XDH or CEBPb
k_1= rate of degradation of mRNA molecules (molecule that represents the activity/expression levels) of gene A
U= constant stimulus or input that regulates the activity of gene A.

The various choices of A lead to the following results: Sum of Square of Error (SSE) of fit for the different genes are SSE_{KLF4} = 0.0488, SSE_{XDH}= 0.5220 and SSE_{CEBPb}=0.3906. This lead to choosing KLF4 as the best fitting gene at the top of the network in window 1. Based on KLF4's position as the first gene in the hierarchy, few different positions in terms of regulations were possible for the other two genes within the window (parallel and serial motifs are shown in fig. 16 below). Due to poor fit of an external input acting on either genes XDH or CEBPb, multiple regulatory motif was not attempted in the present window.

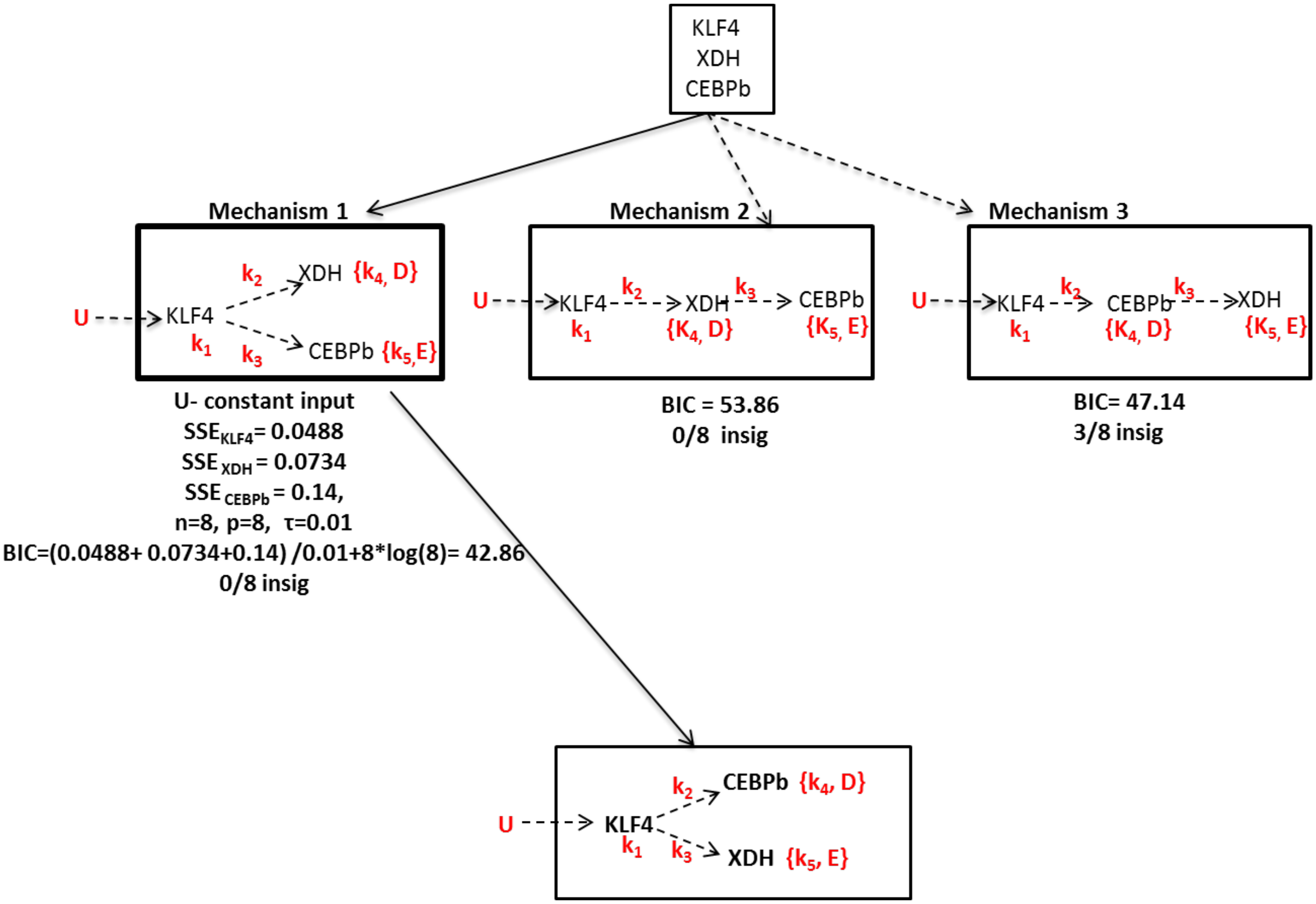


Figure 16. Window #1 network inference. BIC was estimated as $(SSE_{KLF4} + SSE_{XDH} + SSE_{CEBPb})/\tau + p_{total}.\ \ln(n)$. Kinetic parameters associated with the motifs that were estimated are highlighted in red (the exact physical role of the kinetic parameters are explained in the mathematical model of the motif shown below). If the parameters represented the magnitude of regulation of the gene, they were located over the edge connecting the genes but if they represented rate of mRNA degradation, they were located under the name or on the side of the gene. If the degradation process had more than one parameter, they were included within {}. Serial and Parallel regulatory motifs were attempted to model the gene data in window 1 and mechanism 1 (Parallel regulation) has been chosen as the best mechanism (bottom) to describe the genes in window 1 because of its smallest BIC score. The final set of parameters estimated for the best mechanism is available in Table 3. The number of parameters estimated (parameters in red) is indicated below the BIC score for each mechanism and the fraction of those estimated parameters that were either over fitted or found to be insignificant in the optimal fit were also noted.

Mechanism #1 in fig.16 is an example of a parallel regulatory mechanism and such a regulatory relationship between genes can be mathematically described in the following way.

$$\frac{d(KLF4)}{dt} = U - k_1.(KLF4) \qquad ...(12.1)$$

$$\frac{d(XDH)}{dt} = k_2.(KLF4) - k_4.(XDH) - D \qquad ...(12.2)$$

$$\frac{d(CEBPb)}{dt} = k_3.(KLF4) - k_5.(CEBPb) - E \qquad ...(12.3)$$

where state variables KLF4, XDH and CEBPb represent the relative # of mRNA molecules of the respective genes, k_2 and k_3 represent the rate of regulation of the activities of genes, XDH and CEBPb, by KLF4. k_4 and D are the rates of degradation of mRNA molecules of gene XDH while k_5 and E are the rates of degradation of mRNA molecules of gene CEBPb.

Table 3. Values of parameters estimated for the chosen mechanism in window #1.

Parameters	**mean +/- standard error (hr^{-1})**
D	0.12 +/- 0.02
E	0.1 +/- 0.02
k_1	0.13 +/- 0.05
k_2	0.28 +/- 0.04
k_3	0.24 +/- 0.05
k_4	0.03+/-0.01
k_5	0.02+/-0.01
U	0.06+/-0.03

Since window #2 inference is similar to window #1, in the interest of space, I have not included the steps but the final motif of regulation that was inferred for window # 2 was

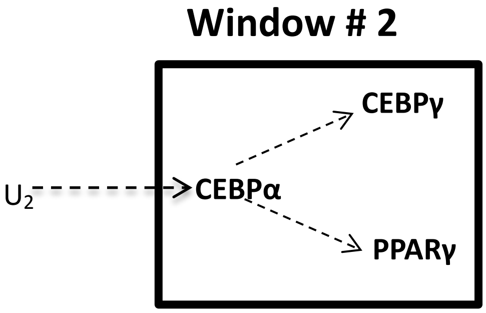


Figure 17. Window #2 network inference.

Simplifications to your model. In the above mathematical model that was used to describe the regulatory relationship within window # 1 (equations 12.1 to 12.3), some simplifying assumptions were made to the model and will be made to future models as well.

a) In the above model, mRNA molecules, that represent the level of activity of a gene, seem to be directly regulating the activity of other genes. However, the true picture is that mRNA molecules do not directly regulate the activity of other genes but instead produce protein molecules that in turn regulates the activity of other genes (Chapter 1). The assumption I have made here is the number of mRNA molecules produced by a gene is correlated to the number of protein molecules also produced from the gene. In other words, in my model, variables representing mRNA molecules and protein molecules have been lumped into a single state variable.

b) The rate of regulation of a gene by other regulating genes in my mathematical model has been approximated as additive and independent. The drawback of using such a model is that it is unable to describe finer aspects of combinatorial (AND regulation, OR regulation etc.) or synergistic regulation. However, using a simplified model as used by the author helps to keep the model from being over fitted and also easily interpretable.

Step 3. Advancing the window(s) forward

Once the optimal motif representing the regulatory relationships between genes within the existing window(s) has been solved, the window(s) is moved forward to include a new gene(s) in the window. A new gene is introduced into the window using One gene In, One gene Out (OIOO) rule. The rule was enforced by adding a new gene to the existing window by finding the gene (among the remaining genes outside the window) that had the highest correlation to any of the genes inside the window and the gene in the window least correlated to the new gene was discarded. By keeping at least one gene and its interactions from the last window, you limit the number of network structures possible in the motif space with respect to the new gene(s) and also limit the size of the window to just 3 genes. If introducing the new gene forms the same window as before (with the exact same set of genes as one of the earlier windows), the rule is relaxed to include the next highest gene in terms of its degree of correlation to the genes in the window. The rule is repeated until a new window is obtained. Sometimes, it led to introducing two new genes instead of just one new gene to the window.

For adipogenesis network inference, having solved the network among genes in the two initial windows (windows # 1 and 2), the two windows were advanced. Therefore, window #1 was advanced by replacing gene XDH with gene CEBPg because correlation of CEBPg to KLF4 (ρ=0.72, refer to the correlation matrix) was highest among the genes outside the window to any gene inside the window. The weakest of the three genes correlated to CEBPg in window #1 was XDH (ρ=0.03). Therefore, window #1 ({KLF4, XDH and CEBPb}) was updated to window #3 ({KLF4, CEBPb and CEBPg}). Similarly window #2 ({CEBPa, CEBPg, PPARg}) was updated to window #4 ({GLUT4, PPARg and CEBPa}).

Step 4. Reconstructing the new window(s) created

For the newly created windows, the network structure was solved for the new gene introduced into the windows with respect to the old genes already present in the window. For example, in window # 3, the position of the new gene, CEBPg (or CEBPγ), was solved for with respect to genes from the previous window, KLF4 and CEBPb (or CEBPβ). The time course expression profile of genes in window # 3 (not shown) indicated noticeable lag for CEBPg compared to genes KL4 and CEBPb. Since CEBPg is the only new gene in the window, the goal was to test the various mechanisms to fit the new gene with respect to the other genes in the window. KL4 and CEBPb will carry their attributes (edge directions and parameters estimated) from their previous window, window # 1. Similarly, CEBPg also will carry its edge (its relationship with gene CEBPa) from its previous window, window # 2 (fig. # 17) to the present window. While trying to fit the possible mechanism of regulation for the new gene, I used the longer lag of the new gene with respect to the other two genes to test the following mechanisms (mechanism based on serial, parallel and serial-parallel motifs) as shown in fig.18. One other alternate motif that was tested for window # 3 apart from the regular list of motifs was the null hypothesis motif. Null hypothesis motif refers to a scenario with no new regulatory edges introduced in the window.

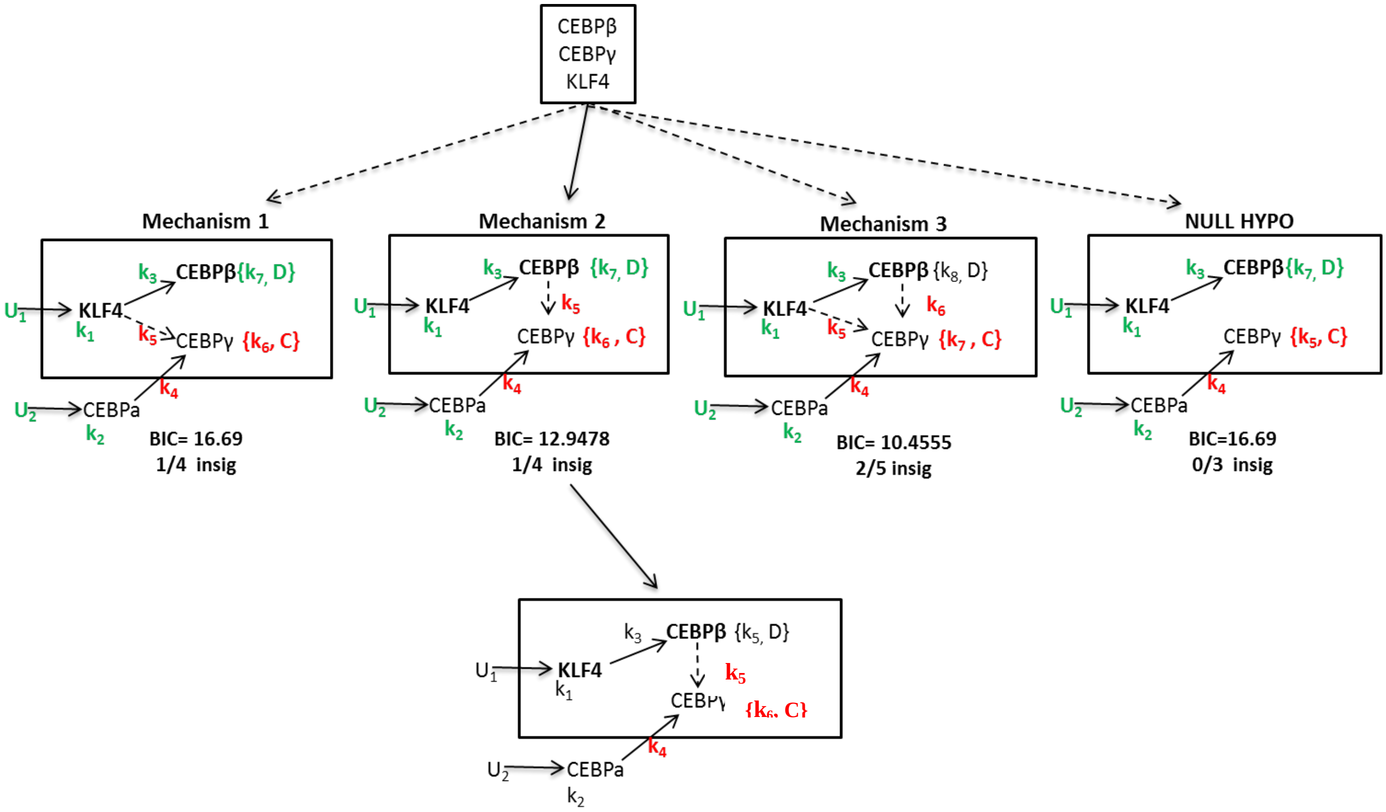


Figure 18. Window #3 network inference. Mechanism 2 (sequential regulation pattern) has been chosen as the best mechanism (bottom) to describe the genes in window #3. The estimated parameters are highlighted in red and the parameters already estimated in previous windows are indicated in green. The edges from previous windows (windows #1 and #2) are shown in solid lines (KLF4-CEBPβ and CEBPa-CEBPγ) and the new edges hypothesized in this window are shown in dotted lines. Though CEBPγ already has been already modeled to be regulated by CEBPa from window 2, this window is an opportunity to identify additional regulations on the gene and observe any improvement in its fit as a result of additional regulations. The 'Null Hypothesis' mechanism representing no additional regulations for CEBPg is titled as NULL HYPO. The values of parameters estimated for the chosen mechanism is available in Table 4.

