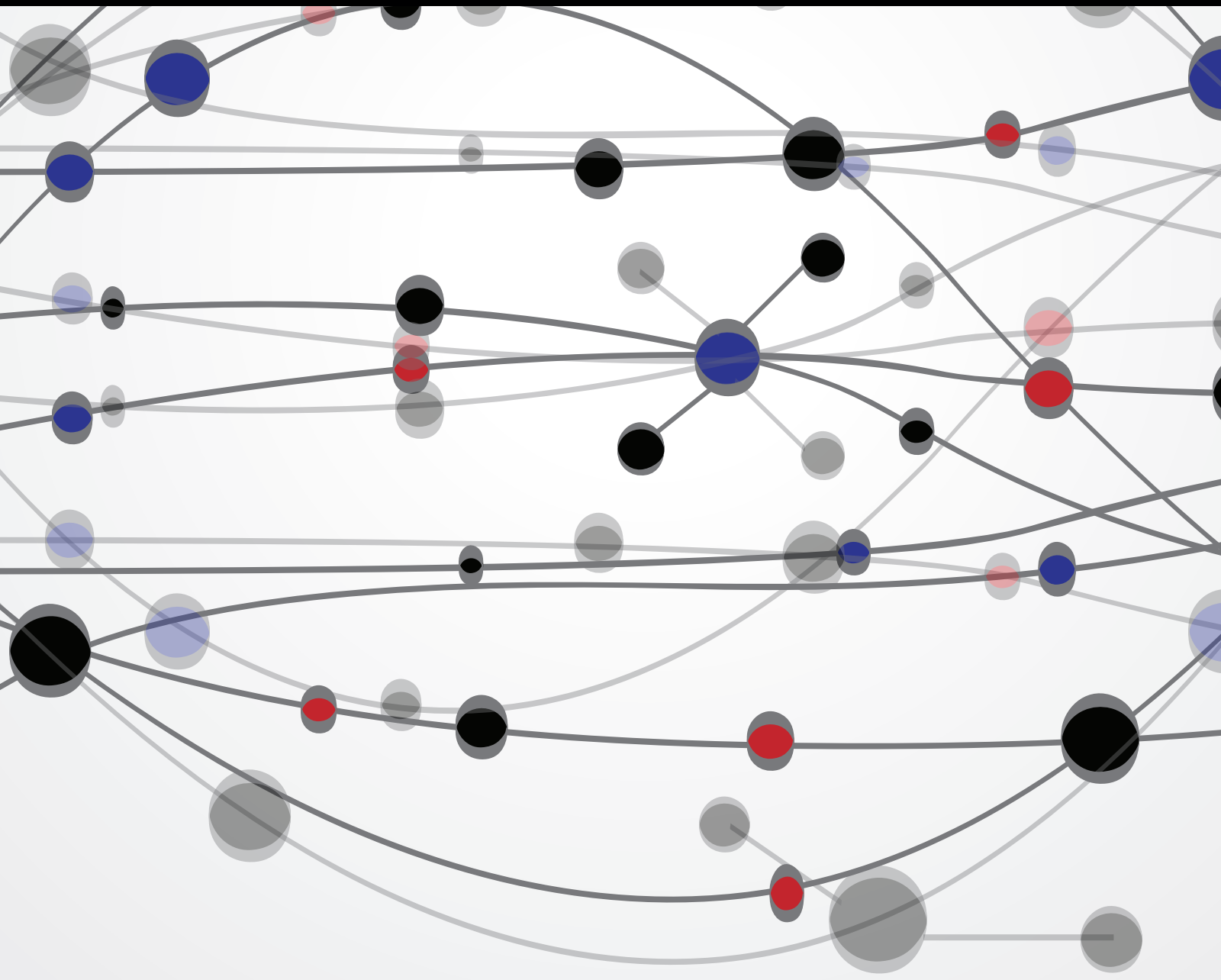
The Scientific World Journal

Computational Systems Biology

Guest Editors: Xing-Ming Zhao, Weidong Tian, Rui Jiang, and Jun Wan



Computational Systems Biology

The Scientific World Journal

Computational Systems Biology

Guest Editors: Xing-Ming Zhao, Weidong Tian, Rui Jiang, and Jun Wan

Contents

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 350358, 2 pages
http://dx.doi.org/10.1155/2013/350358

Editorial

Computational Systems Biology

Xing-Ming Zhao,[1] Weidong Tian,[2] Rui Jiang,[3] and Jun Wan[4]

[1] *School of Electronics and Information Engineering, Tongji University, Shanghai 201804, China*
[2] *Institute of Biostatistics, Fudan University, Shanghai 200433, China*
[3] *Department of Automation, Tsinghua University, Beijing 100084, China*
[4] *Wilmer Eye Institute, Johns Hopkins University School of Medicine, Baltimore, MD 21287, USA*

Correspondence should be addressed to Xing-Ming Zhao; zhaoxingming@gmail.com

Received 5 February 2013; Accepted 5 February 2013

The complex biological systems consist of distinct molecules that exert their functions by interacting with each other, which makes it a big challenge to understand how the cellular machinery works. Recently, the accumulation of a large amount of multiscale omics data, such as next-generation sequencing data and protein interaction data, provides opportunity to investigate the functions of molecules from a systematic perspective. On the other hand, the analysis of these huge datasets demands efficient and robust computational methods. In this special issue, we reported the recent progress made in developing new computational methodologies to analyze the genomics data, construct gene networks, and identify disease genes.

Understanding the Functions of Molecules in the Postgenomic Age. In recent years, the advance of next-generation sequencing (NGS) technology makes it more easier for researchers to access and analyze genetics data and has influential effects on the biomedical research community. However, compared with sequencing, computational analysis of the flooding sequencing data with appropriate tools is becoming a more important task when interpreting the data. In their review paper, M. P. Dolled-Filhart et al. described the pipeline for bioinformatics analysis of the NGS data, starting from alignment to variant calling as well as filtering and annotation. In each step, they discussed the tools or software that should be used as well as their advantages and caveats. This survey of the bioinformatics analysis of NGS data can help researchers to choose appropriate tools when dealing with the sequencing data.

Along with the sequencing technology, lines of evidence show that a lot of noncoding RNAs (ncRNAs) play important roles in various biological processes. Unlike the protein-coding genes that are well studied, the functions of most ncRNAs are not clear. Therefore, it is highly desirable to develop computational methods to predict the functions of the ncRNAs. H. Ma et al. conducted a survey about the computational approaches developed to predict and annotate the long noncoding RNAs (lncRNAs), which can help researchers to learn the progress in this filed and future directions in which bioinformaticists should work while annotating lncRNAs.

While annotating the functions of molecules, standard and controlled vocabularies are required. Hence, the ontologies that are represented as abstract description systems of knowledge are becoming more and more popular recently. At the same time, it is becoming a difficult task to calculate the semantic similarity between ontology terms quantitatively. M. Gan et al. introduced popular methods in quantitating the semantic similarity between ontology terms and their software implementations. Furthermore, they classified these methods into distinct categories and discussed their advantages and shortcomings, which can help researchers to select appropriate tools and methods when working on ontologies.

Gene expression profiles can describe the molecular mechanisms that underlie certain phenotypes. However,

while analyzing the gene expression data, it is inappropriate to treat genes independently considering genes interact with each other within the cell. O. Frings et al. proposed a network-based approach to analyze the gene expression data and applied it to investigate the development of sex-specific chicken gonad and brain tissues. By combining the chicken network and the gene expression data, they identified some sex-biased characteristics, for example, same sex-biased genes tend to be tightly connected in the network, and provided new insights into the molecular underpinnings of sex-biased genes.

Construction and Analysis of Gene Networks. Construction of gene regulatory networks (GRNs) is a crucial step in systems biology, where gene expression data is widely explored to infer the GRNs. However, the high dimensionality and notorious noise of the gene expression data makes it a nontrivial task to infer the GRNs. N. You et al. presented a new Laplace error penalty (LEP) model to calculate the partial correlation coefficients between genes and construct the GRNs. Compared with the popular least absolute shrinkage and selection operator (LASSO) and smoothly clipped absolute deviation (SCAD) approaches, the LEP method reached the highest precision. Except for gene expression data, integration of different data sources may improve the accuracy of inferred GRNs. H. Chen et al. surveyed the strategies to integrate distinct data sources and their effectiveness and recommended how to choose an appropriate strategy while integrating distinct data sources. N. Nakajima et al. proposed a novel network completion approach, DPLSQ, to infer gene networks. Benchmarking on artificial datasets, their proposed DPLSQ outperforms popular ARACNE and GeneNet with the highest accuracy. By investigating a 2-gene network, A. V. Spirov et al. found that gene cooption can affect the robustness of GRNs, and the findings provide new insights into the evolvability and robustness of GRNs.

Network modules are found to be functional blocks of gene networks, the identification of which is becoming a hot research topic. By taking the hierarchical modular structure into account, S. Zhang presented a new stochastic block model to detect the hierarchical modules. Applied to the real yeast gene coexpression network, the proposed method can efficiently detect the hierarchical modular structures that are consistent with biological functions. Recently, it is found that a particular type of ncRNAs, microRNAs, plays important roles in gene regulation by working together with transcription factors. W. Mu et al. proposed a new local genetic algorithm to predict condition-specific regulatory modules that consist of microRNAs, transcription factors, and their commonly regulated genes, and these modules provide useful insights into the regulatory mechanisms underlying gene expression.

Computational Approaches to Hunting Disease-Associated Genes. The identification of genetic variants that are responsible for human diseases is critical for understanding the development of diseases and designing new effective drugs. Thanks to the genome-wide association studies (GWASs), some genetic variants that drive diseases have been identified, among which single nucleotide polymorphisms (SNPs) and nonsynonymous single nucleotide polymorphisms (nsSNPs) are receiving more and more attention. In this issue, J. Wu and R. Jiang reviewed the databases that collect nsSNPs and summarized popular computational methods that identify deleterious nsSNPs. In addition, they introduced machine learning models that are useful in predicting deleterious nsSNPs. Beyond SNP-based association analysis, gene-based association analysis is receiving increasing attention. X. Guo et al. comprehensively compared these two approaches on the data from the study of addiction and found that these two approaches complement with each other and can get better results when used together.

The differentially expressed genes identified from microarray data are generally regarded as candidate disease genes. However, the number of differentially expressed genes may reach hundreds or even thousands, thereby making it difficult to identify the potential disease genes. In this issue, L. Li et al. proposed a new hybrid approach to predict disease genes based on estimation of distribution algorithm and support vector machine. Benchmarking on B-cell lymphoma and colon cancer datasets, their method outperforms two other popular approaches and identify some new candidate genes for future validation.

Acknowledgments

We would like to thank all reviewers for their invaluable contributions to the peer review process which have made this special issue possible.

Xing-Ming Zhao
Weidong Tian
Rui Jiang
Jun Wan

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 793091, 11 pages
http://dx.doi.org/10.1155/2013/793091

Review Article

From Ontology to Semantic Similarity: Calculation of Ontology-Based Semantic Similarity

Mingxin Gan,[1] Xue Dou,[1] and Rui Jiang[2]

[1] *Dongling School of Economics and Management, University of Science and Technology Beijing, Beijing 100083, China*
[2] *Department of Automation, Tsinghua University, Beijing 100084, China*

Correspondence should be addressed to Mingxin Gan; ganmx@ustb.edu.cn and Rui Jiang; ruijiang@tsinghua.edu.cn

Received 27 October 2012; Accepted 16 January 2013

Academic Editors: Y. Cai, S. Mohan, C. Proctor, K. Spiegel, and J. Wang

Advances in high-throughput experimental techniques in the past decade have enabled the explosive increase of omics data, while effective organization, interpretation, and exchange of these data require standard and controlled vocabularies in the domain of biological and biomedical studies. Ontologies, as abstract description systems for domain-specific knowledge composition, hence receive more and more attention in computational biology and bioinformatics. Particularly, many applications relying on domain ontologies require quantitative measures of relationships between terms in the ontologies, making it indispensable to develop computational methods for the derivation of ontology-based semantic similarity between terms. Nevertheless, with a variety of methods available, how to choose a suitable method for a specific application becomes a problem. With this understanding, we review a majority of existing methods that rely on ontologies to calculate semantic similarity between terms. We classify existing methods into five categories: methods based on semantic distance, methods based on information content, methods based on properties of terms, methods based on ontology hierarchy, and hybrid methods. We summarize characteristics of each category, with emphasis on basic notions, advantages and disadvantages of these methods. Further, we extend our review to software tools implementing these methods and applications using these methods.

1. Introduction

Recent technical innovation in high-throughput experiments has been successfully bringing about a revolution in modern biological and biomedical studies. With microarrays, expression levels of thousands of genes can be simultaneously measured [1]. With yeast two-hybrid assays, pairwise interactions between thousands of proteins can be systematically detected [2, 3]. With tandem mass spectrometry, a large number of proteins can be sequenced and characterized rapidly [4]. Indeed, high-throughput experimental techniques have enabled the collection of a vast volume of omics data, while how to organize, interpret, and use these data has now become a serious issue [5]. Each type of data explains the biological system under investigation from a specific point of view. In order to get full understanding of the system, however, one needs to integrate multiple types of data—typically coming from different laboratories and obtained using different experimental techniques. Consequently, the data should be organized in such a way that is standard across different techniques and interpretable across different laboratories. In other words, information and knowledge included in the data should be described using a set of controlled vocabulary that is standardized. Fortunately, an ontology provides us with such a standard means of organizing information [5].

An ontology is an abstract description system for knowledge composition in a certain domain [6]. By organizing concepts (terms) in a domain in a hierarchical way and describing relationships between terms using a small number of relational descriptors, an ontology supplies a standardized vocabulary for representing entities in the domain [7]. Particularly, in biological and biomedical domains, there have been quite a few ontologies available [5]. For example, the gene ontology (GO), including three separate domains (biological process, molecular function, and cellular component), has been widely used as a standard vocabulary for annotating functions of genes and their products across

different species [8]. The human phenotype ontology (HPO) has been explored to facilitate the description of human disease phenotypes with a set of standard terms [9]. The plant ontology (PO) has been utilized to describe plant structures and growth stages [10]. Particularly, in order to achieve the goal of providing standard annotations of multiple heterogeneous data sources using common controlled vocabularies, The open biological and biomedical ontologies (OBO) Foundry has been proposed to coordinate the development of ontologies in different biological and biomedical domains [5]. Up to October 20, 2012, there have been 8 mature ontologies and 107 candidate ontologies included in the OBO Foundry, covering 25 domains, including anatomy, health, phenotype, environment, and many others [5].

Many applications using domain ontologies need to quantify the relationship between two terms [11, 12]. A suitable measure of such relationship is the semantic similarity between the terms, given the underlying domain ontology [13]. Considering the hierarchical structure of an ontology [6], the semantic similarity between two terms is in general defined as a function of distance between the terms in a graph corresponding to the hierarchical structure of the underlying ontology. However, the concrete form of the function may be refined with further knowledge about the ontology or even entities that are already annotated by using the ontology, yielding a wide variety of approaches for calculating semantic similarities of terms [14–19]. More specifically, we classify these approaches into five categories: (1) methods based on semantic distance between terms, (2) methods based on information contents of terms, (3) methods based on features of terms, (4) methods based on the hierarchical structure of an ontology, and (5) hybrid methods. Since each category of methods has its own traits, it is indispensable to know which method is suitable for the application of interest. Motivated by this consideration, we summarize characteristics of each category of methods in this paper, provide a brief review of available software implementation of these methods, and introduce typical biological and biomedical applications that rely on ontologies.

Table 1: Domains in the OBO Foundry.

Index	Domain	Number
1	Adverse events, health	1
2	Algorithms	1
3	Anatomy	39 (3)
4	Anatomy and development	1
5	Anatomy, immunology	1
6	Behavior	1
7	Biochemistry	3 (1)
8	Biological function	1 (1)
9	Biological process	3 (1)
10	Biological sequence	1
11	Environment	3
12	Experiments	8
13	Genomic	1
14	Health	12
15	Information	1
16	Lipids	1
17	Medicine	2
18	Molecular structure	1
19	Neuroscience	3
20	Phenotype	8 (1)
21	Proteins	6 (1)
22	Provenance	1
23	Resources	1
24	Taxonomy	4
25	Other	11
	Total	115 (8)

2. Biological and Biomedical Ontologies

The rapid development of high-throughput biological experimental techniques has enabled the explosive increase of a wide variety of omics data, while the integrated use of these data appeals for the standard annotation of multiple heterogeneous data sources using common controlled vocabularies. To achieve this goal and coordinate the development of ontologies in different domains, the open biological and biomedical ontologies (OBO) Foundry has been proposed [5]. The OBO Foundry is a collaborative experiment that aims at creating controlled vocabularies for shared use across different biological and medical domains. Participants of the OBO Foundry have agreed in advance on the adoption of a set of principles that specify the best practices for the development of ontologies, for the purpose of developing a set of interoperable humanly validated reference ontologies for all major domains of biomedical research. As shown in Table 1, up to October 20, 2012, there have been 8 mature ontologies and 107 candidate ontologies included in the OBO Foundry. These ontologies can further be classified into 25 domains, including anatomy, health, phenotype, and environment.

The 8 mature ontologies are listed in Table 2. Biological process, cellular component, and molecular function belong to the gene ontology (GO), which aims at standardizing representation of characteristics of genes and gene products across species via providing a controlled vocabulary of terms for describing annotations of gene products [20]. Specifically, biological process describes operations or sets of molecular events with a defined beginning and end. Molecular function describes elemental activities of gene products at the molecular level. The cellular component describes parts of a cell or its extracellular environment. The chemical entities of biological interest (ChEBI) provide a controlled vocabulary mainly for describing small chemical compounds, which are either products of nature or synthetic products used to intervene in the processes of living organisms [21]. The phenotypic quality (PATO) can be used in conjunction with phenotype annotations provided by other ontologies to describe qualities (such as red, ectopic, high temperature, fused, small, and edematous) for phenotypes [5, 22]. The protein ontology (PRO) is used to describe protein-related entities such as specific modified

Table 2: Mature ontologies in OBO.

Title	Domain	Prefix
Biological process	Biological process	GO
Cellular component	Anatomy	GO
Chemical entities of biological interest	Biochemistry	CHEBI
Molecular function	Biological function	GO
Phenotypic quality	Phenotype	PATO
Protein ontology	Proteins	PR
Xenopus anatomy and development	Anatomy	XAO
Zebrafish anatomy and development	Anatomy	ZFA

forms, orthologous isoforms, and protein complexes [23]. This ontology is separated into three domains: proteins based on evolutionary relatedness, protein forms produced from a given gene locus, and protein-containing complexes. The Xenopus anatomy and development (XAO) is designed to describe annotations of the model organism African clawed frog (*Xenopus laevis*) [24]. In this ontology, the lineage of tissues and the timing of their development are organized in a graphical view, hence facilitating the annotation of gene expression patterns, mutants, and morphant phenotypes of Xenopus. Similarly, the Zebrafish anatomy and development (XAO) provides a controlled vocabulary for annotating the anatomy of the model organism Zebrafish (*Danio rerio*) [25].

Many of the candidate ontologies have also been widely used in a variety of research areas. For example, in medical research, the human phenotype ontology (HPO) provides a means of describing phenotypic abnormalities encountered in human diseases [9]. This ontology is developed based on the Online Mendelian Inheritance in Man (OMIM) database [26] and medical literature, currently containing more than 10 thousand terms and over 50 thousand annotations to human-inherited diseases. In environmental science, the environment ontology (EnvO) is designed to support annotations of organisms or biological samples with environment descriptions [5].

3. Derivation of Semantic Similarity between Terms in an Ontology

3.1. Hierarchical Structure of an Ontology. Typically, an ontology is represented as a directed acyclic graph (DAG), in which nodes correspond to terms and edges represent relationships between the terms. In some ontologies, there is only one relationship between nodes, while in more general case, there exist more than one relationship between nodes. For example, the gene ontology defines 5 relationships between nodes: is_a, part_of, regulates, negatively_regulates, and positively_regulates [8], while the OBO relational ontology defines 13 relationships between nodes: is_a, part_of, integral_part_of, proper_part_of, located_in, contained_in, adjacent_to, transformation_of, derives_from, preceded_by, has_participant, has_agent, and instance_of [5].

In the DAG corresponding to an ontology, there is a node specified as the root. For every node in the ontology, there exists at least one path pointing from the root to the node. Every node in such a path is called an ancestor of the node, and the ancestor that immediately precedes the node in the path is called the parent of the node. Inversely, if a node is a parent of another node, the node is called a child of the parent. There might be more than one path from the root to a node. Consequently, a node may have several parent nodes, and vice versa. Given two nodes in an ontology, they must share a set of common ancestor nodes, and the one represents the most concrete concept is typically referred to as the lowest common ancestor of the two nodes. Discarding the direction of the edges in an ontology, there exists at least one path between every pair of two nodes.

3.2. Methods Based on Semantic Distance between Terms. Given a pair of two terms, c_1 and c_2, a well-known method with intuitive explicitness for assessing their similarity is to calculate the distance between the nodes corresponding to these terms in an ontology hierarchy; the shorter the distance, the higher the similarity. In the case that multiple paths between the nodes exist, the shortest or the average distance of all paths may be used. This approach is commonly referred to as the semantic distance method, since it typically yields a measure of the distance between two terms. The distance can then be easily converted into a similarity measure. Four main factors are normally considered in distance-based methods as follows

(1) density in the ontology graph: the higher the density, the nearer the distance between nodes;

(2) depths of nodes: the deeper the nodes located in, the more obvious the difference between the nodes;

(3) types of links: the normal type is is-a relation, and other relations such as part-of and substance-of are associated with the weight for edges;

(4) weights of links: edges connecting a certain node with all its child nodes can vary among different semantic weights.

In the last two decades, many efforts have been devoted to building various models to measure such distance in calculating similarities. Some representative algorithms include shortest path [27], connection weight [28], and Wu and Palmer [29].

Rada et al. proposed the shortest path method to calculate semantic similarity based on the ontology hierarchy, suggesting that the shortest path between two nodes was the simplest approach for measuring distance between two terms [27]. In mathematics, the formula for the distance between two nodes by the shortest path was denoted by $\mathrm{Sim}(c_1, c_2) = 2\mathrm{MAX} - L$, where c_1 and c_2 were the compared nodes, MAX the maximum path on the hierarchy, and L the shortest path. The main advantage of this method was its low complexity in calculation. Rada et al. hypothesized that when only the *is-a* relationship existed in a semantic network, semantic relatedness and semantic distance were equivalent. However,

this method was short of consideration for different kinds of edges as well as the semantic relatedness representing these edges.

Sussna proposed an edge weight determination scheme, which considered the first three factors: the density of the graph, depths of nodes, and types of connections [28]. In their method, the distance or weight of the edge between adjacent nodes c_1 and c_2 was defined as

$$wt\left(c_1, c_2\right) = \frac{wt\left(c_1 \longrightarrow_r c_2\right) + wt\left(c_2 \longrightarrow_{r'} c_1\right)}{2d},$$
$$\text{given } wt\left(x \longrightarrow_r y\right) = \max_r - \frac{\max_r - \min_r}{n_r\left(x\right)}, \quad (1)$$

where $\rightarrow_r$ was a relation of type r, $\rightarrow_{r'}$ its inverse, d the depth of the deeper node, $\max_r$ and $\min_r$ the maximum and minimum weights for a relation of type r, respectively, and $n_r(x)$ the number of relations of type r leaving node x. This method exhibited an improvement in reducing the ambiguousness of multiple sense words by discovering the combination of senses from a set of common terms that minimizes total pairwise distance between senses. However, depth factor scaling and restricting the type of a link to a strictly hierarchical relation apparently impaired the performance of the method.

Alternatively, the common path technique calculated the similarity directly by the length of the path from the lowest common ancestor of the two terms to the root node [29]. In detail, Wu and Palmer [29] took into account the position relation of c_1, c_2 to their nearest common ancestor c to calculate similarity. Here, c was the node with fewest *is-a* relationship as their ancestor node which appeared at the lowest position on the ontology hierarchy. In mathematics, the formula calculating similarity between c_1 and c_2 was denoted as

$$\text{Sim}\left(c_1, c_2\right) = \frac{2H}{D_1 + D_2 + 2H}, \quad (2)$$

where D_1 and D_2 were, respectively, the shortest paths from c_1 and c_2 to c, and H the shortest path from c to the root. However, the calculation of similarity only cumulated shortest paths together with the consideration that all the edges were of the same weight. Hence, it might also potentially lose information of semantics represented by various types of edges existing in the ontology hierarchy.

However, in practical application, terms at the same depth do not necessarily have the same specificity, and edges at the same level do not necessarily represent the same semantic distance, and thus the issues caused by the aforementioned assumptions are not solved by those strategies [13]. Moreover, although distance is used to identify the semantic neighborhood of entity classes within their own ontologies, the similarity measure between neighborhoods is not defined based on such a distance measure.

3.3. Methods Based on Information Contents of Terms. A method based on information content typically determines the semantic similarity between two terms based on the information content (IC) of their lowest common ancestor (LCA) node. The information content (IC) gives a measure of how specific and informative a term is. The IC of a term c can be quantified as the negative log likelihood $\text{IC}(c) = -\log P(c)$, where $P(c)$ is the probability of occurrence of c in a specific corpus (such as the UniProt Knowledgebase). Alternatively, the IC can be also computed from the number of children a term has in the ontology hierarchical structure [30], although this approach is less commonly used. On the ontology hierarchy, the occurrence probability of a node decreases when the layer of the node goes deeper, and hence the IC of the node increases. Therefore, the lower a node in the hierarchy, the greater its IC. There have been quite a few methods belonging to this category. For instance, Resnik put forward a first method that is based on information content and tested the method on WordNet [18]. Lin proposed a theoretic definition of semantic similarity using information content [15]. Jiang and Conrath improved the method of Resnik by introducing weights to edges [14]. Schlicker et al. proposed a method that is applicable to the gene ontology [31]. As mentioned by Wang et al. [32], methods based on information content may be inaccurate due to shallow annotations. Lee et al. also pointed out this drawback [33].

Resnik [18] used a taxonomy with multiple inheritance as the representational model and proposed a semantic similarity measure of terms based on the notion of information content. By analogy to information theory, this method defined the information content of a term as the negative algorithm of the probability of its occurrence and the similarity between two terms c_1 and c_2 as the maximal information content of all terms subsuming both c_1 and c_2, calculated by

$$\text{Sim}\left(c_1, c_2\right) = \max_{c \in S(c_1, c_2)} \left[-\log P\left(c\right)\right], \quad (3)$$

where $S(c_1, c_2)$ was the set of all the parents for both c_1 and c_2. Since the lowest common ancestor (LCA) had the maximum value of information content, recognizing the LCA of both c_1 and c_2 can be supported by this measure. The information content-based similarity measure was symmetric and transitive. Obvious advantages of this method were its simple calculation and easy formulation. However, in contrast to distance by Rada et al., the minimality axiom did not hold for Resnik's similarity measure. The similarity between a term and itself was the negative logarithm of its information content. Only the single term on top of the hierarchy reached the self-similarity of one. In addition, this method was only suitable for the ontology hierarchy with single relations; for example, all edges connecting terms represent only the same relationship, so it cannot be applied to the terms with either part-of relations or inferior relations.

Lin [15] proposed an alternative information theoretic approach. This method took into account not only the parent commonality of two query terms, but also the information content associated with the query terms. Three basic assumptions were normally given by Lin [15] in calculating the similarity between two terms as follows.

(1) The similarity between two terms was associated with their common properties: the more the common properties, the higher their similarity.

(2) The similarity between two terms was associated with their difference: the more the difference, the lower their similarity.

(3) The similarity between two terms reached the maximum value when they were totally the same.

Based on the above assumptions, given terms, c_i and c_j, their similarity was defined as

$$\mathrm{Sim}\left(c_i, c_j\right) = \frac{2\log P\left(c_0\right)}{\log P\left(c_i\right) + \log P\left(c_j\right)}, \tag{4}$$

where c_0 was the lowest common ancestor (LCA) of c_i and c_j, and $P(c_i)$ and $P(c_j)$ were the probabilities of occurrence. Not only the information content of LCA was considered in the calculation, but also their information content was taken into account in Lin's method. This measure could be seen as a normalized version of the Resniks method. Lin's values also increased in relation to the degree of similarity shown by two terms and decreased with their difference. However, the consideration of information content of two terms themselves caused a strong dependence on the high precision of the annotation information. Consequently, exact result can be generated only when mapping relationships between compared terms and other terms in the ontology hierarchy were precisely described, while the result would be near to 0 when annotations were abstract, yielding the problem of shallow semantic annotations. In fact, the difference between two terms with abstract annotations could be large, so it might be misleading to produce similarity values according to Lin's method.

Jiang and Conrath [14] proposed a combined approach that inherited the edge-based approach of the edge counting scheme, which was then enhanced by the node-based approach of the information content calculation. The factors of depths of nodes, the density around nodes, and the type of connections were taken into account in this measure. The simplified version of the measure was given as

$$\mathrm{Dist}\left(w_1, w_2\right) = \mathrm{IC}\left(c_1\right) + \mathrm{IC}\left(c_2\right) - 2 \times \mathrm{IC}\left(\mathrm{LCA}\left(c_1, c_2\right)\right). \tag{5}$$

However, being relative measures, both the method of Lin and that of Jiang and Conrath were proportional to the IC differences between the terms and their common ancestor, independently of the absolute IC of the ancestor. To overcome this limitation, Schlicker et al. [31] proposed the relevance similarity measure. This method was based on Lin's measure but used the probability of annotation of the most informative common ancestor (MICA) as a weighting factor to provide graph placement as follows:

$$\mathrm{Sim}\left(c_1, c_2\right) = \max_{c \in S(c_1, c_2)} \left(\frac{2 \times \log p\left(c\right)}{\log p\left(c_1\right) + \log p\left(c_2\right)} \times \left(1 - p\left(c\right)\right) \right). \tag{6}$$

All these measures overlooked the fact that a term can have several disjoint common ancestors (DCAs). To overcome this limitation, Couto et al. [34] proposed the GraSM method, in which the IC of the MICA was replaced by the average IC of all DCA. Bodenreider et al. [35] developed a node-based measure that also used annotation data but did not rely on information theory. Focusing on the gene ontology, their method represented each term as a vector of all gene products annotated with the term and measured similarity between two terms by calculating the scalar product of their vectors. Riensche et al. used coannotation data to map terms between different GO categories and calculated a weighting factor, which could then be applied to a standard node-based semantic similarity measure [36].

3.4. Methods Based on Features of Terms. In feature-matching methods, terms are represented as collections of features, and elementary set operations are applied to estimate semantic similarities between terms. A feature-matching model in general consists of three components: distinct features of term A to term B, distinct features of term B to term A, and common features of terms A and B.

Using set theory, Tversky [37] defined a similarity measure according to a matching process, which generated a similarity value based on not only common but also distinct features of terms. This approach was in agreement with an information-theoretic definition of similarity [15]. Unlike the above-mentioned models based on semantic distance [27–29], this feature-matching model was not forced to satisfy metric properties. A similarity measure based on the normalization of Tversky's model and the set-theory functions of intersection ($D_1 \cap D_2$) and difference (D_1/D_2) was given as

$$\mathrm{Sim}\left(c_1, c_2\right) = \frac{\left|D_1 + D_2\right|}{\left|D_1 \cap D_2\right| + \mu\left|D_1/D_2\right| + \left(\mu - 1\right)\left|D_2/D_1\right|}, \quad \text{for } 0 \le \mu \le 1, \tag{7}$$

where D_1 and D_2 corresponded to description sets of c_1 and c_2, $||$ the cardinality of a set, and μ a function that defines the relative importance of the noncommon features. The first term of a comparison (i.e., c_1) was referred to as the target, while the second term (i.e., c_2) was defined as the base. Particularly, intersections or subtractions of feature sets were based only on entire feature matches. This feature model allowed for representing ordinal and cardinal features, but the similarity measure did not account for their ordering.

In addition, the Matching-Distance Similarity Measure (MDSM) by Rodríguez et al. [38] and Rodríquez and Egenhofer [7, 39] was another feature model developed for similarity measurement of geospatial terms. This category of models was based on the ratio model that extends the original feature model by introducing different types of features and applying them to terms.

3.5. Methods Based on Hierarchical Structure of an Ontology. Typically, an ontology is represented as a directed acyclic graph (DAG), in which nodes correspond to terms, and edges represent relationships between the terms. A parent node may have several child nodes while a child node may have

several parent nodes. Some nodes have high density around them while some have low density in the hierarchy. A method based on the structure of an ontology typically uses a distance measure to quantify the similarity between two nodes in the corresponding DAG of the ontology and then uses this measure to assess the relatedness between the corresponding terms in the ontology.

There have been quite a few methods that belong to this category. For example, Rada et al. converted the shortest path length between two terms into their semantic similarity [27]. Wu and Palmer calculated the distance from the root to the lowest common ancestor (LCA) node of two terms as their semantic similarity [29]. Leacock and Chodorow calculated the number of nodes in the shortest path between two terms and then used the number with the maximum depth of an ontology to quantify the relatedness of the terms [40]. Al-Mubaid and Nguyen quantified the commonality of two terms as their similarity [41]. Wang et al. proposed to aggregate contributions of common ancestor terms to semantic values of two terms in the calculation of their semantic similarity [19]. Zhang et al. improved the method of Wang et al. and proposed the combined use of the shortest path length and the depth of the LCA node [42]. The strategies that these methods employed included lengths of shortest paths, depths of nodes, commonalities between terms, semantic contributions of ancestor terms, and many others. Although the use of these strategies has enabled the successful application of these methods to a variety of problems, the existence of a drawback in these methods is also obvious. It is common that a term in an ontology has more than one parent node in the corresponding DAG, and thus two terms may have two or more LCA nodes. However, none of the above methods take such a situation of multiple LCA nodes into consideration in their calculation of semantic similarity.

Wang et al. evaluated measures proposed by Jiang and Conrath, Lin, and Resnik and tested these measures against gene coexpression data using linear correlation [19]. They pointed out that the distance from a term to the closest common ancestor might fail in accurately representing the semantic difference between two GO terms, since two terms near to the root of the ontology and sharing the same parent should have larger semantic difference than those far away from the root and having the same parent. In addition, considering that a GO term may have multiple parent terms with different semantic relationships, they also suggested that measuring the semantic similarity between two GO terms based only on the number of common ancestor terms might fail in recognizing semantic contributions of the ancestor terms to the two specific terms. In addition, from human perspectives, an ancestor term far away from a descendant term in the GO graph should contribute less to the semantics of the descendant term, while an ancestor term closer to a descendant term in the GO graph should contribute more.

According to the above understanding, Wang et al. presented GO as directed acyclic graphs (DAGs) in which terms form nodes and two kinds of semantic relations is-a and part-of form edges. They further defined the contribution of a GO term t to the semantics of GO term A as the S-value of GO term t related to term A. Formally, a GO term A was defined as a graph $\mathrm{DAG}_A = (A, T_A, E_A)$, where T_A was the set of GO terms in DAG_A, including A and all of its ancestors in the GO graph, and E_A was the set of edges connecting GO terms in DAG_A. For any term t in $\mathrm{DAG}_A = (A, T_A, E_A)$, the S-value related to term A, $S_A(t)$ was then defined as

$$S_A(A) = 1,$$
$$S_A(t) = \max\left\{w_e \times S_A\left(t'\right) t' \in \text{children of } (t)\right\} \quad (t \neq A), \tag{8}$$

where w_e was the semantic contribution factor for edge $e \in E_A$ that links term t and its child term t'. Given $\mathrm{DAG}_A = (A, T_A, E_A)$ and $\mathrm{DAG}_B = (A, T_B, E_B)$, for terms A and B, respectively, the semantic similarity between these two terms, $S_{\mathrm{GO}}(A, B)$, was defined as

$$S_{\mathrm{GO}}(A, B) = \frac{\sum_{t \in T_A \cap T_B} \left(S_A(t) + S_B(t)\right)}{\mathrm{SV}(A) + \mathrm{SV}(B)}, \tag{9}$$

where $S_A(t)$ and $S_B(t)$ are S-values of term t related to terms A and B, respectively, and $\mathrm{SV}(A)$ and $\mathrm{SV}(B)$, defined as $\mathrm{SV}(A) = \sum_{t \in T_A} S_A(t)$ and $\mathrm{SV}(B) = \sum_{t \in T_B} S_B(t)$, were semantic values of terms A and B, respectively. Wang et al. further compared their measure against Resnik's method by clustering gene pairs according to their semantic similarity and showed that their measure produced more reasonable results. However, in Wang's method, the weights of the *is-a* and the *part-of* relations were empirically determined as 0.8 and 0.6, respectively, without theoretical analysis. Moreover, this method did not take into account the factor of the amount of nodes. In a subsequent study, Zhang et al. [42] pointed out that Wang's method overlooked the depth of the GO terms and proposed a measure to overcome this limitation.

