BILATERAL NSF/BIO-BBSRC.
ABI INNOVATION. MULTI-SCALE GENE FUNCTION PREDICTION USING BIOLOGICAL NETWORKS

1. Specific Aims

In recent years, the numerous large-scale sequencing projects combined with fast sequencing techniques have generated enormous amounts of sequence data. This has led to the identification of thousands of previously unseen genes (e.g. protein coding and non-coding RNAs). A fundamental goal is therefore to identify the function of uncharacterized genes on a genomic scale. It is difficult to design functional assays for genomic elements that have not been previously described. Currently, a major challenge in bioinformatics is to devise algorithmic methods that, given a gene or ncRNA, can predict a hypothesis for its function that can then be validated experimentally.

In this project, we shall focus on understanding the various aspects of the gene function as well as the key elements that define and determine it. Our goal is to build a general system that, given a gene, can predict its function. This multi-scale prediction will be carried out exploiting the structure of biological networks.

This project will be developed as a bilateral collaboration between the groups of Dr Mark Gerstein (US NSF PI) at Yale University and Dr Alberto Paccanaro (UK BBSRC PI) at Royal Holloway University of London. The two PIs have a long history of successful collaborations on many network based approaches for biological problems. They have developed methods for predicting networks from heterogeneous biological datasets including genome features \cite{14564010,15998909,16413578,19015141}, protein function prediction \cite{23353650,19402753} and semantic similarity between genes \cite{17540677,24659104} as well as numerous software tools to address these problems.

AIM 1: We plan to develop a computational framework to identify and characterize gene functions using logic-circuit models and regulatory networks. Specifically we propose to develop a method to analyse logic operations of small regulatory triplets using a two-in-one-out logic gate model. We will use a binarized gene expression data to score how well each triplet matches each of all 16 possible logic gates. A high score implies that the logic operation describes accurately the interactions between elements forming the regulatory triplet. As such a similarity in logic gate matches between various triplets implies a similarity in function between the corresponding elements.

AIM 2: We will develop a computational workflow to infer phenotypic function using as input network neighbourhoods and data mining. For this we will use machine-learning techniques on a graph model that can explain the association between the data. Here we make the presumption that attributes associated to characterized-entities can be extended to all the uncharacterized entities that are directly connected to them in a graph model. In this project the graphs will be constituted by large-scale biological networks. Thus for any given genome we will construct a relational network and predict phenotypes of uncharacterized genes using a guild by association model for genes directly connected in the network.

AIM 3: We plan to integrate our results from **AIM 1,** specifically synthesizing the circuit elements and their domains of influence within a regulatory network into logic modules, with phenotypic function predictions from **AIM 2** to better demarcate regions of the network associated with distinct phenotypic functions. Thus will develop a recursive computational method to optimize the phenotypic predictions**.**

2. Background and Preliminary Results

2.1 General Background

The past decade has seen fast grow of genomic data becoming available providing a rich and fertile medium for the study of gene function. Numerous clues regarding the various aspects of gene function are hidden in a vast array of gene expression, metabolite expression and protein-protein interaction data. However, as gene databases grow in size the diversity among the sequences increases and classical homology based methods become less effective \cite{16772267}. Thus the scientists tried new approaches to mine this data for improving the function predictions. As many of these types of data have a natural representation as networks the scientific community has focused on developing methods that make use of network topology for functional inference.

This approach has been pioneered in a most interesting work by Marcotte et al. \cite{10573421}, in which the authors built a network where each node corresponded to a protein in the *S. cerevisiae* genome, and the links between two proteins represented correlated evolution (through phylogenetic profiles), patterns of domain fusion, co-expression and protein-protein interaction. Treating these links as independent, their method consisted in assigning to an uncharacterized protein the function shared by the proteins it was connected to. Since this work appeared, other approaches have been developed, which use networks topologies to infer functional annotation. Most of them use networks built from protein-protein interaction (PPI) data and they could be broadly divided into two categories. A first group of methods breaks the networks into modules and then identifies the function of an unknown protein based on the function of the known member in its module \cite{12538875,15374873,14517352} to name a few. A second group of methods, similar to Marcotte's, assign a function to a protein by directly considering the function of its neighbours (e.g. \cite{12740586,14980019,12855458,15961472}).

2.2 Background on Networks

2.2.1 Networks Biology: A Growing Field

Biological systems are mediated by interactions between thousands of molecules. Network-based statistical models are particularly useful in unlocking the complex organization of biological systems. In the last decade, biological network analysis has blossomed into a new scientific discipline. Examples are numerous, ranging from protein–protein to genetic interaction networks \cite{17473168}. Usually, networks are depicted as graphs with nodes and edges, where nodes denote biological entities such as proteins or genes, and edges represent interactions between nodes.

Cellular networks are organized in the form of interacting modules, whereby nodes in a module tend to have a larger density of edges connecting them. Biologically, the genes within a module of a genetic regulatory network are co-regulated. Graph models can reveal interesting new features of the analysed biological system \cite{11034217,10521342,10935628,12202830,12399590,16730024}, while network topologies can be used to address fundamental biological questions \cite{18421347,15190252,12134151,17274682,19372386,16311037}.

2.2.2 Preliminary Results on Networks

Gerstein lab key papers: [[CSDS – to add]]

Gerstein lab has carried out projects in biological networks for over a decade. We have made extensive contributions in the analysis of genomic data, especially with regard to network prediction and analysis\cite{14564010}. We have also integrated regulatory networks with gene expression to uncover different kinds of dynamic sub-networks \cite{15372033}. We developed methods to determine the hierarchical organization of regulatory networks and applied them to analyze the regulatory networks of a variety of species from yeast to human,

including networks constructed from ENCODE, modENCODE and MCF7 data \cite{22125477,20439753,22955619,21177976}.

Network Construction

We have developed several methods to construct networks based on various genome features \cite{14564010}. We extended this work by combining several heterogeneous biological datasets \cite{12350343,15998909,16413578} and developing new machine learning techniques \cite{19656385} to increase the prediction power. In 2008, our work placed first in the Dialogue for Reverse Engineering Assessments and Methods (DREAM, www.the-dream-project.org) competition for the in silico network prediction challenge. In addition, we have participated in many experimental network determination projects, to refine and keep our methodologies at the cutting edge \cite{16449570,16554755,14704431}.