Table 4. Values of parameters estimated for the chosen mechanism in window #3

Parameters	mean +/- standard error (hr^{-1})
k_4	0.07 +/- 0.02
k_5	0.08 +/- 0.02
k_6	0.03 +/- 0.01
C	0.08 +/- 0.03

The difference between the NULL HYPO mechanism and the chosen mechanism (Mechanism #2) is the additional regulation by CEBPb. The fit of CEBPg were compared between the two cases (Fig.19) and it can be inferred that the fit of CEBPg improved when jointly regulated by CEBPa and CEBPb than when regulated only by CEBPa. This example of window highlights the strength of MANI to uncover multiple regulations on a gene. Window# 4 inference was not discussed in detail but the same principles were applied.

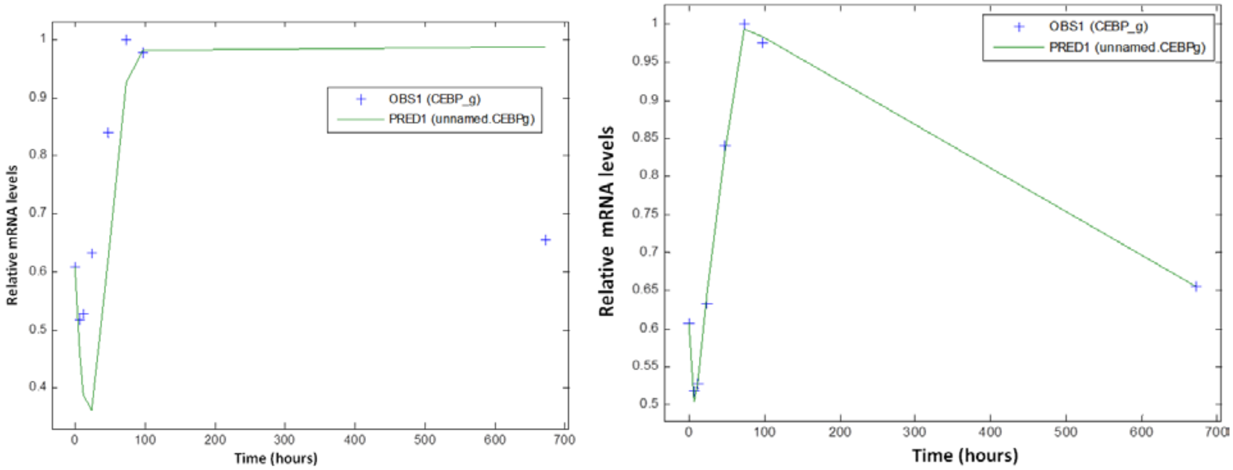


Figure 19. Comparison of fit of gene CEBPg, with (right) and without (left) the additional regulation by CEBPb.

The windows (windows # 3 and 4) were again moved forward as described in step 3 and the new windows were reconstructed as in Step 4. This loop of steps 3 and 4 was repeated till all the 7 in the network were covered at least once. Table 5 shows all the windows of genes generated by MANI till it covered all the genes in the network at least once.

Table 5. The complete set of windows generate by MANI till all the 7 genes in the network were covered at least once.

Window #	G_a	G_b	G_c
1	KLF4	XDH	CEBPb
2	CEBPa	CEBPa	PPARg
3	KLF4	CEBPb	CEBPg
4	GLUT4	PPARg	CEBPa
5	CEBPa	CEBPg	GLUT4

The regulatory relationships within each window were inferred in the order they were created and, in the end, all the gene relationships were accumulated to form the full network as shown in fig.20.

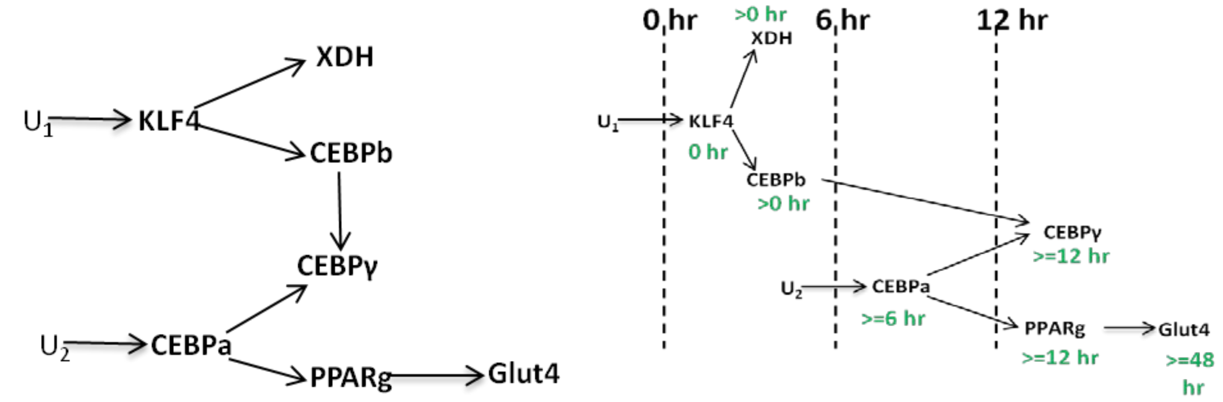


Figure 20. The complete Adipogenesis network as constructed by MANI approach. **Left.**The summary of all the gene relationships constructed by the MANI approach after sliding the window to cover all the 7 genes. **Right.** The same network but with the added information of duration of lag observed in the expression profile of each gene. Arranging the network in order of lag helps to show the network as a dynamical and hierarchical regulatory network and how it switches itself on at various time intervals. For example, gene CEBPa is only triggered more than 6 hours into the cascade, which it follows up by driving expression of genes, PPARg and GLUT4, downstream. Once the cascade gets activated, adipogenesis moves forward in this sequential-parallel fashion. The two inputs of the network are U_1 (time invariant constant input) and U_2 (sigmoid input).

To judge the accuracy of the network constructed by MANI (Fig. 20 left), I used the literature to validate some of the constructed relationships and found evidence for two of them. KLF4 has been shown to regulate the expression of gene CEBPb (Birsoy et al., 2008) and PPARg regulates the expression of gene GLTU4 (Im et al., 2007). It is impossible to classify any of the other edges in the constructed network that currently do not have any literature validation as false positives because of the ongoing nature of biological research that can potentially validate the other gene relationships in the future. Therefore, a more objective validation of MANI is desired. One approach would be to construct a network based on time series data generated by perturbing an artificially developed network. In that way, edges constructed in the network can be objectively judged. One such artificial network was developed and time series data generated by perturbing the network was made available by the DREAM challenge.

3.3. DREAM consortium

Dialogue on Reverse Engineering Assessment and Methods (DREAM) is a yearly community wide challenge for objective assessment of network inference methods developed in the field of biological networks. Inspiration for DREAM was drawn from the field of protein structure prediction and transferred to the field of genomics to predict gene network structures.

The approach taken by DREAM is to develop insilico benchmarks as predominant approach for performance assessment of different network inference methods. By developing and simulating artificial networks, the ground truth is known, predictions can be systematically evaluated against the true network and the performances of various network inference methods can be objectively compared. Among the different editions of DREAM, DREAM3 (year 2008) is the best benchmark for network inference algorithms in biological networks. DREAM 3 challenge consisted of few different types of data simulated from artificial networks of varying sizes (n=10, 50 and 100) that are listed below that the participant could use to construct the source network.

(i) **Knock-down static data.** In a gene relationship, gene A -> gene B, if activity of gene A is altered (reduced), then activity of gene B is similarly affected as illustrated by property of correlation/co-expression property in the introduction chapter. DREAM3 made available knock-down data where every gene in the network is separately knocked down to half its activity and its impact on the activities of other genes in the network as a result of the knock-out was recorded and made available.

(ii) **Knock-out static data.** Besides knock-down data, DREAM3 also made available knock-out data where instead of reducing the activity of the gene by half as in knock-down, the activity of the gene was completely eliminated. As in knock-down, activity of each of the gene in the network is separately eliminated and its impact on the activities of other genes in the network was recorded and made available.

(iii) **Time Series data.** Gene network were perturbed randomly using different initial conditions and were simulated from t=0 to t=200 and the activities of genes in the network were recorded at time intervals of 10 for a total of 21 time points. For a gene network of sizes 10, 50 and 100, DREAM 3 provided 4, 23 and 46 different time series; each time series produced from a unique perturbation of the networks. Since this is a time series data, the property of temporal lag in their expression profile of genes could be used to decipher hierarchy in the network.

3.4. Why I chose to validate my MANI algorithm using only 1 time series data to reconstruct network of size 10?

Among the three sized networks that were available to be constructed, I chose to reconstruct network of size n=10 (network consisting of genes, G1 to G10) since it was comparable in size to the 7 gene adipogenesis network. Among the three different types of data available, the two kinds of static data, knock-out and knock-down data, while are effective for teasing out gene regulatory relationships, are expensive to obtain as real experimental data because it is expensive to reduce and knock out the activity of gene completely and the knocking out has to be done for every gene in the network. Since I developed my MANI approach on a time series data, I picked to use the DREAM3 time series data to construct my network. Time series gene expression data, obtained by perturbing the genes in the network and collecting their activity after perturbation, is easier to obtain experimentally. This makes the MANI approach a more practical network inference approach and also widely usable.

Of the four perturbation Time Series (TS) data available from four random perturbations of the network, I used only 1 TS because I developed my MANI approach using only 1 set of TS data. I also felt it would be better to construct the network using the least amount of data possible since collecting gene activity data from multiple perturbations could also become expensive. The TS DREAM 3 data that was used for constructing the 10 gene network is as shown below.

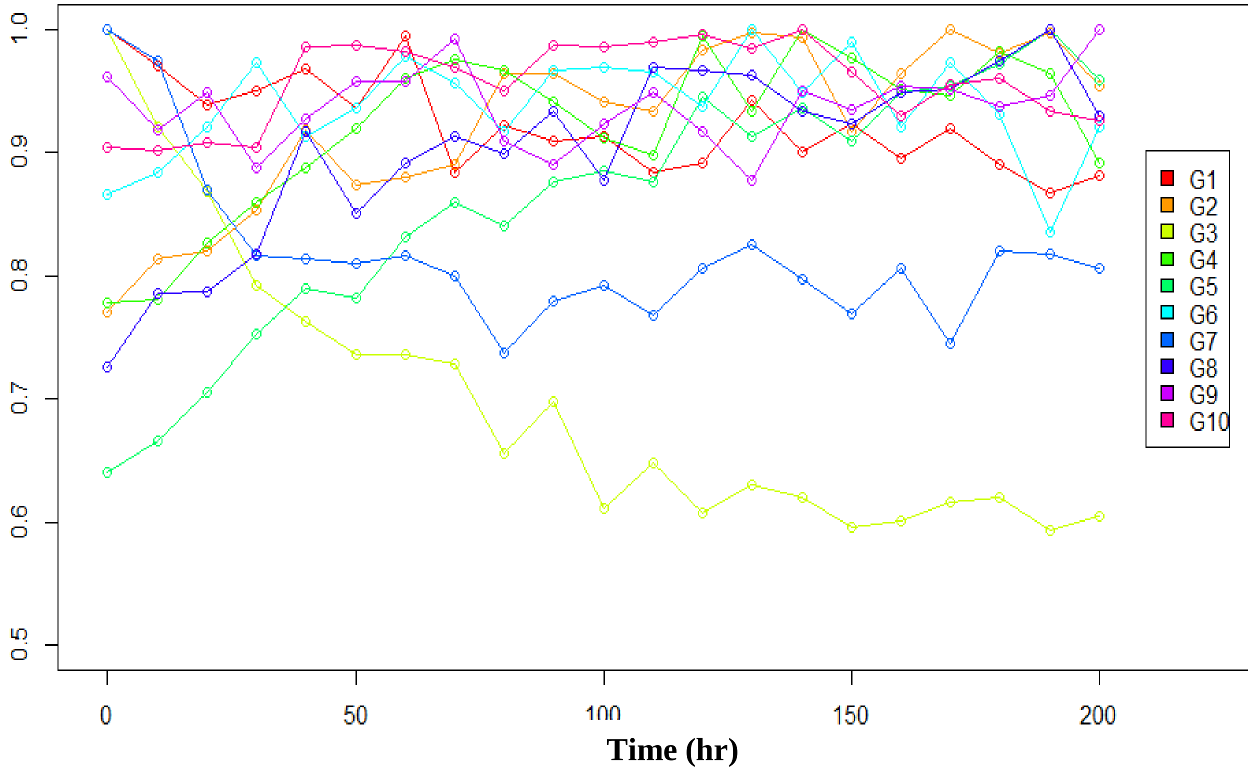


Figure 21. DREAM3 Time Series data. Time Series Gene expression data obtained by perturbing a 10- gene network and collecting the activity of genes for 21 time points in 10 hour intervals. The data presented was normalized such that maximum gene activity possible was 1.