Schickel-Zuber and Faltings [43] defined a similarity measure for hierarchical ontologies called Ontology-Structure-based Similarity (OSS). They pointed out that a quantitative measure of similarity should represent the ratio of numerical scores that may be assigned to each term, and thus the score of a term should be defined as a real-valued function normalized to the range of $[0, 1]$ and should satisfy three assumptions. First, similarity scores depended on features of the terms. Second, each feature contributed independently to a score. Third, unknown and disliked features made no contribution to a score. In detail, the OSS measure first inferred the score of the term b from a, $S(b \mid a)$, by assigning terms in the ontology an a-priori score (APS) and computing relationships between scores assigned to different terms. Then, this method computed how much had been transferred between the two terms, $T(a, b)$. Finally, this method transformed the score into a distance value $D(a, b)$. Mathematically, the *a-priori score* of a term c with n descendants was calculated as

$$\mathrm{APS}(c) = \frac{1}{n+2}, \tag{10}$$

implying that leaves of an ontology have an APS equal to 1/2, the mean of a uniform distribution in $[0, 1]$. Conversely,

the lowest value was found at the root. It also implied that the difference in score between terms decreased when one traveled up towards the root of the ontology, due to the increasing number of descendants. Given two terms x and z in an ontology and their lowest common ancestor y, the distance value was calculated as

$$D(x,z) = \frac{\log(1+2\beta(z,y)) - \log(\alpha(x,y))}{\max D}, \quad (11)$$

where $\alpha(x,y)$ was a coefficient calculated as $\alpha(x,y) = \mathrm{APS}(y)/\mathrm{APS}(x)$, $\beta(z,y)$ a coefficient estimated by $\beta(z,y) = \mathrm{APS}(z) - \mathrm{APS}(y)$, and $\max D$ the longest distance between any two terms in the ontology.

Al-Mubaid and Nguyen [41] proposed a measure with common specificity and local granularity features that were combined nonlinearly in the semantic similarity measure. Compared with other measures, this method produces the highest overall correlation with human judgments in two ontologies. In mathematics, the semantic similarity between two terms was calculated as:

$$\mathrm{Sem}(C_1,C_2) = \log\left((\mathrm{Path}-1)^{\alpha} \times (D - \mathrm{depth}((\mathrm{LCS}(C_1,C_2))))^{\beta} + k\right), \quad (12)$$

where $\alpha > 0$ and $\beta > 0$ were contribution factors of two features, Path the length of the shortest path between the two terms, D the maximum depth, LCS the closest common ancestor of the two terms, and k a constant. Compared with other measures, this measure produced the highest overall correlation results with human judgments in two ontologies.

3.6. Hybrid Methods. Hybrid methods usually consider several features such as attribute similarity, ontology hierarchy, information content, and the depth of the LCA node simultaneously. One of the representative methods was OSS in which a priori score was used to calculate the distance berween two terms, and then the distance was transformed into semantic similarity [43]. Another example was the method proposed by Yin and Sheng [44], which combined term similarity and description similarity.

4. Derivation of Semantic Similarity of Entities Annotated with an Ontology

With the semantic similarity scores between terms in an ontology calculated using either of the above methods, the derivation of semantic similarity of entities annotated with the ontology was typically conducted using either the average rule [15] or the mean-max rule [19].

Given two sets of terms T and S, the average rule calculated the semantic similarity between the two sets as the average of semantic similarity of the terms cross the sets as

$$\mathrm{Sim}(T,S) = \frac{1}{|T| \times |S|} \sum_{t \in T} \sum_{s \in S} \mathrm{Sim}(s,t). \quad (13)$$

Since an entity can be treated as a set of terms, the semantic similarity between two entities annotated with the ontology was defined as the semantic similarity between the two sets of annotations corresponding to the entities.

The mean-max rule defined the semantic similarity between a term t and a set of terms T in the ontology as the maximum similarity between the term and every term in the set as

$$\mathrm{Sim}(t,T) = \max_{t' \in T} \mathrm{Sim}\left(t,t'\right). \quad (14)$$

Then, the semantic similarity between two sets of terms T and S was calculated as

$$\mathrm{Sim}(S,T) = \frac{1}{|S| + |T|}\left(\sum_{s \in S} \mathrm{Sim}(s,T) + \sum_{t \in T} \mathrm{Sim}(t,S)\right). \quad (15)$$

Finally, the semantic similarity between two entities annotated with the ontology was calculated as the semantic similarity between the two sets of annotations corresponding to the entities.

5. Software for Deriving Semantic Similarity Profiles

With the above methods for calculating semantic similarity of terms in an ontology and that of entities annotated with an ontology available, a natural demand in research is the development of user-friendly software tools that implement these methods. So far, there have been quite a few such software tools available, with examples including GOSemSim [45], seGOsa [46], DOSim [47], and many others.

Yu et al. developed GOSemSim [45] for calculating semantic similarity between GO terms, sets of GO terms, gene products, and sets of gene products. This tool was developed as a package for the statistical computing environment *R* and released under the GNU General Public License (GPL) within the Bioconductor project [48]. Consequently, GOSemSim was easy to install and simple to use. However, GOSemSim heavily depended on a number of packages provided by Bioconductor. For example, package GO.db was used by GOSemSim to obtain GO terms and relationships; packages org.Hs.eg.db, org.Rn.eg.db, org.Mm.eg.db, org.Dm.eg.db, and org.Sc.sgd.db were required in order to obtain annotations of gene products for human, rat, mouse, fly, and yeast, respectively. Although such a design scheme greatly alleviated the requirement of understanding specific formats of these annotations, the frequent access of annotation databases was typically the bottleneck of large-scale calculation of semantic similarity profiles for thousands of gene products.

Zheng et al. proposed seGOsa [46], a user-friendly cross-platform system to support large-scale assessment of gene ontology- (GO-) driven similarity among gene products. Using information-theoretic approaches, the system exploited both topological features of the GO and statistical features of the model organism databases annotated to the GO to assess semantic similarity among gene products. Meanwhile, seGOsa offered two approaches to assessing the

similarity between gene products based on the aggregation of between-term similarities. This package has been successfully applied to assess gene expression correlation patterns and to support the integration of GO-driven similarity knowledge into data clustering algorithms. This package has also assessed relationships between GO-driven similarity and other functional properties, such as gene coregulation and protein-protein interactions in *Saccharomyces cerevisiae* and *Caenorhabditis elegans*. A database consisting of semantic similarity between gene products in both *Saccharomyces cerevisiae* and *Homo sapiens* has been successfully established using seGOsa and applied to the prediction of protein interaction networks.

Li et al. developed an R-based software package (DOSim) to compute the similarity between diseases and to measure the similarity between human genes in terms of diseases [47]. DOSim incorporated an enrichment analysis function based on the disease ontology (DO) and used this function to explore the disease feature of an independent gene set. A multilayered enrichment analysis using GO and KEGG [49] annotations that helped users to explore the biological meaning implied in a newly detected gene module was also included in the DOSim package. This package has been applied to calculate relationships between 128 cancer terms, and hierarchical clustering results of these cancers have shown modular characteristics. This package has also been used to analyse relationships of 361 obesity-associated genes, and results have shown the complex pathogenesis of obesity.

6. Applications of Semantic Similarity Profiles

Biological entities can be described using an ontology as a common schema as well as compared by means of semantic similarity to assess the degree of relatedness via the similarity in meaning of their annotations. In recent years, there has been a growing trend towards the adoption of ontologies to support comprehensive, large-scale functional genomics research. For example, it has been shown that incorporating knowledge represented in the gene ontology may facilitate large-scale predictive applications in functional genomics [7, 32, 50] and disease studies [12]. It has also been shown that phenotype ontologies benefit the understanding of relationship between human phenotypes [9, 11].

6.1. Inference of Disease Genes Based on Gene Semantic Similarity Networks. Uncovering relationships between phenotypes and genotypes is a fundamental problem in genetics. In the context of human-inherited diseases, pinpointing causative genes that are responsible for a specific type of disease will greatly benefit the prevention, diagnosis, and treatment of the disease [51]. Traditional statistical methods in this field, including family-based linkage analysis and population-based association studies, can typically locate the genetic risk to a chromosomal region that is 10–30 Mb long, containing dozens of candidate genes [52]. The inference of causative genes from these candidates hence receives more and more attention.

The inference of causative genes is typically modeled as a one-class novelty detection problem [51]. With annotations of a set of seed genes that are known to be responsible for a query disease of interest, candidate genes can be scored according to their functional similarity to the seeds and further prioritized according to their scores. To facilitate the discovery of causative genes for diseases that have no seed genes available, phenotypic similarity between diseases is incorporated. For example, [53] proposed to measure functional similarity between two genes using their proximity in a protein-protein interaction network and further designed a regression model to explain phenotypic similarity between two diseases using functional similarity between genes that were associated with the diseases. However, a protein-protein interaction network can typically cover less than half of known human genes, and thus greatly restricts the scope of application of their method.

To overcome this limitation, Jiang et al. calculated pairwise semantic similarity scores for more than 15,000 human genes based on the biological process domain of the gene ontology [12]. They demonstrated the positive correlation between semantic similarity scores and network proximity scores for pairs of proteins. Moreover, through a comprehensive analysis, they concluded that pairwise semantic similarity scores for genes responsible for the same disease were significantly higher than random selected genes. With these observations, they constructed a semantic similarity network for human genes according to a nearest neighbor rule, and they proposed a random walk model to infer causative genes for a query disease by integrating the phenotype similarity network of diseases and the semantic similarity network of human genes. They compared their methods with a number of the state-of-the-art methods and demonstrated the superior performance of their approach.

6.2. Inference of Drug Indications Based on Disease Semantic Similarity Profiles. The inference of potential drug indications is a key step in drug development [11]. This problem can be defined as follows: given a query disease, a set of small chemical compounds (potential drugs) and known associations between drugs and diseases rank small molecules such that drugs more likely to be associated with the query disease appear higher in the final ranking list. Bearing an analogy to the above problem of inferring causative genes for diseases, the inference of drug indications can greatly benefit from phenotypic similarity profiles of diseases.

A typical method for the derivation of phenotypic similarity profiles of diseases is text mining. For example, van Driel et al. [54] used the anatomy (A) and the disease (C) sections of the medical subject headings vocabulary (MeSH) to extract terms from the OMIM database and further represented the OMIM record (disease) as a vector of the corresponding phenotype features. Then, they defined the similarity score between two disease phenotypes as the cosine of angle between the two corresponding feature vectors. It has been shown that such similarities are positively correlated with a number of measures of functions of genes that are

known to be associated with the diseases, suggesting the effectiveness of this approach.

Recently, the availability of the human phenotype ontology (HPO) [9] provides another means of deriving the phenotypic similarity profile of diseases. Given the ontology and annotations of diseases, Gottlieb et al. [11] proposed to first calculate semantic similarity between terms in the ontology using the method of Resnik [18]. Then, treating a disease as a set of terms in the ontology, they calculated pairwise similarity between OMIM diseases. Further analysis has shown the consistent clustering of diseases according to the semantic similarity profile derived this way (Hamosh et al., 2002). With the semantic similarity profile of diseases ready, Gottlieb et al. [11] further proposed a logistic regression model to predict drug indications for diseases and showed the effectiveness of this profile.

7. Conclusions and Discussion

The explosive increasing of a wide variety of omics data raises the demand of standard annotations of these data using common controlled vocabularies across different experimental platforms and different laboratories. Biological and biomedical ontologies [5], as abstract description systems for knowledge composition in the domain of life sciences, provide structured and controlled representations of terms in this field and, thus, reasonably meet this end. Targeting on the problem of quantifying the relationships between terms in an ontology, and relationships of entities annotated with an ontology, we have summarized a number of existing methods that calculate either semantic similarity between terms using structures of an ontology, annotations of entities, or both. We have further extended the review to the calculation of semantic similarity between entities annotated with an ontology and summarized typical applications that made use of biological and biomedical ontologies.

Although there have been quite a few methods for calculating semantic similarity between terms in biological and biomedical ontologies, the correctness of these methods largely depends on two factors: the quality of the annotation data and the correct interpretation of the hierarchical structure of an ontology. Particularly, for methods that depend on information contents of terms, noise existing in annotation data can adversely affect the correct estimation of the information contents and further bring noise into the resulting semantic similarity. For example, in gene ontology, a large proportion of annotations is inferred electronically by sequence similarity of gene products or other annotation databases. Whether such inferred annotations should be used in the calculation of information contents or not is still an open question. Furthermore, some gene products have been studied in more detail, while knowledge about some gene products is very limited. As a result, available annotations are biased towards heavily studied gene products, and quality of annotations is also biased. Such biased in annotations will also adversely affect the correctness of the derived information contents.

On the other hand, many biological and biomedical ontologies have multiple types of relationships between terms (e.g., is_a, part_of, etc.), and thus methods rely on structure of an ontology need to properly weigh different types of relationships between terms. How to determine such weight values, however, is an open question. For example, although Wang et al. [19] have suggested the weights of 0.6 and 0.8 for is_a and part_of relationships in gene ontology, respectively, whether these values are suitable for other ontologies is not systematically evaluated. Furthermore, for ontologies that have even more types of relationships, the determination of the weight values becomes a more serious problem.

As for applications that make use of ontologies, the problem needs to be cared about is the circularity. For example, information contents are calculated by using annotations, and thus using similarity in annotations to evaluate the goodness of semantic similarity derived from information contents is not appropriate. A direct consequence of overlooking such circularity will be the overestimation of the performance of an application—good in validation but poor in real situation.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China under Grants nos. 71101010 and 61175002, the Fundamental Research Funds for the Central Universities under Grant nos. FRF-BR-11-019A, and the Open Research Fund of State Key Laboratory of Bioelectronics, Southeast University.

References

[1] A. Schulze and J. Downward, "Navigating gene expression using microarrays—a technology review," *Nature Cell Biology*, vol. 3, no. 8, pp. E190–E195, 2001.

[2] T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, and Y. Sakaki, "A comprehensive two-hybrid analysis to explore the yeast protein interactome," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, no. 8, pp. 4569–4574, 2001.

[3] P. Uetz, L. Giot, G. Cagney et al., "A comprehensive analysis of protein'protein interactions in Saccharomyces cerevisiae," *Nature*, vol. 403, pp. 623–627, 2000.

[4] R. Aebersold and M. Mann, "Mass spectrometry-based proteomics," *Nature*, vol. 422, no. 6928, pp. 198–207, 2003.

[5] B. Smith, M. Ashburner, C. Rosse et al., "The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration," *Nature Biotechnology*, vol. 25, no. 11, pp. 1251–1255, 2007.

[6] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowledge Acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[7] M. A. Rodríguez and M. J. Egenhofer, "Determining semantic similarity among entity classes from different ontologies," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 2, pp. 442–456, 2003.

[8] The Gene Ontology Consortium, "The Gene Ontology project in 2008," *Nucleic Acids Research*, vol. 36, pp. D440–D444, 2008.

[9] P. N. Robinson, S. Köhler, S. Bauer, D. Seelow, D. Horn, and S. Mundlos, "The human phenotype ontology: a tool for

annotating and analyzing human hereditary disease," *American Journal of Human Genetics*, vol. 83, no. 5, pp. 610–615, 2008.

[10] P. Jaiswal, S. Avraham, K. Ilic et al., "Plant Ontology (PO): a controlled vocabulary of plant structures and growth stages," *Comparative and Functional Genomics*, vol. 6, no. 7-8, pp. 388–397, 2005.

[11] A. Gottlieb, G. Y. Stein, E. Ruppin, and R. Sharan, "PREDICT: a method for inferring novel drug indications with application to personalized medicine," *Molecular Systems Biology*, vol. 7, article 496, 2011.

[12] R. Jiang, M. Gan, and P. He, "Constructing a gene semantic similarity network for the inference of disease genes," *BMC Systems Biology*, vol. 5, supplement 2, article S2, 2011.

[13] C. Pesquita, D. Faria, A. O. Falcão, P. Lord, and F. M. Couto, "Semantic similarity in biomedical ontologies," *PLoS Computational Biology*, vol. 5, no. 7, Article ID e1000443, 2009.

[14] J. J. Jiang and D. W. Conrath, "Semantic similarity based on corpus statistics and lexical taxonomy," in *Proceedings of the IInternational Conference on Research in Computational Linguistics*, pp. 19–33, 1997.

[15] D. Lin, "An information-theoretic definition of similarity," in *Proceedings of the 15th International Conference on Machine Learning*, pp. 296–304, Morgan Kaufmann, 1998.

[16] A. Maedche and S. Staab, "Measuring similarity between ontologies," in *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, pp. 15–21, 2002.

[17] T. Pedersen, S. V. S. Pakhomov, S. Patwardhan, and C. G. Chute, "Measures of semantic similarity and relatedness in the biomedical domain," *Journal of Biomedical Informatics*, vol. 40, no. 3, pp. 288–299, 2007.

[18] P. Resnik, "Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language," *Journal of Artificial Intelligence Research*, vol. 11, pp. 95–130, 1999.

[19] J. Z. Wang, Z. Du, R. Payattakool, P. S. Yu, and C. F. Chen, "A new method to measure the semantic similarity of GO terms," *Bioinformatics*, vol. 23, no. 10, pp. 1274–1281, 2007.

[20] M. Ashburner, C. A. Ball, J. A. Blake et al., "Gene ontology: tool for the unification of biology," *Nature Genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[21] K. Degtyarenko, P. de matos, M. Ennis et al., "ChEBI: a database and ontology for chemical entities of biological interest," *Nucleic Acids Research*, vol. 36, no. 1, pp. D344–D350, 2008.

[22] G. A. Thorisson, J. Muilu, and A. J. Brookes, "Genotype-phenotype databases: challenges and solutions for the post-genomic era," *Nature Reviews Genetics*, vol. 10, no. 1, pp. 9–18, 2009.

[23] A. S. Sidhu, T. S. Dillon, E. Chang, and B. S. Sidhu, "Protein ontology: vocabulary for protein data," in *Proceedings of the 3rd International Conference on Information Technology and Applications (ICITA '05)*, pp. 465–469, IEEE, July 2005.

[24] E. Segerdell, J. B. Bowes, N. Pollet, and P. D. Vize, "An ontology for Xenopus anatomy and development," *BMC Developmental Biology*, vol. 8, article 92, 2008.

[25] R. J. Bryson-Richardson, S. Berger, T. F. Schilling et al., "FishNet: an online database of zebrafish anatomy," *BMC Biology*, vol. 5, article 34, 2007.

[26] V. A. McKusick, "Mendelian inheritance in man and its online version, OMIM," *American Journal of Human Genetics*, vol. 80, no. 4, pp. 588–604, 2007.

[27] R. Rada, H. Mili, E. Bicknell, and M. Blettner, "Development and application of a metric on semantic nets," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 19, pp. 17–30, 1989.

[28] M. Sussna, "Word sense disambiguation for free-text indexing using a massive semantic network," in *Proceedings of the 2nd International Conference on Information and Knowledge Management*, pp. 67–74, ACM, Washington, DC, USA, November 1993.

[29] Z. Wu and M. Palmer, "Verbs semantics and lexical selection," in *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics*, pp. 133–138, Association for Computational Linguistics, Las Cruces, NM, USA, 1994.

[30] N. Seco, T. Veale, and J. Hayes, "An intrinsic information content metric for semantic similarity in WordNet," *ECAI*. Citeseer, p. 1089, 2004.

[31] A. Schlicker, F. S. Domingues, J. Rahnenführer, and T. Lengauer, "A new measure for functional similarity of gene products based on gene ontology," *BMC Bioinformatics*, vol. 7, article 302, 2006.

[32] H. Wang, F. Azuaje, O. Bodenreider, and J. Dopazo, "Gene expression correlation and gene ontology-based similarity: an assessment of quantitative relationships," in *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB '04)*, pp. 25–31, IEEE, October 2004.

[33] W. N. Lee, N. Shah, K. Sundlass, and M. Musen, "Comparison of ontology-based semantic-similarity measures," in *Proceedings of the American Medical Informatics Association Annual Symposium Proceedings*, pp. 384–388, American Medical Informatics Association, 2008.

[34] F. M. Couto, M. J. Silva, and P. M. Coutinho, "Semantic similarity over the gene ontology: family correlation and selecting disjunctive ancestors," in *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, pp. 343–344, ACM, November 2005.

[35] O. Bodenreider, M. Aubry, and A. Burgun, "Non-lexical approaches to identifying associative relations in the gene ontology," in *Proceedings of the Pacific Symposium on Biocomputing*, pp. 91–102, NIH, 2005.

[36] R. M. Riensche, B. L. Baddeley, A. P. Sanfilippo, C. Posse, and B. Gopalan, "XOA: web-enabled cross-ontological analytics," in *Proceedings of the IEEE Congress on Services*, pp. 99–105, July 2007.

[37] A. Tversky, "Features of similarity," *Psychological Review*, vol. 84, no. 4, pp. 327–352, 1977.

[38] M. A. Rodríguez, M. Egenhofer, and R. Rugg, "Assessing semantic similarities among geospatial feature class definitions," in *Interoperating Geographic Information Systems*, pp. 189–202, 1999.

[39] M. A. Rodríguez and M. J. Egenhofer, "Comparing geospatial entity classes: an asymmetric and context-dependent similarity measure," *International Journal of Geographical Information Science*, vol. 18, no. 3, pp. 229–256, 2004.

[40] C. Leacock and M. Chodorow, "Combining local context and WordNet similarity for word sense and WordNet similarity for word sense identification," in *WordNet: An Electronic Lexical Database*, C. Fellbaum, Ed., The MIT Press, Cambridge, Mass, USA, 1998.

[41] H. Al-Mubaid and H. A. Nguyen, "A cluster-based approach for semantic similarity in the biomedical domain," in *Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '06)*, pp. 2713–2717, IEEE, September 2006.

[42] S. Zhang, X. Shang, M. Wang, and J. Diao, "A new measure based on gene ontology for semantic similarity of genes," in *Proceedings of the WASE International Conference on Information Engineering (ICIE '10)*, pp. 85–88, IEEE, August 2010.

[43] V. Schickel-Zuber and B. Faltings, "OSS: a semantic similarity function based on hierarchical ontologies," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, pp. 551–556, Morgan Kaufmann, 2007.

[44] Y. Guisheng and S. Qiuyan, "Research on ontology-based measuring semantic similarity," in *Proceedings of the International Conference on Internet Computing in Science and Engineering (ICICSE '08)*, pp. 250–253, IEEE, January 2008.

[45] G. Yu, F. Li, Y. Qin, X. Bo, Y. Wu, and S. Wang, "GOSemSim: an R package for measuring semantic similarity among GO terms and gene products," *Bioinformatics*, vol. 26, no. 7, pp. 976–978, 2010.

[46] H. Zheng, F. Azuaje, and H. Wang, "seGOsa: software environment for Gene Ontology-driven similarity assessment," in *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM '10)*, pp. 539–542, December 2010.

[47] J. Li, B. Gong, X. Chen et al., "DOSim: an R package for similarity between diseases based on disease ontology," *BMC Bioinformatics*, vol. 12, article 266, 2011.

[48] R. C. Gentleman, V. J. Carey, D. M. Bates et al., "Bioconductor: open software development for computational biology and bioinformatics," *Genome Biology*, vol. 5, no. 10, p. R80, 2004.

[49] H. Ogata, S. Goto, K. Sato, W. Fujibuchi, H. Bono, and M. Kanehisa, "KEGG: kyoto encyclopedia of genes and genomes," *Nucleic Acids Research*, vol. 27, no. 1, pp. 29–34, 1999.

[50] J. L. Sevilla, V. Segura, A. Podhorski et al., "Correlation between gene expression and GO semantic similarity," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 2, no. 4, pp. 330–338, 2005.

[51] Y. Moreau and L. C. Tranchevent, "Computational tools for prioritizing candidate genes: boosting disease gene discovery," *Nature Reviews Genetics*, vol. 13, no. 8, pp. 523–536, 2012.

[52] A. M. Glazier, J. H. Nadeau, and T. J. Aitman, "Genetics: finding genes that underline complex traits," *Science*, vol. 298, no. 5602, pp. 2345–2349, 2002.

[53] X. Wu, R. Jiang, M. Q. Zhang, and S. Li, "Network-based global inference of human disease genes," *Molecular Systems Biology*, vol. 4, article 189, 2008.

[54] M. A. van Driel, J. Bruggeman, G. Vriend, H. G. Brunner, and J. A. M. Leunissen, "A text-mining analysis of the human phenome," *European Journal of Human Genetics*, vol. 14, no. 5, pp. 535–542, 2006.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 393570, 7 pages
http://dx.doi.org/10.1155/2013/393570

Research Article

A Robust Hybrid Approach Based on Estimation of Distribution Algorithm and Support Vector Machine for Hunting Candidate Disease Genes

Li Li,[1,2] Hongmei Chen,[1] Chang Liu,[1] Fang Wang,[1] Fangfang Zhang,[1] Lihua Bai,[2] Yihan Chen,[2] and Luying Peng[1,2]

[1] *Devision of Medical Genetics, Tongji University School of Medicine, Shanghai 200092, China*
[2] *Key Lab for Basic Research in Cardiology, Ministry of Education, Tongji University, Shanghai 200092, China*

Correspondence should be addressed to Yihan Chen; yihanchen@tongji.edu.cn and Luying Peng; luyingpeng@tongji.edu.cn

Received 23 October 2012; Accepted 25 November 2012

Academic Editors: R. Jiang, W. Tian, J. Wan, and X. Zhao

Microarray data are high dimension with high noise ratio and relatively small sample size, which makes it a challenge to use microarray data to identify candidate disease genes. Here, we have presented a hybrid method that combines estimation of distribution algorithm with support vector machine for selection of key feature genes. We have benchmarked the method using the microarray data of both diffuse B cell lymphoma and colon cancer to demonstrate its performance for identifying key features from the profile data of high-dimension gene expression. The method was compared with a probabilistic model based on genetic algorithm and another hybrid method based on both genetics algorithm and support vector machine. The results showed that the proposed method provides new computational strategy for hunting candidate disease genes from the profile data of disease gene expression. The selected candidate disease genes may help to improve the diagnosis and treatment for diseases.

1. Introduction

Complex diseases are frequently accompanied by changes in gene expression patterns which can serve as secondary endpoints or biomarkers [1]. Microarray technology, which allows researchers to simultaneously measure expression levels of thousands or tens of thousands of genes in a single experiment, has been widely used to explore the gene expression pattern of complex diseases [2]. Typically, there are only a small number of genes associated with diseases. Thus, the selection of feature genes that possess discriminatory power for disease phenotypes is a common task for mining microarray data that are usually high dimension (with thousands of genes) and have small sample size (with usually a few dozens of samples) [3].

The method of gene selection generally falls into one of the following three categories: the filter, wrapper, and embedded approaches. The filter approach collects the intrinsic characteristics of genes in discriminating the targeted phenotype class and usually employs statistical methods, such as mutual information, statistical tests (t-test, F-test), and Wilcoxon's rank test, to directly select feature genes [4, 5]. This approach is easily implemented, but ignores the complex interaction between genes. The "wrapper" approach [6] aims at selecting a subset of feature genes, typically with an induction algorithm to search for an initial gene subset which can then be used for further evaluating new feature gene subsets. The wrapper method is usually superior to the filter one since it involves intercorrelation of individual genes in a multivariate manner. The wrapper method can automatically determine the optimal number of feature genes for a particular classifier. The embedded method is similar to the wrapper method, while multiple algorithms can be combined in the embedded method to perform feature subset selection [6, 7]. In the embedded method, genetic algorithms (GAs) [8, 9] are generally used as the search engine for feature subset, while other classification methods, such as KNN/GA (K nearest neighbors/genetic algorithms) [10], GA-SVM (genetic algorithms-support vector machine) [11], and so forth, are used to select feature subset. Estimation of

Step 1. $M_0 \leftarrow$Read gene expression profile matrix from database, m is the number of genes in M_0.
Step 2. $D_0 \leftarrow$Generate N individuals (the initial population) randomly. Each individual has an m-length vector of bits of either 1 or 0.
Step 3. For each individual j in D_0, determine:
$G_j \leftarrow$a gene subset corresponding to individual j. If bit i equals to 1, include g_i in the subset.
$M_j \leftarrow$gene expression profile submatrix.
Fitness$_j \leftarrow$ eval(M_j).
Step 4. $D_l^r \leftarrow$retain $N/2$ individuals with the highest evaluations.
Step 5. M arginal$(z_i, l) \leftarrow$calculate marginal distribution of variable z_i of bit i based on D_l^r by using the formula: M arginal$(z_i, l) = \left(\sum_{j=1}^{N/2} z_i^j\right) / (N/2)$, where z_i^j is the value of the variable z_i in individual j.
$M_{\text{weight}}(z_i, l) \leftarrow$ calculate weight of z_i corresponding to feature i based on D_l^r.
$M_{\text{weight}}(z_i, l) = \left\{\sum_{j=1}^{N/2} \text{Pre}_{\text{weight}}(z_i^j)\right\} / (N/2)$, where $\text{Pre}_{\text{weight}}(z_i^j)$ is weight of bit i in individual j.
Pro$b(z_i, l+1) \leftarrow$compute probability distribution z_i of each bit i, which is written mathematically as:
Pro$b(z_i, l+1) = lr\beta_i *Prob(z_i, l) + (1 - lr) * (1 - \beta_i) * M$ arginal$(z_i, l) * M_{\text{weight}}(z_i, l)$.
$lr \in (0, 1)$ is learning rate. $\beta_i \in (0, 1)$ is generated at random.
Step 6. $D_{l+1}^{\text{new}} \leftarrow$generate new $N/2$ individuals by sampling the probability distribution.
Step 7. $D_{l+1} \leftarrow D_l^r \cup D_{l+1}^{\text{new}}$.
Step 8. $D_0 \leftarrow D_{l+1}$.
Step 9. End$\leftarrow$output the optimal individual based on the evaluation with: fitness$_j$ = eval(M_j).

Algorithm 1: The step-by-step recipe for the computational algorithm of the EDA-SVM approach.

distribution algorithm (EDA) [12] is a general framework of GA. Compared to traditional GA that employs crossover and mutation operators to create new population, EDA creates new populations by using a statistical approach to estimate the probability distribution of all promising individual solutions for the previous generation. EDA can also explicitly take into account specific interactions among the variables. When EDA is used to search for feature subsets, classification methods, such as Support vector machine (SVM) [13–19], which can deal with the high-dimension data in a limited sample space, can be used to select feature subsets.

In this study, we have developed a hybrid approach that combines both EDA and SVM (termed EDA-SVM) for selecting key feature genes. Here, EDA acts as the search engine, while SVM serves as the classifier, namely, the evaluator. We have applied EDA-SVM to two well-known microarray datasets: a colon data [20] and a diffuse large B cell lymphoma data [3]. Our results have shown that EDA-SVM can be used to identify a smaller number of informative genes with better accuracy in comparison to GA-SVM [11] and an estimation of distribution algorithm named PMBGA [21].

2. Materials and Methods

2.1. Description of DLBCL Datasets. We have applied the EDA-SVM method to the two following data sets: the diffuse large B cell lymphoma (DLBCL) data [3], available at http://llmpp.nih.gov/lymphoma/data.shtml, and the colon data [20], available at http://microarray.princeton.edu/oncology/affydata/index.html. The colon data set consists of 62 tissue samples including 40 tumors and 22 normal tissues, which cover 2000 human gene expression. The DLBCL data set harbors preprocessed expression profile of 4026 genes in tissues derived from 21 activated B-like DLBCL (AB-like DLBCL) samples and 21 germinal center B-like DLBCL (GCB-like DLBCL) samples.

2.2. Data Preprocessing. In DLBCL dataset, among 4026 genes, 6% genes have missing values and are imputed by the KNN Impute algorithm [22] prior to the EDA-SVM analysis. The KNN Impute algorithm uses the expression profiles of K nearest neighbors (here $K = 5$) to impute the missing values for the target gene. Therefore, in colon data M_0 is a matrix with 62 rows and 2000 columns. In DLBCL data, M_0 is a matrix with 42 rows and 4026 columns.

2.3. EDA-SVM. Figure 1 shows the main flowchart of the EDA-SVM. EDA acts as the search engine, while SVM serves as the classifier, namely, the evaluator. The computational procedures are described in Algorithm 1. The major elements of the EDA include feature subset coding, population initialization, fitness computation, estimation probability distribution, generation of offspring and control of parameter assignment. At the beginning, we randomly generated the N fixed-length binary strings (individuals) to build up the initial population. Then, we calculated the fitness for each feature subset. Classification accuracy acted as the fitness index (fitness) that was evaluated using a linear SVM. The algorithm is an iterative process in which each successive generation is produced by estimating the probability distribution model of the selected individuals (parents) in the current generation and sampling the probability distribution to generate new offsprings. In this manner, reasonable subsets are developed successively until the terminal condition is fulfilled. In two

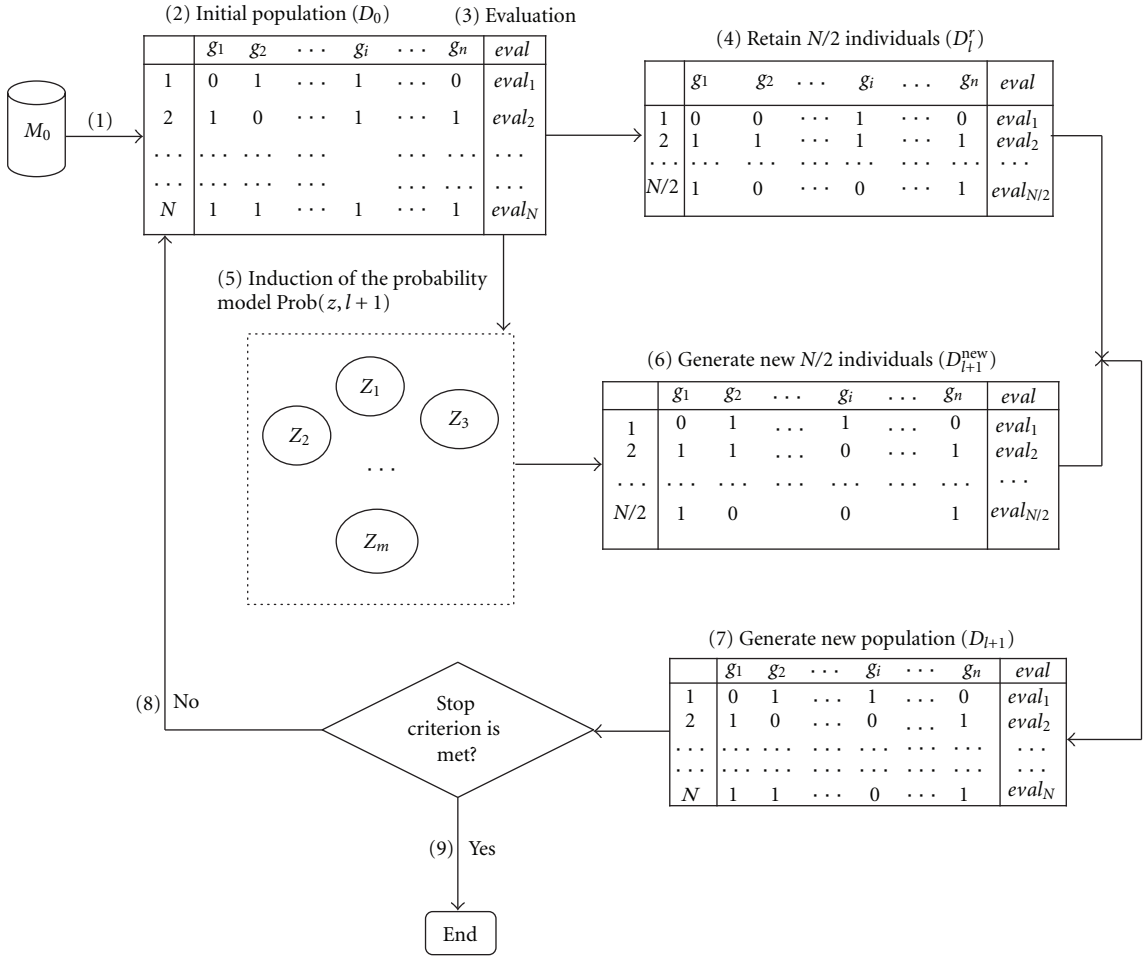


Figure 1: The main flow of EDA-SVM algorithm. M, D, G, and *eval* denote gene expression profile matrix, population, gene subset, and evaluation index, respectively.

data sets, lr is a learning rate and is assigned 0.08. Population size (N) is set as 40 and the maximal generations of 50 are determined, such that the solution space can be sufficiently searched while the best minimal subset can be obtained within the evolution time.