Recently, we have completed the ambitious goal of constructing draft regulatory networks for humans and model organisms based on the mod/ENCODE datasets \cite{21177976,21430782,22955619}. We have successfully completed this challenge through the development of novel approaches for identifying individual proximal and distal edges, as well as creating new miRNA target prediction algorithms.

Leveraging the richness of the next-generation sequencing we have developed several computational approaches to help construct and analyze proximal and distal regulatory networks \cite{19122651,22039215,20126643,22950945}. When analyzed together, the proximal and distal regulatory network elements provide the complete multi dimensional image of the transcriptional regulatory network.

Network constructions using ENCODE, modENCODE and other system-wide data. Using the machine-learning approaches we have constructed highly integrated regulatory networks for humans and model organisms based on ENCODE \cite{22955619} and modENCODE datasets \cite{21430782}. These integrated networks consist of three major types of regulation: TF-gene, TF-miRNA and miRNA-gene, showing rich statistical patterns. For instance, the human regulatory network uniquely displays distinct preferences for binding at proximal and distal regions. The proximal-distal binding preference is a property of the intergenic space in the human genome, which is much larger relative to the genomes of other model organisms. Furthermore, in the human regulatory network, the more highly connected TFs are more likely to exhibit allele-specific binding and gene expression. More recently, we built a regulatory map for 24 nuclear receptors and 14 breast-cancer-associated TFs that are expressed in the breast cancer cell line MCF-7. The resulting network reveals a highly interconnected regulatory matrix with extensive “crosstalk” between NRs and other breast-cancer-associated TFs. We show that large numbers of factors bind in a coordinated fashion to target regions throughout the genome. The highly occupied targets are associated with active chromatin state and hormone-responsive gene expression.

Network Analysis

Biological networks, normally large in scale, are organized with topological structures in the form of interacting modules. Statistics such as 'eccentricity' and 'betweenness' are helpful to explain the connectivity and behaviour of nodes in a network. We have developed a number of tools \cite{15145574,17447836} to analyse the organization and structure of biological networks including identifying the importance of a node in a single network and identifying the modular structure inherent within several biological networks.

Nodes in networks tend to work together as small structures called network motifs. We found that in many of the regulatory networks we constructed in human, worm and fly, the small modular motifs have been evolutionarily reused to create complex transcriptional regulatory networks. The feed-forward loop is over represented in these networks and is used to filter the input stimuli regulating the transcriptional machinery across different hierarchical levels to modulate the expression level of different genes.

Networks and Cellular Function

Cellular networks are organized in the form of interacting modules, whereby nodes in a module tend to have a larger density of edges connecting them. Biologically, the genes within a module of a genetic regulatory network are co-regulated. We developed various methods to identify the functional modules of various networks. For example, by mapping gene-expression data onto the regulatory network of yeast, we identified different sub-networks that are active in different conditions \cite{1537203}. We developed a method to extract metabolic modules from metagenomic data, enabling us to identify pathways that are expressed under different environmental conditions \cite{19164758} (Figure 2). We have also developed a way to identify nearly complete, fully connected modules (cliques) present in network interactions \cite{16455753} and we have been using networks to map various kinds of functional genomics data \cite{22955619}.

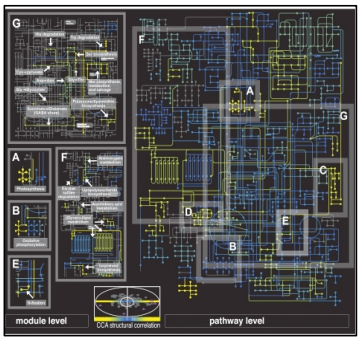


Figure 1 Metabolic network modules active in a specific environment \cite{19164758}.

Integrating Networks with Other Biological Data

To further illustrate the value of the network concept, we have also combined network analyses with many other types of biological data. Recently, we used networks to improve our understanding of genomic variants \cite{24092746}. In \cite{23505346}, we built a multi-layered network that incorporated information from heterogeneous data sources such as protein-protein interactions and metabolic, phosphorylation, signaling, genetic, and regulatory networks. In general, population variants are more likely to be deleterious when they occur in genes or in regulatory elements associated with hubs in the multi-layered networks, indicating that a gene's interactions likely influence the selective pressures on acting on it. Connectivity is also related to selective pressure in noncoding regions, as transcription binding motifs with greater connectivity tend to be under stronger evolutionary pressure \cite{24092746}. We built a workflow model to prioritize noncoding mutations in disease variants based on these patterns of negative selection in functional variants. In addition, we showed that proteins under positive selection are found on the network and on the cellular periphery, an indication of how human variation is arranged with respect to the interactome.

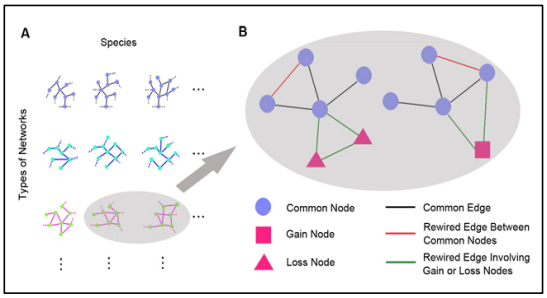


Figure 3. Measuring network rewiring by comparing networks of species pairs. A. Types of biological networks with currently available data for different species are collected. Selected types of commonplace networks with multiple time-point data are also collected. **B.** For each network type, we perform edge rewiring analysis for pairs of species. Three types of nodes are first identified as CNs, GNs and LNs. Four types of rewired edges are then identified and counted including gain/loss edges between CNs (red) and those involving GNs or LNs (green). Rewiring rate from comparing the networks is calculated.