While constructing the network from the above TS data using MANI approach, I made sure the source network (that is, the correct answer shown in fig 22 below) that generated the time series data by perturbation was kept confidential so that, at the end of the inference, I can truly and objectively evaluate the accuracy of the network reconstructed by MANI approach by comparing it to the correct answer without any bias.

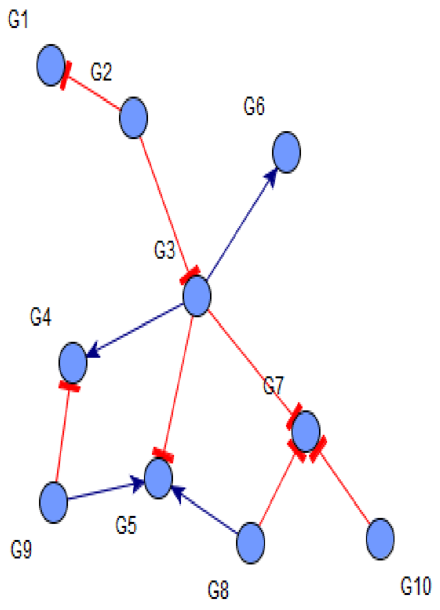


Figure 22. DREAM3 source network (size n=10) that produced the TS data in fig. 21

3.5. How will I evaluate the accuracy of my network inference?

The performance of the MANI approach depends on the accuracy of the network constructed.

The accuracy of the network constructed was judged using the following measures. Note that when edges of the networks were compared between MANI constructed network and the source network (fig. 22), the edges between the networks were matched just for their presence or absence between genes without comparing for their directions or regulatory nature (positive regulation or inhibitory regulation).

A) True Positives (TP), correct identification of the presence of an edge,

B) False Positives (FP), incorrect identification of the presence of an edge,

C) True Negatives (TN), correct identification of the absence of an edge,

D) and False Negatives (FN), incorrect identification of the absence of an edge.

The performance will be judged on two key parameters: Sensitivity= TP/(TP+TN) and Precision= FP/(FP+FN)

3.6. How did I implement my MANI approach to construct the network from the DREAM3 TS data?

Step 1. Selecting initial windows.

As stated before, the initial window(s) was created by selecting pair(s) of genes with maximum correlation between their time series expression profiles and then selecting a third gene to complete the window by choosing a gene (outside the pair of genes) with maximum correlation to either of the genes forming the pair. An alternate application of the rule could be to select a trio of genes that had the highest average degree of correlation between them. The correlation matrix between the 10 genes shown below in table X can help to select such strongly correlated windows among the genes.

Table 6. Correlation matrix of the 10 genes belonging to the DREAM3 network

	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10
G1	1	0.55974	0.709091	0.501299	0.75974	0.1	0.432468	0.716883	0.190909	0.272727
G2	0.55974	1	0.71039	0.614286	0.849351	0.17013	0.398701	0.846753	0.244156	0.406494
G3	0.709091	0.71039	1	0.616883	0.911688	0.146753	0.446753	0.735065	0.015584	0.236364
G4	0.501299	0.614286	0.616883	1	0.62987	0.246753	0.44026	0.601299	0.014286	0.488312
G5	0.75974	0.849351	0.911688	0.62987	1	0.096104	0.318182	0.862338	0.046753	0.232468
G6	0.1	0.17013	0.146753	0.246753	0.096104	1	0.357143	0.120779	0.242857	0.415584
G7	0.432468	0.398701	0.446753	0.44026	0.318182	0.357143	1	0.3	0.057143	0.485714
G8	0.716883	0.846753	0.735065	0.601299	0.862338	0.120779	0.3	1	0.114286	0.418182
G9	0.190909	0.244156	0.015584	0.014286	0.046753	0.242857	0.057143	0.114286	1	0.118182
G10	0.272727	0.406494	0.236364	0.488312	0.232468	0.415584	0.485714	0.418182	0.118182	1

Therefore, applying the rule, window # 1 was created using the genes G2, G5 and G8. The expression profiles of genes in the window are as shown below in fig. 23.

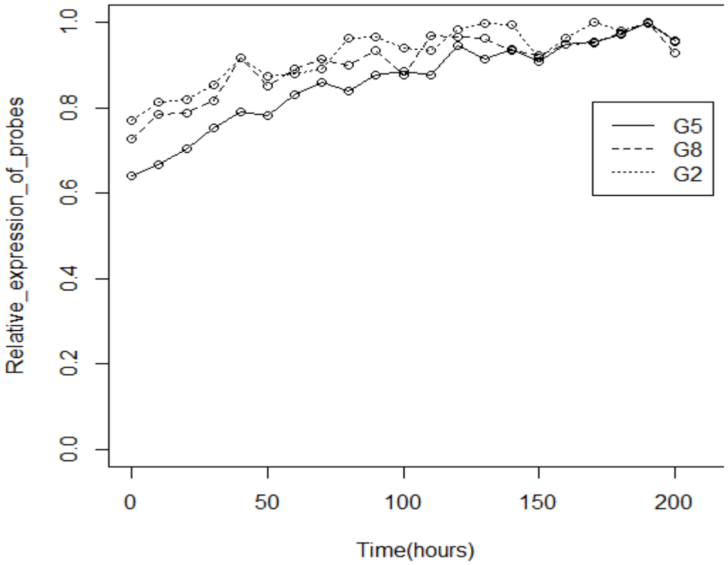
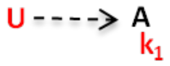


Figure 23. The normalized expression profiles of the 3 genes in window #1 created for DREAM3 network inference

Step 2. Fitting the best possible mechanism for genes within window # 1.

There appears no differentiating lag between the three genes, G2, G5 and G8. Therefore, I checked each gene at top with an input as was done before in window # 1 for adipogenesis network inference. The mathematical description to test a gene with an input is as shown below.



$$\frac{dA}{dt} = U - k_1.A \qquad ...(13)$$

Where
A=G2, G5 or G8
k_1= rate of degradation of mRNA molecules (molecule that represents the activity/expression levels) of gene A
U= constant stimulus or input that regulates the activity of gene A.

Therefore, the three genes were tested for top of the hierarchy test with an input as done before for window # 1 for adipogenesis network. The result of such testing is shown below in Table 7.

Table 7. Testing genes at the top of the hierarchy for window # 1 created for DREAM3 network inference

Gene at the top	U (hr^{-1})	k_1(hr^{-1})	SSE
G2	0.12 ± 0.03	0.12 ± 0.03	0.015
G5	0.08 ± 0.01	0.07 ± 0.01	0.008
G8	0.142 ± 0.03	0.145 ± 0.03	0.016

Since it was not clear which gene is at the top in the motif, among the list of mechanisms (fig.24 below) that were attempted to model the data for genes in window # 1, I tried all three genes at the top (mechanisms # 1, 2 and 3). Among the different motifs that could be tested for the three genes within the present window, I tested parallel (mechanisms # 1, 2 and 3) and multiple regulatory mechanisms (mechanism # 4). However, since there was not enough lag between genes, I did not test a serial regulatory mechanism.

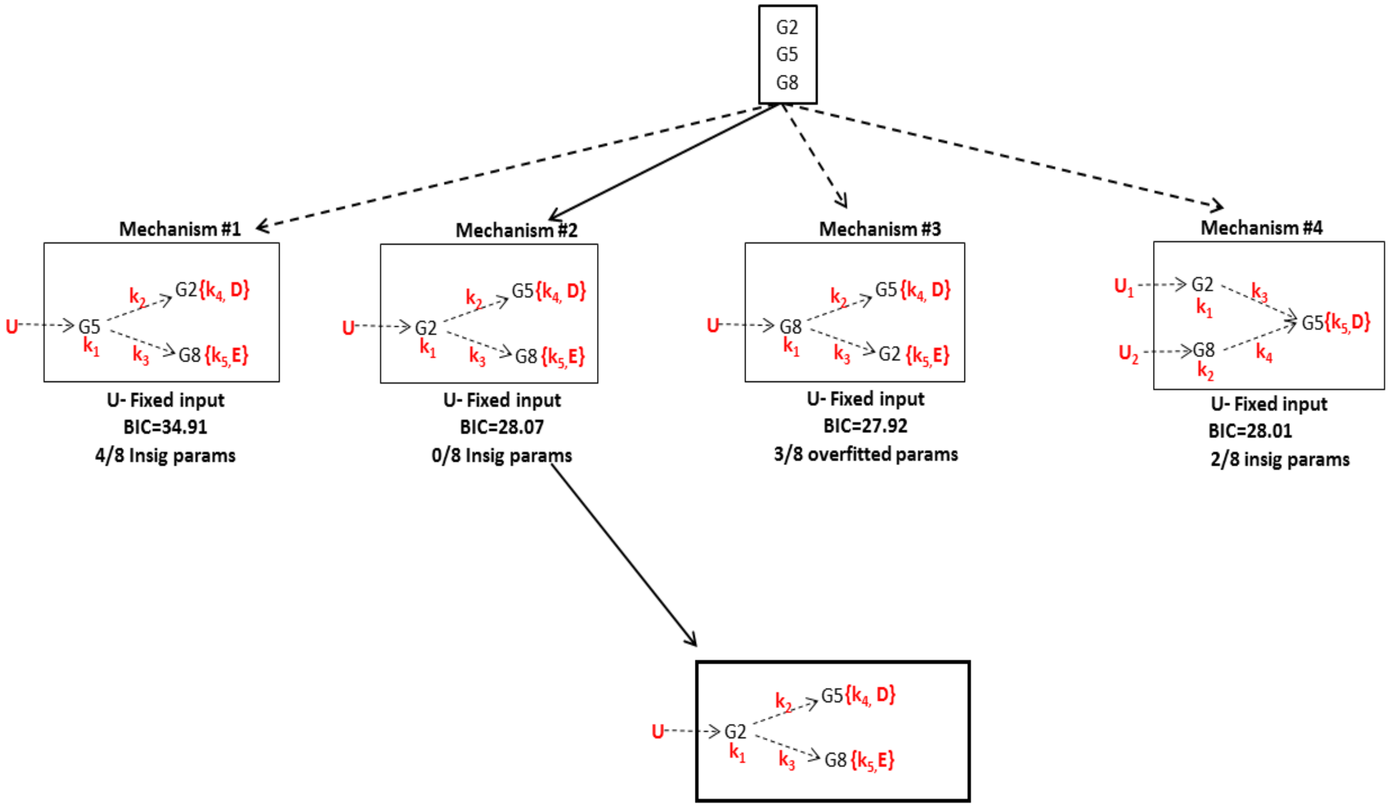
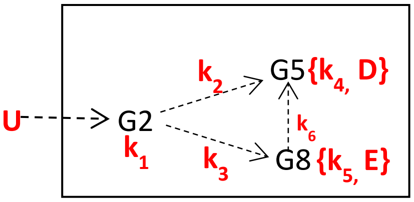


Figure 24. Window #1 network inference. Mechanism 2 (parallel regulation pattern) has been chosen as the best mechanism (bottom) to describe the genes in window #1. For the best mechanism, SSE_{G2}=0.0146, SSE_{G5}=0.006 and SSE_{G8}=0.0165. BIC= (0.0146+0.006+0.0165)/0.01+8*log(21)=28.07. The final set of parameters estimated for the best mechanism is available in table below. The number of parameters estimated (parameters in red) for each mechanism is indicated and the fraction of those estimated parameters that were either over fitted or found to be insignificant in the optimal fit were also noted.

Parameters	**mean +/- standard error (hr^{-1})**
U	0.12 +/-0.03
k_1	0.12 +/-0.03
k_2	2.44 +/- 0.71
k_3	0.92 +/- 0.67
k_4	1.43 +/- 0.02
k_5	0.93 +/- 0.68
D	1.00+/-0.32
E	0

In the above window, though mechanism # 3 had the smaller BIC score, mechanism # 2 was chosen as the optimal mechanism because of the fewer number of insignificant parameter estimated under that mechanism. One other mechanism (based on serial-parallel regulatory motif) that was considered towards solving the network for window # 1 is shown below but the estimation resulted in many insignificant parameters.



Step 3. Advancing the window(s) forward.

One gene In One gene Out (OIOO) rule was used to advance the window forward. The most correlated gene outside the window to any gene inside the window was introduced and the gene inside the window least correlated to the incoming gene was discarded. Window # 2 was created that way by introducing G3 to window # 1 and removing G2 to form a window consisting of genes G3, G5 and G8 (the expression profile of genes within the window in fig. 25 below).

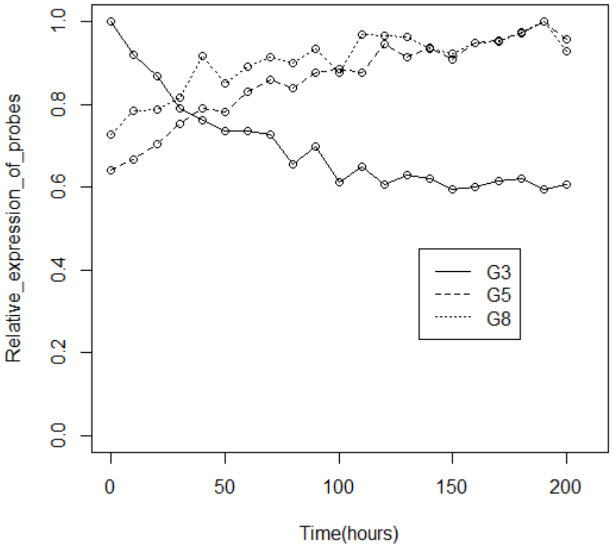


Figure 25. The normalized expression profiles of the 3 genes in window #2 created for DREAM3 network inference

Step 4. Inferring the network within the new window(s) created.