For each gene expression submatrix M_j, we classify the microarray samples with genes contained in individual j using a linear SVM. The classifier, [18], is

$$\widehat{y} = f(x) = \operatorname{sgn}\left(\sum_{i=1}^{L} a_i y_i K(x_i \cdot x) - b\right), \tag{1}$$

then, the accuracy of classification is

$$\operatorname{acc} = \frac{\left(\sum_{t=1}^{T} I(y_t, \widehat{y_t})\right)}{T}, \tag{2}$$

where T is the number of test samples and

$$I(y_t, \widehat{y_t}) = \begin{cases} 1, & \text{if } y_t = \widehat{y_t}, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

The weight of each feature i in individual j is

$$\operatorname{Pre}_{\text{weight}}\left(z_i^j\right) = \begin{cases} 0, & \text{if } z_i^j = 0, \\ \left(\sum_{h=1}^{L} \alpha_h y_h x_h\right)^2, & \text{if } z_i^j \neq 0, \end{cases} \tag{4}$$

where x is a test sample vector and x_i is the learning sample vector. L is the number of learning samples. y_i is a class indicator (for a two-class application, +1 for the first class, −1 for the second class), and a_i is a nonnegative Lagrange multiplier associated with x_i and $a_i \neq 0$ for support vectors. sgn( ) is the sign function and $K(x_i \cdot x)$ is the kernel function: linear kernel ($K(x_i \cdot x) = x_i \cdot x$, i.e., their inner product).

In this study, a fivefold cross-validation (CV) resampling approach is used to construct the learning and test sets. First, the two-class samples are randomly divided into 5 nonoverlapping subsets of roughly equal size, respectively. A random combination of the subsets for the two classes constitutes a test set, and the rest of subsets is totally used as the learning set. The 5-fold CV resampling produces 25

pairs of learning and test sets. Individual j is evaluated by the averaged value over the 25 pairs, that is,

$$\text{Fitness}_j = \frac{\left(\sum_{k=1}^{25} \text{acc}_k\right)}{25},$$
$$\text{weight}\left(z_i^j\right) = \frac{\left(\sum_{k=1}^{25} \text{Pre}_{\text{weight}k}\left(z_i^j\right)\right)}{25}, \quad (5)$$

where k is the replicate number and acc_k is the classification accuracy for the kth replicate.

In the EDA-SVM algorithm, the optimization of the feature gene subset(s) is realized via survival competitions. For each generation, we retain 50% of the high-valued individuals that will directly enter next generation in order to keep these optimal solutions unchanged. On the other hand, in order to avoid the loss of the putative important feature genes, we initially contained about half of genes in each individual or preserving informative gene. Then, we adopt a stepwise data reduction procedure to minimize the feature subsets with more reliable classification accuracy. These gene expression matrices from the optimal individuals serve as the data on which the new round of iteration is performed. The data reduction process is completed once a stable gene subset is obtained.

2.4. GA-SVM. GA-SVM was previously developed [11] by us as a feature selection method. In GA-SVM, better feature subsets have a greater chance of being selected to form a new subset through crossover or mutation. Mutation changes some of the values (thus adding or deleting features) in a subset randomly. Crossover combines different features from a pair of subsets into a new subset. The algorithm is an iterative process in which each successive generation is produced by applying genetic operators to the members of the current generation. In this manner, good subsets are "evolved" over time until the stopping criteria are met. Thus, coding feature subset, population initialization, fitness computation, genetic operation, and control parameter assignment (population size, the maximal number of generations, and the selection probability) are the major elements of the GA-SVM method.

2.5. PMBGA. PMBGA can be applied for selection of a smaller size gene subset that would classify patient samples more accurately [21]. PMGBA generates initial population and builds a probability model and then selects individuals from the population. Probability distribution can be estimated based on the collection of selected individuals, and probability model can accordingly be amended so that a population is generated by sampling from the model. Instead of applying crossover and mutation operators in the process of generating new possible solutions (offspring), population can be updated in whole or in part relied on probability model.

3. Results

3.1. Benchmark EDA-SVM. The EDA-SVM method was applied firstly to the DLBCL data set. We started analysis with all 4026 genes and progressively reduced the dimension of the feature genes successively for 8 iterations after convergence. The accuracy of EDA-SVM increased from 0.9339 initially to 0.9982 at convergence (Figure 2(a)), while the number of feature genes at the successive generations is 4026, 460, 66, 17, 11, 7, 6, and 6, respectively (Figure 2(b)). For the colon data set, EDA-SVM reached accuracy of 1.0 after 7 iterations, and the final gene subset includes only 5 genes (Figure 3).

We compared the performance of EDA-SVM with two alternative methods: GA-SVM and PMBGA (Figures 2 and 3). The convergence speed of EDA-SVM is the fastest among the three methods. EDA-SVM converged after 8 and 7 iterations for the DLBCL and colon datasets, respectively. In contrast, it took 13 and 10 iterations for GA-SVM to converge, and 10 and 10 iterations for PMBGA to converge. Moreover, both the accuracy and the stability of EDA-SVM also show advantages among the three methods. EDA-SVM quickly reaches high accuracy after only a couple of iterations, while both the other two methods took more iteration to reach high accuracy. In addition, the accuracy of the other two methods had large variation during the iteration, while the accuracy of EDA-SVM kept stable during the iteration after it reached the high accuracy.

3.2. Biological Analysis of the Selected Genes in the DLBCL Data. To understand the biological significance of the selected genes, we have analyzed the annotations of selected genes according to Gene Ontology (GO) (http://www.geneontology.org/) [23] and KEGG (http://www.genome.jp/kegg/kegg2.html) [24, 25] database. We selected six genes in the DLBCL data, which are SPIB, IRF8, NFKB2, LMO2, FCGRT, and BCL7B. The GO annotations of these six genes are shown in Table 1. Literature reviews of these six genes suggested that they are highly related to DLBCL. SPIB is an oncogene involved in the pathogenesis of AB-like DLBCL [26]. NFKB2 is a subunit of NF-κB whose signaling pathway might contribute to the biological and clinical differences between the GCB-like and the AB-like DLBCL [27]. LMO2 was found to be located in the most frequent regime of chromosomal translocation in childhood T cell acute lymphoblastic leukemia. It was reported that LMO2 expressed at high level in germinal center B cell lymphocytes and at low level in AB-like DLBCL, respectively [3]. LMO2 is also one of the six genes in a multivariate model previously developed for prolonged survival in the diffusive large b-cell lymphoma [28]. BCL7B was found to be directly involved in a three-way gene translocation together with Myc and IgH in a Burkitt lymphoma cell line, and the disruption of the N-terminal region of BCL7B was thought to be related to the pathogenesis of a subset of high-grade B cell non-Hodgkin lymphoma [29]. BCL2 contributes to the pathogenesis in AB-like DLBCL [10] and is the common target gene of miR-21 and miR-221, both of which are overexpressed in AB-like than GCB-like cell lines [30]. Based on the above evidences, EDA-SVM successfully identified genes that may play role in the pathogenesis of DLBCL.

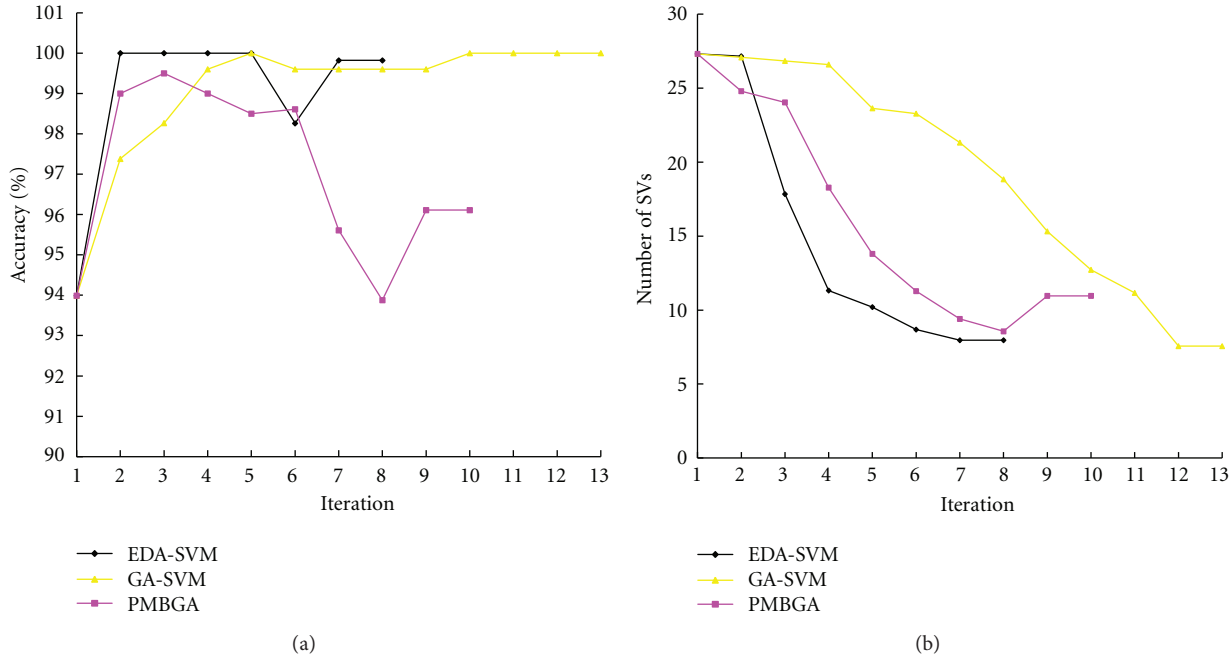


Figure 2: The changes of accuracy of the SVM classifier (a) and the changes of support vectors (b) over iterations in EDA-SVM, GA-SVM, and PMBGA based on DLBCL data set.

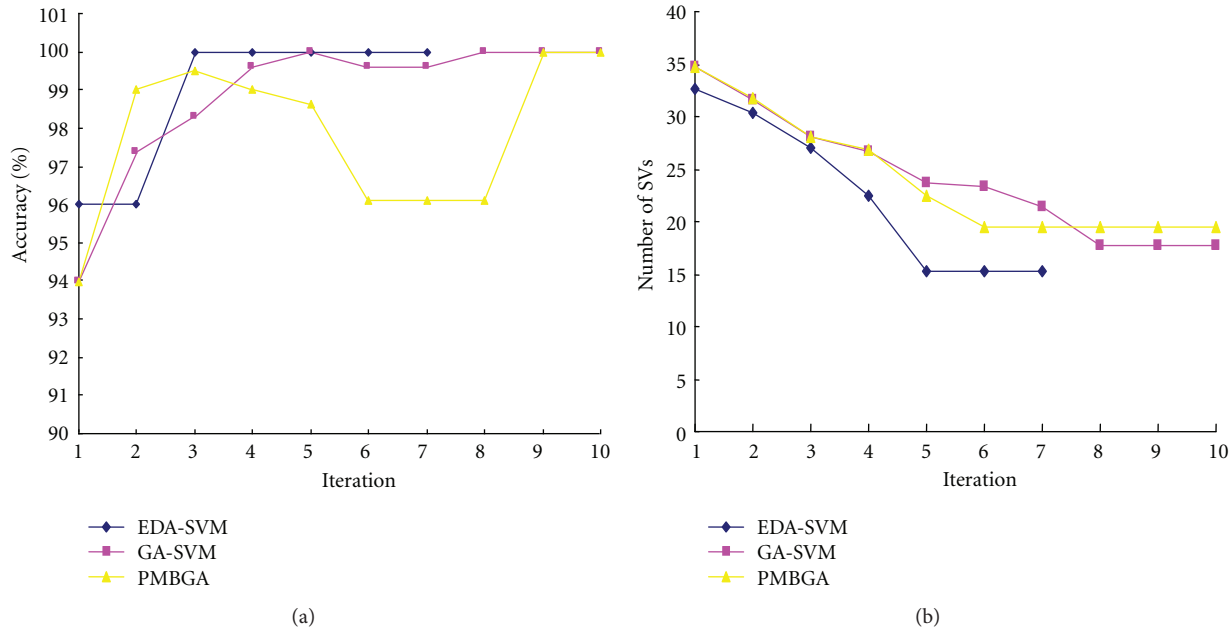


Figure 3: The changes of accuracy of the SVM classifier (a) and the changes of support vectors (b) over iterations in EDA-SVM, GA-SVM, and PMBGA based on colon data set.

4. Discussions and Conclusions

In this study, we have developed a hybrid method, EDA-SVM, which combines the estimation of distribution algorithms (EDA) with support vector machine (SVM) for selecting key feature genes from microarray data. Although similar combination strategies have been explored previously [21], EDA-SVM shows unique advantages compared with the alternative methods, GA-SVM or PMBGA. For example, EDAV-SVM not only converged more quickly, but also achieved higher accuracy with stable performance than the other two methods did. Both EDA-SVM and PMBGA [21] use EDA as the search engine, and SVM acts as evaluation classifier in feature selection procedure. However, there are

Table 1: The GO annotations of EDA-SVM feature genes.

Gene name Uigene ID	Biological process	Cellular component	Molecular function
SPIB (Hs.437905)	GO:0006350 Transcription GO:0006357 Regulation of transcription from RNA polymerase II promoter	GO:0005634 Nucleus GO:0005737 Cytoplasm	GO:0003700 Transcription factor activity GO:0003702: RNA polymerase II transcription factor activity
IRF8 (Hs.137427)	GO:0000122 Negative regulation of transcription from RNA polymerase II promoter GO:0006355 Regulation of transcription, DNA-dependent GO:0006350 Transcription GO:0006955 Immune response	GO:0005634 Nucleus	GO:0003705: RNA polymerase II transcription factor activity, enhancer binding
NFKB2 (Hs.73090)	Go:0006355 Regulation of transcription, DNA-dependent GO:0007165 Signal transduction	GO:0005634 Nucleus GO:0005737 Cytoplasm	GO:0005515 Protein binding GO:0003713 Transcription coactivator activity GO:0003700 Transcription factor activity
LMO2 (Hs.34560)	GO:0008270 Development	GO:0005634 Nucleus	GO:0008270 Zinc ion binding GO:0005515 Protein binding GO:0046872 Metal ion binding
FCGRT (Hs.111903)	GO:0019882 Antigen presentation GO:0007565 Pregnancy GO:0006955 Immune response	GO:0042612 MHC class I protein complex GO:0016021 Integral to membrane	GO:0019864 IgG binding GO:0004872 Receptor activity GO:0030106 MHC class I receptor activity
BCL7B (Hs.408219)	Unknown	Unknown	GO:0003779 Actin binding

several key differences between the two methods. First, EDA-SVM weights each feature using "M_{weight}", so that the contribution of each feature was fully considered during the update of each generation. In contrast, PMBGA assigns only a small random number to each feature. Second, for selecting minimal feature genes, EDA-SVM reduced the feature number step by step, while PMBGA did so by tuning the learning rate. Finally, the way to create the next generation in GA is also different between the two methods. As for the differences between EDA-SVM and GA-SVM, GA-SVM employs the traditional GA, while EDA-SVM generates new possible solutions (individuals) by sampling the probability distribution calculated from the selected solutions of previous generation.

The structure of genes in a microarray data can be described by a Bayesian network. However, microarray data usually contains the expression of thousands or tens thousands of genes, making it virtually impossible to build a Bayesian network with so many genes. In this study, we have shown with EDA-SVM that proper combination of machine learning algorithms can overcome the high-dimension problem, and quickly converge to a small set of feature genes strongly related to target phenotype. The success of EDA-SVM thus made it readily applicable for hunting disease genes in microarray data.

List of Abbreviations

DLBCL: Diffuse large B-cell lymphoma
EDA-SVM: Estimation for distribution algorithm-support vector machine
GO: GeneOntology
KEGG: Kyoto Encyclopedia of Genes and Genomes
GAs: Genetic algorithms
EDA: Estimation of distribution algorithm
AB-like DLBCL: Activated B-like DLBCL
GCB-like DLBCL: Germinal center B-like DLBCL
PMBGA: Probabilistic Model Building Genetic Algorithm
GA-SVM: Genetic algorithm-support vector machine.

Acknowledgments

This work is supported in part by National Natural Science Foundation of China (30971621, 81270231, and 31170791), the National Basic Research Program of China (973 Program) (2012CB9668003 and 2010CB945500), International Science and Technology Cooperation Program of China (2011DFB30010), the Fundamental Research Funds for the

Central Universities to L. Li, and Shanghai Municipal Health Bureau Project to L. Li. We thank Dr. Weidong Tian for critical review of the paper.

References

[1] W. Yang, D. Ying, and Y. L. Lau, "In-depth cDNA library sequencing provides quantitative gene expression profiling in cancer biomarker discovery," *Genomics, Proteomics and Bioinformatics*, vol. 7, no. 1-2, pp. 1–12, 2009.

[2] S. S. Shen-Orr, R. Tibshirani, P. Khatri et al., "Cell type-specific gene expression differences in complex tissues," *Nature Methods*, vol. 7, no. 4, pp. 287–289, 2010.

[3] A. A. Alizadeh, M. B. Elsen, R. E. Davis et al., "Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling," *Nature*, vol. 403, no. 6769, pp. 503–511, 2000.

[4] P. J. Park, M. Pagano, and M. Bonetti, "A nonparametric scoring algorithm for identifying informative genes from microarray data," *Pacific Symposium on Biocomputing*, pp. 52–63, 2001.

[5] Y. Su, T. M. Murali, V. Pavlovic, M. Schaffer, and S. Kasif, "RankGene: Identification of diagnostic genes based on expression data," *Bioinformatics*, vol. 19, no. 12, pp. 1578–1579, 2003.

[6] R. Kahavi and G. H. John, "Wrapper for feature subset selection," *Artificial Intelligence*, vol. 97, pp. 273–324, 1997.

[7] X. Li, S. Rao, Y. Wang, and B. Gong, "Gene mining: a novel and powerful ensemble decision approach to hunting for disease genes using microarray expression profiling," *Nucleic Acids Research*, vol. 32, no. 9, pp. 2685–2694, 2004.

[8] S. J. Cho and M. A. Hermsmeier, "Genetic algorithm guided selection: variable selection and subset selection," *Journal of Chemical Information and Computer Sciences*, vol. 42, no. 4, pp. 927–936, 2002.

[9] X. M. Zhao, Y. M. Cheung, and D. S. Huang, "A novel approach to extracting features from motif content and protein composition for protein sequence classification," *Neural Networks*, vol. 18, no. 8, pp. 1019–1028, 2005.

[10] L. Li, T. A. Darden, C. R. Weinberg, A. J. Levine, and L. G. Pedersen, "Gene assessment and sample classification for gene expression data using a genetic algorithm/k-nearest neighbor method," *Combinatorial Chemistry and High Throughput Screening*, vol. 4, no. 8, pp. 727–739, 2001.

[11] L. Li, W. Jiang, X. Li et al., "A robust hybrid between genetic algorithm and support vector machine for extracting an optimal feature gene subset," *Genomics*, vol. 85, no. 1, pp. 16–23, 2005.

[12] Y. Saeys, S. Degroeve, D. Aeyels, P. Rouzé, and Y. Van de Peer, "Feature selection for splice site prediction: a new method using EDA-based feature ranking," *BMC Bioinformatics*, vol. 5, p. 64, 2004.

[13] M. P. S. Brown, W. N. Grundy, D. Lin et al., "Knowledge-based analysis of microarray gene expression data by using support vector machines," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 97, no. 1, pp. 262–267, 2000.

[14] J. H. Oh and J. Gao, "A kernel-based approach for detecting outliers of high-dimensional biological data," *BMC Bioinformatics*, vol. 10, supplement 4, p. S7, 2009.

[15] S. Hua and Z. Sun, "A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach," *Journal of Molecular Biology*, vol. 308, no. 2, pp. 397–407, 2001.

[16] Y. Zhu, X. Shen, and W. Pan, "Network-based support vector machine for classification of microarray samples," *BMC Bioinformatics*, vol. 10, supplement 1, p. S21, 2009.

[17] L. Evers and C. M. Messow, "Sparse kernel methods for high-dimensional survival data," *Bioinformatics*, vol. 24, no. 14, pp. 1632–1638, 2008.

[18] V. Vapnik, *Statistical Learning Theory*, Wiley, New York, NY, USA, 1998.

[19] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[20] U. Alon, N. Barka, D. A. Notterman et al., "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 96, no. 12, pp. 6745–6750, 1999.

[21] T. K. Paul and H. Iba, "Gene selection for classification of cancers using probabilistic model building genetic algorithm," *BioSystems*, vol. 82, no. 3, pp. 208–225, 2005.

[22] O. Troyanskaya, M. Cantor, G. Sherlock et al., "Missing value estimation methods for DNA microarrays," *Bioinformatics*, vol. 17, no. 6, pp. 520–525, 2001.

[23] M. Ashburner, C. A. Ball, J. A. Blake et al., "Gene ontology: tool for the unification of biology. The Gene Ontology Consortium," *Nature Genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[24] M. Kanehisa, S. Goto, S. Kawashima, Y. Okuno, and M. Hattori, "The KEGG resource for deciphering the genome," *Nucleic Acids Research*, vol. 32, pp. D277–D280, 2004.

[25] M. Kanehisa, S. Goto, S. Kawashima, and A. Nakaya, "Thed KEGG databases at GenomeNet," *Nucleic Acids Research*, vol. 30, no. 1, pp. 42–46, 2002.

[26] G. Lenz, G. W. Wright, N. C. T. Emre et al., "Molecular subtypes of diffuse large B-cell lymphoma arise by distinct genetic pathways," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 36, pp. 13520–13525, 2008.

[27] R. E. Davis, K. D. Brown, U. Siebenlist, and L. M. Staudt, "Constitutive nuclear factor kappaB activity is required for survival of activated B cell-like diffuse large B cell lymphoma cells," *The Journal of Experimental Medicine*, vol. 194, pp. 1861–1874, 2001.

[28] I. S. Lossos, D. K. Czerwinski, A. A. Alizadeh et al., "Prediction of survival in diffuse large-B-cell lymphoma based on the expression of six genes," *The New England Journal of Medicine*, vol. 350, no. 18, pp. 1828–1837, 2004.

[29] S. Amenta, M. Moschovi, C. Sofocleous, S. Kostaridou, A. Mavrou, and H. Fryssira, "Non-Hodgkin lymphoma in a child with Williams syndrome," *Cancer Genetics and Cytogenetics*, vol. 154, no. 1, pp. 86–88, 2004.

[30] C. H. Lawrie, S. Soneji, T. Marafioti et al., "MicroRNA expression distinguishes between germinal center B cell-like and activated B cell-like subtypes of diffuse large B cell lymphoma," *International Journal of Cancer*, vol. 121, no. 5, pp. 1156–1161, 2007.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 675851, 10 pages
http://dx.doi.org/10.1155/2013/675851

Review Article

Prediction of Deleterious Nonsynonymous Single-Nucleotide Polymorphism for Human Diseases

Jiaxin Wu and Rui Jiang

MOE Key Laboratory of Bioinformatics and Bioinformatics Division, TNLIST/Department of Automation, Tsinghua University, Beijing 100084, China

Correspondence should be addressed to Rui Jiang; ruijiang@tsinghua.edu.cn

Received 27 October 2012; Accepted 11 December 2012

Academic Editors: C. Proctor and R. Rivas

The identification of genetic variants that are responsible for human inherited diseases is a fundamental problem in human and medical genetics. As a typical type of genetic variation, nonsynonymous single-nucleotide polymorphisms (nsSNPs) occurring in protein coding regions may alter the encoded amino acid, potentially affect protein structure and function, and further result in human inherited diseases. Therefore, it is of great importance to develop computational approaches to facilitate the discrimination of deleterious nsSNPs from neutral ones. In this paper, we review databases that collect nsSNPs and summarize computational methods for the identification of deleterious nsSNPs. We classify the existing methods for characterizing nsSNPs into three categories (sequence based, structure based, and annotation based), and we introduce machine learning models for the prediction of deleterious nsSNPs. We further discuss methods for identifying deleterious nsSNPs in noncoding variants and those for dealing with rare variants.

1. Introduction

Understanding the relationship between phenotype and genotype is a fundamental problem in genetics. Of particular interest, the identification of genetic risk factors underlying human inherited diseases has long been a goal in human and medical genetics. Since genetic variation is believed to be the major factor that stimulates the diversity between individuals [1], considerable efforts have been taken to understand associations between human genetic variants and their phenotypic effects [2]. A number of successful stories have shown that such efforts are helpful in capturing the causative variants which affect human inherited diseases, providing important information for grasping genetic bases of complex diseases, and further promoting the prevention, diagnosis, and treatment of these diseases [3]. Nevertheless, recent studies have shown that the number of genetic variants is huge, more than 3.5 million variants in the whole genome for a single individual, roughly corresponding to 1,000 variants per megabase pair [4, 5], making the identification of causative variants a task of finding needles in stacks of needles. Furthermore, it has also been shown that although most genetic variants exist common in a population, there also exists a nonnegligible number of variants that occur in very low frequency, making established statistical methods for identifying such rare variants ineffective. Hence, the development of novel computational methods to identify causative variants now receives more and more attentions.

Genetic variants can typically be classified into several categories, including single-nucleotide polymorphisms (SNPs), small insertions and deletions, and structural variants [4]. Among these types of variants, single-nucleotide polymorphisms (SNPs) that occur in single bases of DNA sequences account for a majority of all genetic variants. It has been estimated that there exist nearly 10 million SNPs in the human genome, nearly one SNP for every 290 base-pairs. The vast number of SNPs along with growing functional annotations of the human genome sequence may provide plenty of knowledge to grasp links between genetic and phenotypic variations [6]. Particularly, as an important type of SNP, a nonsynonymous single-nucleotide polymorphism (nsSNP) occurring in a protein coding region alters the encoded amino acid sequence, potentially affects protein structure and function, and further causes human inherited diseases. It has

been reported that nsSNPs constitute more than 50% of the mutations known to be involved in human inherited diseases [7] and each person may hold 24,000–40,000 nsSNPs [8]. It is also believed that although most of the susceptible deleterious nsSNPs are related to individual Mendelian diseases, functional changes aroused by nsSNPs will be of importance for complex diseases [8]. Therefore, more effort should be paid for studying the candidate deleterious nsSNPs [9].

The identification of genetic variants that are associated with human diseases is often undertaken using either a family-based linkage analysis or a population-based association study. In a linkage analysis, susceptible disease-causing loci (usually between 1 and 5 million bp in length) are mapped by identifying genetic markers that are coinherited with a query phenotype. Linkage analysis has poor prediction power for difficultly collecting family-based sequence data and poor performance for complex diseases which are caused by the combination of effects of several susceptible genetic variants and their interactions with environmental factors [10]. An association study compares frequencies of occurrence of genetic variants between a case population and a control population to detect associations between genetic variants and phenotypes [11]. With recent advances in high-throughput experimental techniques, association studies are now often conducted in genomewide scale, often referred to as genomewide association (GWA) studies. Although such a GWA study has shown some success in the past few years, it suffers from serious multiple testing problem when applied to a number of markers in a large population, and its basic hypothesis of Common Disease Common Variant (CDCV) has been challenged by the fact that both common variants and rare variants may be involved in the pathogenesis of common diseases.

To overcome these limitations and serve as a complementary category of these traditional statistical methods, computational approaches that rely on properties of variants instead of experimental data of patients have been designed for the detection of deleterious variants, with the growing functional annotations of the human genome sequence. Although such methods may never be accurate enough to replace wet-lab experiments, they may help in identifying and prioritizing a small number of susceptible and tractable candidate nsSNPs from pools of available data [1]. Recent studies [9–21] have shown that computational methods are capable of well estimating the functional effects of nsSNPs. These approaches may take advantage of structure information, sequence information, and annotations as classification features, as well as logistic regression [21], neural networks [1], Bayesian models [5], and other statistical approaches [18] as classifiers.

In this paper, we first summarize the databases for collecting nsSNP data and provide a framework of nsSNP function prediction methodology. We survey existing deleterious nsSNPs prediction methods and summarize the prediction features conducted in prediction models and the prediction algorithms to distinguish the deleterious nsSNPs. Then, we discuss computational methods that use comparative genomics to predict deleteriousness of nsSNPs in both coding and noncoding regions. We also look at prioritization methods for disease-specific nsSNPs detection and discuss deleterious nsSNPs prediction methods for rare variants detection. Finally, we suggest using multiple prediction algorithms to enhance the prediction power and discuss challenges and likely future improvements of such methods.

2. Databases for nsSNPs

Many popular databases present useful information of nsSNPs. Particularly, as shown in Table 1, deleterious nsSNPs are mainly collected in four databases: the Online Mendelian Inheritance in Man (OMIM) database [22], the Human Gene Mutation Database (HGMD) [12], the UniProt/Swiss-Prot database [13], and the Human Genome Variation database (HGVbase) [14]. Other popular databases like the single-nucleotide polymorphism database (dbSNP) [15], the Protein Mutant Database (PMD) [16], and the database for nonsynonymous SNP's function prediction (dbNSFP) [9] are also important for collecting nsSNP data (also shown in Table 1).

The Online Mendelian Inheritance in Man (OMIM) is a powerful, comprehensive, and widely used database for collecting molecular relations between genetic variations and phenotypes. OMIM contains information of all known Mendelian disorders and their associated genes. Updated to October 23, 2012, OMIM has collected 21,458 entries of possible links between 4,753 phenotypes and over 12,000 genes, and 2,883 genes with phenotype-causing mutations.

The Human Gene Mutation Database (HGMD) records all germ-line disease-causing mutations and deleterious polymorphisms published in the literature. HGMD provides two versions of databases, one is for academic or nonprofit users, and the other is for professional usage. Updated to March 2012, the total mutation data collected in HGMD nonprofit version is 92,715, while the total mutation data in HGMD Professional version is 130,522.

The UniPROT/SWISS-PROT database is a high quality, manually curated, comprehensive protein sequence database, integrating information from the scientific literature and computational analysis. SWISS-PROT provides convincing protein sequences and annotations, such as protein function descriptions and domain structures. Updated to September 2012, UniProtKB/Swiss-Prot contains 538,010 sequence entries and 190,998,508 amino acids abstracted from 213,490 documents, including more than 67,000 nsSNPs.

The Human Genome Variation database (HGVbase) is an accurate, high-quality, and nonredundant database for comprehensive catalog of normal human gene and genome variation, especially SNPs. HGVbase provides both neutral polymorphisms and disease-related mutations. Updated to July 2005 (released 16.0), HGVbase contains 8,924,237 entries, including more than 20,000 coding SNPs and about 11,000 nsSNPs.

The single-nucleotide polymorphism database (dbSNP) is a comprehensive repository for single-nucleotide substitutions, short deletion, and insertion polymorphisms. Data in dbSNP can be combined with other available NCBI genomic data and freely downloaded in a variety of forms. Updated to February 2010, dbSNP has collected over 184 million

Table 1: Database for collecting nsSNP data.

Database	Website	Reference ID
Online Mendelian Inheritance in Man (OMIM)	http://www.omim.org/	[22]
Human Gene Mutation Database (HGMD)	http://www.hgmd.cf.ac.uk/ac/index.php	[12]
UniPROT/SWISS-PROT database	http://www.uniprot.org/	[13]
Human Genome Variation database (HGVbase)	http://hgvbase.cgb.ki.se	[14]
Single-nucleotide polymorphism database (dbSNP)	http://www.ncbi.nlm.nih.gov/snp	[15]
Protein Mutant Database (PMD)	http://pmd.ddbj.nig.ac.jp	[16]
Database for nonsynonymous SNPs' functional predictions (dbNSFP)	http://sites.google.com/site/jpopgen/dbNSFP	[9]

submissions representing more than 64 million distinct variants for 55 organisms, including more than 70,000 SNPs.

The Protein Mutant Database (PMD) [16] is a literature-based database for protein mutants, providing information of amino acid mutations at specific positions of proteins and the structural alterations. Each entry in the database corresponds to one article which may describe one or several protein mutants. Updated to 26 Mar 2007, PMD collects 45,239 entries and 218,873 mutants, including 54,975 nsSNPs occurring in 4,675 proteins.

The database for nonsynonymous SNPs' functional predictions (dbNSFP) [9] is a newly published database, providing both the information about nsSNPs and prediction scores from four popular algorithms (SIFT [17], PolyPhen-2 [18], LRT [19], and MutationTaster [5]) along with a conservation score (PhyloP) [10]. The dbNSFP is the first known integrated database of functional predictions from multiple algorithms for broad collection of human nsSNPs. Updated to March 27, 2009, dbNSFP includes a total of 75,931,005 entries, which covers 64,646,969 nsSNPs in the human genome.

3. Software Tools for Predicting Functional Implication of nsSNPs

With the accelerating advancement of high-throughput experimental techniques, annotations about functional elements in the human genome now become widely available; accordingly a variety of information can be used to study the deleteriousness of an nsSNP. A number of methods have been proposed for the prediction of deleterious nsSNPs, along with friendly web-based interactive software for users to facilitate their own research. In Table 2, we list eleven widely used tools, including SIFT [17], PolyPhen [2], SNAP [1], MSRV [11], LRT [19], PolyPhen-2 [18], MutationTaster [5], KGGSeq [23], SInBaD [21], GERP [24], and PhyloP [10]. The input data for a prediction tool usually requires the protein sequence or protein ID, the amino acid substitution, position of the substitution, chromosome, and/or sequence alignment. After providing all the required input data in the right format, the tools can run automatically and return the predication results, which are usually predictive scores ranging from 0 to 1.

Taking MSRV as an example, the input data for predicting a single amino acid substitution that results from a single base alternation in protein coding sequence includes the protein name, the amino acid substitution, and position of the substitution in protein sequence, and the output data includes the prediction score ranging from 0 to 1, where 0 stands for neutral nsSNP and 1 means deleterious nsSNP. For prioritizing multiple amino acid substitutions, users can directly paste their substation lists in the required format to the website or upload their data from local computer. The outputs are the ranking list containing all the attached substitution and their scores (as shown in Figure 1).

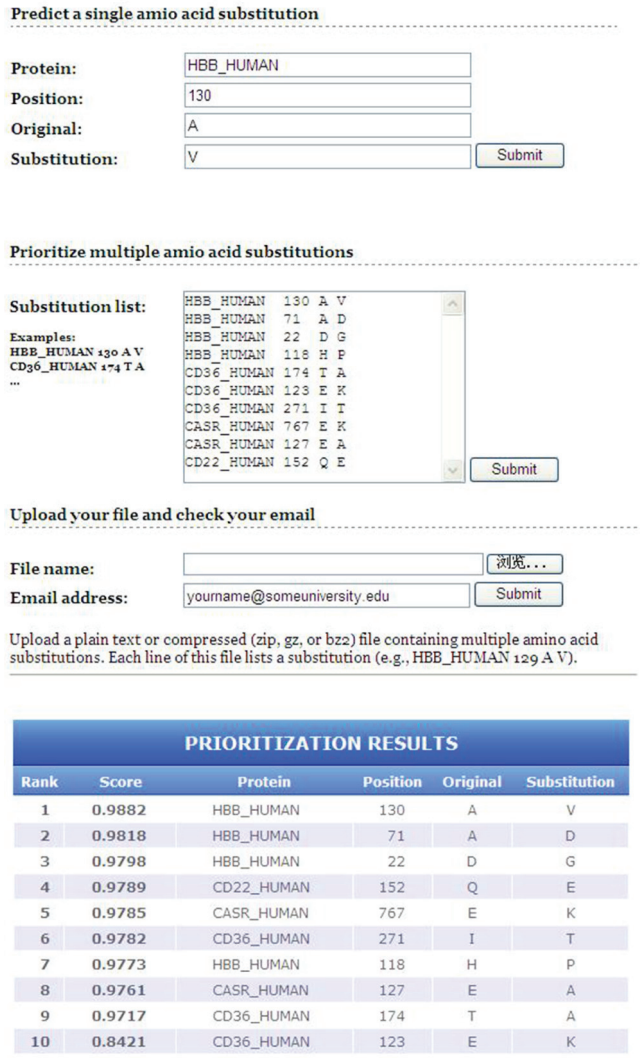


Figure 1: Web interface of MSRV.

Table 2: Tools for deleterious variant detection.