Networks Evolution and Rewiring

We have also explored the evolution of networks and studied the conservation and variability of differ-rent parts of the network. We defined "interologs" and showed how to compare interaction net-works between organisms. We also defined "regulogs" for transferring regulatory relation-ships between organisms \cite{15173116}. In addition, we developed a method to study network rewiring on all currently available biological networks (Figure 3).

We noted that biological networks show a decreased rate of change over large time intervals. However, different types of biological networks consistently rewire at different rates \cite{21253555}.

Web Tools for Network Analysis

We have developed numerous network analysis web tools such as TopNet \cite{14724320}, tYNA \cite{17021160}, and PubNet \cite{16168087}. These tools have been widely used by the research community to analyze network topology—i.e., to calculate hubs, "betweenness", shortness of paths and degree of modularity.

2.3 Background for Phenotypic Function Prediction

Even for genes whose molecular function and cellular roles are known, understanding their role in affecting a certain phenotype remains a challenge. Apart from the Mendelian single gene traits, a substantial portion of the phenotypes we observe in nature are an effect of complex interplays between numerous genes in addition to various environmental factors. Such 'complex traits' are hard to predict and the development of methods for uncovering genotype-phenotype relationships has been identified as one of the major post-genomic challenges \cite{9790834}.

Comparative genomics has been proposed for uncovering such gene-trait relationships \cite{9790834,9598967}. This approach begins by constructing phenotypic profiles, which indicate which organism exhibits a particular phenotype – this is similar to the concept of phylogenetic profiles \cite{10200254}. Then causal relationships between genes and traits can be deduced from the co-occurrence of genes and phenotypes across a large number of genomes. The underlying principle is that species sharing a phenotype are likely to utilize orthologous genes in the involved biological process. These ideas were applied to predict genes involved in well characterised traits such as hyperthermophily \cite{12683966} and flagellar motility \cite{12546786}. Several approaches have been developed for this comparative analysis. For example, Tamura et al. \cite{18467347} proposed a rule based data mining algorithm to associate Clusters of Orthologues Groups of proteins (COGs) with phenotypes; Slonim et al. \cite{ 6732191} proposed an information-theoretic approach to extract preferentially co-inherited clusters of genes having significant association with an observed phenotype. Paccanaro and Gerstein have developed a correlation-based method \cite{17038185} that was able to discover genotype-phenotype associations combining phenotypic information from a biomedical informatics database, GIDEON, with the molecular information contained in Clusters of Orthologous Groups of proteins (COGs) \cite{12969510}.

3. Research Plan and Methods

3.1 AIM 1: Developing a Method to Infer Gene Cellular Role Using Logic Circuit Models and Biological Networks

Our aim is to develop a novel method of inferring a gene cellular role from the analysis of biological networks. More specifically, we will integrate regulatory networks with gene expression data. This will allow us to analyse the interactions between the regulatory factors and target genes using a logic operations based algorithm, Loregic. This study will highlight common behaviour patterns between various RFs as well as groups of genes under similar regulatory constraints. We plan to make available Loregic available as an online tool as well as a stand-alone application that can be downloaded and used on various input datasets.

3.1.1 Background on Logic Circuit Models in Biological Networks

Gene expression is a complex process controlled by regulatory factors on multiple dimensions. An increasing number of recent experimental and computational studies suggest that gene transcription is regulated cooperatively by numerous factors (i.e. TFs and miRNAs)

\cite{24009496,22955619}. These studies analyse the relationships between the regulatory factors (RFs) from various aspects such as protein-protein interactions, sequence motifs in cis-regulatory modules, co-associations of TFs in binding sites, and co-expressions of TF target genes \cite{14627835,22705667,21828005}. However, they focus solely on the identification of the wiring relationships between RFs (e.g. co-binding, co-association and co-expression) leaving untouched the cooperative patterns that drive the biological functions behind the wiring diagrams.

At a high level, the gene regulatory network can be regarded as an electronic circuit, with TFs and miRNAs acting as resistors and capacitors. Just as wiring different circuit elements can generate various electrical functions, connecting various regulatory factors as functional modules will result in different biological functions. Thus, in order to obtain a comprehensive map of gene regulation, it is necessary to go beyond identifying the wiring relationships among individual RFs. Here we propose to study the RFs cooperative patterns, and further regulatory functional modules resulting from those cooperative patterns.

In numerous cases gene regulation can be regarded as a logic process where RFs are the input variables while the target gene expression is the output \cite{12782112,19180174,14530388,21414487,22927416,23412653, 21885784}. In this respect, a common regulatory triplet, with two RFs regulating the same gene, can be formally described by a two-in-one-out logic gate.

Combinatorial logics are numerous extending beyond the three basic logic operations: AND, OR, and NOT \cite{14530388}. For example, for any two-in-one-out scenario, there are 16 possible logic gates (including all possible combinations between positive and negative regulators). In order to capture all possible combinatorial cooperations between regulatory factors we need a more comprehensive model. Previous studies took advantage of binarized regulatory data provided by perturbation experiments (i.e. TF knock-outs) and used a Boolean model to capture this logic processing \cite{Somogyi}. However, previous efforts focused only on a small set of genes, missing the genome-wide identification and characterization of logic operations in gene regulation.

Here we propose a novel approach, Loregic, that will allow a comprehensive analysis of all possible regulatory logic operations from a genome-wide perspective.

3.1.2 Development of the Logic Circuit Models Approach - Loregic

Loregic is a logic operation based method that requires two types of input data: a regulatory network (defined by regulatory factors and their target genes) and a binarized gene expression dataset across multiple samples. The binarized gene expression data (on – 1 and off – 0) is the direct result of the network's regulatory factors activity on the target genes. The inputs can be chosen from different resources to meet the user's interests. The regulatory network is decomposed into regulatory modules formed by triplets consisting of 2 RFs and a common target gene T. Loregic algorithm comprises of five steps (Figure 4).

- **Step 1:** Input gene regulatory network consisting of regulatory factors and their target genes;
- **Step 2:** Identify all RF1-RF2-T triplets where RF1 and RF2 co-regulate the target gene T;
- **Step 3:** Given a particular triplet (RF1, RF2 and T) query the binarized gene expression data

- **Step 4:** Match the triplet's gene expressions against all possible two-in-one-out logic gates based on the binary values;
- **Step 5:** Find the consistent logic gate(s) that best matches the expressions and calculate the consistency score. Test the score significance against random effects;
- Repeat **Step 3-5** for all triplets in the regulatory network.