Window # 2 {G3, G5, G8} inference and parameters

I decided to check if the property of lag could be exploited to identify hierarchy of G3 with respect to G5 and G8. Since, there was equal lag for all three genes (0 hours) and the goal of this inference is to locate the position of the new gene, G3, with respect to older genes, G5 and G8. I tested the performance of G3 as a gene at the top of hierarchy with input as done in window # 1 and compared its performance (measured by SSE) with respect to genes G5 and G8.

Gene at the top	U (hr^{-1})	k_1(hr^{-1})	SSE
G3	0.08 ± 0.01	0.15 ± 0.02	0.012
G5	0.08 ± 0.01	0.07 ± 0.01	0.008
G8	0.142 ± 0.03	0.145 ± 0.03	0.016

Based on SSE values, it was not conclusive if the relationship of G3 with respect to genes, G5 and G8 was either a regulator genes or is being regulated by the two genes. Therefore a simple test was conducted to check what the performance would be if G3 was the regulator of genes G5 and G8 with an external input, U.

Mechanism # 2.1: U $\underset{k_2}{->}$ G3 -> G8 {k_4, D}

k_2=0.06 ± 0.01 hr^{-1}, k_4=0.03 ± 0.01 hr^{-1}, D=0, SSE_{G8}=0.0158

Mechanism # 2.2: U $\underset{k_2}{->}$ G3 -> G5 {k_4, D}

k_2=0.046 ± 0.01 hr^{-1}, k_4=0.017 hr^{-1}, D=0, SSE_{G5}=0.0088

From the above testing of a few basic hypotheses, the goodness of fit of G8 improved slightly from window # 1 ($SSEG_8$ =0.0165) to 0.0158 when regulated by G3. Therefore, mechanism 2.1 could be part of the mechanisms considered for window # 2. On the other hand, gene G5 was already modeled satisfactorily (SSE_{G5} = 0.006) as being a gene regulated by G2 in window # 1 and did not improve when regulated by G3 (SSE_{G5} = 0.0088). Motifs such as G3 being regulated by G5 or G8 or by both were also considered as part of potential mechanisms for window #2. Based on the various hypotheses suggested above based on preliminary motif tests, the mechanisms tested for window #2 were aimed at getting the best fits for gene G8 (either through the single input already available from window # 1 or through multiple regulatory input that includes additional regulation from gene G3), G3 (regulation by G8 or by both G8 and G5) or a null hypothesis (that is G3 is regulated by a separate input independent of the input regulating G5 and G8). The various mechanisms tested for window # 2 are shown in fig. 26.

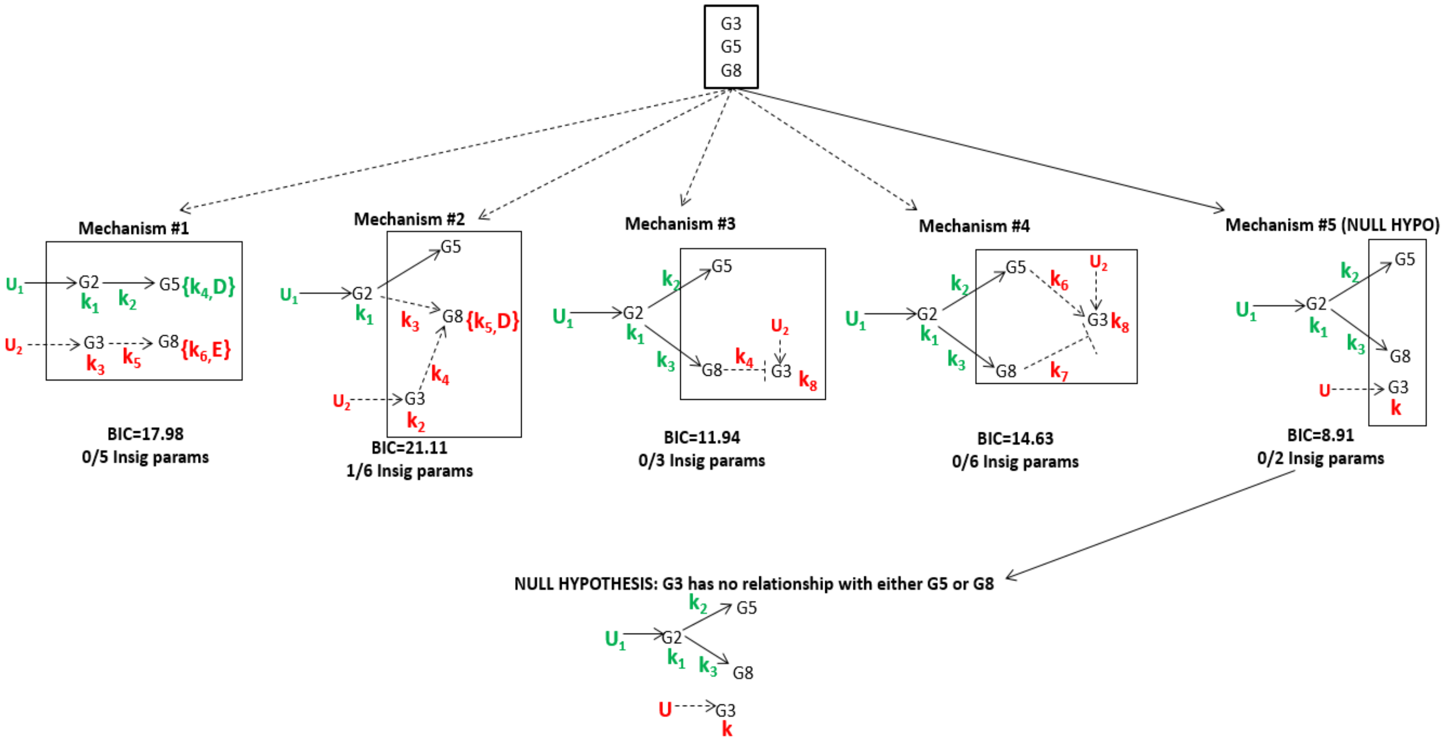


Figure 26. Window #2 network inference. The parameters estimated for the different mechanisms tested for window # 2 are shown in red and the parameters carried over from previous windows are shown in green. Among the different edges forming the mechanism, solid edges refer to edges between genes created from previous windows and dotted edges refer to edges proposed in the present mechanism. Edges that are inhibitory or have negative regulatory effect on genes are shown in -|. Among the different mechanisms fitted, null hypothesis with G3 being regulated by a separate input 'U' seemed to be the most optimal network on account if its lowest BIC score due to parsimonity of the network. Fitted parameters are shown in the below table.

Parameters	mean +/- standard error (hr^{-1})
U	0.08 ± 0.01
K	0.15 ± 0.02

Window # 2 is subsequently advanced forward (step 3) and constructed (step 4). These two steps were repeated in a loop until all the 10 genes in the network were covered at least once. One of the problems encountered in step 3 in advancing the window forward in DREAM3 network was the formation of the same window without covering all the genes in the network at least once (did not experience this problem while constructing adipogenesis network). In such cases, the OIOO rule in step3 was relaxed to allow for newer combinations of genes in the moving window. If the OIOO rule formed the same window without fully covering the network at least once, the rule was relaxed to allow the selection of the next highly correlated gene to the genes within the window so as to admit new genes not previously captured by the moving window. If it still formed the same window, the rule was further relaxed to allow the next most correlated gene. The relaxation was carried out until the window could move forward with newer combination of genes within its selection. The window was halted only when it had covered all the genes in the network at least once. The list of gene sets selected by the window during its journey through the network at least once is as shown below in Table 8 (37 windows ordered on the basis of their average degree of correlation among its genes).

Table 8. The sets of genes selected by MANI as the window moved through the DREAM 3 network

Window #	a in Ga	b in G_b	c in G_c	Average correlation in the window
1	**5**	**8**	**2**	**0.85**
2	**3**	**5**	**8**	**0.84**
3	3	5	2	0.82
4	**3**	**5**	**1**	**0.79**
5	5	8	1	0.78
6	8	2	3	0.76
7	2	5	1	0.72
8	8	3	1	0.72
9	**3**	**4**	**5**	**0.72**
10	2	1	8	0.71
11	5	8	4	0.7
12	2	4	5	0.7
13	8	4	2	0.69
14	3	1	2	0.66
15	8	3	4	0.65
16	3	2	4	0.65
17	**7**	**3**	**5**	**0.56**
18	**2**	**8**	**10**	**0.56**
19	3	1	7	0.53
20	2	7	5	0.52
21	2	7	8	0.52
22	2	4	10	0.5
23	8	4	10	0.5
24	3	4	7	0.5
25	7	8	3	0.49
26	7	8	5	0.49
27	2	1	7	0.46
28	3	10	8	0.46
29	3	10	5	0.46
30	2	10	7	0.43

31	7	8	10	0.4
32	7	3	10	0.39
33	**7**	**3**	**6**	**0.32**
34	2	7	6	0.31
35	3	10	6	0.27
36	7	8	6	0.26
37	**10**	**5**	**9**	**0.13**

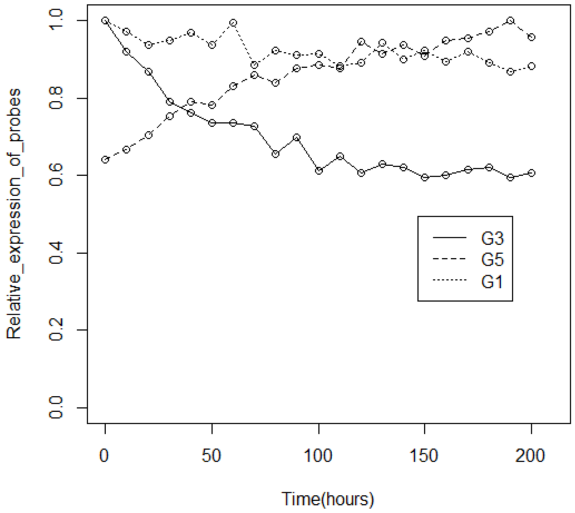
Since there were 37 windows selected, if local network were to be constructed within each of the above windows, it would lead to the formation of a hair ball network. Since parsimonity in the network was preferred, I, therefore, decided that among the windows selected, I would restrict myself to constructing networks only among those windows where a gene gets selected for the first time as the window moved through the network. In other words, among the various windows a gene would appear, I select only those windows which had the highest average degree of correlation among its genes (average degree of correlation between genes within each window is also shown in table 8). Therefore, among the 37 windows selected by MANI in its journey through the network, network inference was simplified to constructing the optimal mechanism within only the 8 windows in the list (window #s 1, 2, 4, 9, 17, 18, 33 and 37 highlighted in Table 9).

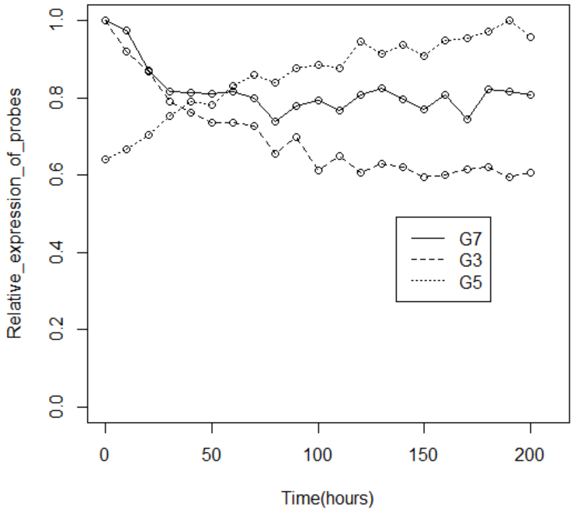
Table 9. Subset of windows selected for inference

Window #	a in Ga	b in G_b	c in G_c
1	5	8	2
2	3	5	8
4	3	5	1
9	3	4	5
17	7	3	5
18	2	8	10
33	7	3	6
37	10	5	9

3.7. Inference of network within windows #4 {G1, G3 and G5} and #17 {G3, G5 and G7} and parameter estimation.

The approach that was used to infer the networks within windows # 4 and # 17 was similar to window # 2. Both involved locating the new gene (gene G1 in window # 4 and gene G7 in window # 17) with respect to genes, G3 and G5, whose position in the network has already been established through windows # 1 and 2. Their expression profiles are as shown in figures below.





The new genes (genes G1 in window # 4 and G7 in window # 17) in the two windows shared the same lag (0 hours) as genes G3 and G5. Since the position of the new genes with respect to G3 and G5 in the hierarchy of gene regulation was not clear from lag, different positions in the hierarchy for the new genes were tested using the approach adopted before in window # 2 and the four advanced motifs (serial, parallel, multiple regulatory and serial-parallel motifs) listed under step 2 of MANI approach were also tested in both the windows. Since the steps involved in testing different motifs for the genes within the two windows were similar to those conducted before for window # 2, I showed here only the final outcome of their inference (Fig. 27).