Method	Website	Features	Method description	Reference ID
SIFT	http://sift.bii.a-star.edu.sg/	Sequence based	Statistical method using PSSM with Dirichlet priors	[17]
PolyPhen	http://genetics.bwh.harvard.edu/pph/index.html	Sequence based, structure based, annotation	Rule-based model	[2]
SNAP	http://www.rostlab.org/services/SNAP/	Sequence based, annotation	Standard feed-forward neural networks with momentum term	[1]
MSRV	http://bioinfo.au.tsinghua.edu.cn/member/ruijiang/english/software.html	Sequence based	Multiple selection rule voting strategy using random forest	[11]
LRT	http://www.genetics.wustl.edu/jflab/lrt_query.html	Sequence based	Log ratio test	[19]
PolyPhen-2	http://genetics.bwh.harvard.edu/pph2/index.shtml	Sequence based, structure based	Naïve Bayes approach coupled with entropy-based discretization	[18]
MutationTaster	http://www.mutationtaster.org/	Sequence based, annotation	Naïve bayes model based on integrated data source	[5]
KGGSeq	http://statgenpro.psychiatry.hku.hk/limx/kggseq/	Sequence based, annotation	A three-level framework to combine a number of filtration and prioritization functions	[23]
SInBaD	http://tingchenlab.cmb.usc.edu/sinbad/	Sequence based	Separate mathematical models for promoters, exons, and introns, using logistic regression algorithm	[21]
GERP (score)	http://mendel.stanford.edu/sidowlab/downloads/gerp/index.html	Sequence based	A "Rejected Substitutions" score computation to infer the constrained region	[24]
PhyloP (score)	http://hgdownload.cse.ucsc.edu/goldenPath/hg18/phyloP44way	Sequence based	An exact P value computation under a continuous Markov substitution model	[10]

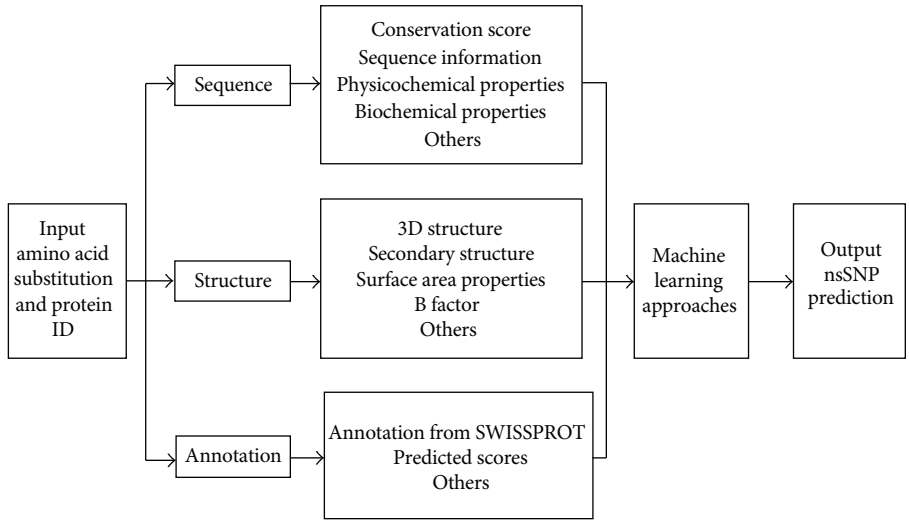


Figure 2: Typical procedure for deleterious nsSNPs detection.

Typically, the deleterious nsSNPs prediction problem is formulated as a binary classification model using diverse genomic data as features to compare the deleterious nsSNP with neutral nsSNP. The typical procedure is shown in Figure 2. Users should provide the information about protein ID or sequence, amino acid substitution, and/or multiple sequence alignment. After inputing all the required information, the classification tools can be implemented by extracting their own features and setting up the new classification model automatically. Finally, the deleterious score or the classification result may output by the tools. Classification features are collected and computed using sequence information, protein structural information, and/or annotations from known databases or prediction results. Sequence-based deleterious nsSNPs prediction methods usually take advantage of biochemical properties, physiochemical properties, sequence information, the evolutionary information of proteins, and the predicted 1D or 2D structure of proteins. Structure-based prediction methods may search a protein structure database and get some structural features for further classification. Annotation-based methods may take annotations from SWISS-PROT database [13] or use some published tools to get the preliminary scores for the query nsSNPs. In the next section, we focus on the eleven computational tools to analyze the deleterious nsSNPs prediction problem from the view of extracted features and classification methods.

4. Features for Characterizing nsSNPs

To fully capture diverse potential properties of deleterious nsSNPs, existing prediction tools take advantage of different types of features including sequencing-based information, structure-based information, and/or annotations to wholesomely carry out the classification of the deleterious nsSNPs from the neutral ones.

4.1. Sequencing-Based Information Provides the Strongest Signal for the Prediction Problem. Once a protein sequence containing the query nsSNP is provided, sequence-based deleterious nsSNP prediction methods calculate some specific features according to the sequence of the gene that contains the nsSNP and the location of the nsSNP in the DNA sequence, and/or look up in some databases to collect biochemical properties or physicochemical properties of the nsSNP or resulting single amino acid polymorphism. The most commonly utilized feature based on protein sequence for the query nsSNP is the conservation information calculated in different ways. Usually, people search the protein sequence against a sequence database to find sequences of homologous proteins. A multiple sequence alignment of the homologous sequences reveals what positions have been conserved throughout evolutionary time, and these positions are inferred to be important for function [8]. There are also many other ways to extract the classification features for nsSNPs according to the protein sequence where the nsSNPs locate [5, 25].

4.1.1. Conservation Scores. As an important feature for studying the deleteriousness of an nsSNP, the conservation score is used by most of prediction methods with their own way of calculation. The estimation of the deleteriousness of an nsSNP is based on the fact that sequences observed among living organisms are those that have not been removed by natural selection. In addition, comparative sequence analysis based on phylogenetic information by quantifying evolutionary changes in genes or genomes to find out the conserved positions that have evolved too slowly to be neutral can be identified [4]. Although evolutionary models may not identify all deleterious mutations, they provide a probabilistic framework in which the subset of deleterious mutations that disrupt highly conserved amino acid positions can be accurately identified [19].

Genome sequencing of a large number of closely related species makes it possible to develop better parameterized evolutionary models that more accurately predict human deleterious mutations [19]. Therefore, given a protein

sequence as input, a sequence database is needed to find homologous sequences for the protein. A multiple sequence alignment of the homologous sequences reveals what positions have been conserved throughout evolutionary time, and these positions are inferred to be important for function [8]. The conservation-based prediction method then scores each nsSNP based on the amino acid appearing in the multiple alignment and the severity of the amino acid change. An amino acid that is not present at the substitution site in the multiple alignment can still be predicted to be neutral if there are amino acids with similar physiochemical properties present in the alignment [8].

There are many ways to compute the conservation score for every query nsSNP. PolyPhen identifies homologues of the input sequences via a BLAST [26] search of the NRDB database and uses the new version of the PSIC (position-specific independent counts) software [27] to calculate the profile matrix, whose elements of the matrix (profile scores) are logarithmic ratios of the likelihood of a given amino acid occurring at a particular site to the likelihood of this amino acid occurring at any site (background frequency). PolyPhen computes the absolute value of the difference between profile scores of both allelic variants in the polymorphic position. Besides the PSIC score, PolyPhen-2 also uses the sequence identity to the closest homologue carrying any amino acid that differs from the wild-type allele at the site of the mutation, congruency of the mutant allele to the multiple alignment, and alignment depth (excluding gaps) at the site of the mutation. PhyloP performs an exact P value computation under a continuous Markov substitution model to compute the conservation score that measures interspecies conservation at each SNP position. MSRV provides an easy and effective way to calculate the conservation scores for the original and substitute amino acid, which are the frequencies of occurrences of the amino acids in the corresponding position of the Pfam multiple sequence alignment. The same features are also used by the MutationTaster algorithm and the SNAP algorithm. The LRT method utilizes the log likelihood ratio of the conserved relative to neutral model to measure the deleteriousness of an nsSNP, with the null model that each codon is evolving neutrally with no difference in the rate of nonsynonymous to synonymous substitution and the alternative model that the codon has evolved under negative selection with a free parameter for the nonsynonymous to synonymous ratio [19].

4.1.2. Sequence Information. Pure sequence information of the protein containing the nsSNP may offer useful indications that helped to identify the deleterious nsSNPs. Different methods adopt different ways to exhibit the usage of protein sequence. PolyPhen uses the characterization of the substitution site as a feature, while PolyPhen-2 employs CpG context of transition mutations. MutationTaster also computes a large number of features to grasp the potential difference between the deleterious nsSNPs and the nondeleterious nsSNPs, one of them is the length of protein, which checks if the resulting protein will be elongated, truncated, or whether nonsense-mediated mRNA decay is likely to occur, another is splice site analysis, which analyzes potential splice site changes.

4.1.3. Physicochemical Properties. It is believed that the physicochemical properties of proteins, especially the changes of physicochemical properties before and after amino acid changes, may present valuable information about how an amino acid substitution may lead to structural or functional changes of a protein. MSRV adopts six physicochemical properties of amino acids, including molecular weight, pI value, hydrophobicity scale, and relative frequencies for the occurrences of amino acids in the secondary structures (helices, strands, and turns) of proteins with known secondary structural information. Six properties are calculated under four situations that are the properties of the original amino acids, properties of the substituted amino acids, properties calculated in a window-sized situation that includes the neighbors of the original amino acids in the query protein sequence, and properties calculated in a column-weighted circumstance in which the query protein sequence is aligned with its homologous proteins. The authors also exploit three more situations which consider the property changes of the substitute amino acid from the original amino acid in a later published paper [20]. Results have shown that the changes of the physicochemical properties are more important than themselves when dealing with the deleterious nsSNP detection problem [20].

4.1.4. Biochemical Properties. Recent studies [28–31] have shown that deleterious substitutions are likely to affect protein structure; therefore, a better understanding about the protein biochemical properties of protein structure changes may accelerate the detection of deleterious nsSNPs. SNAP computes a series of biochemical properties and uses them as important features to construct classification models [1]. The properties contain several binary features, such as whether there is an inflexible proline into an alpha-helix, and some continuous features, such as mass of wild-type and mutant residues.

4.2. Structure-Based Information Facilitates the Prediction of Deleterious nsSNPs. Given a protein sequence data, structure-based deleterious nsSNPs prediction methods find the best match against a protein structure database. Some structural features are extracted using the information surrounding the site of substitution instead of detailed information at the atomic level; therefore, if there is not a perfect match for a query protein in the protein structure database, the structure of a homologous protein can be used.

Mapping of an nsSNP to a known 3D structure reveals whether the replacement is likely to destroy the hydrophobic property of a protein, electrostatic interactions, interactions with ligands, or other important features of a protein [2]. Structural features performed by PolyPhen are based on the use of several structural parameters suggested previously [32–34]. PolyPhen uses the Dictionary of Protein Secondary Structure (DSSP) database [35] to obtain some structural parameters for the mapped amino acid residues, such as secondary structure, solvent accessible surface area absolute value. The solvent accessible surface area (SASA) is the surface area of a molecule which is accessible to a solvent and

used to improve prediction of protein secondary structure [36, 37]. PolyPhen-2 refines the structural parameters using a feature selection mechanism and chooses three important structural features among thirteen candidate structural features. The selected features are normalized accessible surface area of amino acid residue, crystallographic beta-factor reflecting conformational mobility of the wild-type amino acid residue, and change in accessible surface area propensity for buried residues [18].

Methods solely based on protein structure features provide fewer predictions than methods using sequence-based features because there are far fewer protein structures than sequences for which homology can be found [8]. It is reported that the ratio of methods using sequence-based features to all the existing methods is as high as 81%, while the ratio for methods using only structure-based features is only 14% [8]. Independently consideration of isolated protein structure sometimes may lead to misleading prediction, because the proteins often interacted with others. Thus, new methods tend to use sequence-based information as the main features and structure-based information as the supporting features to operate the deleterious nsSNP detection problem.

4.3. Annotations Can Enhance the Prediction Power for Identifying Deleterious nsSNPs. Annotations can be used as supplementary features to enhance the prediction power for identifying deleterious nsSNPs. The SwissProt database annotates the positions of a protein that are located in the active site, involved in ligand binding, part of a disulfide bridge, or involved in other protein-protein interactions. Annotations can enhance the prediction power when incorporating with other features, such as sequence-based predictions of secondary structure and solvent accessibility [38, 39]. PolyPhen adds the SwissProt feature table terms to the final prediction rules, and MutationTaster and SNAP also utilize the SwissProt annotations as features to predict the deleteriousness of the query nsSNPs.

Besides annotations from published databases, the predicted deleterious score given by existing wide-accepted prediction tools can be treated as preliminary annotation for the prediction. For example, SNAP algorithm makes use of SIFT and PolyPhen prediction scores as classification features to enhance the prediction power. SNAP also determines whether the correct predictions made by their method overlapped with those covered by PolyPhen and SIFT [1].

5. Machine Learning Models for Classifying nsSNPs

Most predicting methods treat the identification of deleterious nsSNPs as a binary classification problem and adopt some popular binary classification machine learning algorithms, such as rule-based prediction model [2, 17], naïve Bayes classifier [5, 18], random forest [11], neural networks [1], and many others. After selecting suitable features, these prediction methods usually train and test on two types of datasets: a deleterious nsSNP set, which contains substitutions assumed to affect protein function, and a neutral set, which contains substitutions assumed to have no effect. During the training procedure, machine learning approaches are adopted to construct a classification and give a prediction score measuring the deleteriousness of an nsSNP. A prediction method should predict the substitutions in the deleterious nsSNP set to be damaging to protein function and predict the substitutions in the neutral set to be not related to protein function. Sometimes, a confident score is also provided to explain how confident the prediction result is. A bigger confident score means that the prediction is more approximate to the truth. Criteria to evaluate the prediction methods are mainly accuracy (ACC), false negative error rate (FN), and false positive rate (FP). False negative error rate is the percentage of nsSNP substitutions incorrectly predicted to be neutral, and false positive error is the percentage of neutral substitutions incorrectly predicted to affect protein function [8].

SIFT incorporates position-specific information by using sequence alignment and is intended specifically for predicting whether an amino acid substitution affects protein function. SIFT starts with a query protein sequence. Relying on the observation that proteins in the same subfamily have high conservation in conserved regions, SIFT selects sequences that are similar to the query sequence by adding the most similar sequence extracted from the PSI-BLAST results iteratively to the growing collection until conservation in the conserved regions decreases [17]. After the collection of similar sequences from multiple sequence alignment by PSI-BLAST, SIFT converts the alignment into a position-specific scoring matrix (PSSM) and calculates the probability of an amino acid appearing at a specified position. Using the position-specific probability estimation, SIFT assigns a decision rule to make the classification model. SIFT also provides a measure of confidence in the prediction. To assess confidence in the prediction, SIFT calculates a conservation value at each position in the alignment. PolyPhen also uses a rule-based decision mechanism to make the prediction for candidate nsSNPs. The rule is based on the analysis of the ability of various structural parameters and profile scores to discriminate between disease mutations and substitutions [2]. The rule-based prediction can be treated as prediction using decision trees, which belongs to the binary classification.

Quite different from PolyPhen, PolyPhen-2 adopts a naïve Bayes approach coupled with entropy-based discretization. The naïve Bayes approach can work as well as some machine learning approaches and contains only one parameter, which is Laplace estimators used for representing factored probabilities and smoothing [18]. MutationTaster also uses a naïve Bayes classifier, which predicts the potential candidate disease-associated nsSNPs. Different from other algorithms, MutationTaster chooses between three different prediction models, which are either aimed at "silent" synonymous or intronic alterations, at alterations affecting a single amino acid, or at alterations causing complex changes in the amino acid sequence.

MSRV provides a more realistic solution for identifying disease-associated nsSNPs. MSRV prioritizes mutations occurring in genetic regions to find those that are most likely to cause diseases. MSRV first partitions the training

set to 20 subsets according to different type of amino acid and utilizes a sequential forward feature selection method to choose the valuable features for each subset among extracted 26 physiochemical and conservation features. Then, MSRV trains a decision tree for each subset and takes advantage of random forest algorithm for the multiple selection strategy.

SNAP could potentially classify all nsSNPs in all proteins into deleterious (effect on function) and neutral (no effect) using sequence-based computationally acquired information alone. For each instance SNAP provides a reliability index, which is a well-calibrated measure reflecting the level of confidence of a particular prediction. SNAP uses an approximation of the rule of thumb for feature selection, and a standard feed-forward neural network with momentum term to build a classification model. SNAP also applies support vector machines (SVMs) for the prediction problem but receives a worse performance than a comparable neural network-based method.

6. Beyond Classification of nsSNPs

6.1. From Coding Mutations to Noncoding Mutations. Methods for predict deleteriousness of nsSNPs mainly focus on protein coding regions and conveniently use the properties derived from protein sequence or structure as classification features. Although nsSNPs in protein coding regions are important for studying the potential causative relationship between genetic variants and human inherited diseases, variants in intergenic regions, promoter regions, and intron regions can also strongly influence the phenotypic outcome [21]. In a recently published paper, Kjong-Van and Ting construct a new model named SInBaD (sequence-information-based decision-model) to evaluate any annotated human variant in all known exons, introns, splice junctions, and promoter regions. SInBaD uses nucleotide sequence conservation across multiple vertebrate species as features to find functional variants in regions other than just the coding regions. SinBaD builds three separate mathematical models for promoters, exons, and introns, using the human disease mutations annotated in human gene mutation database as the training dataset for functional variants. The authors perform deleterious variant analysis on four of the currently available individual human genomes and find out that there is considerable amount of predicted deleterious variants in promoter and intron region, especially the number of predicted deleterious variants in promoter region is almost 40% of the number of predicted deleterious variants in all regions.

Besides SinBaD, GERP also tries to overcome the limitation for noncoding mutation prediction. GERP identifies constrained elements in multiple alignments by quantifying substitution deficits, which represent that substitutions may occur if the element is neutral DNA and do not occur if the element is under functional constraint. These deficits, referred to as "Rejected Substitutions," are a natural measure of constraint that reflects the strength of past purifying selection on the element [24]. Although GERP is an algorithm to infer the constrained region, it gives each location a substitution rejection score which can be further used as a conservation score for identifying deleterious variants.

6.2. From Deleterious Classification to Disease-Specific Prioritization. The nsSNP deleterious prediction becomes more and more wholesome when using valuable feature information, sequence information, and annotations from known database. Though effective, all these methods formulate the identification of nsSNPs that are associated with diseases as a classification problem and give no information about what specific disease the nsSNP is associated with. Therefore, the classification results of these methods can only provide limited information to practical applications. For example, an nsSNP i with the highest deleterious prediction score may totally change the function of the corresponding protein and have a strong relation with disease A. However, it is impossible that the nsSNP i is strongly related to all the other diseases. Therefore, disease-specific prediction models are constructed according to the features of variants, information of diseases, known disease-related variants, or other available disease-specific information. Wu et al. use ensemble learning methods to construct a prediction mechanism for disease-specific nsSNPs identification [40] and demonstrate high accuracy of their method. A biological knowledge-based mining platform for genomic and genetic studies using sequence data (KGGSeq) is also a disease-specific prioritization method which makes effort to find the causal mutations for a particular Mendelian disease among millions of variants.

6.3. From Common Variant Detection to Rare Variant Detection. Recently, the popular common-disease common-variant (CDCV) hypothesis that assumes the etiology of common diseases is intervened by commonly occurring genetic variants with small to modest effects has been challenged by the fact that both common variants and rare mutations may be involved in the pathogenesis of common diseases. In fact, studies have already revealed that the presence of multiple rare variants may augment the risk of some diseases. Corresponding to these findings, a common-disease rare-variant (CDVR) hypothesis that indicates that multiple rare variants can also serve as the main factor to influence some common diseases has been proposed. Therefore, deleterious rare variant prediction becomes a new challenge.

KGGSeq is modeled to a comprehensive three-level framework to combine a number of filtrations and prioritization functions into one analysis procedure for exome sequencing-based discovery of human Mendelian disease genes. The framework is composed by several rules to filter and prioritize variants at three different levels: genetic level, variant-gene level, and knowledge level. KGGSeq can implement rare variant detection for Mendelian disease. During a rare variant detection, KGGSeq uses some genetic information and rules to filter the candidate variants, as well as a mechanism to delete common variants deposited in public databases (including the 1000 Genomes Project and NCBI dbSNP) as well as existing in the in-house datasets according to an adjustable allele frequency threshold. KGGSeq also

incorporates PPI, pathway, and literature information to narrow down the candidate rare variants.

6.4. From Single Prediction Score to Integration of Multiple Prediction Scores. More and more deleterious variant prediction methods are developed using different types of features and different training set. As each method has its own strength and weakness, it has been suggested that the combination of some of the prediction scores may enhance the accuracy for predicting a variant. Database for nonsynonymous SNPs' functional predictions (dbNSFP) follows this idea to first build an integrated database of functional predictions from multiple algorithms for the comprehensive collection of human nsSNPs. KGGSeq adopts the prediction scores from four popular algorithms (SIFT, PolyPhen-2, LRT, and MutationTaster) along with a conservation score (PhyloP) published by dbNSFP. KGGSeq uses these five scores as prediction features to train a logistic regression model and find these scores are in weak or moderate correlation. When individually operating the algorithm, MutationTaster outperforms than the other four prediction algorithms. In addition, the combination of predictions by all the five deleterious scores can provide better performance than individual scores as well as combined prediction by part of the deleteriousness scores.

7. Conclusions and Discussion

Deleterious variants detection becomes a more and more popular issue for research and guiding real experiment. In this paper we summarize the database for collecting nsSNP data, existing deleterious nsSNPs prediction methods, prediction features conducted in prediction model, and prediction algorithms to distinguish the deleterious nsSNPs. We discuss computational methods that use comparative genomics to predict deleteriousness in both coding and noncoding DNA, methods for disease-specific nsSNP detection, and methods for rare variant detection. We suggest using multiple prediction algorithms, as well as more available molecule level information may help to enhance the prediction power.

Although the prediction of deleterious nsSNPs seems to be more and more accurate when integrating more valuable information of nsSNPs, there still exist some challenges to deal with. Conservation scores are used by most of the prediction methods as main features to predict the functional effects of a candidate variant. However, not all the deleterious variants are in the constraint region or conserved among multiple sequence alignment. As a result, the nonconserved variants are difficult to identify using existing methods. Accuracy assessment is another problem. During the prediction, deleterious variants and neutral variants are collected from published database, such as OMIM and SWISSPROT. However, whether the so-called deleterious variants are really deleterious or not and whether the so-called neutral variants are really neutral or not may strongly affect the construction of predict model and the final accuracy measurement of the model. In addition, even if a variant is predicted to be deleterious with a strong confidence, the information about which disease the variant is related to and which disease the variant has a casual relation with is still missing. Facts show that variants in noncoding region can strongly influence the phenotypic outcome, and more algorithms for noncoding region deleterious variant detection using more available features besides conservation scores should be designed for further studying the casual relationship between noncoding variants and diseases. Furthermore, standard evaluation rules should be proposed for better comparing the existing deleterious variant prediction methods.

As a suggestion, more molecule level protein information or gene information should be merged with existing features to further strengthen the prediction power. As the protein products of genes responsible for the same or phenotypically similar disorders tend to physically interact with each other so as to carry out certain biological functions [23], PPI data could be considered. Genes sharing similar GO terms trend to have similar functions; thus, gene-gene similarity calculated using GO terms could provide another choice. Moreover, pathway information can be included based on the fact that causative genes of the same (or phenotypically similar) diseases are inclined to distribute within the same pathways.

Acknowledgments

This research was partially supported by the National Basic Research Program of China (2012CB316504), the National High Technology Research and Development Program of China (2012AA020401), the National Natural Science Foundation of China (61175002 and 60805010), the Tsinghua University Initiative Scientific Research Program, and the Tsinghua National Laboratory for Information Science and Technology (TNList) Academic Exchange Foundation, and the Open Research Fund of State Key Laboratory of Bioelectronics, Southeast University.

References

[1] Y. Bromberg and B. Rost, "SNAP: predict effect of non-synonymous polymorphisms on function," *Nucleic Acids Research*, vol. 35, no. 11, pp. 3823–3835, 2007.

[2] V. Ramensky, P. Bork, and S. Sunyaev, "Human non-synonymous SNPs: server and survey," *Nucleic Acids Research*, vol. 30, no. 17, pp. 3894–3900, 2002.

[3] J. Wu, M. Gan, and R. Jiang, "Prioritisation of candidate single amino acid polymorphisms using one-class learning machines," *International Journal of Computational Biology and Drug Design*, vol. 4, no. 4, pp. 316–331, 2011.

[4] G. M. Cooper and J. Shendure, "Needles in stacks of needles: finding disease-causal variants in a wealth of genomic data," *Nature Reviews Genetics*, vol. 12, pp. 628–640, 2011.

[5] J. M. Schwarz, C. Rödelsperger, M. Schuelke, and D. Seelow, "MutationTaster evaluates disease-causing potential of sequence alterations," *Nature Methods*, vol. 7, no. 8, pp. 575–576, 2010.

[6] S. Nakken, I. Alseth, and T. Rognes, "Computational prediction of the effects of non-synonymous single nucleotide polymorphisms in human DNA repair genes," *Neuroscience*, vol. 145, no. 4, pp. 1273–1279, 2007.

[7] M. Krawczak, E. V. Ball, I. Fenton et al., "Human gene mutation database—a biomedical information and research resource," *Human Mutation*, vol. 15, no. 1, pp. 45–51, 2000.

[8] P. C. Ng and S. Henikoff, "Predicting the effects of amino acid substitutions on protein function," *Annual Review of Genomics and Human Genetics*, vol. 7, pp. 61–80, 2006.

[9] X. Liu, X. Jian, and E. Boerwinkle, "dbNSFP: a lightweight database of human nonsynonymous SNPs and their functional predictions," *Human Mutation*, vol. 32, no. 8, pp. 894–899, 2011.

[10] A. Siepel, K. Pollard, and D. Haussler, "New methods for detecting lineage-specific selection," in *Proceedings of the 10th International Conference on Research in Computational Molecular Biology (RECOMB '06)*, pp. 190–205, Venice, Italy, 2006.

[11] R. Jiang, H. Yang, L. Zhou, C. C. J. Kuo, F. Sun, and T. Chen, "Sequence-based prioritization of nonsynonymous single-nucleotide polymorphisms for the study of disease mutations," *American Journal of Human Genetics*, vol. 81, no. 2, pp. 346–360, 2007.

[12] P. D. Stenson, E. V. Ball, M. Mort et al., "Human Gene Mutation Database (HGMD): 2003 update," *Human Mutation*, vol. 21, no. 6, pp. 577–581, 2003.

[13] T. U. Consortium, "The universal protein resource (UniProt) in 2010," *Nucleic Acids Research*, vol. 38, pp. D142–D148, 2010.

[14] D. Fredman, M. Siegfried, Y. P. Yuan, P. Bork, H. Lehväslaiho, and A. J. Brookes, "HGVbase: a human sequence variation database emphasizing data quality and a broad spectrum of data sources," *Nucleic Acids Research*, vol. 30, no. 1, pp. 387–391, 2002.

[15] S. T. Sherry, M. H. Ward, M. Kholodov et al., "DbSNP: the NCBI database of genetic variation," *Nucleic Acids Research*, vol. 29, no. 1, pp. 308–311, 2001.

[16] T. Kawabata, M. Ota, and K. Nishikawa, "The protein mutant database," *Nucleic Acids Research*, vol. 27, no. 1, pp. 355–357, 1999.

[17] P. Kumar, S. Henikoff, and P. C. Ng, "Predicting the effects of coding non-synonymous variants on protein function using the SIFT algorithm," *Nature Protocols*, vol. 4, no. 7, pp. 1073–1081, 2009.

[18] I. A. Adzhubei, S. Schmidt, L. Peshkin et al., "A method and server for predicting damaging missense mutations," *Nature Methods*, vol. 7, no. 4, pp. 248–249, 2010.

[19] S. Chun and J. C. Fay, "Identification of deleterious mutations within three human genomes," *Genome Research*, vol. 19, no. 9, pp. 1553–1561, 2009.

[20] W. Jiaxin, G. Mingxin, Z. Wangshu, and J. Rui, "Prediction of disease-associated single amino acid polymorphisms based on physiochemical features," *International Journal of Bioscience, Biochemistry and Bioinformatics*, vol. 1, no. 2, pp. 102–108, 2011.

[21] L. Kjong-Van and C. Ting, "Exploring functional variant discovery in non-coding regions with SInBaD," *Nucleic Acids Research*, vol. 41, no. 1, p. e7, 2013.

[22] A. Hamosh, A. F. Scott, J. S. Amberger, C. A. Bocchini, and V. A. McKusick, "Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders," *Nucleic Acids Research*, vol. 33, pp. D514–D517, 2005.

[23] M. X. Li, H. S. Gui, J. S. H. Kwan et al., "A comprehensive framework for prioritizing variants in exome sequencing studies of Mendelian diseases," *Nucleic Acids Research*, vol. 40, no. 7, p. e53, 2012.

[24] G. M. Cooper, D. L. Goode, S. B. Ng et al., "Single-nucleotide evolutionary constraint scores highlight disease-causing mutations," *Nature Methods*, vol. 7, no. 4, pp. 250–251, 2010.

[25] A. Schlessinger, G. Yachdav, and B. Rost, "PROFbval: predict flexible and rigid residues in proteins," *Bioinformatics*, vol. 22, no. 7, pp. 891–893, 2006.

[26] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of Molecular Biology*, vol. 215, no. 3, pp. 403–410, 1990.

[27] S. R. Sunyaev, F. Eisenhaber, I. V. Rodchenkov, B. Eisenhaber, V. G. Tumanyan, and E. N. Kuznetsov, "PSIC: profile extraction from sequence alignments with position-specific counts of independent observations," *Protein Engineering*, vol. 12, no. 5, pp. 387–394, 1999.

[28] D. Chasman and R. M. Adams, "Predicting the functional consequences of non-synonymous single nucleotide polymorphisms: structure-based assessment of amino acid variation," *Journal of Molecular Biology*, vol. 307, no. 2, pp. 683–706, 2001.

[29] P. C. Ng and S. Henikoff, "Predicting deleterious amino acid substitutions," *Genome Research*, vol. 11, pp. 863–874, 2001.

[30] P. Yue, Z. Li, and J. Moult, "Loss of protein structure stability as a major causative factor in monogenic disease," *Journal of Molecular Biology*, vol. 353, no. 2, pp. 459–473, 2005.

[31] S. R. Sunyaev, V. Ramensky, I. Koch, W. I. Lathe, A. S. Kondrashov, and P. Bork, "Prediction of deleterious human alleles," *Human Molecular Genetics*, vol. 10, no. 6, pp. 591–597, 2001.

[32] S. Sunyaev, V. Ramensky, I. Koch, W. Lathe III, A. S. Kondrashov, and P. Bork, "Prediction of deleterious human alleles," *Human Molecular Genetics*, vol. 10, no. 6, pp. 591–597, 2001.

[33] Z. Wang and J. Moult, "SNPs, protein structure, and disease," *Human Mutation*, vol. 17, no. 4, pp. 263–270, 2001.

[34] D. Chasman and R. M. Adams, "Predicting the functional consequences of non-synonymous single nucleotide polymorphisms: structure-based assessment of amino acid variation," *Journal of Molecular Biology*, vol. 307, no. 2, pp. 683–706, 2001.

[35] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, vol. 22, no. 12, pp. 2577–2637, 1983.

[36] A. Momen-Roknabadi, M. Sadeghi, H. Pezeshk, and S. A. Marashi, "Impact of residue accessible surface area on the prediction of protein secondary structures," *BMC Bioinformatics*, vol. 9, article 357, 2008.

[37] R. Adamczak, A. Porollo, and J. Meller, "Combining prediction of secondary structure and solvent accessibility in proteins," *Proteins*, vol. 59, no. 3, pp. 467–475, 2005.

[38] C. Ferrer-Costa, M. Orozco, and X. de la Cruz, "Sequence-based prediction of pathological mutations," *Proteins*, vol. 57, no. 4, pp. 811–819, 2004.

[39] F. Cambien, O. Poirier, V. Nicaud et al., "Sequence diversity in 36 candidate genes for cardiovascular disorders," *American Journal of Human Genetics*, vol. 65, no. 1, pp. 183–191, 1999.

[40] J. Wu, W. Zhang, and R. Jiang, "Comparative study of ensemble learning approaches in the identification of disease mutations," in *Proceedings of the 3rd International Conference on Biomedical Engineering and Informatics (BMEI '10)*, vol. 6, pp. 2306–2310, Yantai, China, October 2010.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 197406, 9 pages
http://dx.doi.org/10.1155/2013/197406

Research Article

A Local Genetic Algorithm for the Identification of Condition-Specific MicroRNA-Gene Modules

Wenbo Mu, Damian Roqueiro, and Yang Dai

Department of Bioengineering, University of Illinois at Chicago, Chicago, IL 60607, USA

Correspondence should be addressed to Yang Dai; yangdai@uic.edu

Received 12 November 2012; Accepted 17 December 2012

Academic Editors: R. Jiang, W. Tian, J. Wan, and X. Zhao

Transcription factor and microRNA are two types of key regulators of gene expression. Their regulatory mechanisms are highly complex. In this study, we propose a computational method to predict condition-specific regulatory modules that consist of microRNAs, transcription factors, and their commonly regulated genes. We used matched global expression profiles of mRNAs and microRNAs together with the predicted targets of transcription factors and microRNAs to construct an underlying regulatory network. Our method searches for highly scored modules from the network based on a two-step heuristic method that combines genetic and local search algorithms. Using two matched expression datasets, we demonstrate that our method can identify highly scored modules with statistical significance and biological relevance. The identified regulatory modules may provide useful insights on the mechanisms of transcription factors and microRNAs.

1. Introduction

Transcription factors (TFs) and microRNAs exert a widespread impact on gene expression. Most genes in genome are regulated by the TFs, which account for about 10% of the protein-coding genes in humans and mice [1]. TFs function by interacting with genomic cis-regulatory DNA elements. MicroRNAs primarily bind to regulatory elements located in the 3′ untranslated region (3′UTR) of their target mRNAs. There are more than 1000 microRNAs, which target 60% of protein-encoding genes in the human genome, and each microRNA regulates about 200 transcripts (miRBase 2011 [2]). The identification of TF and microRNA targets is a key in understanding their roles in gene regulation. However, it is a laborious task. The availability of large amount of matched condition-specific microRNA and mRNA expression data for a specific cell or tissue type has provided a good resource for the prediction of microRNA functional target. Various methods using matched expression profiles coupled with sequence-based predictions of targets of microRNAs have been proposed [3]. On the other hand, the interplay between TFs and microRNAs was recently recognized [4]. However, there are only a limited number of integrated analysis tools [5–7]. Integrated analysis tools for identifying functional regulatory modules involving microRNAs and TFs targets are still needed.

2. Materials and Methods

The proposed method starts with a matched global mRNA and microRNA expression dataset; that is, mRNA and microRNA expression levels were measured from the same sample. The method consists of four steps. (1) Perform differential expression analyses for microRNA and mRNA profiles. (2) Calculate correlations of expression for pairs of microRNAs, pairs of mRNAs, and pairs of microRNAs and mRNAs. (3) Predict TF and microRNA targets. (4) Predict microRNA-gene modules based on the information obtained from (1) to (3) by a heuristic method, which is the combination of a genetic algorithm and a local search. The framework of our proposed method is presented in Figure 1.

2.1. Datasets and Preprocessing. Two datasets were used in our study. The first dataset contains the expression profiles of 98 primary cancer, 13 metastatic cancer, and 28 normal prostate samples [8]. The mRNA expression profiles were measured using the Affymetrix Human Exon 1.0 ST

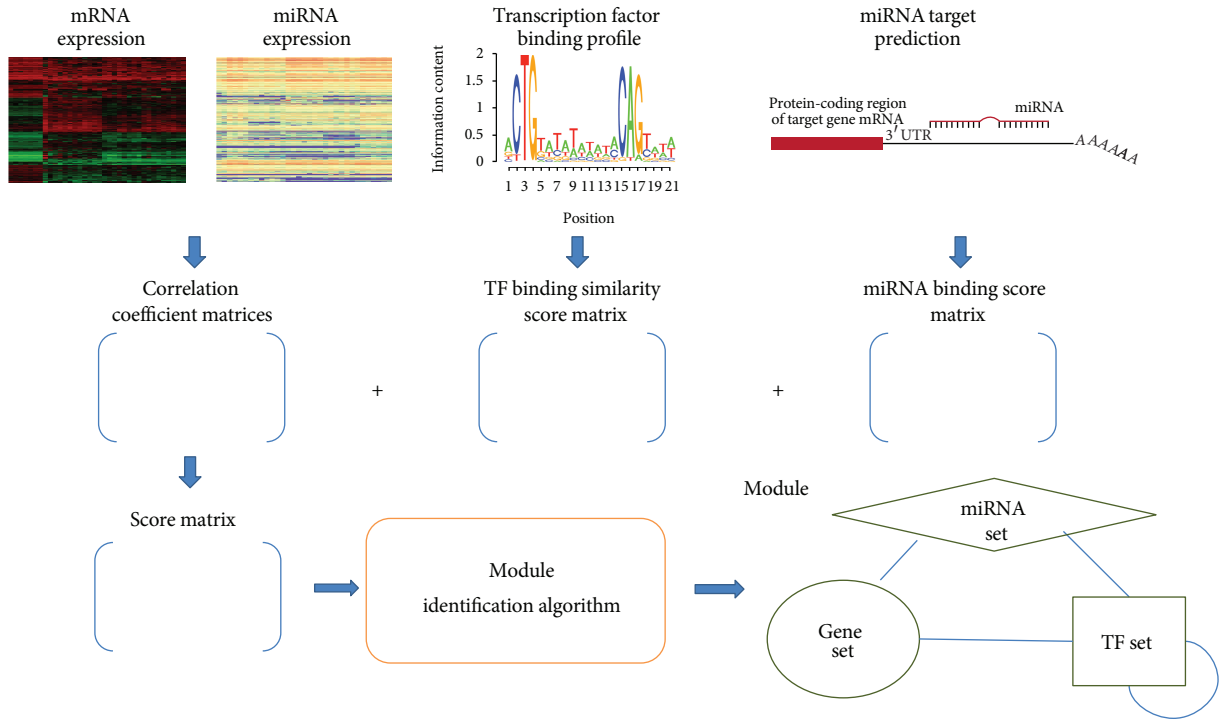


Figure 1: Method scheme.