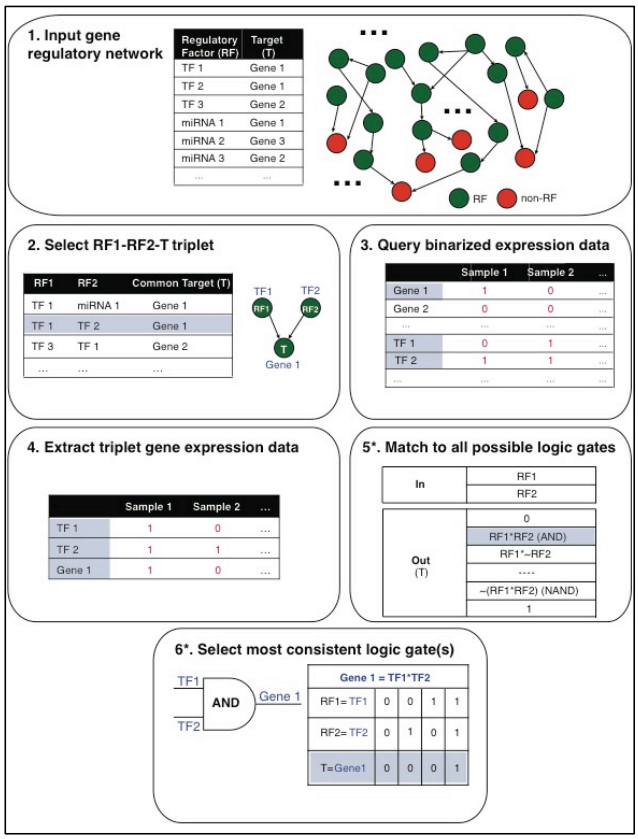


Figure 4 Loregic workflow.

Loregic describes each regulatory module (triplet) using a particular type of logic gate – i.e. the logic gate that matches best the binarized expression data for that triplet across all samples. If such a logic gate is found, we claim that the regulatory triplet is defined by a **consistent logic gate**. Next Loregic calculates the corresponding **consistency score** for the selected gate (Figure 5). Logic gate consistency score is calculated as follows.

- Create the truth table. A logic gate with two inputs (RF1, RF2) and one output (T) can be determined by a combination of four (RF1, RF2, T) binary vectors, ***v1***=(RF1=0, RF2=0, T), ***v2***=(RF1=0, RF2=1, T), ***v3***=(RF1=1, RF2=0, T), and ***v4***=(RF1=1, RF2=1, T) with specific values (0 or 1) for T.
- Given a RF1-RF2-T triplet, match output T (0 or 1) for each of four input combinations of RF1 and RF2, and find the logic gate(s) that describes best the truth tables.
- Calculate the consistency score: For any triplet with m binary vectors and any gate g the gate **consistency score** of the triplet is, $S(g)=(n_1+n_2+n_3+n_4)/m$, where n_i as number of vectors matching $v_i(g)$ with i=1,2,3,4.

Also in order to verify the authenticity of the consistent logic gate given a triplet of (RF1, RF2, T), we calculate its significances over the 16 logic gates' scores. We suppose that it matches the k^{th} logic gate, G_k. We replace the target gene, T by a randomly selected gene N times ($N=1000$), and calculate its significance score, as $p(Gk)=(Gk)/N$. A high significance score implies that random effects may cause the matched logic gate. We suggested to select the consistent logic gates within top 2% of consistency and significance scores.

In the case where there is no consistent logic gate found, we claim that the triplet is inconsistent with all logic gates. The negative result indicates that the activity relationship between the two RFs cannot be described by a standard logic operation. There are several potential explanations. First, the cooperative patterns of two RFs might follow a more complex mechanism, which can't be simply modelled as Boolean logics. Second, the target gene can be regulated by more than two RFs, thus we might need a higher-order logic circuit model with multiple inputs (>2) to capture the RF logics to the target. Finally, the target gene expression may also be impacted by stochastic signals, which can't be described as deterministic models such as logic gates.

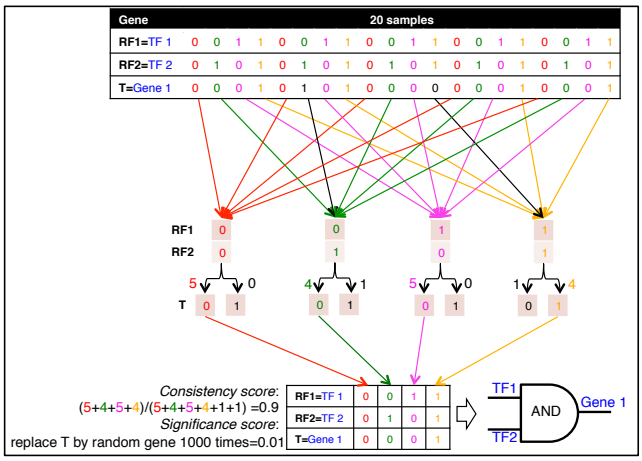


Figure 5 Consistency score evaluation

All the triplets that can be described by logic gates can be further mapped onto other regulatory networks. As such the logic gates information brings a new dimension to the interaction between regulatory elements and targets. We tested and validated the enhancement in the information obtained through Loregic on the study of yeast regulatory network. We studied the TF promoter sequence motifs and used predicted TF logics to infer potential indirect bindings.

3.1.3 Preliminary Results on Logic Circuit Models

Cooperative behaviour between yeast TFs during cell cycle

As an initial application to test the effectiveness of the proposed method, we have used logic circuit models to study the cooperation between yeast TFs during cell cycle. We identified ~39k TF-TF-target triplets from 176 different TFs using TF-target assignments in \cite{15343339,19690563}. We used Loregic to characterize the TF-TF-target logics during yeast cell cycle across 59 time points. We found 4126 TF-TF-target triplets with consistent logic gates (Figure 6). Among those, we found that "T=RF1*RF2" (AND), "T=~RF1*RF2", and "T=RF1*~RF2" logic gates, have more triplets matched than all the others. The AND triplets mean that both TFs have to be present to activate the expression of their target gene.