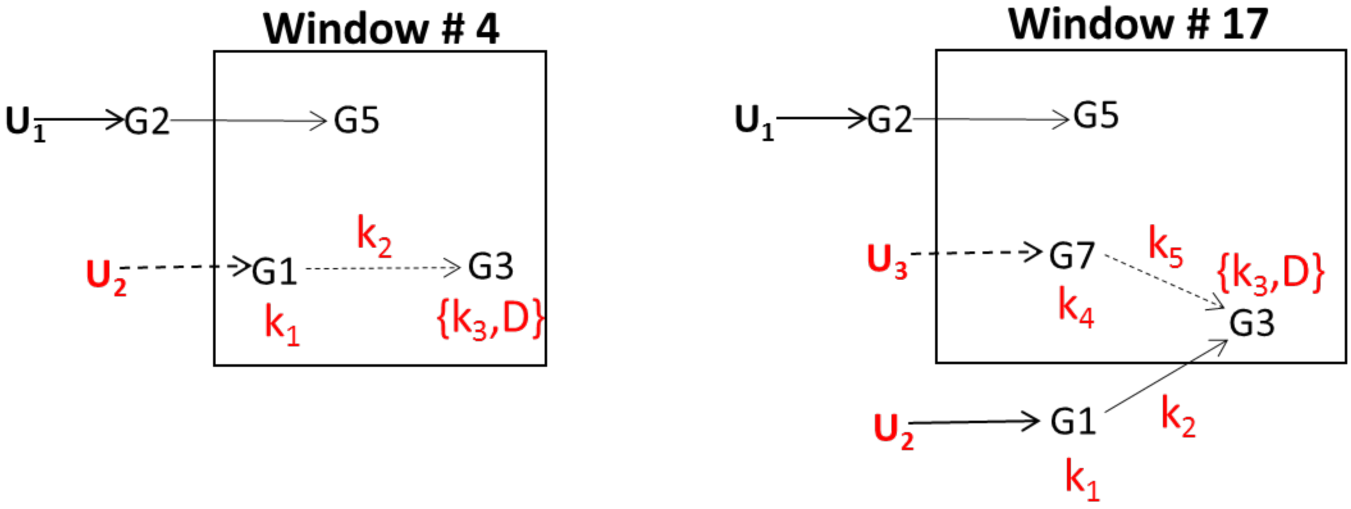


Figure 27. Windows #4 and #17 network inference. The dotted edges showed the new edges that were constructed within the windows and the parameters estimated in each of the windows were shown in red. Parameters from window # 4 were involved in window # 17 since it was inferred from the network inference that both the new genes, G1 and G7, regulate G3. The final values of parameters estimated from the network inference from the two windows are shown below in table.

Parameters	**mean +/- standard error (hr^{-1})**
U_2	0.24 ± 0.06
U_3	0.10 ± 0.04
k_1	0.11 ± 0.05
k_2	6.85 ± 2.26
k_3	2.81 ± 0.11
k_4	0.3 ± 0.07
k_5	3.78 ± 1.24
D	7.64 ± 1.1

3.8. Window #9 {G3, G4 and G5} inference and parameters.

The goal of reconstructing window # 9 is to locate the position of G4 with respect to older genes G3 and G5. G4 had longer lag (10 hours) compared to G3 and G5 (Fig. X) and therefore appear lower in the hierarchy compared to G3 and G5.

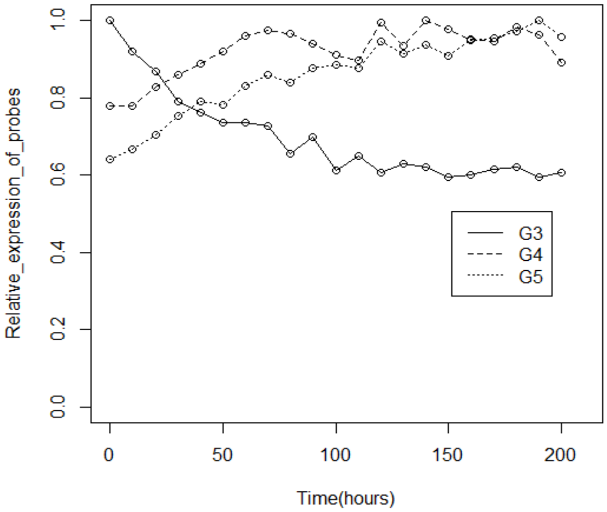


Figure 28. The normalized expression profiles of the 3 genes in window #9 created for DREAM3 network inference

Since G4 had longer lag compared to other genes, we focused only on mechanisms where G4 is regulated by other genes: by G3 (mechanism # 1) or G5 (mechanism # 2) or both (multiple regulatory mechanism, mechanism #3) or G4 being regulated by a separate independent input (null hypothesis, mechanism #4). The different mechanisms tested are shown in the below figure (Fig. 29).

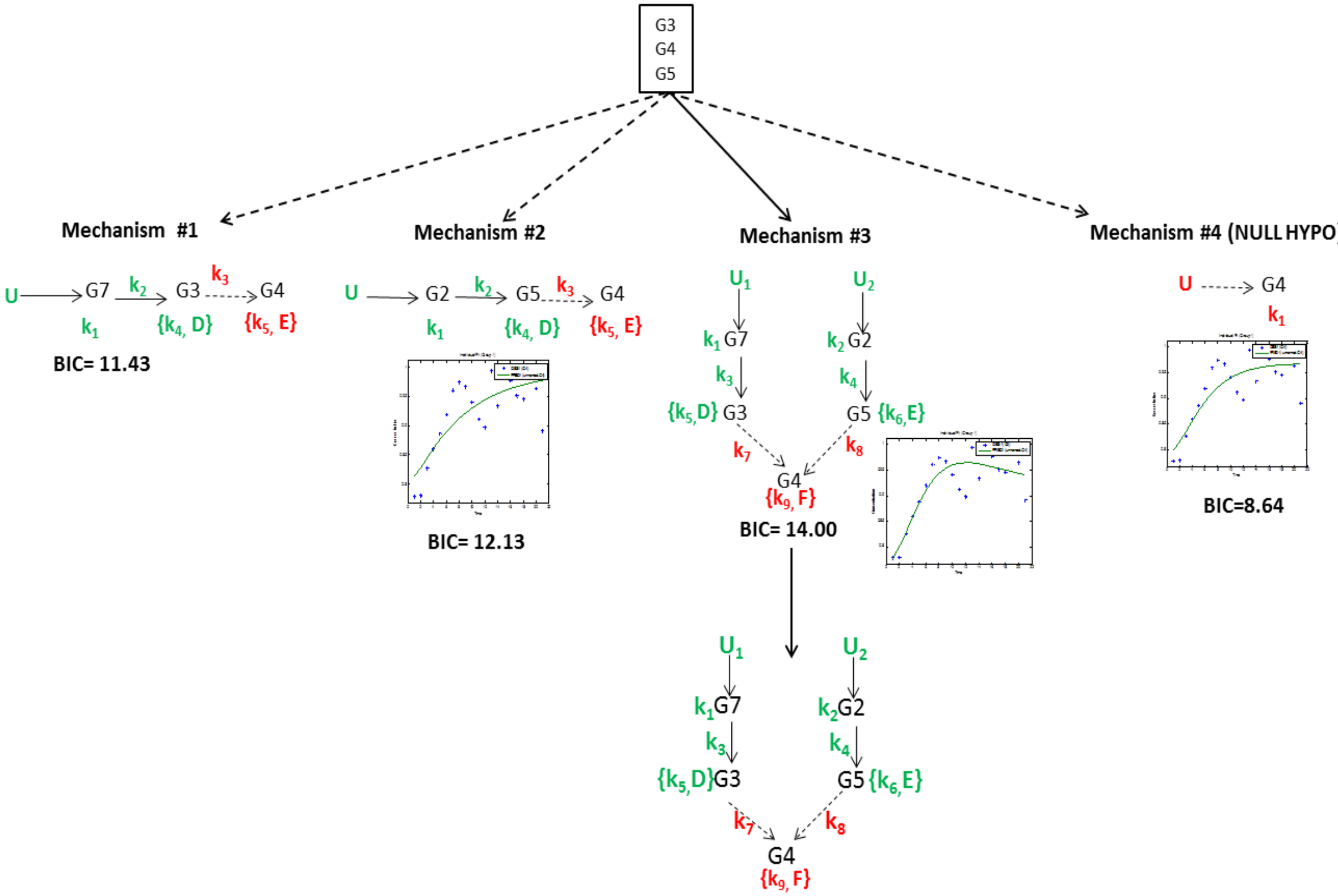


Figure 29. Windows #9 network inference. The parameters estimated for the different mechanisms tested for window # 9 are shown in red and the parameters carried over from previous windows are shown in green. Multiple regulatory mechanism (mechanism #3), G4 being regulated by both G3 and G5, has been chosen as the optimal mechanism of regulation for gene G4. The values of parameters estimated (parameters in red) for mechanism # 3 are shown in the table below. For mechanisms #2, #3 and #4, there were small insets included showing the degree of fit achieved by the model for the expression profile of G4 in each mechanism. The degree of fit achieved by mechanism #1 (inset not shown) was similar to the fit achieved by mechanism #2.

Parameters	**mean +/- standard error (hr^{-1})**
k_7	-3.0 ± 0.02
k_8	-2.15 ± 0.51
k_9	2.27 ± 0.81
F	-6.1 ± 0.37

Explanation for mechanism selection for window #9.

In fig. 29 above, though the null hypothesis (mechanism # 4 or in other words separate input for the new gene, G4, independent of other genes in the window) should have been the natural selection for the most optimal mechanism for G4 based on the smallest BIC score, I chose mechanism # 3 (multiple regulatory mechanism) to describe the regulation of G4 because the optimal mechanism of regulation was chosen based on the quality of fit rather than only based on BIC score. As can be seen in the insets of the above fig. X, mechanism # 3 could capture more features in the expression data of G4, the rise and decline in the expression levels of the G4 and the lag phase of the gene's expression. The high BIC score is because of the presence of higher number of model parameters that had inflated the score. In this case, quality of fit played a stronger role than BIC score in determining the optimal mechanism.

Therefore, it is advisable to select the optimal mechanism of gene regulation by not just using the criterion of smallest BIC score but also based on quality of fit. In appropriate cases, applying quality of fit to choose optimal mechanism can result in capturing more features in the data and can add to the interpretability of the model. From the optimal mechanism chosen to describe the regulation of G4 based on the quality of fit, it can be inferred that the increase and the subsequent decrease in the expression level of G4 was due to the regulatory activities of G3 and G5, genes which have opposing trends in their own activities over time.

3.9. Window #18 {G2, G8 and G10} and Window # 37 {G5, G9 and G10} inference and parameters

Inferring the network between genes within window # 18 had many similar characteristics to how the network was inferred within windows # 4 and # 9 in that the location of a new gene (G10's location) had to be determined with respect to older genes from previous windows (G2 and G8). G10 had longer lag (t=30 hours) compared to genes G2 and G8 (0 hours, see fig. 30 below). The property of lag could be now be used to infer that G10 will appear lower in the hierarchy of regulation compared to G2 and G8. Since the hierarchy of regulation among the genes was known from lag, different mechanisms were tested to determine if G10 was regulated by G8 or G2 or by both. The final mechanism that was deduced to be optimal is shown below and the value of the kinetic parameters can be found in a later section when the full network inference is summarized.

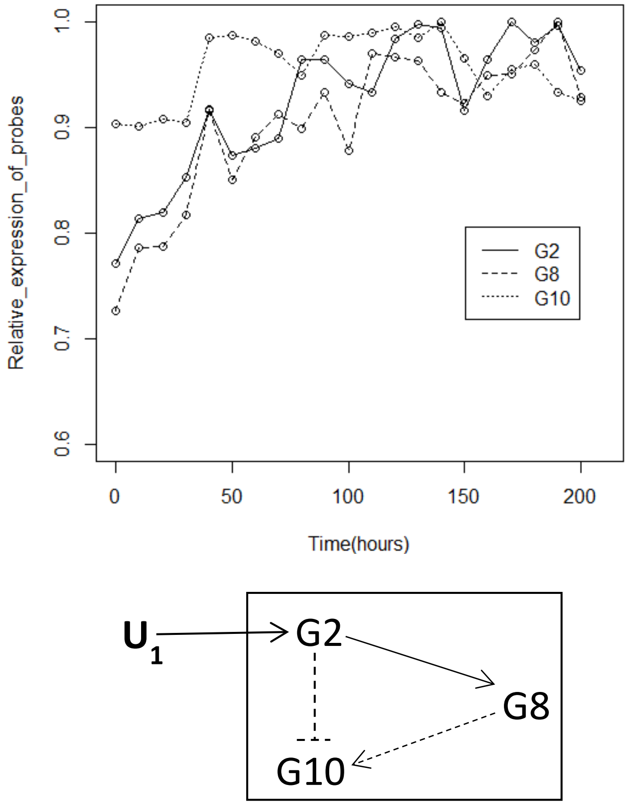
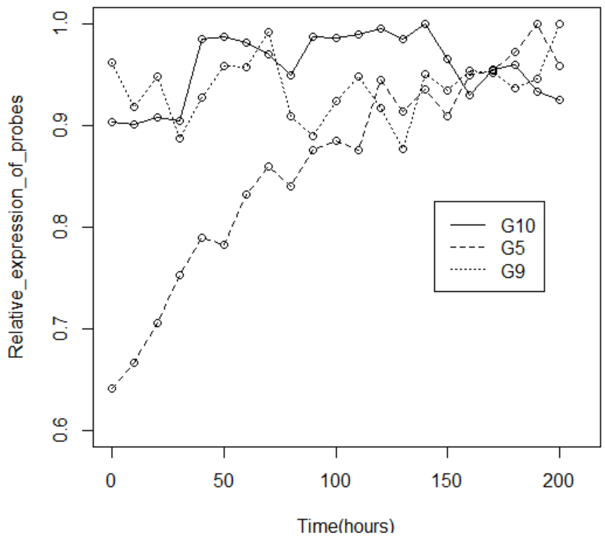


Figure 30. Windows #18 network inference. Edge between G2 and G8 has been known before. Dotted edges has been inferred from the above chosen optimal mechanism. G2 and G8 both regulate G10 in a serial-parallel regulatory framework. G2 has inhibitory role while G8 has a positive role regulating G10.