Array which includes 26,447 mRNAs, and the microRNA expression profiles were measured by the Agilent Human microRNA Microarray 2.0 which includes 368 microRNAs. The normalized data were obtained from the NCBI Gene Expression Omnibus (GEO) [9] through GEO accession number GSE21032. The second dataset includes a wide variety of tumor and normal tissue types: 218 tumor samples of 14 common tumor types and 90 normal tissue samples [10]. The mRNA expression profiles were measured with the Affymetrix Hu6800 and the Hu35KsubA Genechips and contained 16,063 genes. The corresponding microRNA expression profiles were measured with the bead-based flow cytometric microRNA expression profiling method on 217 mammalian microRNAs and 334 samples [11]. Among them, 68 cancer tissue samples on 11 tumor types and 21 normal samples have both mRNA expression profile and microRNA expression profile. These matched profiles were selected in our study. The normalized and log2-transformed data were obtained from the Broad Institute website (http://www.broad.mit.edu/cancer/pub/migcm/).

The differential expression analysis was performed on both mRNA and microRNA expression profiles. Prior to the analysis, 25% probes with the lowest variation (measured by coefficient of variation) for both mRNAs and microRNAs were discarded. The differential expression analysis was performed using *limma* package in Bioconductor, and the false discovery rate (FDR) was controlled by adjusting P values based on the Benjamini and Hochberg multiple testing procedure [12]. Since functional TFs are not necessarily differentially expressed, all genes whose protein products are TFs (TF genes) were kept in our analysis. For the rest of the genes (nTF genes), a stringent cutoff of 0.001 for the adjusted P values was applied. Since a slight change of microRNA expresses can affect gene expression drastically, microRNAs with the adjusted P values less than 0.05 were defined as differentially expressed.

Pearson correlation coefficients (PCCs) were used to measure correlations of expression of (1) mRNA pairs, (2) microRNA pairs, and (3) mRNA-microRNA pairs. A permutation test on PCCs was employed for significance analysis. Specifically, random expression profiles were generated by shuffling the mRNA labels in the original datasets for 10,000 times, and the PCC was recalculated for each shuffled dataset. The P value was determined as the percentage of times that the PCC obtained from a shuffled dataset exceeded that obtained from the observed data.

Predicted microRNA targets were retrieved from the http://www.microRNA.org/ website, which provides access to the comprehensive database of predicted and experimentally validated microRNA targets [13–15]. The predicted targets for the conserved microRNAs with P value less than 0.05 were selected, resulting in a final set of 879,049 microRNA-gene pairs. The corresponding alignment scores associated with the microRNA targets were scaled to $(0, 1)$.

The predicted transcription factor binding sites (TFBSs) were obtained by mapping position weight matrices (PWMs) from TRANSFAC (ver. 2010.1) [16] of transcription factors to the promoter regions of genes using the MATCH algorithm [17]. We defined 10 KB upstream and 2 KB downstream of the transcription start site (TSS) as the promoter region of a gene. TFBSs were obtained from bindSDb [18], a database

Input: Parameter list of the genetic algorithm
Parameter list of local search
Score matrix *ScoreM*.
Number of Modules *ModN*
Output: Module chromosomes *M*
Formal steps:
(1) Set $m = 0$
(2) **while** m != *ModN* **do**
(3) Perform the genetic algorithm to identify co-expressed gene set and save to *Gco*
(4) Apply the local search to *Gco* and save solution to *M*
(5) Update PCC matrix
(6) **end while**
(7) Return *M*

Algorithm 1: Pseudocode for module identification.

developed to store both experimentally validated and predicted TFBSs based on the RefSeq gene information from the UCSC RefSeq track of the Human Genome Assembly (hg19) and the NCBI mRNA annotations. In case there are multiple PWMs for a TF, the maximum alignment score of all its PWMs to the predicted TFBSs was used to determine the unique relation between the TF and its multiple PWMs. The matching information between a TF and its gene symbol was obtained from TRANSFAC. Even with the stringent threshold for the alignment scores, the MATCH algorithm still produced a large number of TFBSs, among which many may be false positives. To reduce the number of false positives, we applied a cutoff value (described later) on the similarity scores to reduce the number of interactions significantly without losing too much information.

2.2. Proposed Algorithm. Our module identification method consists of two steps. (1) Identify coexpressed gene sets which include TF genes and nTF genes by the genetic algorithm (GA). This step located the highly plausible region of "good" solution in the searching space. (2) Search coregulators for the coexpressed gene sets obtained by the GA using the local search algorithm. All direct regulators of genes were candidates for the local search. In order to guarantee no duplicated modules to be considered in the future generations, after a module was identified from the local search, the correlation coefficient matrix of mRNAs was updated by removing the pairs involving the mRNAs in the current module. The pseudocode of our algorithm is given in Algorithm 1.

2.2.1. Design of the Genetic Algorithm. A binary string of fixed length was used to represent a chromosome, that is, a candidate of coexpressed gene sets in the GA. The value 1 stands for the gene included in the set and 0 for otherwise. Three setups with different percentages of genes included in the initial chromosomes were considered: 2%, 20%, and 80% of total genes. The roulette wheel selection was used for the selection of parent chromosomes for producing offspring. For the selected parents, the crossover was carried out separately for TF genes and nTF genes. The crossover probability P_{co} was in the range of (0.5–0.9) with an incremental size of 0.1. The mutation probability P_{mu} was varied at four values: 0.00001, 0.0001, 0.001, and 0.01. In addition to these genetic operators, randomly generated chromosomes were introduced as new immigrants into the population pool to substitute the worst chromosome at each generation. Three immigration rates, 0.01, 0.001, and 0.0001, were considered.

The average of the absolute PCCs over all pairs of genes included in a chromosome was defined as the fitness score of the chromosome. Two termination conditions were considered: 5,000 generations limitation or the highest fitness score remains unchanged for 200 generations.

2.2.2. Design of Local Search Algorithm. After the best coexpressed gene set was obtained from the GA, the candidates for the local search were determined to be all regulators (microRNAs and TFs) that were either predicted to target the genes in the coexpressed gene set or had significant PCCs with them. The initial solution for the local search was constructed by the TF genes in coexpressed genes and the randomly added 1% microRNAs from the candidate pool of regulators. The fitness score of a local search solution, or module, was defined as follows.

Let M' and T' represent the set of microRNAs and TF genes in the module, respectively, G' the union of both TF genes and nTF genes, N the total number of interactions among the members in the module.

Define MGI as a score for the predicted targeting interactions between microRNAs and genes; MS_{ij} and Cor_{ij} as the binding score and the correlation coefficient between microRNA i and gene j, respectively:

$$\text{MGI} = \sum_{i \in M'} \sum_{j \in G'} \left(k_1 \text{MS}_{ij} + k_2 \left| \text{Cor}_{ij} \right| \right). \quad (1)$$

Here k_1 and k_2 are two parameters. In our study we used $k_2 = 1$ and $k_1 = 1, 2, 3$.

Define TGI as a score for the predicted target interactions between TF genes and all genes; TS_{ij} and Cor_{ij} as the binding

score and correlation coefficient between TF-gene i and nTF-gene j, respectively:

$$\mathrm{TGI} = \sum_{i \in T'} \sum_{j \in G'} \left(k_1 \mathrm{TS}_{ij} + k_2 \left| \mathrm{Cor}_{ij} \right| \right). \quad (2)$$

The total PCCs among microRNAs in M' were denoted by $\mathrm{Cor}_{M'}$:

$$\mathrm{Cor}_{M'} = \sum_{i,j \in M'} \sum_{i \neq j} \left| \mathrm{Cor}_{ij} \right|. \quad (3)$$

To prevent the size of modules from unlimited increasing, the fitness score for a module was defined as the averaged value over the four sets of interaction scores described above:

$$F = \frac{\mathrm{MGI} + \mathrm{TGI} + \mathrm{Cor}_{M'}}{N}. \quad (4)$$

The interaction scores of TF-gene and microRNA-gene and all absolute PCCs were further scaled in the range of (0.5–1). The local search was terminated either when it reached 1000 iterations or the fitness scores remained unchanged for 100 iterations.

At each iteration of the local search, a local change to either microRNAs or TFs was made. For the user's convenience, we added a user option that specifies a preferred size of regulators in local search, since in most circumstances a user may be only interested in several most important regulators. For study reported here, the numbers of microRNAs and TFs in the modules are controlled at less than 1% and 4% of candidate regulators, respectively. After microRNAs/TF genes were determined to change, a microRNA/TF-gene was chosen from all candidates if the restriction of size had not been reached. A chosen microRNA/TF-gene was removed from the solution if it was already in the solution. If the number of the current regulators in the solution had reached the limit, a microRNA/TF-gene in the candidate searching space but not belonging to the current solution was chosen to substitute one microRNA/TF-gene in the current solution.

2.3. Validation and Evaluation Criteria. In order to evaluate the overall quality of the identified modules, we defined a score by combining the fitness measurements used in the GA and local search. In addition to the fitness measurement used in local search, a term of total correlation coefficients among nTF genes in the module, $\mathrm{Cor}_{R'}$, was added:

$$\mathrm{Cor}_{R'} = \sum_{i,j \in R'} \sum_{i \neq j} \left| \mathrm{Cor}_{ij} \right|. \quad (5)$$

The final score for an identified module was defined as below:

$$F = \frac{\mathrm{MGI} + \mathrm{TGI} + \mathrm{Cor}_{M'} + \mathrm{Cor}_{R'}}{N}, \quad (6)$$

where N is the total number of interactions among the members in the module.

In order to show our method can successfully identify modules with high fitness scores, we compared specific scores of randomly generated modules with the identified modules. For each module, 1,000 randomized controls were generated and each control has the identical number of microRNAs, TF genes, and nTF genes with the identified modules. To evaluate the significance of our modules, we performed the permutation test for each module to determine P values. For each module at each permutation, a number of microRNAs/genes in module were substituted by the same number of randomly selected microRNAs/genes. The size of substitutions follows a discrete uniform distribution between 0 and the number of genes for each identified module. The P value was evaluated by the chance of obtaining a permutated module better than the original one. To evaluate the biological relevance of our modules, we performed the enrichment analyses for gene ontology (GO) terms and KEGG pathways for the identified modules using DAVID [19].

3. Results and Discussion

In this section, we first show how to determine the parameter values in our algorithm using Dataset I. Subsequently, we present the predicted modules based on the determined parameters for Dataset I. Most of the results were derived based on $k_1 = k_2 = 1$ unless otherwise is specified.

We identified 1,933 differentially expressed nTF genes and 144 differentially expressed microRNAs for Dataset I. These 1,933 nTF genes, 189 TF genes with mRNA measurements, and 144 microRNAs were used to calculate PCCs of their expression levels. Only those PCCs with P value less than 0.0001 were considered to be significant and were retained for the subsequence analysis (See Table S1 in Supplementary Material available online at http://dx.doi.org/10.1155/2013/197406.)

To determine the cutoff value on the TFBS similarity scores, we checked the effect of different thresholds on the predicted number of TF-gene pairs. A total of 16,292,671 alignments between PWMs and TFBSs were obtained from bindSDb based on the TRANSFAC threshold for the minimum false positives, and 3,469,371 TF-gene pairs were specified after determining the unique TFBS for a TF as described in Section 2. The different numbers of predicted TF-gene pairs and the numbers of involved TFs based on different thresholds for similarity scores were summarized in Table S2. We applied a cutoff 0.99 for the similarity scores, which significantly reduced the number of predicted pairs without drastically changing the numbers of TFs and target genes. Finally 1,705,837 predicted pairs between 260 TFs and 21,054 genes were retained for the module identification.

3.1. Determination of GA Parameters. We examined the average sizes of coexpressed gene sets obtained from the GA at three different sizes for the initial chromosomes setups, that is, inclusion of 2%, 20%, and 80% of genes. The average sizes of the coexpressed gene sets obtained from the GA were 54, 224, and 401, respectively. However, in the latter two cases, the fitness scores are far from converging at the termination. Therefore, we set the initial chromosomes with only 2% of randomly selected genes.

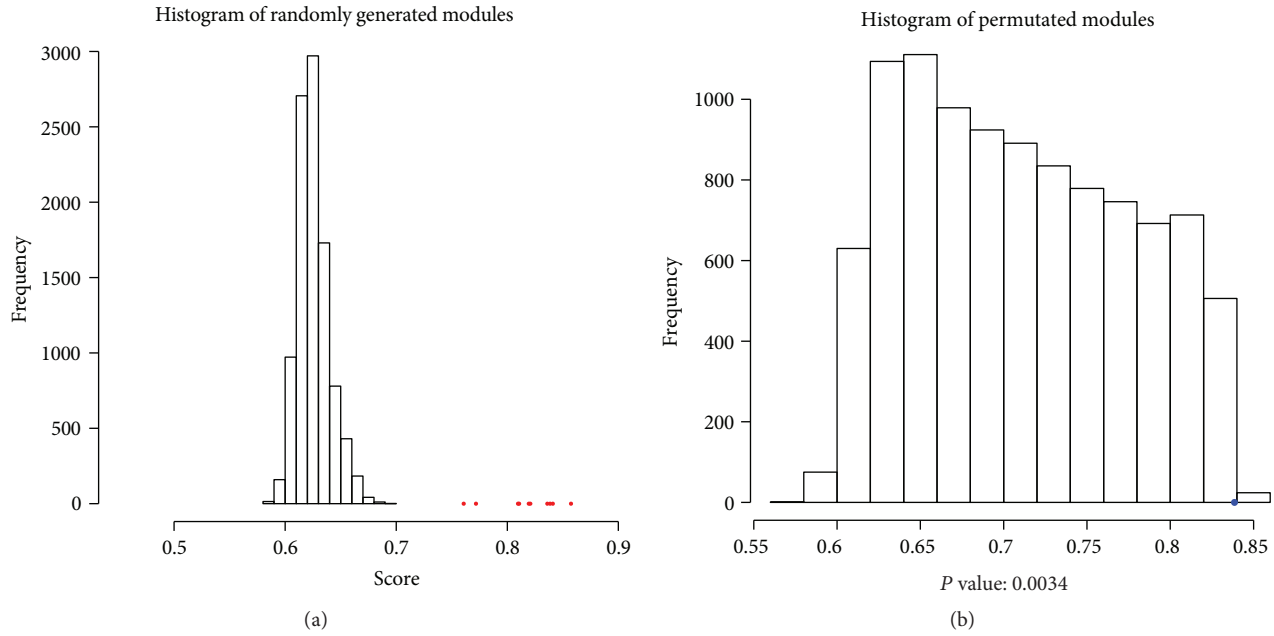


Figure 2: Histogram of control scores: (a) randomly generated modules; (b) permutated modules for module 1.

The proper choice of values for P_{co}, P_{mu}, and P_{new} is important to the performance of a GA. To find the good value for each genetic operator, we ran the GA by changing the value of one operator while keeping the other two fixed. For each value of a specific operator, we ran genetic algorithm for 10 times, 1000 generations each, and evaluated the performance by convergence rate. The convergence rate was defined as average incensement of fitness score per iteration. The GA performed better with $P_{co} = 0.7$, $P_{mu} = 0.001$, and $P_{new} = 0.01$ (Table S3). We used these values for the subsequent analysis.

3.2. Evaluation of Local Search. In order to demonstrate that the local search can find a local optimal solution, we recorded the start and end scores and calculated the convergence rate of the scores. The results for 10 modules (Figure S1) show that the local search did improve the fitness score and locate the local optimal solutions efficiently.

3.3. Module Evaluation. Figure 2(a) shows the histogram of fitness scores for 10,000 randomized modules and modules identified by our method (red dots). It suggests that our method was able to successfully identify modules with significantly higher scores. The identified modules, the corresponding scores, and the P values were listed in Table S4(a) (Supplementary file). All the modules were significant with P values less than 0.005 based on the permutation test. Figure 2(b) shows the distribution of scores for the 10,000 permuted modules of module 1. It indicates that the local optimal solution was found by our method.

Table 1 provides a summary of the 10 regulatory modules found by our method. The interactions were divided into three categories based on the evidence of support: sequence-based binding prediction only, PCC only, and both. Most interactions predicted by sequence information also have significant PCCs, indicating the direct regulations. However, considerable fractions of interactions in the modules only have PCC support, implying indirect regulation between the regulators and targets.

The details of genes and microRNAs in the identified modules, enriched KEGG pathways and GO terms (adjusted $P < 0.01$) were included in Tables S4(b) and S4(c) (Supplementary File). The enriched GO terms that annotate at least 5 genes were summarized. Compared to the results of enrichment analysis for the modules identified with a lasso model for the same dataset [20], most of the common KEGG pathways related to cancers were found, including focal adhesion, MAPK signaling pathway, hypertrophic cardiomyopathy, vascular smooth muscle contraction, regulation of actin cytoskeleton, pathways in cancer, and Wnt signaling pathway.

3.4. Control of Interaction Types in the Predicted Modules. The definition of the fitness score is a key factor to control the type of interactions one wishes to include in the modules. In the previous section we reported the results when an equal weight, that is, $k_1 = k_2 = 1$, was imposed on the alignment scores of TFs/microRNAs and the correlation coefficients of expression in the fitness function. We examined if the increase of the weight on the alignment scores could lead to the increase of the number of interactions with support from both the predicted binding and significant PCC values. We performed the experiment using $k_2 = 2, 3$ and $k_1 = 1$. It can be observed that an increasing trend in the proportion of interactions was supported by the predicted binding and expression correlation between the regulators and targets

Table 1: Summary of regulatory interactions in the 10 predicted modules for Dataset I.

Module ID	# Nodes[a]	# Interactions[b]	# PCC and Binding[c]	# PCC[d]	# Binding[e]
1	3/7/22	264	53	197	14
2	3/3/36	704	33	665	6
3	3/7/39	823	60	751	12
4	3/15/21	384	135	228	21
5	3/7/17	233	74	149	10
6	3/3/49	1284	7	1273	4
7	3/4/46	1181	42	1127	12
8	3/7/42	988	99	877	12
9	3/7/49	1316	74	1232	10
10	3/7/53	1431	83	1339	9

[a]The numbers of miRNAs, TF-genes, and nTF-genes.
[b]The number of interactions.
[c]The number of interactions with support of both significant PCC and predicted binding.
[d]The number of interactions with support of only significant PCC.
[e]The number of interactions with support of only predicted binding.

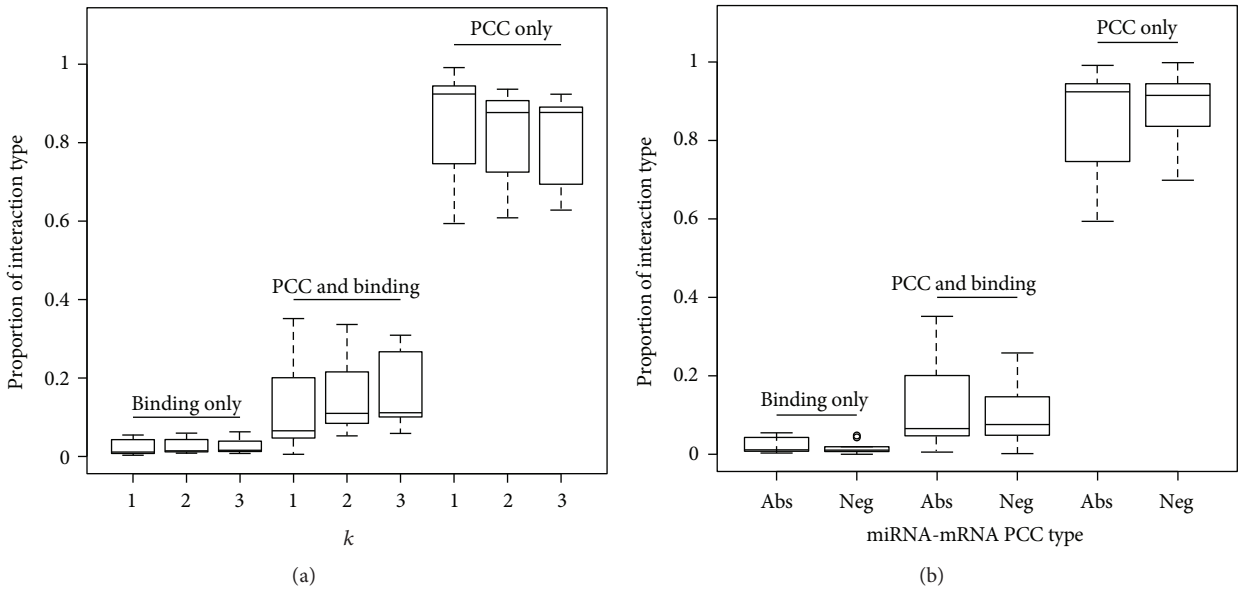


Figure 3: Boxplots of the proportion of three interaction types in the identified 10 modules with definitions for the fitness score. (a) The three boxplots in each type represent the results for ($k_1 = 1, k_2 = 1$), ($k_1 = 2, k_2 = 1$), and ($k_1 = 3, k_2 = 1$), respectively. (b) The two boxplots in each type represent the results using (1) both positive and negative microRNA-mRNA PCCs and (2) negative microRNA-mRNA PCCs, respectively.

(Figure 3(a)) when k_2 increases. This result shows the flexibility of our method in finding regulatory modules according to user's preference on the interaction types.

We also examined the ability of our method in finding regulatory modules when only including microRNAs that were negatively correlated with the predicted genes in the coexpressed set in the local search step. Our algorithm was able to successfully identify significant modules (Supplementary File 1). Compared with the case where both negatively and positively expressed microRNA regulators were considered in a module, there was a slight increase in the proportion of the interaction type with support from both predicted binding and significant PCCs (Figure 3(b)).

3.5. Literature Validation. The interactions in the identified module 1 to module 10 were shown in Figure 4 and Figures S2 and S3. In module 1, no microRNAs genes become isolated, and the main network structure is not changed after removing those predicted by the PCC interactions. In module 10, however, the targets of MEIS1 become isolated, and many potential regulatory relationships between MEIS1 and target genes also disappear after removing the PCC predicted interactions. MEIS1, which encodes a homeobox protein belonging to the TALE "three amino acid loop extension" family of homeodomain-containing proteins, as well as MEIS2 and PBX1 are found to have a critical function to suppress prostate cancer initiation and progression [21].

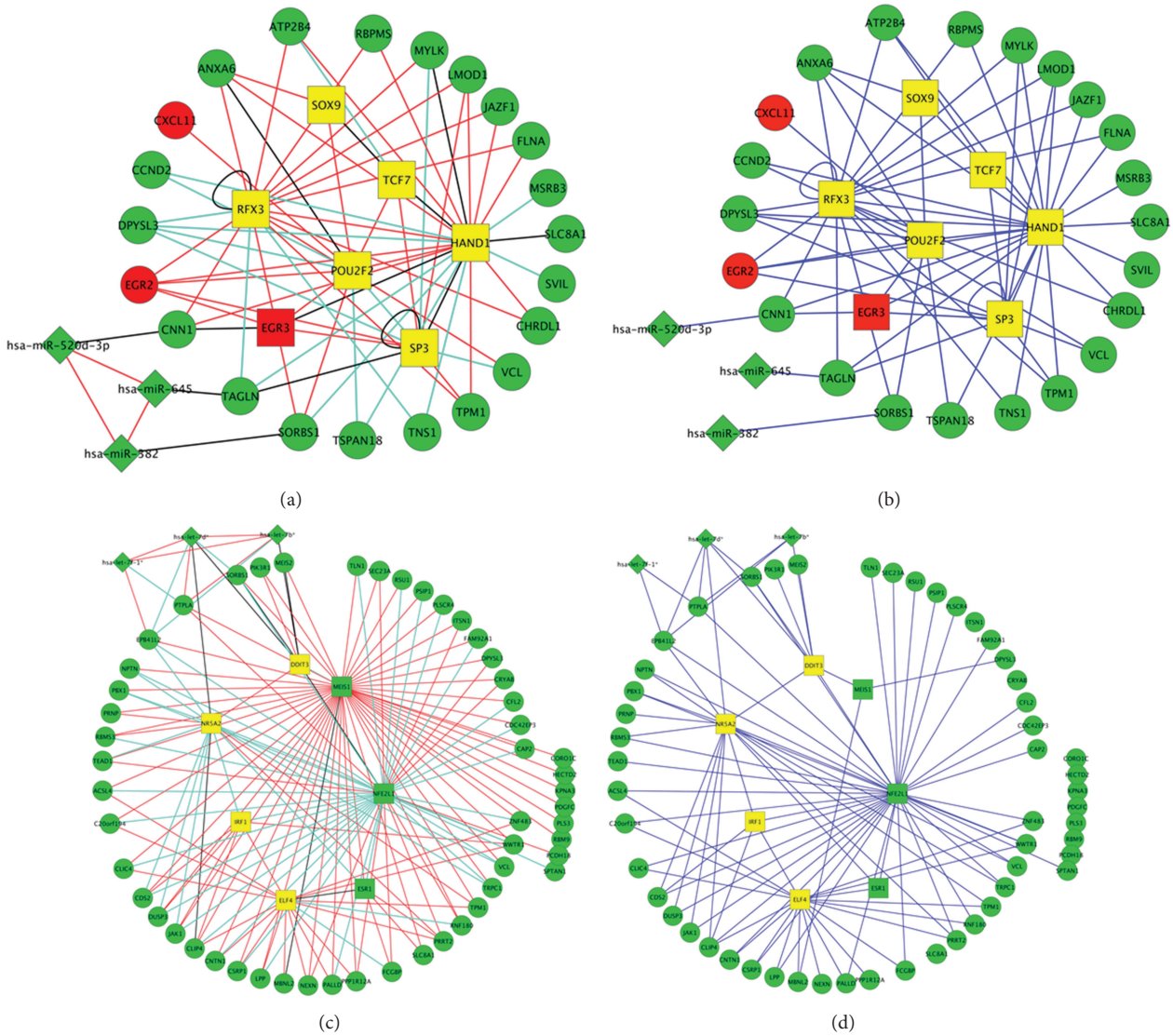


FIGURE 4: Two types of visualization of selected modules: (a) general representation of module 1; (b) sequence-based predictions only of module 1; (c) general representation of module 10; (d) sequence-based predictions only of module 10. Diamond, rectangle, and eclipse represent microRNA, TF genes, and nTF genes, respectively. Red nodes and green nodes represent overexpressed and underexpressed microRNAs/genes in tumor samples. Red lines and light green lines stand for positive correlations and negative correlations, respectively, while interactions that were predicted only by sequence information are drawn as black lines. For clear visualization, the links between nTF genes were not plotted.

The difference between Figures 4(c) and 4(d) suggests that MEIS1 may be a coactivator to regulate genes without directly binding to the promoters of the targets.

We also explored the literatures about other core regulators and regulatory relationships in identified modules. For example, hsa-miR-7f-1, which was identified as a core regulator in both modules 8 and 10, was found to be associated with lung cancer, breast cancer, colorectal cancer [22], pancreatic cancer [23], and pituitary adenomas [24]. Another microRNA, hsa-miR-328, identified in modules 1, 3, 7, and 9, was found to be dysregulated in both breast cancer [25] and colorectal cancer [26]. But their functional mechanisms to cancer development are still unknown. Our predicted modules may be used to facilitate further experiments for functional study.

Our method also identified several important TFs and their regulatory relationships, such as EGR3, RFX3 and MYLK. EGR3 was found overexpressed in tumor cells and was identified as a core regulator in modules 1, 3, and 10. The protein encoded by EGR3 participates in the transcriptional regulation of genes in controlling biological rhythm and may also play a role in a wide variety of processes including muscle development, lymphocyte development, endothelial cell growth, and migration and neuronal development (Ref-Seq December 2010). EGR3 was found to be closely associated with the genesis and malignant progression of

breast cancer by being involved in the estrogen-signaling pathway. Recently it was shown that EGR3 plays a pivot role in mechanism of prostate cancer initiation or early progression [27]. RFX3 is a transcriptional activator with highly conserved winged helix DNA-binding domain and can bind DNA as a monomer or as a heterodimer with other RFX family members. Its function in prostate cancer has not been well explored, but its regulation on the same set of genes together with MEIS1 in modules 7 and 9 suggests it may present as a coregulator with MEIS1 to be functional. MYLK was known to be involved in many biological processes including the inflammatory response (e.g., apoptosis, vascular permeability, and leukocyte diapedesis), cell motility and morphology and MARK signaling pathway. It was identified to be coregulated by MEIS1 and RFX3 in modules 7 and 9. It was involved in a total of 5 modules, suggesting its importance in cancer development, especially prostate cancer. A thorough literature search on all of the predicted interactions and core regulators for prostate cancer is not possible here. However, we demonstrated that our method is likely to be useful for identifying functional regulatory modules in specific diseases.

3.6. Prediction in Dataset II. Since Dataset II includes expression on a wide variety of tissue and normal samples, we applied our method to identify cancer-related common regulatory modules. Because of the multiple cell types and intrinsic complication of tumor cellular environment, we used the same procedure for differential expression analysis with a relatively loose cutoff for P values. The threshold of 0.05 for the adjusted P values was applied to the nTF genes and microRNAs. All TF genes were retained. This step resulted in 94 microRNAs, 162 TF genes, and 1,410 nTF genes. The sequence-based prediction for TF and microRNA targets led to a set of 74 microRNAs, 148 TF genes, and 1,194 nTF genes for module identification. We performed the same test to determine the optimal values for genetic operators. Crossover probability 0.7, mutation probability 0.001, and random immigrant probability 0.01 were obtained. These values were the same as those used for Dataset I, showing that the choice of parameters was not biased to a particular dataset.

All 10 modules achieved the significance level based on our permutation test. The numbers of interactions in the identified modules are showed in Table S5. All sequence-base predicted regulations were with significant PCC values between the regulators and regulated genes. The enriched GO terms and KEGG pathways include cancer relevant GO terms and KEGG pathways, such as positive regulation of RNA metabolic process and Wnt signaling pathway, suggesting the method also predicted microRNA-gene regulatory modules for Dataset II (Table S6(b), Supplementary File).

4. Discussion

Several related methods and databases for the identification of microRNA-TF-gene regulatory modules have been published. The method we proposed has a number of advantages over other module identification methods. For example, CircuitDB [7] and MIR@NT@N [5] utilized sequenced-based target predictions and protein-protein interactions to constrict microRNA-TF-gene module. But they are static databases and could not answer the question about alteration of gene expressions in a specific type of disease or lack of ability to incorporate the expression values into analysis. MAGIA2 [28] and miRGator 2.0 [29] provided tools for prediction of microRNA-gene modules by combining the sequence-based target prediction and user-supplied expression profiles, but they did not separate TF genes from the entire set of genes. In regulatory modules, many TF genes that are not differentially expressed could be as important as differentially expressed TF genes as coactivators. mirConnX [6] took both the sequence-based predictions and the specified TFs into consideration to construct condition-specific mRNA-microRNA networks. However, the resulting networks were often too large to pinpoint the most important functional modules in a disease. Our method bridged the gap between the above methods by utilizing both sequence-based predictions and expression profiles and emphasizing the transcription factor's effect for the detection of condition-specific regulatory modules.

Our method and other similar methods that identify microRNA-gene regulatory modules were based on the assumption that microRNAs are posttranscriptional regulators that regulate TFs' expressions. But several studies have proposed that TFs can regulate the transcription of microRNA directly [30, 31]. Currently those databases were built to predict the regulation of transcription factors on microRNA precursors. A possible extension of our method is to transfer it into relationships between transcription factors and mature microRNAs and to incorporate this knowledge into our module identification method.

As more information is incorporated, not all of them should be considered equally in evaluation; for example, experimentally validated regulation may be more valuable for the user. In addition, the microRNA's regulation on TF genes and nTF genes are measured equally in current method, but the microRNA's regulation on TF genes may be more interesting. We can improve our method by adding control parameters to emphasize specific types of relationships.

5. Conclusion

We proposed a computational method that combines the sequence-based target predictions and matched microRNA-gene expression profiles. Our method independently processes measurement of interactions, identification of coexpression gene sets, and regulatory modules. The major characteristics are (1) easy integration of other methods for identification of gene coexpression set, (2) easy refinement by including updated information of target prediction, and (3) easy setup of parameters to emphasize interest of research. It is a candidate tool for clinical researchers to use user-supplied data to perform further investigation and exploration.

Acknowledgment

The research was partially supported by the Chancellor's Discovery Fund, University of Illinois at Chicago.

References

[1] N. J. Martinez and A. J. M. Walhout, "The interplay between transcription factors and microRNAs in genome-scale regulatory networks," *BioEssays*, vol. 31, no. 4, pp. 435–445, 2009.

[2] miRBase, http://www.mirbase.org/.

[3] Y. Dai and X. Zhou, "Computational methods for the identification of microRNA targets," *Open Access Bioinformatics*, vol. 2010, no. 2, pp. 29–39, 2010.

[4] N. J. Martinez and A. J. M. Walhout, "The interplay between transcription factors and microRNAs in genome-scale regulatory networks," *BioEssays*, vol. 31, no. 4, pp. 435–445, 2009.

[5] A. Le Béchec, E. Portales-Casamar, G. Vetter et al., "MIR@NT@N: a framework integrating transcription factors, microRNAs and their targets to identify sub-network motifs in a meta-regulation network model," *BMC Bioinformatics*, vol. 12, no. 1, article 67, 2011.

[6] G. T. Huang, C. Athanassiou, and P. V. Benos, "mirConnX: condition-specific mRNA-microRNA network integrator," *Nucleic Acids Research*, vol. 39, supplement 2, pp. W416–W423, 2011.

[7] O. Friard, A. Re, D. Taverna, M. De Bortoli, and D. Corá, "CircuitsDB: a database of mixed microRNA/transcription factor feed-forward regulatory circuits in human and mouse," *BMC Bioinformatics*, vol. 11, no. 1, article 435, 2010.

[8] B. S. Taylor, N. Schultz, H. Hieronymus et al., "Integrative genomic profiling of human prostate cancer," *Cancer Cell*, vol. 18, no. 1, pp. 11–22, 2010.

[9] R. Edgar, M. Domrachev, and A. E. Lash, "Gene Expression Omnibus: NCBI gene expression and hybridization array data repository," *Nucleic Acids Research*, vol. 30, no. 1, pp. 207–210, 2002.

[10] S. Ramaswamy, P. Tamayo, R. Rifkin et al., "Multiclass cancer diagnosis using tumor gene expression signatures," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, no. 26, pp. 15149–15154, 2001.

[11] J. Lu, G. Getz, E. A. Miska et al., "MicroRNA expression profiles classify human cancers," *Nature*, vol. 435, no. 7043, pp. 834–838, 2005.

[12] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: a practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society B*, vol. 57, no. 1, pp. 289–300, 1995.

[13] D. Betel, M. Wilson, A. Gabow, D. S. Marks, and C. Sander, "The microRNA.org resource: targets and expression," *Nucleic Acids Research*, vol. 36, supplement 1, pp. D149–D153, 2008.

[14] P. Landgraf, M. Rusu, R. Sheridan et al., "A mammalian microRNA expression atlas based on small RNA library sequencing," *Cell*, vol. 129, no. 7, pp. 1401–1414, 2007.

[15] B. John, A. J. Enright, A. Aravin, T. Tuschl, C. Sander, and D. S. Marks, "Human microRNA targets," *PLoS Biology*, vol. 2, no. 11, article e363, 2004.

[16] V. Matys, O. V. Kel-Margoulis, E. Fricke et al., "TRANSFAC and its module TRANSCompel: transcriptional gene regulation in eukaryotes," *Nucleic Acids Research*, vol. 34, pp. D108–D110, 2006.

[17] A. E. Kel, E. Gößling, I. Reuter, E. Cheremushkin, O. V. Kel-Margoulis, and E. Wingender, "MATCH: a tool for searching transcription factor binding sites in DNA sequences," *Nucleic Acids Research*, vol. 31, no. 13, pp. 3576–3579, 2003.

[18] D. Roqueiro, J. Frasor, and Y. Dai, "BindSDb: a binding-information spatial database," in *Proceedings of IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW '10)*, pp. 573–578, December 2010.

[19] D. W. Huang, B. T. Sherman, and R. A. Lempicki, "Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources," *Nature Protocols*, vol. 4, no. 1, pp. 44–57, 2009.

[20] Y. Lu, Y. Zhou, W. Qu, M. Deng, and C. Zhang, "A lasso regression model for the construction of microRNA-target regulatory networks," *Bioinformatics*, vol. 27, no. 17, pp. 2406–2413, 2011.