TF1-TF2-target triplets in Yeast Cell Cycle

Triplet ID	RF1	RF2	Common Target Gene (T)	Consistent logic gate (score)
1	YHR084W	YBR083W	YBR082C	AND (0.9)
2	YKL112W	YIL131C	YMR198W	OR (0.85)
…	…	…	…	…
39011	YMR043W	YER111C	YDR507C	OR (0.81)

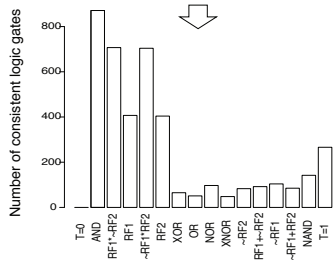


Figure 6 Loregic Validation.

We tested our TF co-operativity results analysing the fold changes in the target gene expression as consequence of deleting one of the regulatory triplet TF.

The yeast TF knockout experiments gave us fold changes in gene expression as a result of deleting a single TF \cite{17417638,20385592}. If a target gene is regulated by two

cooperative TFs in an "AND" relationship, deletion of either TF may corrupt the co-operativity and that impacts gene expression. For example, for the triplets with high significant scores at "AND" gate, we found that deleting either of their TFs gave rise to considerably down-regulated target genes, i.e., negative expression fold changes (t-test p-value =0.068). For non-cooperative TFs such as "T=RF1" or "T=RF2" gates, i.e., one of TFs (dominate TF) fully determines target gene expression, we found that target genes are more affected (down-regulated) by the removal of the dominant TFs rather than by deleting the other TFs (t-test p-value < 0.05 for T=RF1, <0.005 for T=RF2).

3.2 AIM 2: Inferring Phenotypic Functions Through Network Mining (Using Graph Models and Latent Variables)

[[AP: I NEED TO WRITE A COUPLE OF SECTIONS IN THE BACKGROUND, to which I refer to here. These will describe the experimental data available and the phenotype ontologies we will use. But first we need to decide on the organisms.]]

The vast array of available data brings a fresh perspective in the area of gene function prediction, offering clues about gene-phenotype associations. Integrating various datasets (from wet-lab experiments \cite{toADD} to phenotype ontologies \cite{toADD}) we believe that we can make statistically significant large-scale phenotypical inferences.

The data available for phenotypic function prediction can be divided into two categories. While some type of data translate directly into a probability of a given phenotype, other types of data describe instead a "relatedness" in the phenotype associated to two genes *in the same genome*. For example, detecting a phenolog provides a probability P that a gene has phenotype F. On the other hand, finding a certain correlation between the profiles of the expression of genes X and Y gives a certain probability Q that the two genes have related phenotype. We shall refer to these two types of data as "unary relations" and respectively "binary relations" (see Table 1).

Table 1. Phenotypic function prediction input data types.

Data Type	Example
UNARY RELATIONS	Experimental evidence; Phenolog (homology)
BINARY RELATIONS	Gene expression Protein expression Protein-protein interaction Genetic interaction Pathway information Linkage maps ?

The binary relation has a natural representation as a graph. Recently there has been a lot of interest in the machine learning community on methods for inferring graphs and relations on graphs. We propose to leverage the on the current knowledge base and develop theoretical graph based methods for large-scale phenotypical inference. The approach makes use of the phenotypical label associated with some genes to infer phenotypes of uncharacterized ones (semi-supervised learning).

In a typical situation, for a given genome there will be genes which have already been associated with a given phenotype, and genes whose associated phenotype is still unknown. We begin by constructing graphs, in which the nodes represent the genes and each edge represents a (binary) relation between the two genes it connects, e.g. co-expression. Each edge is labelled with a value that quantifies the relation it represents (e.g. their level of co-expression); similarly each node is labelled with their kwon phenotypical assignment or "NA" otherwise.

The two different types of relations described above will be treated differently for inference: binary relations will allow the characterization of the unknown genes by *diffusing* the information of the labelled nodes over the graph, through the links; while unary relations will be thought of as representing a "tendency" (or a prior probability) of a gene to be associated with a given phenotype.

Binary relations inference by diffusion. Intuitively the process of binary relations inference by diffusion can be visualized by considering a network where each node is a gene and each edge is a binary relation characterizing the relationship between the connected genes. We label each edge with the phenotypes describing the gene that it connects too. As such, the

diffusion process allows to assign phenotypes to uncharacterized genes based on the edge labels that connects that gene with a gene of known phenotype.

In summary the diffusion of information over graphs offers a natural framework for integrating datasets which are themselves graphs. This process produces evidence for phenotypical assignments which can then be integrated with the evidence coming from the unary relations using a statistical method, e.g. the Bayesian model. The strength of the methodology proposed here lies in its ability to use diverse sets of noisy data, and to combine them to obtain sound statistical inferences of gene phenotypes; the weak signals contained in each dataset is enhanced by integrating various types of data.

3.2.1 Algorithm Development

The phenotype inference method will contain several parameters that will be learned from the data. Here we assume that, for a given genome, this will be done by applying various machine learning techniques (as described below) to subsets of genes for which the phenotypic assignment is known (training sets). The method development will have to solve two main issues: (i) how to integrate information coming from different experimental sources; and (ii) how to properly diffuse the information over the graphs. The study of solutions for these two problems will constitute most of the algorithmic research of **AIM 2**. In the remaining of this section we shall analyze each one in turn, proposing some possible ideas for their solution.