In window # 37, G9 and G10 have the same lag (30 hours) in comparison to lag of gene G5 (0 hours, see fig.31 below), which implies G9 and G10 are lower in hierarchy compared to G5. Since the location of G9 has to be determined with respect to G5 and G10, mechanisms were proposed to test (i) if G9 was being regulated by G5 and (ii) if G5 could be an additional regulator of G10 besides G2 and G8. Of these two major mechanisms tested above, G5 as a regulator of G9 was the most optimal and hence the optimal network constructed for this window is shown below and the value of the kinetic parameters of the regulatory model can be found in a later section when the full network inference is summarized.

.



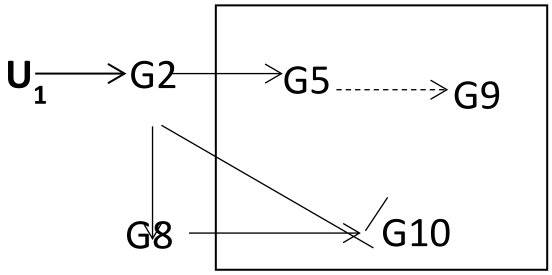
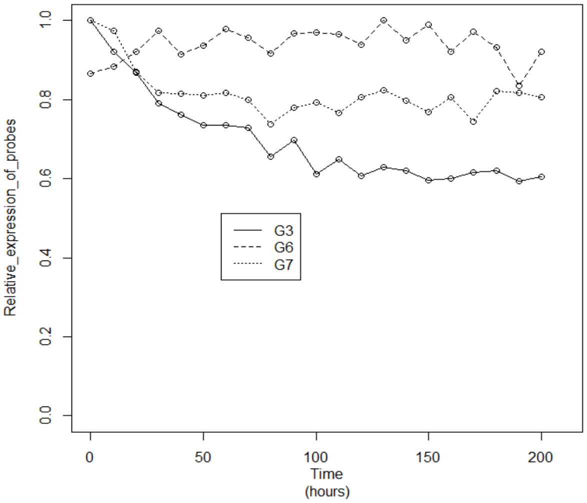


Figure 31. Windows #37 network inference. Duration of lag in expression profile of a gene has been defined as the last time point after which the expression profile sees a continuous increase or decrease in gene expression for 2 or more time points. Applying the definition, lag of gene G9 and G10 was estimated to be 30 hours.

3.10. Window #33 {G3, G6 and G7} inference and parameters

The expression profiles of genes within this window are as shown in fig. 32 below. Since G3 and G7 are genes from previous windows, the location of the new gene, G6, has to be determined with respect to G3 and G7. Since it was also previously inferred (from window # 17) that G7 regulates G3, mechanisms were tested to check the possibility of G6 being regulated by G3 or G7 or by both (fig. 32). The optimal mechanism was chosen purely based on BIC score though the quality of fit for the expression profile of G6 was compared across the different mechanisms. No particular mechanism offered any superior quality of fit. The mechanism of regulation of G6 by G3 has been chosen as the optimal mechanism and the value of the kinetic parameters can be found in a later section when the full network inference is summarized



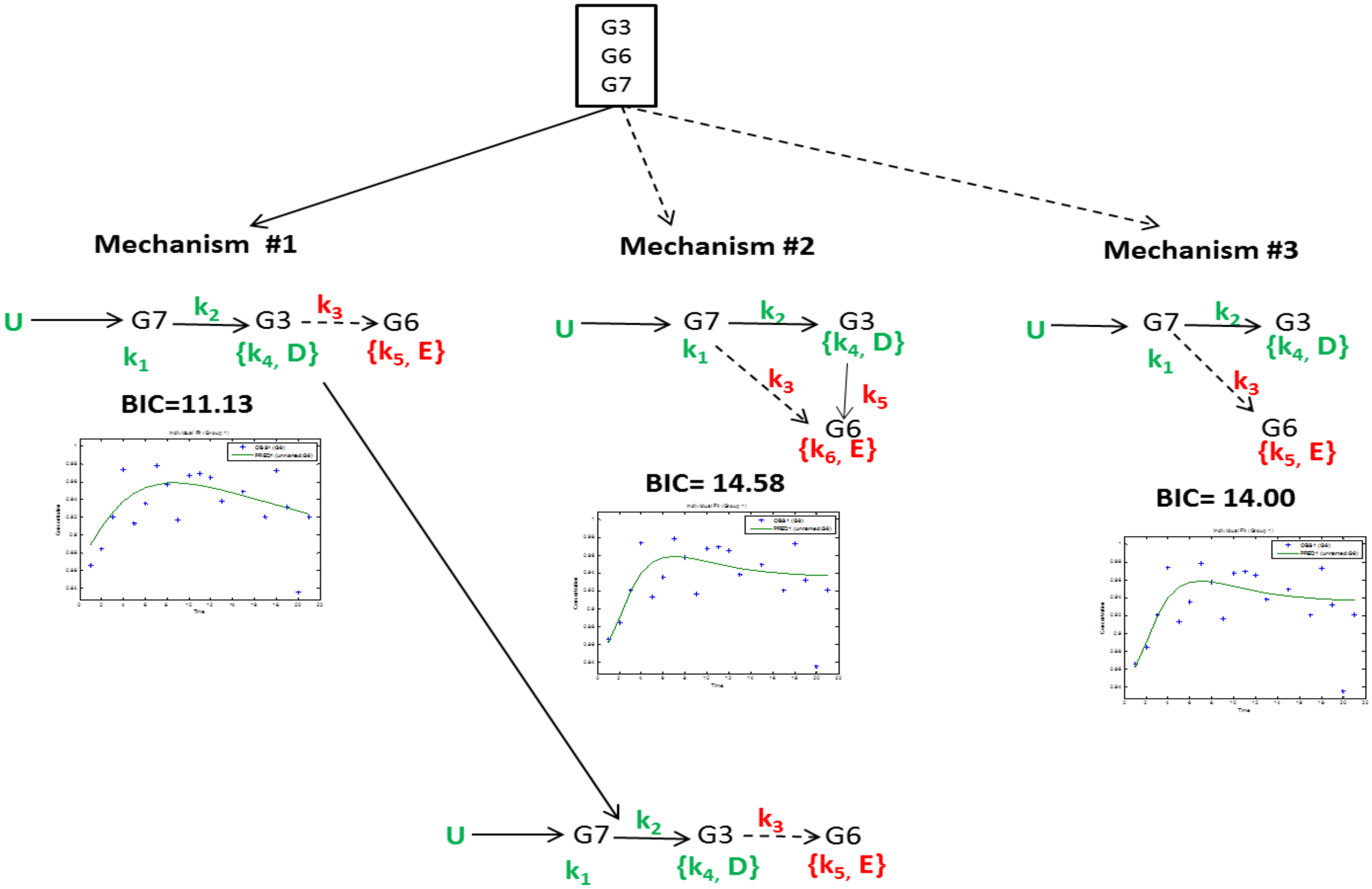


Figure 32. Windows #33 network inference. The solid edges and the parameters in green were from a previous window (window # 17). The dotted edges were edges proposed in the current window and parameters in red are parameters to be estimated. Quality of fit produced by the model for the expression data of G6 by each mechanism is indicated by the inset figures.

3.11. Complete inference

Having gathered the inference from various windows, the summary of inference of the network is as shown below and the values of the parameters estimated is shown in the table below.

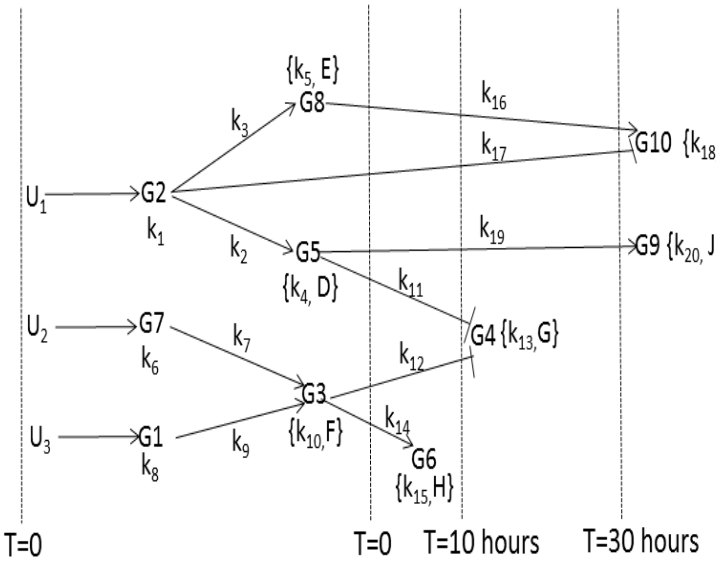


Figure 33. Complete summary of network inference. The inputs regulating the gene network are U_1, U_2 and U_3. The gene regulatory parameters representing the magnitude of regulation of a gene were located over the edge connecting the genes but if they represented degradation of mRNA produced from the gene, they were located under the name or on the side of the gene. If the degradation process had more than one parameter, they were included within {}. The network, besides representing edges and nodes in the network, is organized in time periods of when the gene gets activated (in other words, genes were arranged in order of their durations of lag estimated from their expression data).

Table 10. Values of kinetic parameters of the constructed DREAM3 network

Parameters	**Mean +/- Standard Error (hr^{-1})**	**Parameters**	**Mean +/- Standard Error (hr^{-1})**
U_1	0.12 ± 0.03	F	7.64 ± 1.1
k_1	0.12 ± 0.03	k_{11}	-2.15 ± 0.51
k_2	2.44 ± 0.71	k_{12}	-3.0 ± 0.02
k_3	0.92 ± 0.67	k_{13}	2.27 ± 0.81

k_4	1.43 ± 0.39	G	-6.1 ± 0.37
k_5	0.93 ± 0.68	k_{14}	0.07 ± 0.01
D	1.00 ± 0.32	k_{15}	0.05 ± 0.01
E	0	H	0
U_2	0.24 ± 0.06	k_{16}	3.66 ± 1.96
K_6	0.3 ± 0.07	k_{17}	-4.06 ± 2.08
U_3	0.10 ± 0.04	k_{18}	0.42 ± 0.33
k_8	0.11 ± 0.05	I	-0.83 ± 0.44
k_7	3.78 ± 1.24	k_{19}	0.04 ± 0.02
k_9	6.85 ± 2.26	k_{20}	0.19 ± 0.39
k_{10}	2.81 ± 0.11	J	-0.15 ± 0.36

3.12. How accurate is my DREAM 3 network inference?

The accuracy of the DREAM 3 network constructed by MANI approach was evaluated by comparing the network constructed in fig. 33 with respect to the correct answer shown in fig. 22. The comparisons were made between the networks by the parameters defined in section 3.5 by purely comparing the presence or absence of edges between genes rather than their direction. The number of edges that were True Positives (TPs, shown as green dotted edges in fig.34), False Negatives (FNs, shown as red dotted edges in fig.34) and False Positives (FPs, shown as black solid edges in fig.34) in my network constructions were 4, 7 and 6. Therefore, the Sensitivity (TP/ (TP+FN)) and Precision (TP/ (TP+FP)) of my network inference by MANI approach were 36.36% and 40% respectively.

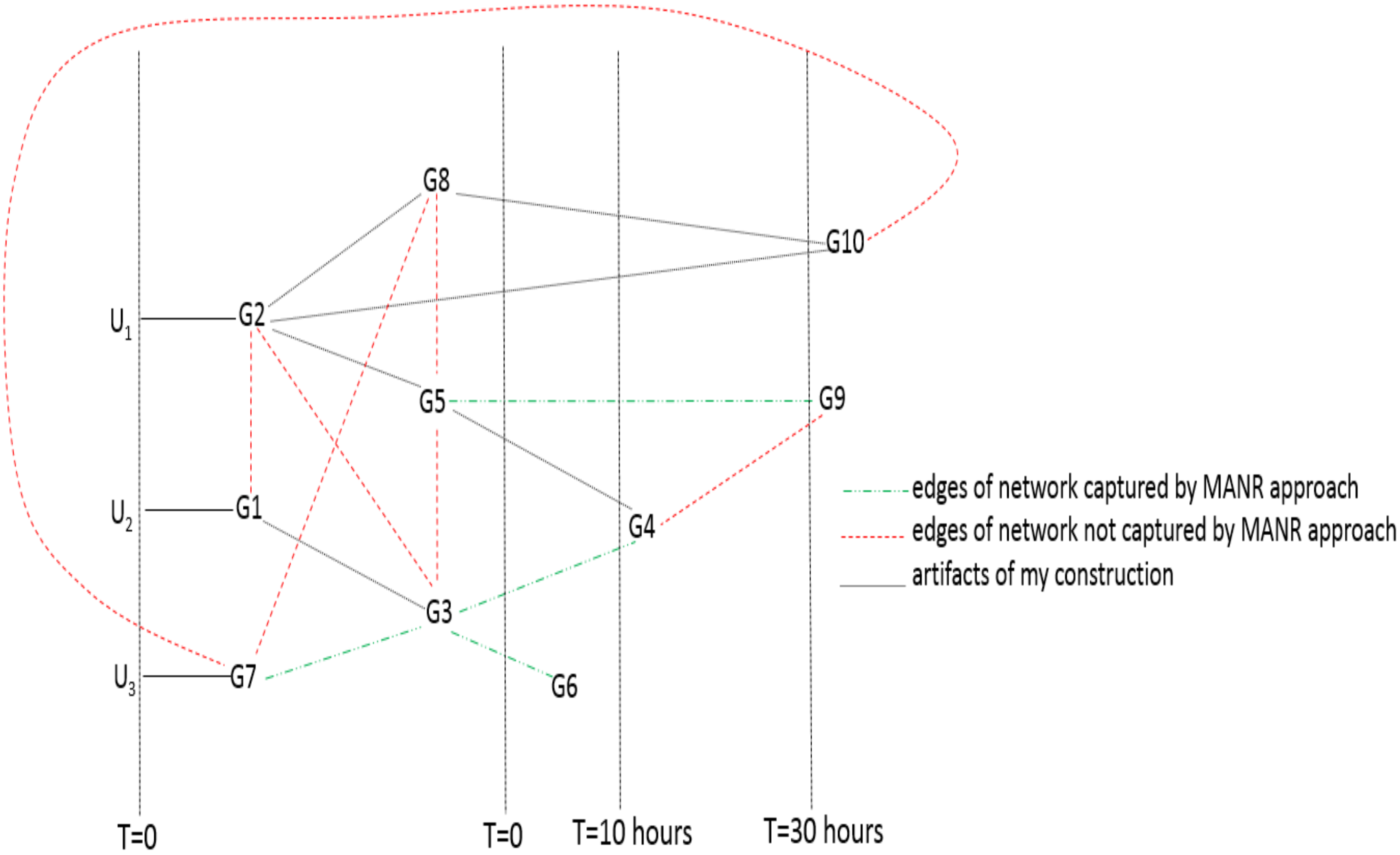

G8
G10
U1
G2
G5
G9
U2
G1
G4
G3
U3
G7
G6
edges of network captured by MANR approach
edges of network not captured by MANR approach
artifacts of my construction
T=0
T=0
T=10 hours
T=30 hours

Figure 34. Evaluation of accuracy of DREAM3 network constructed by MANI approach. The directions from edges in Fig. X were removed indicating only presence or absence of edges. The green dotted lines represented edges that were True Positives (TPs), red dotted lines represented edges that were False Negatives (FNs) and black solid lines represented edges that were False Positives (FPs). The time scales on the network indicates the time points at which the gene gets activated, which was estimated from the duration of lags in the expression profiles of the respective genes.