[21] J. L. Chen, J. Li, K. J. Kiriluk et al., "Deregulation of a Hox protein regulatory network spanning prostate cancer initiation and progression," *Clinical Cancer Research*, vol. 18, no. 16, pp. 4291–4302, 2012.

[22] J. Jiang, E. J. Lee, Y. Gusev, and T. D. Schmittgen, "Real-time expression profiling of microRNA precursors in human cancer cell lines," *Nucleic Acids Research*, vol. 33, no. 17, pp. 5394–5403, 2005.

[23] E. J. Lee, Y. Gusev, J. Jiang et al., "Expression profiling identifies microRNA signature in pancreatic cancer," *International Journal of Cancer*, vol. 120, no. 5, pp. 1046–1054, 2007.

[24] G. A. Calin, A. Cimmino, M. Fabbri et al., "MiR-15a and miR-16-1 cluster functions in human leukemia," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 13, pp. 5166–5171, 2008.

[25] Y.-Z. Pan, M. E. Morris, and A.-M. Yu, "MicroRNA-328 negatively regulates the expression of breast cancer resistance protein (BCRP/ABCG2) in human cancer cells," *Molecular Pharmacology*, vol. 75, no. 6, pp. 1374–1379, 2009.

[26] E. Bandrés, E. Cubedo, X. Agirre et al., "Identification by real-time PCR of 13 mature microRNAs differentially expressed in colorectal cancer and non-tumoral tissues," *Molecular Cancer*, vol. 5, article 29, 2006.

[27] R. L. Pio, *The role of early growth response gene Egr3 in prostate cancer [Ph.D. thesis]*, University of California, Irvine, Calif, USA, 2012.

[28] A. Bisognin, G. Sales, A. Coppe, S. Bortoluzzi, and C. Romualdi, "MAGIA^2: from miRNA and genes expression data integrative analysis to microRNA-transcription factor mixed regulatory circuits (2012 update)," *Nucleic Acids Research*, vol. 40, no. 1, pp. W13–W21, 2012.

[29] J.-H. Cho, R. Gelinas, K. Wang et al., "Systems biology of interstitial lung diseases: integration of mRNA and microRNA expression changes," *BMC Medical Genomics*, vol. 4, no. 1, article 8, 2011.

[30] J. Wang, M. Lu, C. Qiu, and Q. Cui, "TransmiR: a transcription factor—microRNA regulation database," *Nucleic Acids Research*, vol. 38, supplement 1, pp. D119–D122, 2010.

[31] C. Qiu, J. Wang, P. Yao, E. Wang, and Q. Cui, "microRNA evolution in a human transcription factor and microRNA regulatory network," *BMC Systems Biology*, vol. 4, no. 1, article 90, 2010.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 730210, 10 pages
http://dx.doi.org/10.1155/2013/730210

Review Article

Computational and Bioinformatics Frameworks for Next-Generation Whole Exome and Genome Sequencing

Marisa P. Dolled-Filhart, Michael Lee Jr., Chih-wen Ou-yang, Rajini Rani Haraksingh, and Jimmy Cheng-Ho Lin

Rare Genomics Institute, 4100 Forest Park Avenue, Suite 204, St. Louis, MO 63108, USA

Correspondence should be addressed to Jimmy Cheng-Ho Lin; jimmy.lin@raregenomics.org

Received 28 October 2012; Accepted 22 November 2012

Academic Editors: R. Jiang, W. Tian, J. Wan, and X. Zhao

It has become increasingly apparent that one of the major hurdles in the genomic age will be the bioinformatics challenges of next-generation sequencing. We provide an overview of a general framework of bioinformatics analysis. For each of the three stages of (1) alignment, (2) variant calling, and (3) filtering and annotation, we describe the analysis required and survey the different software packages that are used. Furthermore, we discuss possible future developments as data sources grow and highlight opportunities for new bioinformatics tools to be developed.

1. Introduction

Without doubt, the development of next-generation sequencing has transformed biomedical research. Multiple second generation sequencing platforms, such as Roche/454, Illumina/Solexa, AB/SOLiD, and LIFE/Ion Torrent, have made high-throughput genetic analysis more readily accessible to researchers and even clinicians [1]. On the horizon, third generation sequencing technologies, such as Oxford Nanopore, Genia, NABsys, and GnuBio, will continue to increase throughput capabilities and decrease the cost of sequencing. With each new generation of sequencing technology, there is an exponential increase in the flood of data. The true challenges of high throughput sequencing will be bioinformatics. As ever larger datasets become more affordable, computational analysis rather than sequencing will be the rate-limiting factor in genomics research. In this paper, we provide an overview of the current computational framework and options for genomic analysis and provide some outlook on future developments and upcoming needs.

In this paper, we will discuss some of the options in each of the steps and provide a global outlook on the software "pipelines" currently in development (Figure 1).

2. Overview

While different sequencing technologies may use different initial raw data (e.g., imaging files or fluctuations in current), the eventual outputs are nucleotide base calls. Short strings of these bases, varying from dozens to hundreds of base pairs for each fragment, are combined together, often in a form of a FASTQ file. From here, bioinformatics analysis of the sequence falls into three general steps: (1) alignment, (2) variant calling, (3) filtering and annotation.

The first step is alignment—matching each of the short reads to positions on a reference genome (for the purposes of this paper, the human genome). The resulting sequence alignment is stored in a SAM (sequence alignment/map) or BAM (binary alignment/map) file [2]. The second step is variant calling—comparing the aligned sequences with known sequences to determine which positions deviate from the reference position. The produces a list of positions or calls recorded in a VCF (variant call format) file [3]. The third step includes both filtering as well as annotation. Filtering takes the tens of thousands of variants and reduces them to a smaller set. For cancers, this involves comparing cancerous cell genomes to normal genomes. For family data,

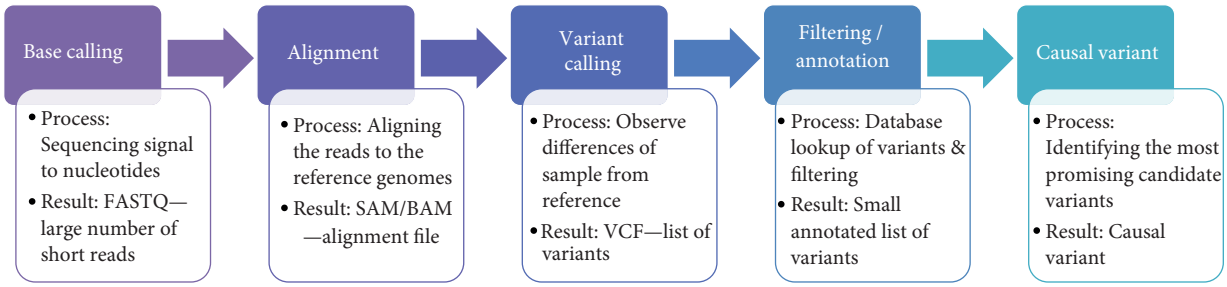


Figure 1: Next-generation sequencing bioinformatics workflow.

it involves selecting variants that conform to a specific genetic inheritance pattern. Annotation involves querying known information about each variant that is detected. Annotation may reveal, for example, that a variant is an already-known single nucleotide polymorphism, that a functional effect has already been predicted, that the function or activity of the gene in question is already known, or even that an associated disease has been identified.

Ultimately, the optimal result from the analysis is a small number of well-annotated variants that can explain a biological phenomenon. For example, for a Mendelian disease, analysis could identify the causative variant or gene. For cancer, analysis may point to driver mutations or targetable genes. Starting from base calls and ending with biologically important genetic variants, each step of analysis may be performed using one of many pieces of software. This paper discusses several of the bioinformatics options for each of these three steps.

3. Alignment

Alignment is the process of mapping short nucleotide reads to a reference genome. Because each of the millions of short reads must be compared to the 3 billion possible positions within the human genome, this computational step is not trivial. Software must assess the likely starting point of each read within the reference genome, and the task is complicated by the volume of short reads, unique versus non-unique mapping, and variation in base quality. This step is thus computationally intense and time consuming [4]. It is also a critical step, as any errors in alignment to the reference genome will be carried through to the rest of the analysis.

The Sequence Alignment/Map (SAM) and Binary Alignment/Map (BAM) formats are the standard file formats for storing NGS read alignments [2]. There are various software programs, some commercially available and others freely available to the scientific community, that can be used to perform sequencing read alignment. Various programs differ in speed and accuracy. Most alignment algorithms use an indexing method in order to more rapidly narrow down potential alignment locations within the reference genome with ungapped alignment, although other algorithms allow for gapped alignment. Different approaches to alignment involve hash tables, spaced seeds, and/or contiguous seeds. This method also enables comparison of differing output structures (single versus multiple possible alignment outputs) [5].

Short reads generated from NGS may either be single-end reads or paired-end reads from the sample, and may range from dozens to hundreds of base pairs [5]; these reads need to be aligned correctly to their appropriate location within the reference genome. Algorithms typically utilize a hash-based index (e.g., MAQ, ELAND), BWT-based index (e.g., BWA, Bowtie, SOAP2), genome-based hash (e.g., Novoalign, SOAP), or a spaced-seed approach (e.g., SHRiMP). Some algorithms report the "best" match using heuristic approaches (e.g., BWA, Bowtie, MAQ), while others allow for all possible matches (e.g., SOAP3, SHRiMP). Algorithms differ in whether they can handle both single-end and paired-end reads, or just one type (e.g., SARUMAN for single-end reads), and whether they can perform gapped alignment (e.g., BWA, Bowtie2) in addition to ungapped alignment (e.g., MAQ, Bowtie). Some algorithms focus on speed (e.g., BWA, Bowtie), some on sensitivity (e.g., Novoalign), and some algorithms aim to the two (e.g., Stampy). Table 1 provides a listing of relevant algorithms for alignment of short reads to the reference genome. While there has been previous comparisons about these algorithms [6], we describe some of the newer programs, such as Bowtie and Bowtie 2, or SOAP/SOAP2/SOAP3, and others below.

3.1. Bowtie/Bowtie 2. The Bowtie algorithm is both ultrafast and memory efficient [7] due to its use of a refinement of the FM Index, which itself utilizes the Burrows-Wheeler transformation for ultrafast and memory-efficient alignment of reads to a reference genome. Bowtie improves upon BWT with a novel quality-aware backtracking algorithm that permits mismatches. However, there may be some tradeoffs between speed and alignment quality using this algorithm [5]. Bowtie2 allows for analysis of gapped reads, which may result either from true insertions or deletions, or from sequencing errors. The newer adaptions utilize full-text minute indices and hardware-accelerated dynamic programming algorithms to optimize both speed and accuracy [8].

3.2. BWA/BWA-SW. The BWA approach, based on BWT, provides efficient alignment of short reads against the reference genome [9]. This is the most commonly used approach for sequence alignment, and followed the development of the first-generation hash-table based alignment algorithm MAQ

Table 1

Program	Source type	Description	Website
Bowtie	Open source	Ungapped alignment Refined use of FM Index using the BWT Fast and memory-efficient alignment Quality value output	http://bowtie.cbcb.umd.edu/
Bowtie2	Open source	Extends Bowtie approach to be useful for gapped alignment	http://bowtie-bio.sourceforge.net/bowtie2/index.shtml
SEAL	Open source	Comparison of alignment algorithms using simulated short read sequencing runs	http://compbio.case.edu/seal/
SOAP3	Open source	Gapped and ungapped alignment Specialized for detecting and genotyping SNPs Hash table accelerates searching using BWT-based index Reports multiple possibilities rather than single best match Increased speed using GPU	http://www.cs.hku.hk/2bwt-tools/soap3/; http://soap.genomics.org.cn/soap3.html
BWA, BWA-SW	Open source	Most common/standard method used Index with BWT that is faster than the hash-based index used for MAQ Quality score reported	http://maq.sourceforge.net/
mrFAST, mrsFAST	Open source	Seed-and-extend alignment method with hash table inex for reference genome Reports all read mappings instead of single best mapping Useful for CNVs, structural variants	—
Novoalign	Commercially available	Novocraft's proprietary software with hashing strategy High accuracy for single end mapping Focused on sensitivity	http://www.novocraft.com/
SHRiMP/SHRiMP2	Open source	Specialized for SOLiD color-space reads using spaced seeds and SWA Also applicable for regular letter-space reads Handles higher level of polymorphisms	http://compbio.cs.toronto.edu/shrimp/
MAQ	Open source	Ungapped alignment Hash-based index with quality score for mapping	http://maq.sourceforget.net/
Stampy	Open source	Hybrid mapping algorithm and statistical model, complementary to BWA	http://www.well.ox.ac.uk/project-stampy/
ELAND	Commercially available	Hash-based alignment program	http://www.illumina.com/
LAST aligner	Open source	Probablistic alignment quality scores as well as usual score matrix Useful for preprocessing for SNP/indel calling	http://last.cbrc.jp/
SARUMAN	Open Source	Mapping approach for single-end reads that returns all possible alignments with given error threshold with high speed using GPUs	http://www.cebitec.uni-bielefeld.de/brf/saruman/saruman.html

[10]. BWA improved upon MAQ by allowing for gapped alignment of single end reads, which is important for longer reads that may contain indels, and allowed for increased speed. BWA-SW allows for matches without heuristics and alignment of longer sequences [11].

3.3. mrFAST/mrsFAST. In contrast to algorithms focused on "unique" alignment of regions of the genome and selection of the "best" match, mrFAST [12] and mrsFAST [13] allow for rapid assessment of copy-number variation and assignment of sequences into both unique and the more complex duplicated regions of the genome [14, 15]. The methodology of these algorithms is a seed-and-extend approach similar to BLAST, which uses hash tables to index the reference genome. These algorithms can handle smaller structural variants (e.g., indels) and larger structural variants such as insertions, deletions, inversions, CNVs, and segmental duplications in a cache-oblivious manner.

3.4. SHRiMP/SHRiMP2. Developed to handle a greater number of polymorphisms by utilizing a statistical model to screen out false positive hits, SHRiMP [16] can be utilized for color-spaced reads from AB SOLiD sequencers and can also be used for regular letter-space reads. SHRiMP2 [17] enables direct alignment for paired-reads and uses multiple spaced seeds, but instead of using indexed reads like SHRiMP, SHRiMP2 switched to an indexing method like Bowtie and BWA.

3.5. SOAP/SOAPv2/SOAPv3. SOAP was developed for use in gapped and ungapped alignment of short reads using a seed strategy for either single-read or pair-end reads, and can also be applied to small RNA and mRNA tag sequences [18]. SOAP2 reduced memory usage and increased speed using BWT for hash-based indexing instead of the seed algorithm, and also includes SNP detection [19]. SOAP3 is a GPU (graphics processing unit) version of the compressed full-text index-based SOAP2, which allows for a speed improvement [20].

4. Variant Calling

After alignment of the short reads to the reference genome, the next step in the bioinformatics process is variant calling. Since the short reads are already aligned, the sample genome can be compared to the reference genome and variants can then be identified. These variants may be responsible for disease, or they may simply be genomic noise without any functional effect. Variant call format (VCF) is the standardized generic format for storing sequence variation including SNPs, indels, larger structural variants and annotations [3]. The computational challenges in SNP (variant) calling are due to the issues in identifying "true" variants versus alignment and/or sequencing errors. Yet the ability to detect SNPs with both high sensitivity and specificity is a key step in identifying sequence variants associated with disease, detection of rare variants, and assessment of allele frequencies in populations.

The difficulty of variant calling is complicated by three factors: (1) the presence of indels, which represent a major source of false positive SNV identifications, especially if alignment algorithms do not perform gapped alignments; (2) errors from library preparation due to PCR artifacts and variable GC content in the short reads unless paired-end sequencing is utilized; and (3) variable quality scores, with higher error rates generally found at bases at the ends of reads [4]. Therefore, the rate of false positive and false negative calls of SNVs and indels is a concern. A detailed review of SNP-calling algorithms and challenges recommends recalibration of per-base quality scores (e.g., GATK, SOAPsnp), use of an alignment algorithm with high sensitivity (e.g., Novoalign, Stampy), and SNP calling using Bayesian procedures or likelihood ratio tests and incorporation of linkage disequilibrium to improve SNP call accuracy [21]. We provide an overview of some of the software packages for variant calling below.

4.1. The Genome Analysis ToolKit (GATK). Developed by the Broad Institute, the Genome Analysis ToolKit (GATK) is one of the most popular methods for variant calling using aligned reads. It is designed in a modular way and is based on the MapReduce functional programming approach [22]. The package has been used for projects such as The Cancer Genome Atlas [23] and the 1000 Genomes Project [24] that have covered analyses of HLA typing, multiple-sequence realignment, quality score recalibration, multiple-sample SNP genotyping and indel discovery and genotyping [22].

4.2. SOAPsnp. Developed by the Beijing Genome Institute, SOAPsnp is an open source algorithm (http://soap.genomics.org.cn/) that requires access to a high-quality variant database using SOAP alignment results as an input [18]. It can be used for consensus calling and SNP detection for the Illumina Genome Analyzer platform and utilizes the phred-like quality score to calculate the likelihood of each genotype based on the alignment results and sequencing quality scores. Building upon the speed of the alignment algorithm Bowtie [7] and using SOAPsnp for SNP calling, an open source cloud-computing tool called Crossbow [7] was developed to perform both alignment and SNP calling.

4.3. VarScan/VarScan2. Developed by the Genome Institute at Washington University in St. Louis, VarScan (http://genome.wustl.edu/tools/cancer-genomics/) is an open source tool for short read variant detection of SNPs and indels that is compatible with multiple sequencing platforms and aligner algorithms such as Bowtie and Novoalign [25]. It can detect variants at 1% frequency, which can be useful for pooled samples; VarScan permits analysis of individual samples as well. VarScan2 [26] includes some improvements upon VarScan, such as the ability to analysis tumor-normal sample pairs for somatic mutations, LOH (loss of heterozygosity) and CNAs (copy number alterations). This program reads tumor and normal sample Samtools pileup or mpileup output simultaneously for pairwise comparisons of base calling and normalized sequence depth at each position.

4.4. ATLAS 2. Developed by the Baylor Genome Center, Atlas 2 can be used for variant calling of aligned data from multiple NGS platforms on a range of computing platforms [27]. Atlas 2 can also be implemented via a web resource called Genboree Workbench (http://www.genboree.org/). A few other web-based analysis tools are available such as DNANexus (http://www.dnanexus.com) and Galaxy [28]. Details of Atlas 2 in comparison to other variant calling algorithms such as SAMtools mpileup and GATK are included in Challis et al. [27], and reviewed by Ji [29].

5. Insertions and Deletions

While the majority of research has focused on diseases associated with SNPs, indel (insertion and deletion) mutations are a common polymorphism that can also demonstrate to biological effects. Studies have shown that small indels might be highly associated with neuropsychiatric diseases such as schizophrenia, autism, mental retardation, and Alzheimer's disease [30].

In addition, the presence of certain indels is associated with the disease progression of HBV-induced hepatocellular carcinoma (HCC) in the Korean population [31]. Indels are also used as genetic markers in natural populations [32]. With the advance of sequencing platforms and analysis tools, detection of indels through NGS has become more common. However, accurate mapping of indels to the reference genome is challenging, because it requires approaches that involve complicated gapped alignment and paired-end sequence inference [9]. Moreover, the occurrence rate of indels is approximately 8-fold lower than that of SNPs [33]. An optimal combination of both alignment and indel-calling algorithms is essential for identifying indels with high sensitivity and specificity. One review evaluated the performance of various alignment tools on microindel detection, and recommended single-end reads gapped alignment mapping tools such as BWA and Novoalign [34]. Various software approaches have been developed to identify indels, including a pattern growth approach (e.g., Pindel) and a Bayesian procedure (e.g., Dindel). A detailed review by Neuman et al. evaluated the performance of several difference indel-calling programs in the presence of varying parameters (read depth, read length, indel size, and frequency). By using both simulated and real data that included the *Caenorhabditis elegans* genome, they observed that Dindel has the highest sensitivity (indels found) at low coverage, although Dindel is only suitable for Illumina data analysis. VarScan and GATK require additional parameter adjustments, such as high coverage for VarScan, to reach their best performance. This review provides information for appropriate tool selection and parameter optimization to assist successful experimental designs and recommends Dindel as a suitable tool for low coverage experiments. Below, we survey the tools that have been commonly used for indel calling.

5.1. Pindel. Pindel is a software program which implements a pattern growth approach to detect breakpoints of large deletions (1–10 kb) and medium sized insertions (1 bp–20 bp) from paired-end short reads in NGS data [35]. A recent, more advanced, version, Pindel2, has been introduced which includes the ability to identify insertions of any size, inversions and tandem duplications [35]. Pindel has been used for the 1000 Genomes Project (http://www.1000genomes.org/) [36], the Genome of the Netherlands project, and the Cancer Genome Atlas [23].

5.2. Dindel. Developed by the Welcome Trust Sanger Institute, Dindel is an open-source program that utilizes a Bayesian approach for calling small (<50 bp) insertions and deletions (http://www.sanger.ac.uk/resources/ software/dindel/) [37]. Principally, this algorithm realigns sequence reads mapped to a variety of candidate haplotypes that represent alternative sequences to the reference. Dindel has been used in the 1000 Genomes Project call sets and can only analyze data from Illumina.

5.3. GATK. As described in the variant calling section, the Genome Analysis ToolKit (GATK), which provides a collection of data analysis tools, can also allow indel calling based on the MapReduce programming approach [22]. Details of GATK in comparison to other indel calling methods including Dindel (VarScan, SAMtools mpileup) are evaluated in Neuman et al. [38].

6. Filtering and Annotation

After alignment and variant calling, a list of thousands of potential differences between the genome under study and the reference genome is generated. The next step is to determine which of these variants are likely to contribute to the pathological process under study. The third step involves a combination of both filtering (removing variants that fit specific genetic models or are not present in normal tissue) as well as annotation (looking up information about variants and identifying ones that fit the biological process).

Filtering can be done with a genetic pedigree or with cancer and normal samples from the same individual. In the instance of cancer, a common method is removing variants that are present in both the cancer sample and the normal sample, leaving only somatic variants, which have mutated from the germline sequence. In the instance of a pedigree, filtering can be done based on the different inheritance patterns. For example, if the inheritance pattern is autosomal recessive, the variants that are heterozygous in the parents and homozygous in the child can be chosen. Similar methods can be done with larger pedigrees based on the inheritance pattern.

In addition to filtering, further selection of causal variants can be based on existing annotation or predicted functional effect. Many tools exist to examine relevant variants by referencing previously known information about their biological functions and inferring potential effects based on their genomic context. In addition, many tools have been developed to identify genetic variants that cause disease pathogenesis or phenotypic variance [39]. Rare nonsynonymous SNPs are SNPs that cause amino acid substitution

(AAS) in the coding region, which potentially affect the function of the protein coded and could contribute to disease.

The advance of exome and genomic sequencing is yielding an extensive number of human genetic variants, and a number of disease-associated SNVs can be identified following alignment and variant calling. Unlike nonsense and frameshift mutations, which often result in a loss of protein function, pinpointing disease-causal variants among numerous SNVs has become one of the major challenges due to the lack of genetic information. For instance, ~1,300 loci are shown to be associated with ~200 diseases by GWASs but only a few of these loci have been identified as disease-causing variants [40]. Exome sequencing enables the identification of more novel genetic variants than previously possible, but it still requires computational and experimental approaches to predict whether a variant is deleterious. To this end, several approaches have been developed to identify rare nonsynonymous SNPs that cause amino acid substitution (AAS) in the coding region. The major principle of the protein-sequence-based methods to predict deleteriousness in the coding sequence is based on comparative genomics and functional genomics. Comparative sequencing analysis assumes that amino acid residues that are critical for protein function should be conserved among species and homologous proteins; therefore, mutations in highly conserved sites are more likely to result in more deleterious effect. Other modalities to predict disease-causing variants include protein biochemistry, such as amino acid charge, the presence of a binding site, and structure information of protein. SNVs that are predicted to alter protein feature (such as polarity and hydropathy) and structure (binding ability and alteration of secondary/tertiary structure) have a higher probability of being deleterious.

Although the majority of research has focused on protein-altering variants, noncoding variants constitute a large portion of human genetic variation. Results obtained from GWAS indicate that ~88% of trait-associated weak effect variants are found in noncoding regions, demonstrating the importance of functional annotation of both coding and noncoding variants [41]. Computational tools for protein-sequence-based prediction of deleteriousness fall into two categories: constraint-based predictors such as MAPP and SIFT, and trained classifiers such as MutationTaster and polyPhen. In addition to protein-sequence-based methods, another way to prioritize disease-casual SNVs is through nucleotide-sequence-based prediction in noncoding and coding DNA. This process also utilizes comparative genomics to predict deleteriousness, and is used by programs such as phastCons, GERP, and Gumby. In one detailed review of disease-causing variant identification, the authors introduced the concepts and tools that allow genetic annotation of both coding and noncoding variants [39]. They also compared the relative utility of nucleotide- and protein-based approaches using exome data, finding that nucleotide-based constraint scores defined by Genomic Evolutionary Rate Profiling (GERP) and protein-based deleterious impact scores provided by PolyPhen were similar for two Mendelian diseases, suggesting that nucleotide-based prediction can be as powerful as protein-based metrics [39]. Below, we survey tools that are helpful identifying disease-causal variants among numerous candidates.

6.1. Sorting Intolerant from Tolerant (SIFT). Sorting Intolerant From Tolerant (SIFT) (http://sift.jcvi.org/) prediction is based on conserved amino acid residues through different species using comparative sequencing analysis through PSI-BLAST [42]. This relies on the presumption that amino acid residues that are essential for protein function should be evolutionally conserved by natural selection. Therefore, SNPs resulting in AAS on the conservative residues are more likely to be deleterious.

6.2. PolyPhen. PolyPhen/PolyPhen2 (http://genetics.bwh.harvard.edu/pph2/) algorithm predicts the potential impact of AAS on the structure and function of human protein based on protein sequence, phylogenetic and structural information [43]. An amino acid replacement might occur at a specific site where binding to other molecules or the formation of a secondary/tertiary structure is disrupted. Therefore, PolyPhen determines if the AAS is found at a site which is annotated as a disulfide bond, an active site, a binding site, or a specific motif such as transmembrane domain. Another function of PolyPhen is to compare the sequence and polymorphic regions of homologous proteins in the same family to identify AASs that are rare or never observed in the family. In addition, PolyPhen also maps of the substitution site to the known 3-dimensional protein structure to assess if an AAS has the potential to destroy protein structure via an alteration of, for example, the hydrophobic core of a protein, electrostatic interactions, or interactions with ligands or other molecules.

6.3. VariBench. VariBench (http://structure.bmc.lu.se/VariBench/) is the first benchmark database that provides testing and training tools for computational variation effect prediction [44]. It comprises experimentally validated variation datasets collected from the literature and relevant databases. The datasets housed in VariBench enable identification of variants that affect protein tolerance, protein stability, transcription factor binding sites, and splice sites. Additionally, VariBench maps variant positions to the DNA, RNA, and protein sequences at RefSeq, and to the 3-dimensional protein structures at Protein Data Bank (PDB).

6.4. snpEFF. snpEFF is an open source, Java-based program that rapidly categorizes SNP, indel, and MNP variants in genomic sequences as having either high, medium, low or modifier functional effects [45]. Variant annotation is based on genomic location (intron, exon, untranslated region, upstream, downstream, splice site, intergenic region) and predicted coding effect (synonymous/nonsynonymous amino acid replacement, gain/loss of start/stop site, frameshift mutations). The program may find several different functions for a single variant due to competing predictions based on alternative transcripts. snpEFF uses a VCF input and output style. Currently snpEFF does not support structural variants but there are plans to incorporate

such support soon. snpEFF is compatible with GATK and Galaxy, which are popular variant-calling toolkits. The program currently supports 260 genome versions and can be used with custom genomes and annotations.

6.5. The SNPeffect Database. The SNPeffect Annotation database (http://snpeffect.switchlab.org/) uses sequence and structure information to predict the effect of protein-coding SNVs on the structural phenotype of proteins [46]. It is primarily focused on disease-causing and polymorphic variants in the human proteome. This program compares variant protein predictions to wild type protein information from the UniProtKB database, which currently contains more than 60,000 variant proteins. Variant characterization is achieved by integrating aggregation, amyloid prediction, chaperone-binding prediction, and protein stability analysis information by applying several algorithms to each wild type and mutant protein. The first algorithm, TANGO, detects regions that are prone to aggregation and calculates a score difference between the mutant and wild type protein. The WALTZ algorithm is applied to predict amyloid-forming regions in protein sequences using a position-specific scoring matrix to deduce amyloid-forming propensity. LIMBO is an algorithm that predicts chaperone binding sites for the Hsp70 chaperones. In cases where structural information is available, the FoldX algorithm is used to calculate the difference in free energy between the mutated protein and the wild type and determine whether the mutation stabilizes or destabilizes the structure. Mutations are also characterized as falling into catalytic sites according to information in the Catalytic Site Atlas or not, and falling into known domains or not. Subcellular information is predicted using PSORT.

6.6. SeattleSeq. SeattleSeq (http://snp.gs.washington.edu/SeattleSeqAnnotation/) annotates known and novel SNPs with biological functions, protein positions and amino-acid changes, conservation scores, HapMap frequencies, PolyPhen predictions, and clinical associations based on an integrated database. Most of the annotation information is derived from the Genome Variation Server (http://gvs.gs.washington.edu/GVS134/), which includes information from dbSNP as well as other sources. The algorithm accepts input files in a number of formats including GATK and VCF output styles. Currently, annotation of indels is limited.

6.7. ANNOVAR. The ANNOVAR software tool (http://www.openbioinformatics.org/annovar/) utilizes up-to date information to rapidly functionally annotate genetic variants called from sequencing data [47]. ANNOVAR works on a number of diverse genomes including hg18, hg19, mouse, worm, fly, and yeast. The annotation system allows the user flexibility in the set of genomic regions that are queried. Annotations can be gene-based (users can select the gene definition system; RefSeq, UCSC, ENSEMBL, GENCODE, etc.), region-based (transcription factor binding sites, DNAse I hypersensitivity sites, ENCODEmethylation sites, segmental duplication sites, DGV sites, etc.), filter-based (e.g., using only variants reported in dbSNP, or only variants with MAF > 1%), or based on any of many other user-driven functionalities.

6.8. The Variant Annotation, Analysis and Search Tool (VAAST). The Variant Annotation, Analysis and Search Tool (VAAST) identifies damaged genes and deleterious variants in personal genome sequences using a probabilistic search method [48]. The tool utilizes both existing amino acid substitution and aggregative approaches to variant prioritization and combines them into a single unified likelihood-framework. This method increases the accuracy with which disease causing variants are identified. VAAST scores both coding and noncoding, and both rare and common, variants simultaneously and aggregates this information to identify disease causing variants.

6.9. The Variant Analysis Tool (VAT). The Variant Analysis Tool, VAT, (http://vat.gersteinlab.org/) functionally annotates variants called from personal genomes at the transcript level and provides summary statistics across genes and individuals [49]. VAT is a computational framework that can be implemented through a command-line interface, a web application, or a virtual machine in a cloud-computing environment. This tool has been utilized extensively to annotate loss-of-function variants obtained as part of the 1000 Genomes Project [50]. The VAT modules *snpMapper*, *indelMapper and svMapper* relate SNPs, indels and SVs to protein-coding genes while the *genericMapper* module relates variants to noncoding regions of the genome. Transcript-level analysis allows identification of affected isoforms. VAT outputs VCF files as well as visualization summarizing the biological impact of annotated variants.

6.10. VARIANT. VARIANT (VARIant ANalysis Tool) (http://variant.bioinfo.cipf.es/) provides annotation of variants from next generation sequencing based on several different databases and repositories including dbSNP, 1000 Genomes Project, the GWAS catalog, OMIM, and COSMIC [36]. The provided annotations also include information on the regulatory or structural roles of the variants as well as the selective pressures on the affected genomic sites. Unlike other such tools, VARIANT utilizes a remote database and operates by interacting with this database through efficient RESTful Web Services. Currently VARIANT supports all human, mouse and rat genes. Analyzing variants generated by exome sequencing of families in which rare Mendelian diseases are segregated can be a time-consuming process.

6.11. VAR-MD. VAR-MD is a software tool to analyze variants derived from exome or whole genome sequencing in human pedigrees with Mendelian inheritance [51]. This algorithm outputs a ranked list of potential disease-causing variants based on predicted pathogenicity, Mendelian inheritance models, genotype quality, and population variant frequency data. This tool is unique in that it uses family-based annotation of sequence data to enhance mutation identification. VAR-MD is a Unix-based tool and is implemented in

Python. Independent functions of the program are usually run sequentially. In order to facilitate parallel analysis of multiple data sets, VAR-MD utilizes Galaxy for distributed resource management.

The various variant annotation tools differ in the types of variants they process. All algorithms process SNPs and indels, but only a few, such as ANNOVAR and VAT, can handle SVs. These tools also differ in the computing environment in which they are implemented. Some rely on command-line operation while others operate using web-based interfaces or virtual machines in the cloud. Some tools utilize local databases while other use up-to-date remote databases. These various tools also differ in the genomic regions that they target. For example, SNPeffect focuses on the proteome while other tools focus on the less obvious, but still functionally relevant regions. From the long list of possible variants, through filtering and annotation, a smaller list of most probably causal variants is generated.

7. Future Outlook and Conclusion

While the current tools in all three stages of the bioinformatics analysis are adequate, more data will enable further significant improvements. New technology and algorithms may significantly shift the field in unforeseeable ways, but several future improvements are predictable as (1) sequencing reads increase in length, (2) more genomes are completed, and (3) annotation databases are better populated.

First, as sequencing technology increases the base pair read length, alignment will become more accurate. Shorter reads match with a greater number of genome sites. As reads grow in length, they can be mapped more precisely with fewer options and thus less room for error. This is especially true in regions with low complexity or a high number of repeats, classically very difficult regions to map. Longer reads will make alignment an easier problem.

Second, the process of variant calling will benefit from larger databases of completed genomes. A variant is derived from comparison to the reference genome, and our set of reference genomes continues to grow. This will enable variant calling based on ethnic background, or based on populations of genomes instead of a single reference genome or a small set of reference genomes.

Third, while filtering appears unlikely to change significantly, annotation and functional prediction will be improved by more data and more-populated databases. For filtering, since the genetic models and removal of normal variants from tumor variants are based only on the genetics and the samples under study, additional information from the databases will not change these aspects much. By contrast, the efficacy of annotation is directly related to what is present in known databases. Different dimensions of data, such as functional, pathway, biochemical, or genetic annotation can all be improved as more genomes are sequenced and annotated. Moreover, current predictive algorithms such as SIFT and Polyphen are dependent on current database annotation. If large numbers of human genomes are sequenced, analysis need not resort to merely predicting the effect of a single position; one can simply query that position in the millions of people that are sequenced and infer the deleterious effect.

Besides the more predictable changes that will follow naturally from more data, there are also opportunities for larger paradigm shifts in bioinformatics tools. First, emerging tools may be able to analyze samples not as a homogenous whole, but in ways that allow for tumor heterogeneity with differing populations of cells. Furthermore, single-cell and single-molecule methods are maturing. It is now more appreciated that the tumors consists of populations of cells, and that being able to determine the quantity and identity of these cells will not only help understand tumor population dynamics, but may also inform treatment and prognosis.

Second, thus far relatively few tools have integrated other high throughput modalities such as proteomics into genomic interpretation. In order to understand whether the mutation has biological significance, it is critical to know whether a gene is expressed on a transcript or protein level. As more multidimensional data is produced through projects such as ENCODE, TCGA, or 1000 Genomes, and high-throughput sample profiling becomes easier on a genomic, transcriptomic, and proteomic level, methods that can incorporate all this data will add power to the analysis.

Third, in additional to multidimensional data, there are also opportunities for systems biology methods to be incorporated to software packages. Protein-protein interaction datasets continue to grow as the human interactome is mapped, and knowledge of these molecular pathways can and should be integrated into genomics analysis. Understanding genes not only as isolated constructs but also as part of a greater system would better model the biological process.

Fourth, as more and more datasets are available and sequencing becomes cheaper, genomics analysis need no longer be based on a single genome, a comparison between an isolated pair of cancer genome samples, or larger, but still isolated, pedigrees. Current tools analyze single samples at a time and compare what is found with databases. Instead, tools that are able to analyze large numbers of genomes at the same time to sizes similar to genome-wide association studies will prove to be powerful.

Undoubtedly, the datasets used in genomics analysis will continue to grow in depth per individual and in the number of samples. Bioinformatics, more than ever before, will be the crucial step in making sense of the data flood. The incremental progress afforded by this flood will be critical and valuable, but researchers can also look forward to the yet-unknown paradigm shifts that loom over the horizon.

References

[1] E. R. Mardis, "Next-generation DNA sequencing methods," *Annual Review of Genomics and Human Genetics*, vol. 9, pp. 387–402, 2008.