Integration of Information from Different Experimental Sources

As anticipated earlier, a possible method for integrating the various types of information is using a statistical Bayesian model. Using the Naïve Bayes assumption, we can rewrite the likelihood of the combined vector of evidences given the phenotype as a product of each evidence given the phenotype. That is, the posterior probability distribution of the phenotypic assignment given the evidence, $P(F_i \mid E_1 \ldots E_n)$, is defined as:

$$P(F_i \mid E_1,\ldots,E_n) = \frac{P(E_1,\ldots,E_n \mid F_i)\cdot P(F_i)}{\sum_j P(E_1,\ldots,E_n \mid F_j)\cdot P(F_j)}$$

and can be approximated by:

$$P(F_i \mid E_1,\ldots,E_n) = \frac{P(E_1 \mid F_i)\cdot\ldots\cdot P(E_n \mid F_i)\cdot P(F_i)}{\sum_j P(E_1 \mid F_j)\cdot\ldots\cdot P(E_n \mid F_j)\cdot P(F_j)}$$

where $(E_1 \ldots E_n)$ is the combined vector of n different evidences or features (E_j), and F_i represents the ith phenotypical assignment. Each E_j represents evidence coming either from a unary relation (e.g. a phenolog) or a binary relation (e.g. co-expression). Since unary and binary relations must be treated differently, their likelihood model $P(E_j|F_i)$ will be built in a different way from the training set.

For unary relations, the likelihood models, $P(E_j|F_i)$, can be approximated directly by using maximum likelihood estimates, that is by using the frequencies of the features in the training set (or alternatively using more robust "smoothed" estimates). In other words, for each value of a given feature, we calculate the ratio of how many times the genes with phenotype i have that value of the feature to the total number of genes in that class (in case of continuous features, these must first be discretized).

In order to estimate a likelihood model for a given binary relation we first need to build a graph, and then we need to run the diffusion process (described in the next sub-section). The graph will be fully connected and will have a node for each gene. The values for the edges controlling the diffusion process would be a non-linear mapping of the experimental data

Cristina Sisu 30/7/14 16:11

Comment [1]: [[AP wrote]] Intuitively the process of binary relations inference by diffusion can be visualized by considering a network of water wheels connected by underground pipes in which water flows: for each node (gene) we have a wheel, and for each edge (binary relation) we have a pipe connecting the corresponding wheels. The pipes have different sizes according to the edge label, thus allowing different amounts of water to flow through them, depending on the strength of the relation. Each different phenotypical assignment of genes in the dataset is represented by a salt of a specific colour. When a salt is dropped in a wheel, it colours the water in it, and we shall assume that waters of different colour don't mix. The diffusion process consists in dropping the coloured salt of each known gene in its corresponding wheel, and then letting the coloured water be transported by the pipes. No salt is dropped in the wheels corresponding to the uncharacterized gene. However, the water in these wheels will also eventually become coloured due to the coloured waters coming from the connected pipes. After the coloured waters have been allowed to circulate in the pipes for some time, the amounts of different coloured waters arriving at such unlabelled wheels will provide the basis for a probabilistic distribution of assignments over the phenotypical classes for the corresponding uncharacterized genes. It is important to notice that the whole process can naturally take into account genes having multiple phenotypes, as salts of different colours can be poured into the same wheel.

which would be learned[1] from the training set using, for example, Support Vector Machines. Thus, for each binary relation there would be a different graph and the diffusion process would be carried out separately. The result of each diffusion process, corresponding to the amount of different phenotypic labels, will constitute the feature for that binary relation. The likelihood models for the binary relations will be approximated by the frequencies of these features in the training set. The prior probabilities of phenotypic assignment, $P(F_i)$, will also be approximated by the relative numerosity of the different phenotypic classes in the training set. Thus having obtained likelihood models for both unary and binary relations and estimates for the priors, we can obtain a phenotypical assignment by computing the numerator of the above equation (notice that the denominator is independent on the phenotypical class).

Finally we note that together with the overall prediction made by the integrated system, the biologists will also have access to the separate predictions coming from the different experimental datasets. In other words, the user would have the possibility of knowing what data supports the phenotypical assignment, e.g. evidence coming from co-expression data or from phenologs. This type of "explanation" can be very important for the biologists when reasoning about the prediction given by the system. Also, the Bayesian model outlined here is not the only possible way for integrating the information coming from the different types of data. Data from unary relations can be included directly, while for each binary relation we would go through the additional step of the diffusion process. However, once the diffusion process has generated a feature for a binary relation, then all the features can be collected into a vector and a unique probability distribution of phenotypical assignments can be obtained as a non-linear mapping of this vector. Such non-linear mapping would also be learned from a well characterized training set. Here we have outlined a Bayesian model as a possible method for learning this non-linear mapping, but various other machine learning techniques will be tested in order to choose the best solution.

The Diffusion of Experimental Information for Phenotypical Assignments

We have already formalized several different approaches that can be used to diffuse the phenotypic label information over the graphs. Here we describe three promising methods that will probably be considered during the project:

● The first approach consists in simply diffusing the phenotypical labels by simulating Markov random walks on the graph. Given a graph, we can derive the Markov transition matrix that controls the Markov diffusion process, and used it to diffuse the normalized vectors of known phenotypic assignments over the graph. Using similar approaches, Paccanaro has recently obtained excellent results clustering protein sequences (*41*).

● Another approach consists in projecting the nodes of the graph onto points in a (low dimensional) space in such a way that the distance between any two points is related to how well connected the two nodes are in the original graph. In other words, we project the nodes in such a way that for any two nodes, the higher the number of short paths existing between them in the original graph, the smaller their distance in the projected space (here the length of a path in a graph is defined as the sum of the values that label the edges along the path). Once the genes have been projected into this space, we need to discriminate between the distinct phenotypical classes. This could be done by learning an appropriate discriminative function using some training data; or by learning a separate probabilistic model for the points in each phenotypical category.

This type of projection, sometimes called *Diffusion Maps*, has been recently successfully applied to solve problems from Computer Vision: lip-reading and image-sequence alignment (*42*). We have used these ideas with very good results for predicting protein-protein

[1] This technique for building the graph is similar to the method that we have already successfully applied to obtain a unique protein-protein interaction network from several independent protein-protein interaction datasets obtained using different experimental techniques in Yeast (*33*).

interactions using the topological properties of networks of interactions observed experimentally (*43*).