The key reason for low sensitivity (~ 40%) or high proportion of missed edges (FNs, ~60%) is because the time series data used for the above network inference was obtained by a single perturbation of the 10 gene network. Perturbation of a network based on a single stimulus may activate only certain genes within the network and, therefore, regulatory edges between genes do not get identified if the relationships between genes do not appear either as (i) differential lag or (ii) as correlated expression pattern in the time series data set. Therefore, in order to improve the performance of MANI approach, additional perturbation based time series data from the same network that can establish relationships between genes should be used.

Among the 11 correct network edges, MANI approach was able to detect 4 correct edges and 6 wrong edges from among 90 possible edges between 10 genes in the network. The possibility of this by random network inference would be $\frac{\left(\frac{11}{4}\right) \text{ x } \left(\frac{79}{6}\right)}{\left(\frac{90}{10}\right)} = 0.02$ (is less than 0.05, threshold for network inference to be considered significant). Therefore, my network inference through MANI approach can be considered better than random network inference.

Though the sensitivity of my network construction approach is low, it is still comparable in terms of overall performance to other contemporary algorithms that were used to construct the DREAM 3 network (see below in Table 11).

Table 11. Comparing the performance of my network inference approach to other algorithms on DREAM3 data.

Parameters of assessment	**Majority of network inference algorithms* in DREAM 3**	**MANI approach**
Sensitivity and Precision (estimate is based only on the presence or absence of an edge in the network and not its direction)	Precision < 50%	Sensitivity = 36.36% and Precision = 40%
How much of Knock-out and Knock-down static data was used during the network inference?	10 knock down and 10 knock out data	0
How many of the perturbation based Time Series (TS) data sets were used during the network inference?	4/4 TS data (each TS data set had 21 time points)	1/4 TS data (each TS data set had 21 time points)

* 3.13. State of art network inference algorithms that participated in DREAM3 challenge.

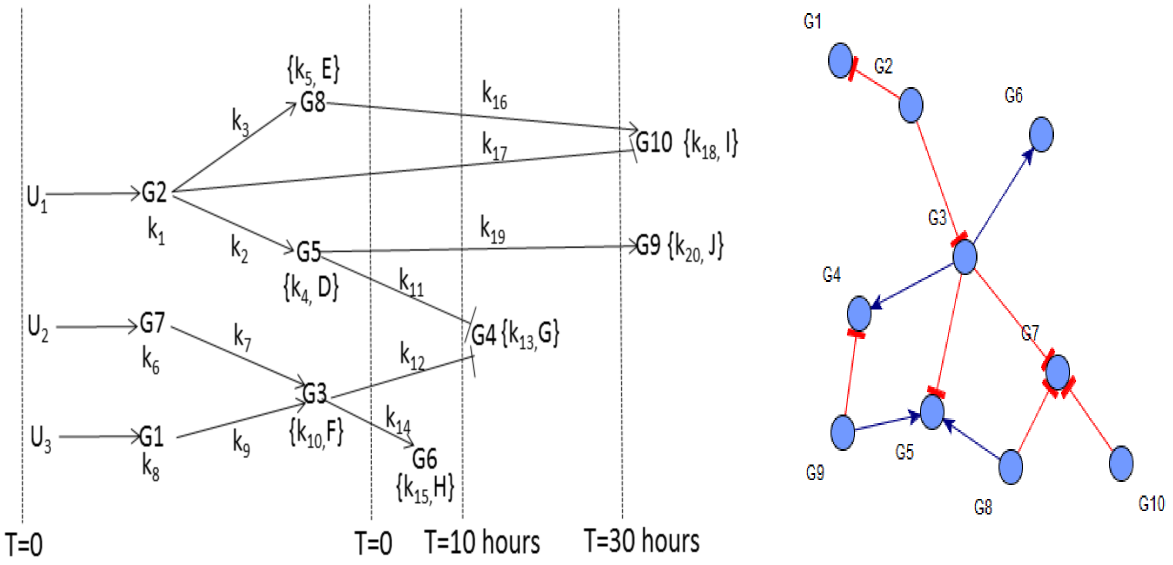
I discussed many of the state of the art network inference techniques/algorithms in section 1.5. Most of them estimated gene relationships using either (i) Mutual Information between genes or (ii) Pair-wise correlation between genes or (iii) conditional probability of interaction between genes or (iv) solved for the interconnectivity matrix in the gene regulatory equation # 5 in section 1.3. There were 29 teams that participated in the DREAM3 challenge to construct the network. Of the 29 teams that participated, I highlight here (see table below) some of the key network inference techniques used by them during the challenge along with references to the original work that published the techniques.

Techniques/principles used in constructing networks	**Source**
Metabolic Control Analysis (MCA) technique that used control coefficients to infer gene relationships.	(de la Fuente et al., 2002)
Parsimonious network created by linear regression using multiple steady state data produced by perturbation.	(Gardner et al., 2003)
Bayesian hierarchical model	(Friedman, 2004)
Conditional correlation analysis to infer gene relationships	(Rice et al., 2005)
Information Theoretic approach to infer gene relationships	(Basso et al., 2005), (Faith et al., 2007)

3.14. What are the contributions of my MANI approach to the field of gene network inference?

If precision of network inference by MANI approach is comparable to already existing network inference algorithms, what is the unique contribution of my algorithm? Precise network inference is a challenging task and an ambitious goal. Time series perturbation data, though can provide insights into the network, would not be adequate to perfectly construct the network. Therefore, my goal was not necessarily to outperform other network inference algorithms but rather focus on obtaining useful knowledge or features ("useful things") that can be learned about the dynamics and overall structure of the gene network using my MANI approach from time series data, which is relatively inexpensive data to obtain in comparison to other kinds of data that can also help infer gene relationships such as knock-out and knock-down static data. I explain the "useful things" that MANI approach can contribute to gene network inference problems in the next two sections.

3.15. “Useful things” that can be learned about the network from MANI



The figure above illustrates the difference in the quality of network constructed from time series data by my MANI approach (left) and the hairball network that is usually generated by other network inference algorithms (right). The construction principle, adopted by MANI approach, was to model the entire network of gene regulations as individual motifs of regulation and gradually build the network through these motifs. This local network inference approach adopted by MANI emphasizing the mechanisms of regulation underlying the network, as opposed to the global inference approach (solving for the network interconnectivity matrix between all the 10 genes at the same time) by other network inference algorithms, enriched the “dynamical” aspects of the constructed network that gave many valuable insights about the network. I highlight below some important “dynamical” features that MANI approach has added to the constructed network.

1. **A key feature of the network captured well in the constructed network is the hierarchy between genes.** MANI inferred regulatory relationship between genes in the network (fig. 33). It was especially successful in delineating hierarchy between genes that had noticeable differences in the durations of lag in their expression profiles (G3, G4, G5, G6 and G9). Most of the edges between these genes were successfully inferred (as shown by green edges indicating true positive in fig. 34) because of the noticeable difference in the durations of lag between these genes as indicated by the time scale.

2. **Time Sensitive Activation Network (TSAN).**

   The time delay or duration of lag in gene expression profile was a key property that was used to infer the hierarchy of genes within each window/motif. In the final constructed network, besides summing up the regulatory relationships between genes from the various windows, genes were also arranged according to their durations of lag in their expression profiles (Figs. 33 and 34 above). Therefore, it was easy to see how the different genes in the regulatory cascade get switched on at different time intervals in response to external inputs using this Time Sensitive Activation Network (TSAN). TSAN is a product that resulted from the unique approach taken by MANI towards network inference due to strong emphasis on lag while inferring the network within each window. TSAN can provide actionable knowledge to biologists because it not only tells the key gene that needs to get switched on for the network to propagate but also the time so that interventions to the network can be timely.

3. **Quality of the network constructed.**

   Since careful emphasis was placed on the mechanism of gene regulation (serial, parallel and serial-parallel etc.) at each step of network inference, the final network (fig. 33) reflected how the overall network cascaded/propagated through its multiple channels over time. The overall structure of the network constructed is easier to interpret for biologists as opposed to a hairball as constructed by other contemporary algorithms.

3.16. Data Economy

One of the striking features of MANI approach is how little data it requires for network inference (Table 11) compared to other network inference algorithms and is still able to produce networks with comparable precision. Therefore, to sum up the key contributions of MANI approach, it is a novel network inference approach that can construct networks with strong, interpretable and actionable “dynamical” features at reasonable precision using limited and cheaper gene expression data (requires only one set of perturbation based time series data, which is cheaper to obtain compared to other more sophisticated experiments such as gene knock-out experiments to tease out gene relationships). Hence MANI can do “more” with “less”.

Chapter 4: Conclusion

Network inference, that is, constructing microscopic gene interactions within a dynamical complex network such as GRN from its macroscopic properties (high volume gene expression data) is a key area of interest in the field of biomedical engineering and systems biology due to interest surrounding identifying target genes in GRN for therapeutic reasons. The current network inference techniques are genome-wide global computational network inference approaches that frequently result in numerous false positive gene interactions whose biological confirmation experiments are impractical and often infeasible. In this study, I have developed two novel network inference approaches, BAKE for static gene expression data and MANI for time series gene expression data, which used a more “local” approach to gene network inference.

BAKE approach used prior knowledge about the network and constructed network locally around that knowledge using high volume static data instead of inferring clusters of genes directly from data. This local approach dramatically reduced false positive findings. I demonstrated BAKE principle by performing network expansion around insulin signaling pathway (prior knowledge) using gene expression data obtained from insulin resistant mice. While expanding network around insulin pathway genes, I discovered and validated a novel regulatory gene, KLF4, in this pathway. KLF4 mediated the down-regulation of IRS2 and TSC2 expression, which was triggered by a high-fat diet in our mouse model and promoted insulin resistance. I also estimated BAKE’s precision (> 44%) by testing its ability to construct a full adipogenesis network using only a partial version of the network as prior knowledge.

MANI approach used a local approach to network inference from time series data by focusing on constructing the network around its motifs, which are the building blocks of the network, and then gradually building the network through the motifs. Each motif represented a subunit of the network and was made up of three genes, and regulatory relationships between genes within each motif were inferred and the network was built through these motifs. Due to MANI’s strong emphasis on mechanism of regulation in its implementation, it produced networks with strong “dynamical” quality at comparable precision (40%). I implemented MANI on a time series data produced by perturbing a 7 gene network belonging to adipogenesis cascade and also subsequently validated MANI on a time series data produced by perturbing a 10 gene network made available by DREAM3 challenge.

As a future extension to my work, I can potentially explore the possibility of using a combination of BAKE and MANI approach to network inference to infer more accurate and dynamical large gene networks but also consider software assisted automation of sequence of steps in BAKE and MANI.

References

ABEGAZ, F. & WIT, E. Sparse time series chain graphical models for reconstructing genetic networks. *Biostatistics*.

ALTAY, G. & EMMERT-STREIB, F. (2011) Inferring the conservative causal core of gene regulatory networks. *BMC Syst Biol,* 4**,** 132.

BALLEZA, E., ALVAREZ-BUYLLA, E. R., CHAOS, A., KAUFFMAN, S., SHMULEVICH, I. & ALDANA, M. (2008) Critical dynamics in genetic regulatory networks: examples from four kingdoms. *PLoS One,* 3**,** e2456.

BAR-JOSEPH, Z. (2004) Analyzing time series gene expression data. *Bioinformatics,* 20**,** 2493-503.

BASSO, K., MARGOLIN, A. A., STOLOVITZKY, G., KLEIN, U., DALLA-FAVERA, R. & CALIFANO, A. (2005) Reverse engineering of regulatory networks in human B cells. *Nat Genet,* 37**,** 382-90.

BIRSOY, K., CHEN, Z. & FRIEDMAN, J. (2008) Transcriptional regulation of adipogenesis by KLF4. *Cell Metab,* 7**,** 339-47.

BLATT, M., WISEMAN, S. & DOMANY, E. (1996) Superparamagnetic clustering of data. *Phys Rev Lett,* 76**,** 3251-3254.

BONABEAU, E. (2002) Agent-based modeling: methods and techniques for simulating human systems. *Proc Natl Acad Sci U S A,* 99 Suppl 3**,** 7280-7.