[2] H. Li, B. Handsaker, A. Wysoker et al., "The Sequence Alignment/Map format and SAMtools," *Bioinformatics*, vol. 25, no. 16, pp. 2078–2079, 2009.

[3] P. Danecek, A. Auton, G. Abecasis et al., "The variant call format and VCFtools," *Bioinformatics*, vol. 27, no. 15, Article ID btr330, pp. 2156–2158, 2011.

[4] A. G. Day-Williams and E. Zeggini, "The effect of next-generation sequencing technology on complex trait research," *European Journal of Clinical Investigation*, vol. 41, no. 5, pp. 561–567, 2011.

[5] M. Ruffalo, T. LaFramboise, and M. Koyuturk, "Comparative analysis of algorithms for next-generation sequencing read alignment," *Bioinformatics*, vol. 27, no. 20, pp. 2790–2796, 2011.

[6] N. Homer and S. F. Nelson, "Improved variant discovery through local re-alignment of short-read next-generation sequencing data using SRMA," *Genome Biology*, vol. 11, no. 10, article R99, 2010.

[7] B. Langmead, C. Trapnell, M. Pop, and S. L. Salzberg, "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome," *Genome Biology*, vol. 10, no. 3, article R25, 2009.

[8] B. Langmead and S. L. Salzberg, "Fast gapped-read alignment with Bowtie 2," *Nature Methods*, vol. 9, no. 4, pp. 357–359, 2012.

[9] H. Li and R. Durbin, "Fast and accurate short read alignment with Burrows-Wheeler transform," *Bioinformatics*, vol. 25, no. 14, pp. 1754–1760, 2009.

[10] H. Li, J. Ruan, and R. Durbin, "Mapping short DNA sequencing reads and calling variants using mapping quality scores," *Genome Research*, vol. 18, no. 11, pp. 1851–1858, 2008.

[11] H. Li and R. Durbin, "Fast and accurate long-read alignment with Burrows-Wheeler transform," *Bioinformatics*, vol. 26, no. 5, Article ID btp698, pp. 589–595, 2010.

[12] C. Alkan, S. Sajjadian, and E. E. Eichler, "Limitations of next-generation genome sequence assembly," *Nature Methods*, vol. 8, no. 1, pp. 61–65, 2011.

[13] F. Hach, F. Hormozdiari, C. Alkan et al., "MrsFAST: a cache-oblivious algorithm for short-read mapping," *Nature Methods*, vol. 7, no. 8, pp. 576–577, 2010.

[14] I. D. Dinov, F. Torri, F. Macciardi et al., "Applications of the pipeline environment for visual informatics and genomics computations," *BMC Bioinformatics*, vol. 12, article 304, 2011.

[15] C. Alkan, J. M. Kidd, T. Marques-Bonet et al., "Personalized copy number and segmental duplication maps using next-generation sequencing," *Nature Genetics*, vol. 41, no. 10, pp. 1061–1067, 2009.

[16] S. M. Rumble, P. Lacroute, A. V. Dalca, M. Fiume, A. Sidow, and M. Brudno, "SHRiMP: accurate mapping of short color-space reads," *PLoS Computational Biology*, vol. 5, no. 5, Article ID e1000386, 2009.

[17] M. David, M. Dzamba, D. Lister, L. Ilie, and M. Brudno, "SHRiMP2: sensitive yet practical short read mapping," *Bioinformatics*, vol. 27, no. 7, Article ID btr046, pp. 1011–1012, 2011.

[18] R. Li, Y. Li, K. Kristiansen, and J. Wang, "SOAP: short oligonucleotide alignment program," *Bioinformatics*, vol. 24, no. 5, pp. 713–714, 2008.

[19] R. Li, C. Yu, Y. Li et al., "SOAP2: an improved ultrafast tool for short read alignment," *Bioinformatics*, vol. 25, no. 15, pp. 1966–1967, 2009.

[20] C. M. Liu, K. F. Wong, E. M. K. Wu et al., "SOAP3: ultra-fast GPU-based parallel alignment tool for short reads," *Bioinformatics*, vol. 28, no. 6, pp. 878–879, 2012.

[21] R. Nielsen, J. S. Paul, A. Albrechtsen, and Y. S. Song, "Genotype and SNP calling from next-generation sequencing data," *Nature Reviews Genetics*, vol. 12, no. 6, pp. 443–451, 2011.

[22] A. McKenna, M. Hanna, E. Banks et al., "The genome analysis toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data," *Genome Research*, vol. 20, no. 9, pp. 1297–1303, 2010.

[23] Cancer Genome Atlas Network, "Comprehensive molecular portraits of human breast tumours," *Nature*, vol. 490, no. 7418, pp. 61–70, 2012.

[24] D. L. Altshuler, R. M. Durbin, G. R. Abecasis et al., "A map of human genome variation from population-scale sequencing," *Nature*, vol. 467, no. 7319, pp. 1061–1073, 2010.

[25] D. C. Koboldt, K. Chen, T. Wylie et al., "VarScan: variant detection in massively parallel sequencing of individual and pooled samples," *Bioinformatics*, vol. 25, no. 17, pp. 2283–2285, 2009.

[26] D. C. Koboldt, Q. Zhang, D. E. Larson et al., "VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing," *Genome Research*, vol. 22, no. 3, pp. 568–576, 2012.

[27] D. Challis, J. Yu, U. S. Evani et al., "An integrative variant analysis suite for whole exome next-generation sequencing data," *BMC Bioinformatics*, vol. 13, article 8, 2012.

[28] J. Hillman-Jackson, D. Clements, D. Blankenberg, J. Taylor, and A. Nekrutenko, "Using Galaxy to perform large-scale interactive data analyses," in *Current Protocols in Bioinformatics*, chapter 10, unit 10.5, 2012.

[29] H. P. Ji, "Improving bioinformatic pipelines for exome variant calling," *Genome Medicine*, vol. 4, no. 1, article 7, 2012.

[30] R. R. Lemos, M. B. Souza, and J. R. Oliveira, "Exploring the implications of INDELs in neuropsychiatric genetics: challenges and perspectives," *Journal of Molecular Neuroscience*, vol. 47, no. 3, pp. 419–424, 2012.

[31] S. A. Lee, H. S. Mun, H. Kim et al., "Naturally occurring hepatitis B virus X deletions and insertions among Korean chronic patients," *Journal of Medical Virology*, vol. 83, no. 1, pp. 65–70, 2011.

[32] U. Väli, M. Brandström, M. Johansson, and H. Ellegren, "Insertion-deletion polymorphisms (indels) as genetic markers in natural populations," *BMC Genetics*, vol. 9, article 8, 2008.

[33] G. Lunter, "Probabilistic whole-genome alignments reveal high indel rates in the human and mouse genomes," *Bioinformatics*, vol. 23, no. 13, pp. i289–i296, 2007.

[34] P. Krawitz, C. Rödelsperger, M. Jäger, L. Jostins, S. Bauer, and P. N. Robinson, "Microindel detection in short-read sequence data," *Bioinformatics*, vol. 26, no. 6, Article ID btq027, pp. 722–729, 2010.

[35] K. Ye, M. H. Schulz, Q. Long, R. Apweiler, and Z. Ning, "Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads," *Bioinformatics*, vol. 25, no. 21, pp. 2865–2871, 2009.

[36] D. G. MacArthur, S. Balasubramanian, A. Frankish et al., "A systematic survey of loss-of-function variants in human protein-coding genes," *Science*, vol. 335, no. 6070, pp. 823–828, 2012.

[37] C. A. Albers, G. Lunter, D. G. MacArthur, G. McVean, W. H. Ouwehand, and R. Durbin, "Dindel: accurate indel calls from short-read data," *Genome Research*, vol. 21, no. 6, pp. 961–973, 2011.

[38] J. A. Neuman, O. Isakov, and N. Shomron, "Analysis of insertion-deletion from deep-sequencing data: software evaluation foroptimal detection," *Briefings in Bioinformatics*. In press.

[39] G. M. Cooper and J. Shendure, "Needles in stacks of needles: finding disease-causal variants in a wealth of genomic data," *Nature Reviews Genetics*, vol. 12, no. 9, pp. 628–640, 2011.

[40] E. S. Lander, "Initial impact of the sequencing of the human genome," *Nature*, vol. 470, no. 7333, pp. 187–197, 2011.

[41] L. A. Hindorff, P. Sethupathy, H. A. Junkins et al., "Potential etiologic and functional implications of genome-wide association loci for human diseases and traits," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 23, pp. 9362–9367, 2009.

[42] P. Kumar, S. Henikoff, and P. C. Ng, "Predicting the effects of coding non-synonymous variants on protein function using the SIFT algorithm," *Nature Protocols*, vol. 4, no. 7, pp. 1073–1081, 2009.

[43] I. A. Adzhubei, S. Schmidt, L. Peshkin et al., "A method and server for predicting damaging missense mutations," *Nature Methods*, vol. 7, no. 4, pp. 248–249, 2010.

[44] P. S. Nair and M. Vihinen, "VariBench: A benchmark database-for variations," *Human Mutation*. In press.

[45] P. Cingolani, A. Platts, L. Wang le et al., "A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w1118, iso-2, iso-3," *Fly*, vol. 6, no. 2, pp. 80–92, 2012.

[46] G. De Baets, J. Van Durme, J. Reumers et al., "SNPeffect 4.0: on-line prediction of molecular and structural effects of protein-coding variants," *Nucleic Acids Research*, vol. 40, pp. D935–D939, 2012.

[47] K. Wang, M. Li, and H. Hakonarson, "ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data," *Nucleic Acids Research*, vol. 38, no. 16, Article ID gkq603, p. e164, 2010.

[48] M. Yandell, C. D. Huff, H. Hu et al., "A probabilistic disease-gene finder for personal genomes," *Genome Research*, vol. 21, no. 9, pp. 1529–1542, 2011.

[49] L. Habegger, S. Balasubramanian, D. Z. Chen et al., "VAT: a computational framework to functionally annotate variants in personal genomes within a cloud-computing environment," *Bioinformatics*, vol. 28, no. 17, pp. 2267–2269, 2012.

[50] D. G. MacArthur, S. Balasubramanian, A. Frankish et al., "A systematic survey of loss-of-function variants in human protein-coding genes," *Science*, vol. 335, no. 6070, pp. 823–828, 2012.

[51] M. Sincan, D. R. Simeonov, D. Adams et al., "VAR-MD: a tool to analyze whole exome-genome variants in small human pedigrees with mendelian inheritance," *Human Mutation*, vol. 33, no. 4, pp. 593–598, 2012.

The Scientific World Journal
Volume 2012, Article ID 523706, 8 pages
doi:10.1100/2012/523706

The ScientificWorldJOURNAL

Research Article

Hierarchical Modular Structure Identification with Its Applications in Gene Coexpression Networks

Shuqin Zhang

Center for Computational Systems Biology, School of Mathematical Sciences, Fudan University, Shanghai 200433, China

Correspondence should be addressed to Shuqin Zhang, zhangs@fudan.edu.cn

Received 2 November 2012; Accepted 25 November 2012

Academic Editors: R. Jiang, W. Tian, J. Wan, and X. Zhao

Network module (community) structure has been a hot research topic in recent years. Many methods have been proposed for module detection and identification. Hierarchical structure of modules is shown to exist in many networks such as biological networks and social networks. Compared to the partitional module identification methods, less research is done on the inference of hierarchical modular structure. In this paper, we propose a method for constructing the hierarchical modular structure based on the stochastic block model. Statistical tests are applied to test the hierarchical relations between different modules. We give both artificial networks and real data examples to illustrate the performance of our approach. Application of the proposed method to yeast gene coexpression network shows that it does have a hierarchical modular structure with the modules on different levels corresponding to different gene functions.

1. Introduction

Networks are widely applied to model complex systems, including biological systems, social organizations, World-Wide-Webs, and so on. In a network, the nodes (vertices) represent the members in the system, while the edges represent the interactions among the members. If two nodes have interactions in a network, there will be an edge connecting them. With such a representation, the complex systems can be analyzed by computational methods.

Module (community) structure is a common property of many different types of networks. Modules are the dense subgroups of a network, where the nodes in the same module are more likely to connect each other than the nodes in other modules. In general, the members in the same module share some common properties or play similar roles. For example, in a gene coexpression network, the genes in the same module may belong to the same functional category such as lipid metabolism and acute-phase response [1]. Since the paper published by [2], module detection and identification becomes a hot research topic in several different areas such as computer science, physics, and statistics. A large number of related works have been published with the physicists making the most contributions [3–12]. Several recent review papers provide details and comparisons of the module identification methods [6, 9, 13]. Reference [13] compares the performance of several existing methods for both computation time and output. Reference0020[6] is a thorough, more recent discussion. Reference [9] contrasts different perspectives of the methods and sheds light on some important similarities of several methods. A recent comparison of some popular methods is shown in [14]. Among the compared methods, the method by maximizing the average degree within modules and minimizing the average connections between different modules outperforms other methods in identification accuracy. Its computational speed is also competitive [14]. Besides these computational methods, theoretical analysis on module identifications is presented very recently. Bickel and Chen gave the first statistical analysis on the properties of modules [15]. There based on the stochastic block model, they gave the sufficient conditions for a modularity to be a consistent estimator of modules and presented a new consistent modularity. However, the computation of maximizing this modularity is very time consuming.

Although so many related works are published, how to choose an appropriate number of modules keeps being an open problem. Different methods output different solutions when they are applied to the same network. In reality, all

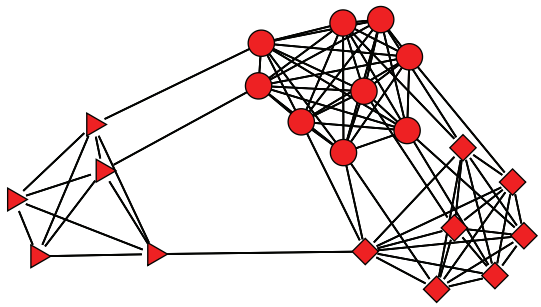


Figure 1: Example of hierarchical modular network structure.

of the different choices may be reasonable because different choices of this number may correspond to the modules on different levels. As explained in [16], some modular networks may have hierarchical structure. For example, in a friendship network, on the large scale, the modules may correspond to people from different countries. On the smaller scales, people in the same module may graduate from the same university, grow up in the same community, or even be born in the same family. Such hierarchical modular structure appears in different kinds of networks. For example, Meunier and colleagues gave an example on hierarchical modular structures in human brains [17]. Figure 1 shows an example of hierarchical modular network. There are two levels of the modules. We can identify three modules corresponding to different shapes of nodes on the lowest level or two modules with nodes represented by cubes and circles being combined together on the higher level.

Compared to the module identification in a partitional way (all the modules are on the same level), there are much fewer works on computational methods for hierarchical modular structure analysis [18–20]. Although these papers present some methods to construct the hierarchical modular structure, they do not give a clear picture on how these modules are organized and what the relationship among the modules is. In this paper, we mainly consider the problem of hierarchical modular structure in unweighted networks. Based on the module identification method presented in [14], we give the method on how to construct the hierarchical structure from all the possible modules in Section 2. Numerical experiments for both simulated networks and real data networks are presented to show the performance of our proposed method in Section 3. The application of the proposed method to yeast gene coexpression network shows that it does have a hierarchical structure, which corresponds to the different levels of gene functions. Conclusion remarks are given finally. By constructing the hierarchical structure, we aim to explore the functions of modules on different levels and explain why the number of modules may differ for different identification methods.

2. Methodology

Before going to the details on how to construct the hierarchical structure, we give its definition first. We consider a network $G(V, E)$ with n nodes, where V denotes the set of nodes and E denotes the set of edges. The adjacency matrix is denoted as A with each entry being 0 or 1. The hierarchical structure of a network is defined based on the stochastic block model, which is a direct extension of the Erdös-Rényi random graph model [21]. The network is obtained by starting with a set of n nodes and adding edges between them in a probabilistic fashion. The presence of an edge between any two nodes is a Bernoulli event where the probability may be vertex-pair dependent. At the beginning, we assume there are K modules in the network. The network is generated in two steps. First, any node is assigned to a module M_i with a probability μ_i, where $\mu = (\mu_1, \mu_2, \ldots, \mu_K)$ satisfies $\sum_{i=1}^{K} \mu_i = 1$. Then any two nodes $u, v \in V$ and $u \in M_i$, $v \in M_j$ are connected with probability $P_{i,j}$ depending on M_i, M_j, and P is symmetric. If there is the modular structure in the network, then $P_{i,j} < \min\{P_{i,i}, P_{j,j}\}$. With this model, the hierarchical structure of a network can be defined recursively. For any three modules M_i, M_j, and M_k, if $P_{i,j} > \max\{P_{i,k}, P_{j,k}\}$, we say there is hierarchical structure among these three modules and M_i, M_j can be combined to a new module parallel to M_k.

To construct the hierarchical structure, we use the bottom-up strategy. We first find all the possible modules on the lowest level and then build the hierarchical structure. We use the method presented in [14] to find all the possible modules. Suppose K is given first. We let N_k denote the number of nodes in subnetwork V_k, L_{kk} denote twice the total number of edges in subnetwork V_k, and L_{kl} denote the total number of connections between the subnetworks V_k and V_l, where $k, l = 1, 2, \ldots, K$. The module identification problem is formulated as

$$\max_{\mathbf{P}} \Phi(\mathbf{P}) = \sum_{k=1}^{K} \frac{L_{kk}}{N_k} - \sum_{k=1}^{K} \sum_{l \neq k} \frac{L_{kl}}{N_k}, \tag{1}$$

where $\mathbf{P}$ is a partition of the network.

In matrix form, if we let

$$S_{ik} = \begin{cases} 1, & \text{if node } i \in V_k \\ 0, & \text{otherwise} \end{cases} \quad i = 1, 2, \ldots, n, \tag{2}$$

the problem is formulated as

$$\begin{aligned} \max \Psi(S) &= \sum_{k=1}^{K} \frac{S_{\cdot,k}^T A S_{\cdot,k}}{S_{\cdot,k}^T S_{\cdot,k}} - \sum_{k=1}^{K} \sum_{l \neq k} \frac{S_{\cdot,k}^T A S_{\cdot,l}}{S_{\cdot,k}^T S_{\cdot,k}} \\ &= \sum_{k=1}^{K} \frac{S_{\cdot,k}^T (2A - D) S_{\cdot,k}}{S_{\cdot,k}^T S_{\cdot,k}} \\ &\text{s.t.} \quad S_{i,j} \in \{0, 1\} \quad \text{for } i, j = 1, 2, \ldots, K, \\ &\sum_{k=1}^{K} S_{\cdot,k} = \mathbf{1}. \end{aligned} \tag{3}$$

Here $\mathbf{1}$ is a vector with all elements being 1.

The objective function aims to both maximize the average degree within each module and minimize the average connections between different modules. We expect to achieve a good balance of the module size and make correct inference

on the modules. The problem (3) is solved with an approximate method similar to the spectral clustering. We first compute the K eigenvectors of the matrix $2A - D$. By clustering these K eigenvectors as a matrix of n objects with K dimensions, we get the assignment of the n nodes into K modules.

Now, we discuss how to determine the lowest level of all the possible modules K. For any node $i \in V$, the degree can be written as

$$d_i = \sum_{k=1}^{K} d_i(V_k), \tag{4}$$

where

$$d_i(V_k) = \sum_{j \in V_k} A_{ij}, \tag{5}$$

which defines the connections that node i has in the subnetwork V_k. To determine the number of possible modules, we compare the average connectivity within a subnetwork and the average connectivity between it and any other subnetwork. If the average connectivity within a subnetwork is greater, we take it as a module, that is,

$$\frac{\sum_{i \in V_k} d_i(V_k)}{N_k} > \frac{\sum_{i \in V_k} d_i(V_l)}{N_k}, \quad l \neq k. \tag{6}$$

Alternatively, it can also be written as

$$L_{kk} > L_{kl}, \tag{7}$$

if we multiply both sides with N_k. This condition is very weak, thus with it, we hope we find all the modules as on the lowest level. We do the clustering for K increasing from two until the condition (6) does not hold and get all the possible modules. The efficiency of the above algorithm can be seen in [14].

Based on the above results, we construct the hierarchical structure in an agglomerative way (bottom-to-up). We directly use connection probability, which is computed from the clustering results through maximum likelihood estimation, to measure the distance between different modules. This connection probability matrix is denoted as $\hat{P}^0$. First the maximum connection probability between different modules is found, and we assume it is $\hat{P}^0_{i_0,j_0}$ with the corresponding two modules i_0, j_0 being recorded. The second largest connection probability for these two modules i_0, j_0 are also found, and we assume they are $\hat{P}^0_{i_0,k_0}$ and $\hat{P}^0_{j_0,l_0}$ with the corresponding modules being k_0 and l_0. To determine whether there is a hierarchical structure for these modules, we use Fisher exact test to see whether the connection probabilities $\hat{P}^0_{i_0,k_0}$ and $\hat{P}^0_{j_0,l_0}$ are the same as $\hat{P}^0_{i_0,j_0}$. That is, we need to test $\hat{P}^0_{i_0,j_0} = \hat{P}^0_{i_0,k_0}$ and $\hat{P}^0_{i_0,j_0} = \hat{P}^0_{j_0,l_0}$. Here we take a P value threshold to be 0.05. Three different cases may occur for these two relations. (1) Both of these two null hypotheses are rejected. In this case, there is hierarchical structure and the modules i_0, j_0 are on the lower level than k_0 and l_0. We combine the two modules i_0 and j_0 and take them as one module. (2) Only one of $\hat{P}^0_{i_0,j_0} = \hat{P}^0_{i_0,k_0}$ and $\hat{P}^0_{i_0,j_0} = \hat{P}^0_{j_0,l_0}$ is accepted. The corresponding modules having the same connection probability are combined together. We look for the next largest connection probability for these three modules and test the relationship again. If two modules are tested to have the same connection probability, they are combined into one group, and the same step is implemented again. (3) Both of these two null hypotheses are accepted. These modules are taken as on the same level and combine together. We search the next largest connection probability to these four modules and do the statistical test until the hierarchical structure occurs or all the modules are combined together. After the above steps are finished, the connection probability between different modules is recalculated and recorded as $\hat{P}^1$. The above search and test steps are repeated for $\hat{P}^1$. Such steps are implemented recursively until all the modules are combined into one big module/network. For the statistical tests, we can also use t-test to test the relations between the connection probabilities if the distribution of the connections between different modules can be approximated by normal distribution. With this method, we can efficiently combine the modules with the same connection probability into the same level.

3. Numerical Experiments

In this section, we evaluate the performance of our proposed method through its application to several examples. We first start with two artificial networks having comparatively clear module structures. We then apply our method to two real networks to evaluate its performance. The first real network is the well-known karate club network and the second one is a yeast gene coexpression network.

3.1. Artificial Networks

3.1.1. A Network Composed of Cliques. We consider a network with 200 nodes, which is composed of 4 cliques. The sizes of the cliques are 90, 30, 40, and 40. The connections between different cliques are randomly generated with the following probability:

$$P = \begin{pmatrix} 1.000 & 0.200 & 0.002 & 0.003 \\ 0.200 & 1.000 & 0.005 & 0.010 \\ 0.002 & 0.005 & 1.000 & 0.030 \\ 0.003 & 0.010 & 0.030 & 1.000 \end{pmatrix}. \tag{8}$$

The pattern of the adjacency matrix is shown in Figure 2(a). From upper-left to lower-right, we denote the four modules as M_1, M_2, M_3, and M_4, which correspond to the position in the connection probability matrix. We can see the hierarchical structure of the network from the adjacency matrix. We apply our proposed method to this network. The condition (6) is satisfied until $K = 4$. The estimated connection probability matrix is

$$\hat{P} = \begin{pmatrix} 1.000 & 0.205 & 0.003 & 0.003 \\ 0.205 & 1.000 & 0.006 & 0.009 \\ 0.003 & 0.006 & 1.000 & 0.029 \\ 0.003 & 0.009 & 0.029 & 1.000 \end{pmatrix}. \tag{9}$$

We apply statistical tests to the corresponding modules, and finally we get the hierarchical structure as shown in

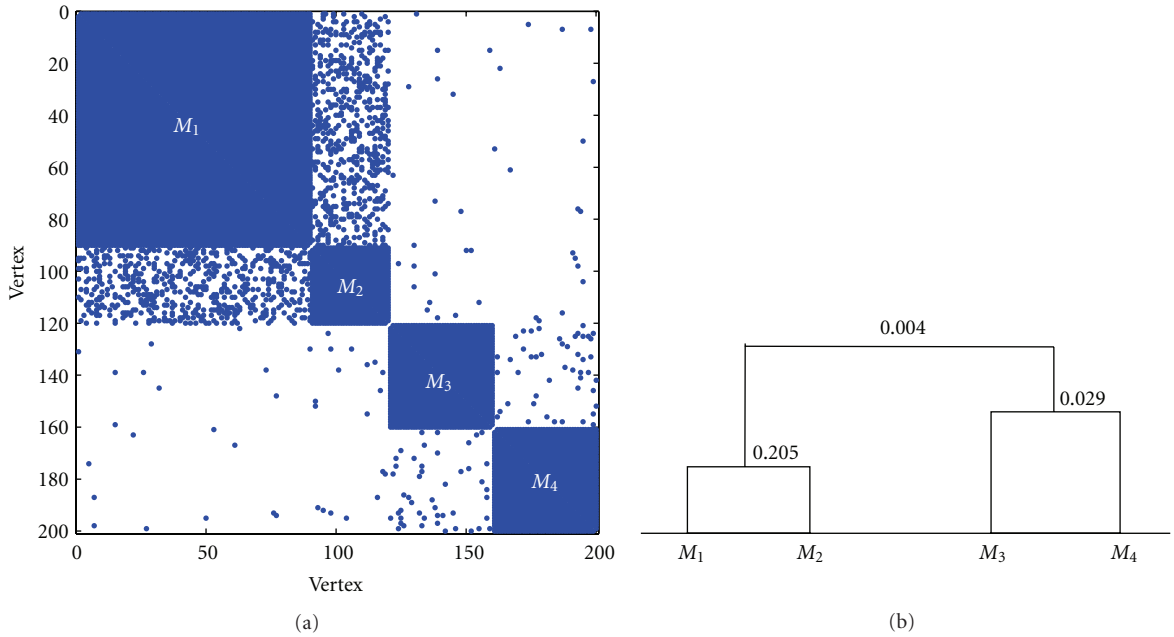


Figure 2: Example of hierarchical modular network structure. (a) Pattern of the adjacency matrix; (b) the hierarchical structure of the network.

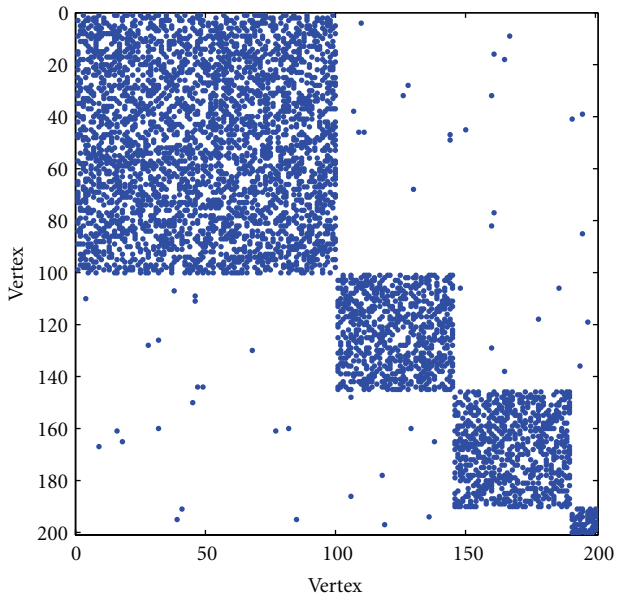


Figure 3: Pattern of the adjacency matrix for the randomly generated network.

Figure 2(b). The values on the hierarchical tree is the estimated connection probability of the corresponding modules. On the lowest level, there are four modules. If the tree is cut between 0.205 and 0.029, there are three modules while if the cutoff is greater than 0.029, there are only two modules. These results are consistent with the network generation strategy.

3.1.2. A Randomly Generated Network. In this example, we also consider a network with 200 nodes and 4 modules. The size of each module is 10, 45, 45, and 100. We set the degree of each node within its module to be 6, 15, 15, and 30. Then the connections between different nodes are randomly generated. We keep all the edges generated for each node. So finally the average degree within each module is greater than the prespecified number. The connection probability between different modules is 0.002. The pattern of the adjacency matrix is shown in Figure 3. From upper-left to lower-right, the four modules are M_1, M_2, M_3, and M_4, respectively. With our proposed method, the network is partitioned into four modules correctly on the lowest level and the estimated connection probability is

$$\hat{P} = \begin{pmatrix} 0.298 & 0.002 & 0.002 & 0.003 \\ 0.002 & 0.328 & 0.002 & 0.004 \\ 0.002 & 0.002 & 0.321 & 0.000 \\ 0.003 & 0.004 & 0.000 & 0.560 \end{pmatrix}. \quad (10)$$

By using the statistical tests, these four modules are determined as parallel modules, which is the same as that in our network generation strategy.

3.2. Karate Club Network. We consider the Zachary's network of karate club members [22] in this example. There are 34 nodes in this network corresponding to the members in a karate club. This dataset has been applied as a benchmark to test many module identification algorithms since the true modules are known in this network. The people in the club were observed for a period of three years. The edges represent connections of the individuals outside the activities of the club. At some point, the administrator and the instructor of the club broke up due to a conflict between them. The club was separated into two groups supporting the administrator and the instructor. Figure 4 shows the network. Originally, there are two modules, which have 16 nodes (squares and

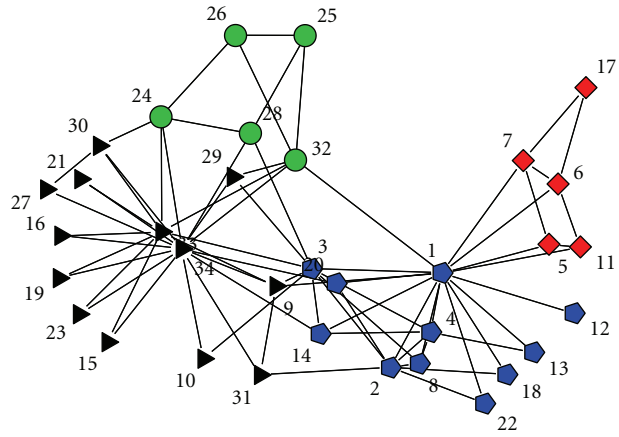


FIGURE 4: Zachary's karate club network. Different shapes show the modules. M_1: pentagon, M_2: square, M_3: triangle, M_4: circle.

pentagons in the figure) and 18 nodes (circles and triangles in the figure), respectively.

We apply our proposed method to this network. The criterion (6) is satisfied until $K = 4$. The result is shown in Figure 4, with different shapes of the nodes denoting different modules. The estimated connection probability matrix is

$$\hat{P} = \begin{pmatrix} 0.364 & 0.073 & 0.056 & 0.036 \\ 0.073 & 0.480 & 0.000 & 0.000 \\ 0.056 & 0.000 & 0.237 & 0.108 \\ 0.036 & 0.000 & 0.108 & 0.480 \end{pmatrix}. \quad (11)$$

From this matrix, it is easy to see that M_3 and M_4 are more likely to connect each other. With statistical tests, we can get that the connection probability among M_3, M_4, and M_1 is the same. Although M_2 has no connections to M_3 and M_4, it has a larger connection probability to M_1 than M_3, M_4 to M_1. Thus these four modules are on the same level. In [19], the authors considered constructing the hierarchical modular structure of this network too. At first, they also found four modules on the lowest level. Then they found that this network has two modules with some nodes (3, 9, 10, 14, 31) belonging to both of them. We did not consider the overlapping nodes in this article. However, we can see that because these overlapping nodes belong to both M_1 and M_3, and they connect both parts closely, our method detect M_1 and M_3, M_3 and M_4 as having the same connectivity.

3.3. Hierarchical Modular Structure in Yeast Gene Coexpression Network. In this section, we apply our proposed approach to analyze a gene coexpression network of yeast. The data set we use was generated by Brem and Kruglyak from a cross between two distinct isogenic strains BY and RM [23]. As described in [23], a total of 5740 ORFs were obtained after data preprocessing. In our analysis, we only use the 1,800 most differentially expressed genes as input to construct coexpression network and derive modules. When constructing the adjacency matrix of the network, we use the hard thresholding, that is: if the absolute value of Pearson correlation coefficient between two genes is greater than some given value, we assign an edge between them; otherwise, there is no edge. We compute the linear regression coefficient between the frequency of degree d ($\log 10(f(d))$) and the log 10 transformed degree d ($\log 10(d)$), and choose the threshold that leads to approximately scale free property of the network as described in [24]. Finally, the threshold is set to be 0.705, $\hat{R}$ is about 0.75. By such a setting, this gene coexpression network is divided into 690 unconnected parts with the largest part of size 788. Here, we only analyze the hierarchical modular structure of the largest connected network.

Starting from $K = 2$, we apply the method in [14] to this network, and the condition (6) holds until $K = 10$. To make the solution more accurate, we do a global maximization by changing the module index of boundary nodes starting from the approximate solution. Since the approximate solution is already good, this step is very fast. The structure of the network is shown in Figure 5(a), with different colors and shapes denoting different modules as described in Table 1. Then we construct the hierarchical modular structure as shown in Figure 5(b). On the lowest level, there are ten modules, while on the highest level, there are four modules.

Since coexpressed genes tend to be coregulated and possibly have similar functions, genes in the same module are expected to be enriched for some function categories. In order to understand the biological basis of the network modules, we consider each identified module for enrichment of annotations from gene ontology (GO) [25]. In our analysis, the enrichment analysis was performed by GO stats from Bioconductor. For each module, the statistically most significant GO categories are analyzed. Table 1 shows the enrichment results for the ten modules. "M-size" and "G-size" are the size of both the modules and the GO categories, respectively. "Overlap" is the overlap size of the module and the GO category. Table 2 shows the enrichment results for the modules on different levels. From the tables, it is easy to see that different gene function categories are enriched most on different levels. For example, module M_2 enriches the GO category "translation" most significantly, while the combined module M_2, M_8 enriches "Ribonucleoprotein complex biogenesis" most significantly, with M_2 containing 42 genes having this function. The combined module M_2, M_8, M_4, and M_1 also enriches this function, while M_4 itself enriches

Table 1: GO enrichment analysis results of the gene modules on the lowest level.

Module	Color, shape	M-size	Enriched GO category	P value	G-size	Overlap
M_1	White, square	190	Cellular carbohydrate metabolic process	3.23×10^{-9}	60	35
M_2	White, circle	126	Translation	4.70×10^{-59}	101	80
M_3	Grey, triangle	135	Organic acid biosynthetic process	5.41×10^{-35}	89	64
M_4	Grey, pentagon	62	Cellular respiration	4.13×10^{-27}	36	28
M_5	Black, circle	12	Amino acid catabolic process to alcohol via Ehrlich pathway	1.76×10^{-7}	5	4
M_6	Black, circle	13	Steroid biosynthetic process	2.20×10^{-15}	13	9
M_7	White, pentagon	19	Branched chain family amino acid metabolic process	4.37×10^{-8}	11	6
M_8	Grey, triangle	209	Ribonucleoprotein complex biogenesis	5.94×10^{-39}	149	106
M_9	Grey, square	11	Protein targeting to membrane	8.91×10^{-6}	4	3
M_{10}	White, square	11	Regulation of translational termination	1.55×10^{-4}	2	2

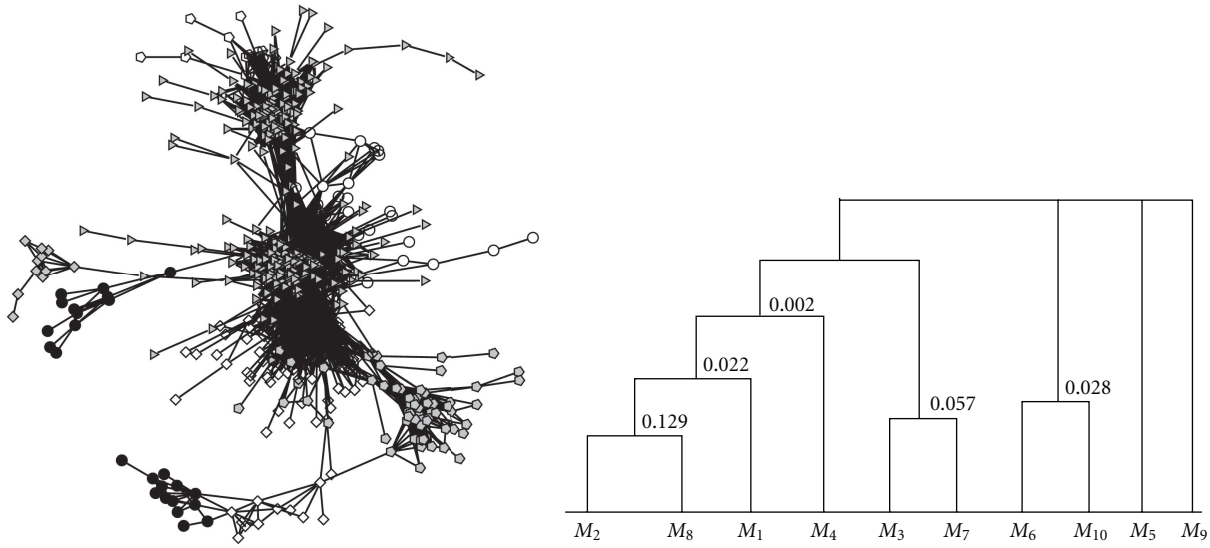


Figure 5: Yeast gene coexpression network. (a) The network structure, (b) the hierarchical structure.