● Finally a third approach is to map the problem of phenotypical assignment onto that of learning a particular classification on a Riemannian manifold. This approach has been shown to be very successful in a variety of classification problems, in the context of semi-supervised learning, by Belkin et al (*44*). The authors modelled the manifold where the data lies as a weighted graph G. Next, they showed that any function on G can be decomposed as a weighted sum of eigenfunctions of the graph Laplacian L, and they learned such coefficients from the training data. For the problem of phenotypical assignment, data from binary relations are already in the form of graphs, and therefore we need to learn the values for the weights for the eigenfunctions of the graph Laplacian. This can be seen as another way to diffuse information, as the Laplacian matrix is related to the Markov random walk (*41*).

3.2.2 Algorithm Implementation

An important effort in this project will be devoted to the design and the implementation of software tools for phenotype prediction which will incorporate the algorithms developed. These tools will be made freely available to the scientific community. There will be two versions: a suite of stand-alone applications and a web-based tool.

Stand-alone Applications. The suite of stand-alone applications will enable the biologists to easily apply the algorithms through a user-friendly interface. These will be available for various operating system. Using these tools, the biologists will provide a list of genes as well as sets of large scale experimental data for a certain organism. Using the data provided, the system will compute a predicted phenotype for each gene in the list. Importantly, some of these tools will also be integrated into Cytoscape (http://www.cytoscape.org), as Java plug-ins. This will allow the user to visualize relevant graphs. Such visualization will make the diffusion process transparent to the user, providing an explanation and a better understanding of predictions.

Web-based Tool. A web-based tool will also be created. Using this tool the biologist will access a web interface where one can upload a list of genes. The system will then report the phenotypes predicted for those genes, together with the evidence supporting the prediction.

We expect that these software tools will have an important impact in the field and that they will become a very useful resource for the scientific community.

3.2.3 Method Validation

In this project we aim to study the following organisms XXX. The phenotype of each organism will be defined by its respective phenotype ontology XXX.

As a proof of principle, we shall validate our algorithm taking advantage of the vast amount of data available for *S. cerevisiae*. Various solutions for diffusing information over graphs and integrating it with information from unary relations will be investigated. Several prototypes will be implemented using MATLAB, a numerical computing environment and high level programming language. These prototypes will be trained, i.e. some of their parameters will be fine tuned using a machine learning procedure as described above. The performance of the algorithms will then be evaluated "*in silico*" by means of test sets (by "cross-validation").

Next we shall then assess the performance of our algorithms when they are applied to different organisms such as *C. elegans*, *D. melanogaster*, *A. thaliana*, and *H. sapiens*.

3.3 AIM 3: Optimizing Phenotypic Function Prediction Using Logic Circuit Models on Biological Networks

We aim to improve the phenotypic function predictions from AIM 2 leveraging on the objective classification of genes based on regulatory logic gate preferences resulted from

AIM 1. For this we need to define two entities: (i) phenotypic distance and (ii) coherent logic gene modules.

3.3.1 Computing the Phenotypic Distance

[[AP- to add]]

3.3.2 Gene Logic Modules Classification

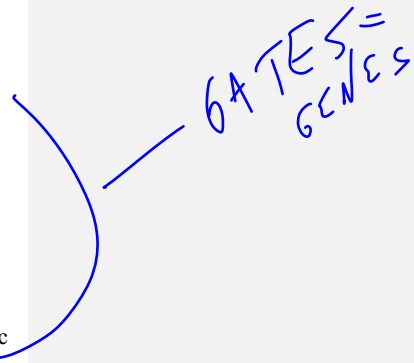
Using Lorergic we are able to characterize all regulatory network triplets (formed of two regulators and one target gene) using one of the sixteen logic gates. As a result we are able to classify different genes based on their logic gate preferences. We distinguish sixteen gene and regulators clusters based on the number of consistent logic gates that define the particular gene independent of regulators. The genes can also be divided based on the various types of logic gates that define the regulatory triplet. We define a logic gene module as a group of genes that share the same principal regulatory logic. The genes that cannot be assigned any consistent regulatory logic are grouped into a non-coherent logic module.

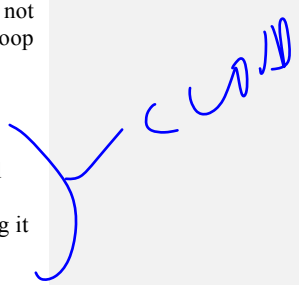
3.3.3 Phenotypic Function Prediction Optimization

In order to optimize the phenotype function prediction we start with the assumption that all genes sharing the same logic module will have similar phenotypes. The optimization workflow can be summarized in the following steps:

- **Step 1:** Assign all genes into logic modules as defined above (see **3.3.2**).
- **Step 2:** Assign all genes a phenotype function using the network mining algorithm as described in **AIM 2** and subdivide the logic modules into phenotypic.
- **Step 3:** Using logic modules as gold standard subdivide them into phenotypic modules grouping together genes with similar phenotypes.
- **Step 4:** Calculate the phenotypic distance between all the phenotype groups in the logic modules.
- **Step 5:** Define a maximum phenotypic distance threshold e.g. genes in the same logic module should not have phenotypes with distances larger than the selected threshold.
- **Step 6:** Extract all genes from all logic modules that do not pass the phenotypic distance threshold and re-cluster them based on their shared phenotypes.
- **Step 7:** Using phenotypic modules as gold standard divide these genes into submodules maximizing the identity between their logic gate preferences.
- **Step 8:** Extract all the genes that do not share logic gate preferences from each phenotypic module and re cluster them based on their preferred logic.
- **Step 9:** Repeat **Steps 3-8** until all the genes have been assigned to coherent logic and phenotypic modules.

This workflow leverages on the fact that each genes can be characterized by a variety of regulatory logic gates as well as numerous phenotypes. In majority of cases a dominant logic or phenotypic function can be defined. However, sometimes, the available data is not adequate enough to accurately describe the gene function, thus using a circular feedback loop approach we are able to predict the potential gene function.