BUTTE, A. J. & KOHANE, I. S. (2000) Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. *Pac Symp Biocomput***,** 418-29.

CHEUNG, K. J., TZAMELI, I., PISSIOS, P., ROVIRA, I., GAVRILOVA, O., OHTSUBO, T., CHEN, Z., FINKEL, T., FLIER, J. S. & FRIEDMAN, J. M. (2007) Xanthine oxidoreductase is a regulator of adipogenesis and PPARgamma activity. *Cell Metab,* 5**,** 115-28.

DE LA FUENTE, A., BING, N., HOESCHELE, I. & MENDES, P. (2004) Discovery of meaningful associations in genomic data using partial correlation coefficients. *Bioinformatics,* 20**,** 3565-74.

DE LA FUENTE, A., BRAZHNIK, P. & MENDES, P. (2002) Linking the genes: inferring quantitative gene networks from microarray data. *Trends Genet,* 18**,** 395-8.

ESPINOSA-SOTO, C., PADILLA-LONGORIA, P. & ALVAREZ-BUYLLA, E. R. (2004) A gene regulatory network model for cell-fate determination during Arabidopsis thaliana flower development that is robust and recovers experimental gene expression profiles. *Plant Cell,* 16**,** 2923-39.

FAITH, J. J., HAYETE, B., THADEN, J. T., MOGNO, I., WIERZBOWSKI, J., COTTAREL, G., KASIF, S., COLLINS, J. J. & GARDNER, T. S. (2007) Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biol,* 5**,** e8.

FEIZI, S., MARBACH, D., MEDARD, M. & KELLIS, M. (2013) Network deconvolution as a general method to distinguish direct dependencies in networks. *Nat Biotechnol,* 31**,** 726-33.

FRIEDMAN, N. (2004) Inferring cellular networks using probabilistic graphical models. *Science,* 303**,** 799-805.

FRIEDMAN, N., LINIAL, M., NACHMAN, I. & PE'ER, D. (2000) Using Bayesian networks to analyze expression data. *J Comput Biol,* 7**,** 601-20.

GABRIELSSON, B. G., OLOFSSON, L. E., SJOGREN, A., JERNAS, M., ELANDER, A., LONN, M., RUDEMO, M. & CARLSSON, L. M. (2005) Evaluation of reference genes for studies of gene expression in human adipose tissue. *Obes Res,* 13**,** 649-52.

GARDNER, T. S., DI BERNARDO, D., LORENZ, D. & COLLINS, J. J. (2003) Inferring genetic networks and identifying compound mode of action via expression profiling. *Science,* 301**,** 102-5.

GUPTA, A., VARNER, J. D. & MARANAS, C. D. (2005) Large-scale inference of the transcriptional regulation of Bacillus subtilis. *Computers and Chemical Engineering,* 29**,** 12.

HASE, T., GHOSH, S., YAMANAKA, R. & KITANO, H. (2013) Harnessing Diversity towards the Reconstructing of Large Scale Gene Regulatory Networks. *PLoS Comput Biol,* 9**,** e1003361.

HUYNH-THU, V. A., IRRTHUM, A., WEHENKEL, L. & GEURTS, P. (2010) Inferring regulatory networks from expression data using tree-based methods. *PLoS One,* 5.

IM, S. S., KWON, S. K., KIM, T. H., KIM, H. I. & AHN, Y. H. (2007) Regulation of glucose transporter type 4 isoform gene expression in muscle and adipocytes. *IUBMB Life,* 59**,** 134-45.

KAUFFMAN, S. (1995) *At Home in the Universe,* Oxford University Press.

KAUFFMAN, S. A. (1969) Metabolic stability and epigenesis in randomly constructed genetic nets. *J Theor Biol,* 22**,** 437-67.

KHOLODENKO, B. N., KIYATKIN, A., BRUGGEMAN, F. J., SONTAG, E., WESTERHOFF, H. V. & HOEK, J. B. (2002) Untangling the wires: a strategy to trace functional interactions in signaling and gene networks. *Proc Natl Acad Sci U S A,* 99**,** 12841-6.

LEBRE, S., DONDELINGER, F. & HUSMEIER, D. (2012) Nonhomogeneous dynamic Bayesian networks in systems biology. *Methods Mol Biol,* 802**,** 199-213.

LUO, W., HANKENSON, K. D. & WOOLF, P. J. (2008) Learning transcriptional regulatory networks from high throughput gene expression data using continuous three-way mutual information. *BMC Bioinformatics,* 9**,** 467.

MARBACH, D., COSTELLO, J. C., KUFFNER, R., VEGA, N. M., PRILL, R. J., CAMACHO, D. M., ALLISON, K. R., KELLIS, M., COLLINS, J. J. & STOLOVITZKY, G. (2012) Wisdom of crowds for robust gene network inference. *Nat Methods,* 9**,** 796-804.

MARBACH, D., PRILL, R. J., SCHAFFTER, T., MATTIUSSI, C., FLOREANO, D. & STOLOVITZKY, G. (2010) Revealing strengths and weaknesses of methods for gene network inference. *Proc Natl Acad Sci U S A,* 107**,** 6286-91.

MARGOLIN, A. A., NEMENMAN, I., BASSO, K., WIGGINS, C., STOLOVITZKY, G., DALLA FAVERA, R. & CALIFANO, A. (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics,* 7 Suppl 1**,** S7.

MARTIN, S., ZHANG, Z., MARTINO, A. & FAULON, J. L. (2007) Boolean dynamics of genetic regulatory networks inferred from microarray time series data. *Bioinformatics,* 23**,** 866-74.

MCCULLOCH, W. S. & PITTS, W. (1943) A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics,* 5**,** 115-133.

MEYER, P. E., KONTOS, K., LAFITTE, F. & BONTEMPI, G. (2007) Information-theoretic inference of large transcriptional regulatory networks. *EURASIP J Bioinform Syst Biol*, 79879.

MODICA, S., MORGANO, A., SALVATORE, L., PETRUZZELLI, M., VANIER, M. T., VALANZANO, R., ESPOSITO, D. L., PALASCIANO, G., DULUC, I., FREUND, J. N., MARIANI-COSTANTINI, R. & MOSCHETTA, A. (2009) Expression and localisation of insulin receptor substrate 2 in normal intestine and colorectal tumours. Regulation by intestine-specific transcription factor CDX2. *Gut,* 58**,** 1250-9.

NEUMANN, J. V. & BURKS, A. W. (1966) *Theory of Self-Reproducing Automata,* University of Illinois Press.

RESHEF, D. N., RESHEF, Y. A., FINUCANE, H. K., GROSSMAN, S. R., MCVEAN, G., TURNBAUGH, P. J., LANDER, E. S., MITZENMACHER, M. & SABETI, P. C. (2011) Detecting novel associations in large data sets. *Science,* 334**,** 1518-24.

RICE, J. J., TU, Y. & STOLOVITZKY, G. (2005) Reconstructing biological networks using conditional correlation analysis. *Bioinformatics,* 21**,** 765-73.

ROSEN, E. D. & SPIEGELMAN, B. M. (2000) Molecular regulation of adipogenesis. *Annu Rev Cell Dev Biol,* 16**,** 145-71.

ROWLAND, B. D. & PEEPER, D. S. (2006) KLF4, p21 and context-dependent opposing forces in cancer. *Nat Rev Cancer,* 6**,** 11-23.

SAYAMA, H. (2015) *Introduction to the Modeling and Analysis of Complex Systems,* Open SUNY Textbooks, Milne Library, State University of New York at Geneseo, Geneseo, NY 14454.

SEGAL, E., SHAPIRA, M., REGEV, A., PE'ER, D., BOTSTEIN, D., KOLLER, D. & FRIEDMAN, N. (2003) Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat Genet,* 34**,** 166-76.

SHIN, Y. J. & BLERIS, L. Linear control theory for gene network modeling. *PLoS One,* 5.

SZE, K. L., LEE, W. M. & LUI, W. Y. (2008) Expression of CLMP, a novel tight junction protein, is mediated via the interaction of GATA with the Kruppel family proteins, KLF4 and Sp1, in mouse TM4 Sertoli cells. *J Cell Physiol,* 214**,** 334-44.

TANIGUCHI, C. M., EMANUELLI, B. & KAHN, C. R. (2006) Critical nodes in signalling pathways: insights into insulin action. *Nat Rev Mol Cell Biol,* 7**,** 85-96.

VAN SOMEREN, E. P., VAES, B. L., STEEGENGA, W. T., SIJBERS, A. M., DECHERING, K. J. & REINDERS, M. J. (2006) Least absolute regression network analysis of the murine osteoblast differentiation network. *Bioinformatics,* 22**,** 477-84.

VEIGA, D. F., VICENTE, F. F., GRIVET, M., DE LA FUENTE, A. & VASCONCELOS, A. T. (2007) Genome-wide partial correlation analysis of Escherichia coli microarray data. *Genet Mol Res,* 6**,** 730-42.

VILLAVERDE, A. F. & BANGA, J. R. (2013) Reverse engineering and identification in systems biology: strategies, perspectives and challenges. *J R Soc Interface,* 11**,** 20130505.

VU, T. T. & VOHRADSKY, J. (2009) Inference of active transcriptional networks by integration of gene expression kinetics modeling and multisource data. *Genomics,* 93**,** 426-33.

YEUNG, M. K., TEGNER, J. & COLLINS, J. J. (2002) Reverse engineering gene networks using singular value decomposition and robust regression. *Proc Natl Acad Sci U S A,* 99**,** 6163-8.

YUAN, Y., STAN, G.-B., WARNICK, S. & GONCALVES, J. (2011) Robust dynamical network structure reconstruction. *Automatica,* 47**,** 6.

ZOPPOLI, P., MORGANELLA, S. & CECCARELLI, M. (2010) TimeDelay-ARACNE: Reverse engineering of gene networks from time-course data by an information theoretic approach. *BMC Bioinformatics,* 11**,** 154.

Supplementary material

Table S1. Genes belonging to prior network, Insulin signaling pathway (L_{path})

Gene_Symbol
IR A/B
Arrb2
IRS1
IRS2
IRS3
Pik3cb (P110 beta)
Pdpk1 (PDK1)
FRAP1
Rictor
Raptor
Akt2
Akt1
glut4
Ins2
Ins1
PDK4
PTP 1B
Socs3
Socs1
Retn
Tnfalpha
SHP2 (ptpn11)
JNK1
JNK2
Pik3r1
Pik3r2
Pten
ship2
Trib3
GYS1
Rab8a
Rab10

Tnfrsf1a
Tnfrsf1b
Phlpp1
pp2ac
Tbc1d1
Tbc1d4
PKC-theta
PHLPP2
Gsk3a
Gsk3b

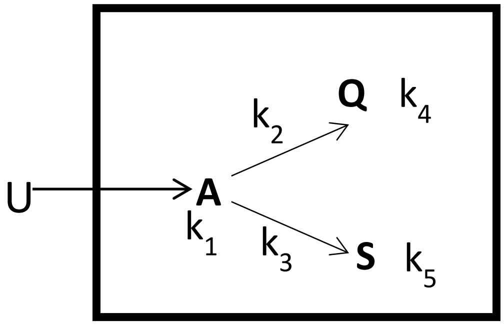
Table S2.List of Anchor genes

Gene_Symbol
IRS2
p110Beta
FRAP1
Rictor
Raptor
AKT2
AKT1
Glut4
SOCS3
SHP2
SHIP2
GYS1
Rab10a
Tnfrsf1a
PP2AC

How to estimate BIC score for a network?

Though the following was derived for a parallel mechanism, it can be applied to other forms of mechanisms as well.

Parallel Mechanism



K=[U, k_1, k_2, k_3, k_4, k_5]
G=[A, Q, S]
P(Mechanism X/Data) α P(Data/Mechanism X) P(Mechanism X)
α P(G/Mechanism X) P(Mechanism X)
α P(G/Mechanism X) (non-informative prior)
If X= parallel mechanism as shown in the figure above
L= P(G/Mech.X) =P(G/K)=P(A,Q,S/K)=P(A/U, k_1). P(Q/U, k_1, $k_{2,}$ k_4). P(S/ U, k_1, $k_{3,}$ k_5)
= P(ε_A/ U, k_1). P(ε_Q/ U, k_1, $k_{2,}$ k_4). P(ε_s/ U, k_1, $k_{3,}$ k_5)
$\varepsilon_i \sim N(0, \tau_i)$ (follows a normal distribution)
=

$$\prod_{i=1}^{n} e^{\frac{-(A\exp - Apred)^2}{2*\tau_A}} \prod_{i=1}^{n} e^{\frac{-(Q\exp - Qpred)^2}{2*\tau_Q}} \prod_{i=1}^{n} e^{\frac{-(S\exp - Spred)^2}{2*\tau_S}}$$

(i) Where Aexp, Qexp, Sexp are the observed values of gene expression of genes A, Q and S respectively while Apred, Qpred and Spred are the predicted values of gene expression of genes A, Q and S respecitvely by the model .

(ii) n is the number of sample points in the time course data.

$$-\ln L = (SSE_A/\tau_A + SSE_Q/\tau_Q + SSE_R/\tau_R)/2$$

$$BIC = -2. \ln (L) + p .\ln (n) = (SSE_A/\tau_A + SSE_Q/\tau_Q + SSE_R/\tau_R) + p_A .\ln (n_A) + p_Q .\ln (n_Q) + p_S .\ln (n_S) \quad ...(1)$$

if $\tau_{A\,=}\ \tau_{Q\,=}\ \tau_{S\,=}\ \tau$ and $n_{A\,=}\ n_{Q\,=}\ n_{S=}$ n then,

$$BIC = (SSE_A + SSE_Q + SSE_R)/\tau + p_{total} .\ln (n) \quad ...(2)$$

where $p_{total} = p_{A\,+}\ p_{Q\,+}\ p_C$