4. Tools and Sustainability

The sustainability of the developed resources is very much contingent on the hardware and servers on which they are stored and run. The existing infrastructure at Yale and Royal Holloway has served investigators well, but we aim to improve the current setup by making it reliable and robust for supporting all the proposed tools, as well as more accessible to the scientific community.

Here, we summarize the various distinct components the function predictions tools, along with the means by which we intend to disseminate each:

- **Source code**, as used in constructing the various software components will be made available from open access repositories, such as sourceforge, github, and/or google code.
- **Web-services** (Loregic, phenotype prediction tools): each of the different servers would be encapsulated and made available as a virtual machines (see below for a description and the advantages of virtual machines), which may be downloaded from our servers, and then stored locally by the user.
- **Databases**: the regulatory network datasets used in the pilot and algorithm development studies will be bundled up into a single virtual machine, and made available through open access repositories.

5. Broader Impacts

5.1 Integration of Research into Education

We propose to integrate the above described research activities into graduate and undergraduate education.

Mark Gerstein is the Co-Director of the Computational Biology and Bioinformatics (CBB) PhD program (cbb.yale.edu) at Yale University, and he has been designing and teaching graduate courses in bioinformatics, genomics, and data mining for almost 20 years. These activities could easily be translated into class projects, which may help recruit undergraduates into Yale labs. In addition, we focus on students of underrepresented groups through a Yale program called "Science, Technology and Research Scholars" or STARS (science.yalecollege.yale.edu/stars-home), which includes Computer Science, Bioinformatics, and Genomics components.

All the tools developed for gene function prediction will be integrated into Computational Biology and Bioinformatics 752 (*Bioinformatics: Practical Application of Simulation and Data Mining*), a course directed by Dr Gerstein, and taught to undergraduates and graduate students. The course is an introduction to the computational approaches used for addressing questions in genomics and structural biology. The function component of the course can be substantially improved by introducing the students to innovative tools to predict gene function using a variety of data. This resource represents the integration of many facets of bioinformatics, including functional data, biological network analysis, programming, as well as sets of algorithms applied to address questions about gene function discovery. It will also be integrated into final year projects, and as part of these projects, students will develop online libraries for gene function.

5.2 Workshops and Webinars: From Our Computers to Everyone's

As a "tool just as useful the consumer's ability to effectively use it" we plan to reach out to the scientific community and popularize our newly developed methods using reach media interactions such as webinars and hands-on workshops. Also we aim to present the function prediction methods and network analysis algorithm at scientific conferences as well as "Open Day" events.

6. Project Management Plan

The research will be conducted by graduate students and early career personnel under the supervision of Dr Mark Gerstein (US NSF PI) at Yale University, and Dr Alberto Paccanaro at Royal Holloway University of London (UK BBSRC PI).

In leading this collaborative project, we will draw on considerable experience we have had with other integrative collabroative projects. In particular, Dr Gerstein has been an integral part of the ENCODE Project as well as the modENCODE Project since its inception. Within these he has had a number of leadership roles, as he has co-directed the Networks/Elements Group. He has co-led high profile papers focusing on networks and was the leader of the numerous collaborative papers.

This project will integrate the biological networks expertise of Dr Gerstein with the of software development expertise of Dr Paccanaro and thus will bring a fresh new perspective to protein function prediction. Dr Gerstein and Dr Paccanaro have been collaborating for over ten years on many network-based approaches for problems in biology. To some degree the collaboration between the two labs will be cemented through knowledge exchange and work visits. As such Dr Sisu (Yale) will have a visiting scientist appointment in Dr Paccanaro's lab and will work closely with his team to integrate the network analysis tool with phenotype predictions. Also Dr Paccanaro will visit the Gerstein Lab and contribute invited lectures to the computational biology and bioinformatics course led by Dr Gerstein.

Dr Gerstein will be responsible for the coordination, designing and development of tools associated with **AIM 1** created by Dr Sisu and Dr Wang at Yale. Dr Paccanaro will be involved in the design and development of phenotype prediction tools associated with **AIM 2**. While these two aims are lead mostly by each lab independently, both groups will collaborate towards their completion. As such, Paccanaro group will help with model development and implementation for **AIM 1**, while Gerstein group will help with assessment of data quality, standardization and biological interpretation of **AIM 2** results. The two groups will work closely together to facilitate the implementation of **AIM 3**.

The overall progress of the project is summarized in yearly milestones. As such, during the first and a half years of the project, the Gerstein lab work will be devoted to the development of logic circuit models for network analysis (**AIM 1**). Dr Paccanaro will provide technical support for the correct implementation and optimization of the algorithm. The successful development of this method will be assessed by a pilot study on yeast regulatory network. In the same time, the Paccanaro's lab will focus on developing machine learning methods for phenotype function predictions (**AIM 2**). Similarly Dr Gerstein lab will provide scientific feedback and validation of the prediction results. The rest of the second year will be focused on developing of a robust and friendly interface for the network analysis and function prediction tools, their deployment on host websites and open access repositories. The final year will be used for the implementation **AIM 3**. This year will also be dedicated to publishing collaborative papers describing the newly developed tools as well as the scientific advances resulted from their use.

The two groups will also coordinate the analysis and writing of collaborative manuscripts. To achieve this, we plan to implement regular conference calls between the two groups, but also open them to the larger networks and protein function community.

We will also work closely with other investigators from UK and US to identify additional regulatory networks datasets for integrative analysis, and coordinate the sharing of information with the larger biological research community. On a regular basis, the project results will be disseminated to a broad audience (from senior researchers to middle and high school teachers) through conferences, public workshops and webinars.

CheckList:
Important Proposal Preparation Information: FastLane will check for required sections of the full proposal, in accordance with Grant Proposal Guide (GPG) instructions described in Chapter II.C.2. The GPG requires submission of:

- Project Summary;
- Project Description;
- References Cited;
- Biographical Sketch(es);
- Budget;
- Budget Justification;
- Current and Pending Support;
- Facilities, Equipment & Other Resources;
- Data Management Plan; and
- Postdoctoral Mentoring Plan, if applicable.

If a required section is missing, FastLane will not accept the proposal.