"cellular respiration" significantly. On the uppermost level, the module composed of M_2, M_8, M_1, M_4, M_3, and M_7 enriches four GO function categories most significantly, and all the genes are overlapped. Three ("cellular component biogenesis," "cellular component biogenesis at cellular level," and "ribosome biogenesis") of them are different from the most enriched gene functions for each of these six modules. All these results indicate that hierarchical modular structure does exist in gene coexpression networks and different gene functions are enriched most on different levels.

We use the software REViGO to check the hierarchical structure of the enriched GO categories [26]. We consider the enriched GO categories in Tables 1 and 2 except the category "regulation of translational termination" because its G-size is very small and the P value is comparatively large. Figure 6 shows the tree map of the most enriched GO categories. The subgraph that we do not mark with the module corresponds to the combined module M_1, M_2, M_3, M_4, M_7, M_8. Here the modules M_6, M_9 and other modules are parallel to each other, which is consistent with our results. M_3 and M_7 belong to a large category, which is "branched chain family amino acid metabolic process". This large category is different from the most enriched category for the combined module M_3 and M_7. This may come from the fact that since M_7 is very small, it does not cover a large part of its enriched category. M_1 and M_4 are parallel to each other which is also consistent with our analysis. All these results show that our proposed method can explain some of the hierarchical structure of the GO categories. Due to the network size, we did not handle all the genes of yeast. This may be a reason why some of our computational results are not consistent with the GO function tree map.

4. Conclusion

Module identification problem has attracted much attention from different fields and it continues being a hot research topic. How to determine the number of modules in a modular network has been an open problem during the study of module identification methods. This problem may come from the hierarchical structure of modular networks. The different numbers correspond to the different levels of

Table 2: GO enrichment analysis results of gene modules on the upper level.

Module	M-size	Enriched GO category	P value	G-size	Overlap
M_2, M_8	335	Ribonucleoprotein complex biogenesis	4.02×10^{-66}	149	148
M_1, M_2, M_8	525	Ribonucleoprotein complex biogenesis	1.33×10^{-29}	149	148
M_1, M_2, M_4, M_8	587	Ribonucleoprotein complex biogenesis	6.04×10^{-23}	149	149
M_3, M_7	154	Organic acid biosynthetic process	9.22×10^{-40}	89	71
M_1, M_2, M_3, M_4	741	Cellular component biogenesis	4.01×10^{-6}	175	175
M_7, M_8		Cellular component biogenesis at cellular level	1.84×10^{-5}	156	156
		Ribonucleoprotein complex biogenesis	3.19×10^{-5}	149	149
		Ribosome biogenesis	3.44×10^{-5}	148	148
M_6, M_{10}	24	Steroid biosynthetic process	2.36×10^{-19}	13	12

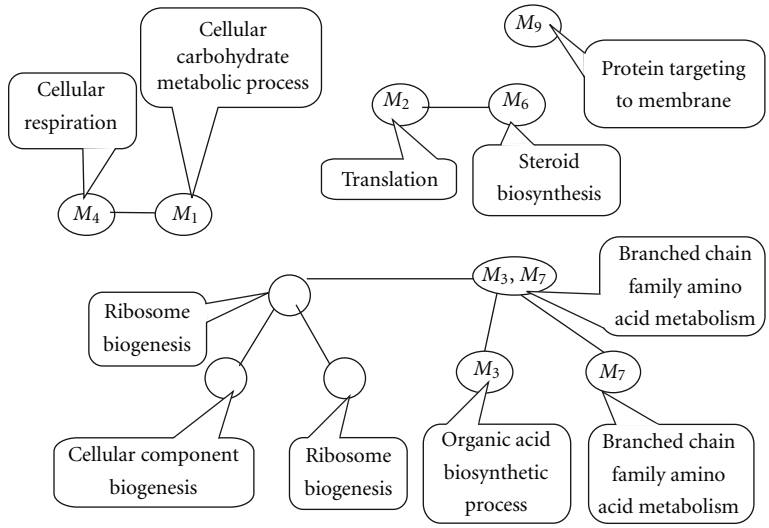


Figure 6: Tree map of the enriched GO categories in yeast gene coexpression network.

the hierarchical structure and they may be all reasonable. In this paper, we proposed a method for constructing the hierarchical modular structure of networks. With statistical tests, we can identify both the parallel modules and the hierarchical structure. According to different cutoffs of the hierarchical tree, different numbers of modules can be identified. This may solve the problem of the number of network modules to some extent. Several examples are given to demonstrate the efficiency of our method. Application of this method to gene coexpression networks shows that there are hierarchical modules in yeast gene coexpression network. On different levels of such networks, the genes in the module belong to different gene functions most. Thus studying the gene function through constructing the hierarchical modular structure instead of specifying the number of modules should perform better. Application of such algorithms to other kinds of networks may also contribute to other research fields.

Acknowledgments

This work was supported in part by NSFC Grants 10901042, 10971075, and 91130032. The primary version of this paper has appeared in IEEE ISB 2012.

References

[1] R. Guimerà and L. A. N. Amaral, "Functional cartography of complex metabolic networks," *Nature*, vol. 433, no. 7028, pp. 895–900, 2005.

[2] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821–7826, 2002.

[3] A. Arenas, J. Borge-Holthoefer, S. Gómez, and G. Zamora-López, "Optimal map of the modular structure of complex networks," *New Journal of Physics*, vol. 12, Article ID 053009, 2010.

[4] J. Dong and S. Horvath, "Understanding network concepts in modules," *BMC Systems Biology*, vol. 1, article 24, 2007.

[5] E. Estrada and N. Hatano, "Communicability in complex networks," *Physical Review E*, vol. 77, no. 3, Article ID 036111, 2008.

[6] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3–5, pp. 75–174, 2010.

[7] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E*, vol. 74, no. 3, Article ID 036104, 2006.

[8] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 23, pp. 8577–8582, 2006.

[9] M. A. Porter, J. P. Onnela, and P. J. Mucha., "Communities in networks," *Notices of the American Mathematical Society*, vol. 56, no. 9, pp. 1082–1102, 2010.
[10] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Paris, "Defining and identifying communities in networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 9, pp. 2658–2663, 2004.
[11] Z. Li, S. Zhang, R.-S. Wang, X.-S. Zhang, and L. Chen, "Quantitative function for community detection," *Physical Review E*, vol. 77, no. 3, Article ID 036109, 2008.
[12] M. Rosvall and C. T. Bergstrom, "An information-theoretic framework for resolving community structure in complex networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 18, pp. 7327–7331, 2007.
[13] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics*, no. 9, Article ID 09008, pp. 219–228, 2005.
[14] S. Zhang and H. Zhao, "Community identification in networks with unbalanced structure," *Physical Review E*, vol. 85, no. 6, Article ID 066114, 2012.
[15] P. J. Bickel and A. Chen, "A nonparametric view of network models and Newman-Girvan and other modularities," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 50, pp. 21068–21073, 2009.
[16] A. Arenas, A. Díaz-Guilera, and C. J. Pérez-Vicente, "Synchronization reveals topological scales in complex networks," *Physical Review Letters*, vol. 96, no. 11, Article ID 114102, 2006.
[17] D. Meunier, R. Lambiotte, and E. T. Bullmore, "Modular and hierarchically modular organization of brain networks," *Frontiers Neuroscience*, vol. 4, no. 200, 2010.
[18] H. Shen, X. Cheng, K. Cai, and M.-B. Hu, "Detect overlapping and hierarchical community structure in networks," *Physica A*, vol. 388, no. 8, pp. 1706–1712, 2009.
[19] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New Journal of Physics*, vol. 11, Article ID 033015, 2009.
[20] E. Ravasz, "Detecting hierarchical modularity in biological networks," *Computational Systems Biology*, vol. 54, pp. 145–160, 2009.
[21] P. Erdős and A. R. Rényi, "On random graphs. I," *Publicationes Mathematicae*, vol. 6, pp. 290–297, 1959.
[22] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. 33, pp. 452–473, 1977.
[23] R. B. Brem and L. Kruglyak, "The landscape of genetic complexity across 5,700 gene expression traits in yeast," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 5, pp. 1572–1577, 2005.
[24] B. Zhang and S. Horvath, "A general framework for weighted gene co-expression network analysis," *Statistical Applications in Genetics and Molecular Biology*, vol. 4, no. 1, article 17, 2005.
[25] M. Ashburner, C. A. Ball, J. A. Blake et al., "Gene ontology: tool for the unification of biology, the gene ontology consortium," *Nature Genetics*, vol. 25, no. 1, pp. 25–29, 2000.
[26] F. Supek, M. Bošnjak, N. Škunca, and T. Šmuc, "REVIGO summarizes and visualizes long lists of gene ontology terms," *PLoS ONE*, vol. 6, no. 7, Article ID 21800, 2011.

The Scientific World Journal
Volume 2012, Article ID 939584, 6 pages
doi:10.1100/2012/939584

The ScientificWorldJOURNAL

Research Article

Large Scale Association Analysis for Drug Addiction: Results from SNP to Gene

Xiaobo Guo,[1,2] Zhifa Liu,[1] Xueqin Wang,[2,3] and Heping Zhang[1]

[1] *Department of Biostatistics, Yale University School of Public Health, New Haven, CT 06520, USA*
[2] *Department of Statistical Science, School of Mathematics and Computational Science, Sun Yat-sen University, Guangzhou, Guangdong, China*
[3] *Zhongshan School of Medicine, Sun Yat-sen University, Guangzhou, Guangdong, China*

Correspondence should be addressed to Heping Zhang, heping.zhang@yale.edu

Received 26 September 2012; Accepted 25 November 2012

Academic Editors: R. Jiang, W. Tian, J. Wan, and X. Zhao

Many genetic association studies used single nucleotide polymorphisms (SNPs) data to identify genetic variants for complex diseases. Although SNP-based associations are most common in genome-wide association studies (GWAS), gene-based association analysis has received increasing attention in understanding genetic etiologies for complex diseases. While both methods have been used to analyze the same data, few genome-wide association studies compare the results or observe the connection between them. We performed a comprehensive analysis of the data from the Study of Addiction: Genetics and Environment (SAGE) and compared the results from the SNP-based and gene-based analyses. Our results suggest that the gene-based method complements the individual SNP-based analysis, and conceptually they are closely related. In terms of gene findings, our results validate many genes that were either reported from the analysis of the same dataset or based on animal studies for substance dependence.

1. Introduction

Genome-wide association studies (GWAS) have become a powerful tool in the identification of susceptible loci for numerous diseases [1]. A typical strategy in GWAS is to analyze single nucleotide polymorphisms (SNPs) individually and select the top SNPs by setting a stringent threshold for the P value. Then the top SNPs were mapped into functional regions such as a gene or pathway to facilitate further investigation of the corresponding gene and disease. Based on SNP-based association analysis, many genetic variants underlying complex diseases or traits were detected [2, 3]. Due to the large number of SNPs with each of which entails an association test, it is essential to control the type I error or false discovery rate [4]. A predefined P value $< 5 \times 10^{-8}$ is usually used as the threshold to declare a genome-wide significance SNP, which also limits the discoveries of the genes that are important to the disease. Also importantly, susceptible SNPs generally explain a small fraction of the risk—a phenomenon commonly referred to as the "missing heritability" [5, 6]. To alleviate this problem, alternative methods have emerged to complement the simple SNP-based methods. Among those methods, gene-based analysis [7–9], which jointly analyzes the SNPs within genes, is a promising solution to improve the power of GWAS. Compared with the SNP-based approach, gene-based association analysis has certain advantages. First, gene is a unit of heredity and function, and hence the gene-based association approaches can provide direct insights into the heredity and functional mechanisms of complex traits [10]. Second, from the statistical perspective, the gene-based association approaches reduce the number of association tests in the order of millions to about 20,000 gene-based tests, which dramatically reduces the chance of false discovery. In addition, the gene-based methods are not affected by the heterogeneity of a single locus. Hence, the results are highly consistent across populations [11], which enhances the likelihood of replication.

Gene-based methods have been successfully applied to GWAS of complex diseases, including Crohn's disease [7], type 1 diabetes [12], and melanoma [8]. Despite the above-noted features of the gene-based association approach, there

are few comparisons of genetic association analyses between SNP and gene-based methods. Here, we compare and relate these two approaches using the data from the Study of Addition: Genetics and Environment (SAGE) [13].

Recent studies show that there are many candidate genes associated with substance dependence. For example, GABRA2, CHRM2, ADH4, PKNOX2, GABRG3, TAS2R16, SNCA, OPRK1, and PDYN are well studied for alcohol addiction and have been replicated in many samples [13–28]. However, other candidate genes, such as KIAA0040, ALDH1A1, DKK2, and MANBA [25, 27, 29, 30], remain illusive. For addiction to nicotine, CHRNA5, CHRNA3, CHRNB4, and CSMD1 have been replicated in many studies [31–39].

Based on the analysis of the SAGE data, we report a number of susceptible loci at the SNP and/or gene levels, which validate many susceptibility loci that have been reported to be associated with substance dependence [13, 14, 25, 27, 29, 37, 38, 40–44]. Meanwhile, both SNP- and gene-based analyses reveal three novel risk genes: NCK2, DSG3, and PUSL1.

2. Materials and Methods

2.1. Dataset and Study Design. The dataset included 4,121 subjects in SAGE with six categories of substance dependence data: alcohol, cocaine, marijuana, nicotine, opiates, and other dependencies on drugs. The data were downloaded from dbGaP (study accession phs000092.v1.p1) [13]. SAGE [13] is a large case-control study which aims to detect susceptible genetic variants for addition. The subjects were recruited from eight study sites in seven states and the District of Columbia in the United States. All subjects' life time dependencies on these six dependencies are diagnosed by using the Diagnostic and Statistical Manual of Mental Disorders, Fourth Edition (DSM-IV). All samples were genotyped on ILLUMINA Human 1 M platform at the Center for Inherited Disease Research in Johns Hopkins University. In this paper, we strictly followed the quality control/quality assurance as we did in our previous analysis [14]. Genome-wide SNP data were filtered by setting thresholds: MAF > 5% and call rate > 90%. In addition, 60 duplicate genotype samples and 9 individuals with ethnic backgrounds other than African origin or European origin were excluded in our analysis. Finally, 3,627 unrelated samples with 859,185 autosomal SNPs passed the quality control procedures. To avoid population stratification, the dataset was stratified into four subsamples: 1,393 white women, 1,131 white men, 568 black women, and 535 black men. To capture most of the gene coding and regulatory variants, SNPs are considered being mapped to a gene if their physical locations are within 20 kilobases (kb) 5′ upstream and 10 kilobases (kb) 3′ downstream of gene coding regions [26]. In addition, SNPs are also assigned to a gene if they are in strong LD ($r^2 > 0.9$) with the initially assigned SNPs within the gene [10]. Together, around 533,639 SNPs were assigned to 18,699 protein coding genes (28.6 ± 47.7 (mean ± SD) SNPs per gene).

Following the conventional standards, we used $5.0E-8$ and $2.5E-6$ as the genome-wide significant thresholds for SNP-based and gene-based methods, respectively [4]. To increase the power of detecting potentially important SNPs that do not meet the stringent thresholds, we also considered relaxed thresholds. Specifically, SNPs with $P < 1.0E-5$ and genes $P < 5.0E-4$ were considered further. These P values are referred to as relaxed significance thresholds below. The selected SNPs were then mapped into the corresponding genes by the mapping rule proposed above.

2.2. Genetic Association Test at SNP and Gene Levels. We took several steps in testing the associations between genetic variants (SNP or gene) and substance dependenice. First, the P value of each SNP was evaluated by the logistic regression, and then the correlation coefficients (r^2) of all SNP pairs were calculated. The computation was performed in PLINK software (version 1.07) [45]. In the second step, we implemented the gene-based analysis in the open-source tool: Knowledge-Based Mining System for Genome-Wide Genetic Studies (KGG, version 2.0) [46] based on the association test results and LD files obtained from PLINK. Simes procedure (GATES) was employed in the gene-based association test [7]. Specifically, assume that m SNPs are assigned to a gene; an association test such as through the traditional logistic regression or linear regression is used to examine the association between the phenotype and each single SNP. This step yields m P values for m SNPs. GATES combines the available m P values within a gene by using a modified Simes test to give a gene-based P value. The summary P value is defined as

$$P_G = \mathrm{Min}\left(\frac{m_e p_{(j)}}{m_{e(j)}}\right), \quad (1)$$

where $p_{(j)}$ is the jth smallest P value among the m SNPs; m_e is the effective number of independent P values among m SNPs within the gene, and $m_{e(j)}$ is the effective number of independent P values among the top j SNPs. The effective number of independent P values was derived by accounting for the LD structure among the specified SNPs; we refer to [7] on the calculation.

In order to compare the performance of the SNP-based and gene-based methods, in the SNP-based method, we selected those SNPs whose P values were less than $1.0E-5$ and then mapped them into the corresponding genes. This allows us to compare the susceptible genes identified by both methods discussed above.

3. Results

3.1. Detecting Susceptibility Loci at the Relaxed Significance Level. Table 1 summarizes the susceptible genes identified by the SNP-based association test and gene-based association test at the relaxed significance level. In total, 207 genes passed the relaxed gene-based threshold, whereas only 64 genes with SNPs passed the relaxed SNP-based threshold.

Next, we performed a literature search on the genetic regions which contain the identified genes and filtered the

Table 1: Summary statistics for susceptibility loci identified by gene-based method and SNP-based method.

	Alcohol		Cocaine		Marijuana		Nicotine		Opiates		Other	
	G	S	G	S	G	S	G	S	G	S	G	S
Black men	4	3	4	1	6	2	5	2	8	2	9	5
Black women	4	3	8	5	9	3	7	3	3	1	6	3
White men	16	3	9	2	10	3	4	1	11	3	3	1
White women	20	5	12	2	10	2	11	1	4	5	24	3

G refers to gene-based method. S refers to SNP-based method.

Table 2: Summary of the candidate genes identified by the gene-based and SNP-based methods.

Chr	Gene	Source	P value (gene-based)[a]	Min P value (SNP-based)[b]	Detected SD[c]	Reported SD	Reference
1	KIAA0040	White women	$3.75E-05$	$2.60E-06$	Alcohol	Alcohol	[27, 44]
2	HAAO	White women	$4.40E-04$	$3.02E-05$	Cocaine	Alcohol	[41]
2	NCK2	Black men	$2.70E-06$	$1.10E-07$	Opiates	NA	NA
3	SH3BP5	White men	$1.20E-04$	$4.24E-06$	Cocaine	Alcohol	[13]
4	MANBA	White men	$4.63E-04$	$3.47E-05$	Alcohol	Alcohol	[29]
7	RELN	White men	$8.53E-04$	$5.32E-06$	Cocaine	Smoking	[37]
8	CSMD1	Black women	$1.23E-02$	$8.50E-06$	Nicotine	Smoking	[37, 38]
11	LRP5	White men	$4.01E-05$	$1.58E-06$	Opiates	Smoking	[42]
11	PKNOX2	White women	$1.84E-04$	$2.20E-06$	Alcohol	Alcohol	[13, 27, 41]
12	IFNG	White women	$1.16E-04$	$1.57E-05$	Opiates	Smoking	[37]
18	FAM38B	Black women	$9.24E-04$	$5.61E-06$	Cocaine	Smoking	[40]
18	PTPRM	Black women	$2.21E-3$	$9.50E-06$	Marijuana	Alcohol	[43]
22	MAPK1	Black women	$2.79E-04$	$3.52E-05$	Marijuana	Alcohol	[25]

[a]P value (gene-based): the P value obtained by the gene-based association test;
[b]min P value (SNP-based): the minimal P value of the SNPs within the corresponding gene;
[c]SD: substance dependence.

susceptible genetic regions which have been reported to associate with substance dependence for further investigation. In Table 2, we listed the filtered genes, their associated substance dependence type, the P values for the gene-based method, the minimal P value of SNPs within a gene, and their literature references and reported substance dependence.

In Figure 1, we plot the filtered genes obtained from the SNP-based and gene-based analyses by the position on the chromosomes against their log-transformed P values, $-\log_{10}(P)$. Each point for the SNP-based analysis in Figure 1 corresponds to the smallest SNP-based P value within the gene.

Overall, five genes, NCK2 (opiates dependence in black men), SH3BP5 (cocaine dependence in white men), LRP5 (opiates dependence in white men), KIAA0040 (alcohol dependence in white women), and PKNOX2 (alcohol dependence in white women), were identified by both the SNP-based and gene-based methods as meeting either of the relaxed significance levels for a specific dependence and within a gender-racial group. Four genes, MAPK1 (marijuana dependence in black women), MANBA (alcohol dependence in white men), HAAO (cocaine dependence in white women), and IFNG (opiates dependence in white women), met the threshold by the gene-based method only. We found that the significant signal of gene MAPK1 was mainly driven by SNPs: rs7290469 ($P = 3.25E-5$), rs9610271 ($P = 4.19E-5$), rs9610417 ($P = 5.38E-5$), and rs2876981 ($P = 7.51E-5$). The P values for these SNPs are slightly greater than the relaxed SNP-based threshold ($P < 1.0E-5$), and hence the SNP-based method failed to detect them.

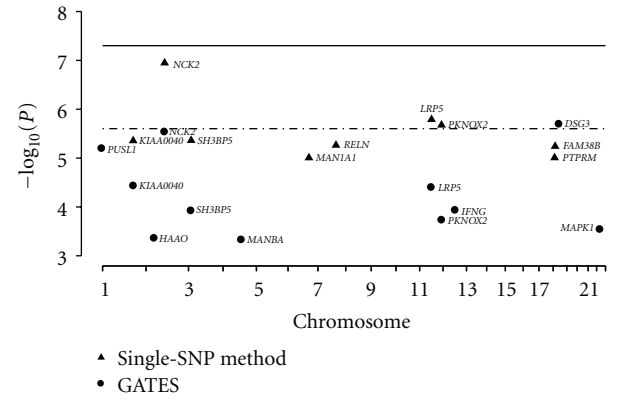


Figure 1: Comparison of candidate genes associated with substance dependence by the SNP- and gene-based analyses. A triangle represents the −log10 transformed P value of the marked gene from the gene-based analysis, and a dot represents the –log10 transformed the minimal P value of the SNPs within the marked gene. The solid and dashed ones are the genome-wide thresholds of SNP- and gene-based significance, respectively.

Table 3: Summary of genome-wide significant genes at the gene level (P value $< 1.0E-5$) and their top SNPs with P value $< 1.0E-3$.

Population	Substance dependence	Gene	Gene's P value	Top SNPs	SNP's P value
Black men	Opiates	NCK2	$2.70E-06$	rs2377339	$1.10E-07$
				rs7589342	$1.45E-04$
				rs12995333	$1.89E-04$
				rs12053259	$2.31E-04$
				rs6747023	$3.90E-04$
				rs879900	$7.72E-04$
White men	Nicotine	DSG3	$1.99E-06$	rs6701037	$1.20E-07$
				rs1057302	$3.93E-07$
				rs6425323	$2.94E-04$
				rs1057239	$3.35E-04$

Furthermore, four other genes, FAM38B (cocaine dependence in black women), PTPRM (marijuana dependence in black women), CSMD1 (nicotine dependence in black women), and RELN (cocaine dependence in white men), contain at least one SNP that met the SNP-based relaxed threshold of significance. The gene-based P values for FAM38B, PTPRM, and RELN are $9.27E-4$, $2.21E-3$, and $8.53E-4$, respectively, which are greater than yet at the same order as the relaxed threshold (P value $< 5.0E-4$). For CSMD1, 1,934 SNPs were mapped into it. Its signal was mainly determined by only five SNPs: rs2624087 (P value $= 8.50E-6$), rs4875371 (P value $= 4.0E-4$), rs2623607 (P value $= 6.89E-4$), rs10503267 (P value $= 7.22E-4$), and rs4875372 (P value $= 8.18E-4$). Because there were only 5.3% of the SNPs (103 SNPs) with P value less than 0.05, the overall association from the gene became less significant.

3.2. Genome-Wide Significant Loci. Since none of the SNPs attained the genome-wide significance for any dependence by the SNP-based method, in this section we will only focus on the results from the gene-based method.

Table 3 presents the genes with gene-based P value $< 1.0E-5$. This method identified one genome-wide significant gene, DSG3 (P value $= 1.99E-6$) for nicotine dependence in white men. The P value of gene NCK2: $2.70E-6$ is very close to the genome-wide significant threshold, which provided very strong evidence for the association of opiates in black men. As shown in Table 3, both NCK2 and DSG3 contained SNPs with strong signals; they are rs2377339 (P value $= 1.09E-7$) for NCK2 gene and rs6701037 (P value $= 1.20E-7$) and rs1057302 (P value $= 3.93E-7$) for DSG3 gene. However, none of these SNPs reached genome-wide significance.

4. Discussion

In this paper, we thoroughly analyzed the SAGE data from the SNP-based and gene-based methods, and compared the results obtained from these two methods. Specifically, for each sex-racial group, we performed association analysis for the six categories of substance dependence separately. The gene-based method appears to be more powerful in detecting susceptibility loci.

Most of the genes identified in our study are supported by various reports in the literature related to the genetics of substance dependence [47, 48]. Based on some of the genes that we identified, here common genetic variants among different substance dependencies may exist [49].

Overall, we did not detect any genome-wide significant SNP when using the SNPs-based method. However, one gene, DSG3, is genome-wide significantly ($P = 2.70E-6$) associated with nicotine dependence in the white men, according to the gene-based method. Another gene, NCK2, is nearly genome-wide significant ($P = 2.7E-6$) in its association with substance dependence.

The SNP-based method and gene-based method are closely related. In fact, the SNP-based method can be viewed as a gene-based method using the extreme function, namely, the minimal P value of the SNPs within a gene, whereas the typical gene-based method uses a weighted approach. The advantages and limitations of these two approaches are similar to those between the extreme function and a weighted average.

We should point out that both the SNP-based and gene-based methods have their own advantages and disadvantages. The SNP-based method has its unique strength in identifying genes with only a small number of significant SNPs. However, since the SNP-based method focuses on a single SNP at a time, it is less powerful to detect a gene whose SNPs have weak marginal effects, but a strong joint effect. In our analysis, 207 genes passed the relaxed gene-based threshold, whereas only 64 genes passed the relaxed SNP-based threshold.

Both the SNP-based and gene-based methods can be conducted conveniently in commonly available software, such as PLINK [45] for the SNP-based method and KGG [46] for the gene-based method. For the SNP-based analysis, PLINK is the most convenient platform. For the SAGE GWAS data, it took about 25 minutes to do a genome-wide SNP scan on a regular desktop computer (Intel Core 2, 4 GB Memory). In our gene-based analysis, we used the SNP-based association results and the linkage disequilibrium (LD) files from PLINK as the input to the KGG software. After this preparation, it took about 30 minutes to perform the gene-based association scan with the same desktop as mentioned above.

Authors' Contributions

X. Guo and Z. Liu are contributed equally to this work.

Acknowledgments

This work was supported by Grant R01 DA016750-09 from the National Institute on Drug Abuse. Funding support for the Study of Addiction: Genetics and Environment (SAGE) was provided through the NIH Genes, Environment and Health Initiative (GEI) (U01 HG004422). SAGE is one of the genome-wide association studies funded as part of the Gene Environment Association Studies (GENEVA) under GEI. Assistance with phenotype harmonization and genotype cleaning, as well as with general study coordination, was provided by the GENEVA Coordinating Center (U01 HG004446). Assistance with data cleaning was provided by the National Center for Biotechnology Information. Support for the collection of datasets and samples was provided by the Collaborative Study on the Genetics of Alcoholism (COGA; U10 AA008401), the Collaborative Genetic Study of Nicotine Dependence (COGEND; P01 CA089392), and the Family Study of Cocaine Dependence (FSCD; R01 DA013423). Funding support for genotyping, which was performed at the Johns Hopkins University Center for Inherited Disease Research, was provided by the NIH GEI (U01HG004438), the National Institute on Alcohol Abuse and Alcoholism, the National Institute on Drug Abuse, and the NIH contract "High throughput genotyping for studying the genetic contributions to human disease" (HHSN268200782096C). The datasets used for the analyses described in this paper were obtained from dbGaP at http://www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/study.cgi?study_id=phs000092.v1.p1 through dbGaP accession number phs000092.v1.p. The authors have no conflict of interests.

References

[1] M. I. McCarthy, G. R. Abecasis, L. R. Cardon et al., "Genome-wide association studies for complex traits: consensus, uncertainty and challenges," *Nature Reviews Genetics*, vol. 9, no. 5, pp. 356–369, 2008.

[2] P. R. Burton, D. G. Clayton, L. R. Cardon et al., "Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls," *Nature*, vol. 447, no. 7145, pp. 661–678, 2007.

[3] D. J. Hunter, P. Kraft, K. B. Jacobs et al., "A genome-wide association study identifies alleles in FGFR2 associated with risk of sporadic postmenopausal breast cancer," *Nature Genetics*, vol. 39, no. 7, pp. 870–874, 2007.

[4] F. Dudbridge and A. Gusnanto, "Estimation of significance thresholds for genomewide association scans," *Genetic Epidemiology*, vol. 32, no. 3, pp. 227–234, 2008.

[5] E. E. Eichler, J. Flint, G. Gibson et al., "Missing heritability and strategies for finding the underlying causes of complex disease," *Nature Reviews Genetics*, vol. 11, no. 6, pp. 446–450, 2010.

[6] T. A. Manolio, F. S. Collins, N. J. Cox et al., "Finding the missing heritability of complex diseases," *Nature*, vol. 461, no. 7265, pp. 747–753, 2009.

[7] M. X. Li, H. S. Gui, J. S. H. Kwan, and P. C. Sham, "GATES: a rapid and powerful gene-based association test using extended Simes procedure," *American Journal of Human Genetics*, vol. 88, no. 3, pp. 283–293, 2011.

[8] J. Z. Liu, A. F. McRae, D. R. Nyholt et al., "A versatile gene-based test for genome-wide association studies," *American Journal of Human Genetics*, vol. 87, no. 1, pp. 139–145, 2010.

[9] X. Guo, Z. Liu, X. Wang, and H. Zhang, "Genetic association test for multiple traits at gene level," *Genetic Epidemiology*. In press.

[10] K. Wang, M. Li, and H. Hakonarson, "Analysing biological pathways in genome-wide association studies," *Nature Reviews Genetics*, vol. 11, no. 12, pp. 843–854, 2010.

[11] B. M. Neale and P. C. Sham, "The future of association studies: gene-based analysis and replication," *American Journal of Human Genetics*, vol. 75, no. 3, pp. 353–362, 2004.

[12] B. Lehne, C. M. Lewis, and T. Schlitt, "From SNPs to genes: disease association at the gene level," *PLoS ONE*, vol. 6, no. 6, Article ID e20133, 2011.

[13] L. J. Bierut, A. Agrawal, K. K. Bucholz et al., "A genome-wide association study of alcohol dependence," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 107, no. 11, pp. 5082–5087, 2010.

[14] X. Chen, K. Cho, B. H. Singer, and H. Zhang, "The nuclear transcription factor PKNOX2 is a candidate gene for substance dependence in European-origin Women," *PLoS ONE*, vol. 6, no. 1, Article ID e16002, 2011.

[15] J. Clarimon, R. R. Gray, L. N. Williams et al., "Linkage disequilibrium and association analysis of α-synuclein and alcohol and drug dependence in two American Indian populations," *Alcoholism: Clinical and Experimental Research*, vol. 31, no. 4, pp. 546–554, 2007.

[16] D. M. Dick, H. J. Edenberg, X. Xuei et al., "Association of GABRG3 with alcohol dependence," *Alcoholism: Clinical and Experimental Research*, vol. 28, no. 1, pp. 4–9, 2004.

[17] H. J. Edenberg, D. M. Dick, X. Xuei et al., "Variations in GABRA2, encoding the α2 subunit of the GABA a receptor, are associated with alcohol dependence and with brain oscillations," *American Journal of Human Genetics*, vol. 74, no. 4, pp. 705–714, 2004.

[18] H. J. Edenberg and T. Foroud, "The genetics of alcoholism: identifying specific genes through family studies," *Addiction Biology*, vol. 11, no. 3-4, pp. 386–396, 2006.

[19] H. J. Edenberg, J. Wang, H. Tian et al., "A regulatory variation in OPRK1, the gene encoding the κ-opioid receptor, is associated with alcohol dependence," *Human Molecular Genetics*, vol. 17, no. 12, pp. 1783–1789, 2008.

[20] T. Foroud, L. F. Wetherill, T. Liang et al., "Association of alcohol craving with α-synuclein (SNCA)," *Alcoholism: Clinical and Experimental Research*, vol. 31, no. 4, pp. 537–545, 2007.

[21] J. Gelernter, R. Gueorguieva, H. R. Kranzler et al., "Opioid receptor gene (OPRM1, OPRK1, and OPRD1) variants and response to naltrexone treatment for alcohol dependence: results from the VA Cooperative Study," *Alcoholism: Clinical and Experimental Research*, vol. 31, no. 4, pp. 555–563, 2007.

[22] T. Reich, "A genomic survey of alcohol dependence and related phenotypes: results from the Collaborative Study on the Genetics of Alcoholism (COGA)," *Alcoholism: Clinical and Experimental Research*, vol. 20, supplement 8, pp. 133A–137A, 1996.

[23] T. Reich, H. J. Edenberg, A. Goate et al. et al., "Genome-wide search for genes affecting the risk for alcohol dependence,"

American Journal of Medical Genetics, vol. 81, no. 3, pp. 207–215, 1998.

[24] J. Song, D. L. Koller, T. Foroud et al., "Association of GABAA receptors and alcohol dependence and the effects of genetic imprinting," *American Journal of Medical Genetics*, vol. 117, no. 1, pp. 39–45, 2003.

[25] B. Tabakoff, L. Saba, M. Printz et al. et al., "Genetical genomic determinants of alcohol consumption in rats and humans," *BMC Biology*, vol. 7, article 70, 2009.

[26] J. C. Wang, A. L. Hinrichs, S. Bertelsen et al., "Functional variants in TAS2R38 and TAS2R16 influence alcohol consumption in high-risk families of African-American origin," *Alcoholism: Clinical and Experimental Research*, vol. 31, no. 2, pp. 209–215, 2007.

[27] K. S. Wang, X. F. Liu, Q. Y. Zhang, Y. Pan, N. Aragam, and M. Zeng :, "A meta-analysis of two genome-wide association studies identifies 3 new loci for alcohol dependence," *Journal of Psychiatric Research*, vol. 45, no. 11, pp. 1419–1425, 2011.

[28] H. Zhang, H. R. Kranzler, B. Z. Yang, X. Luo, and J. Gelernter, "The OPRD1 and OPRK1 loci in alcohol or drug dependence: OPRD1 variation modulates substance dependence risk," *Molecular Psychiatry*, vol. 13, no. 5, pp. 531–543, 2008.

[29] G. Kalsi, P. H. Kuo, F. Aliev et al., "A systematic gene-based screen of chr4q22-q32 identifies association of a novel susceptibility gene, DKK2, with the quantitative trait of alcohol dependence symptom counts," *Human Molecular Genetics*, vol. 19, no. 20, Article ID ddq326, p. 4121, 2010.

[30] P. H. Kuo, G. Kalsi, C. A. Prescott et al., "Association of ADH and ALDH genes with alcohol dependence in the Irish Affected Sib Pair Study of alcohol dependence (IASPSAD) Sample," *Alcoholism: Clinical and Experimental Research*, vol. 32, no. 5, pp. 785–795, 2008.

[31] L. J. Bierut, "Genetic variation that contributes to nicotine dependence," *Pharmacogenomics*, vol. 8, no. 8, pp. 881–883, 2007.

[32] L. J. Bierut, P. A. F. Madden, N. Breslau et al., "Novel genes identified in a high-density genome wide association study for nicotine dependence," *Human Molecular Genetics*, vol. 16, no. 1, pp. 24–35, 2007.

[33] N. Caporaso, F. Gu, N. Chatterjee et al., "Genome-wide and candidate gene association study of cigarette smoking behaviors," *PLoS ONE*, vol. 4, no. 2, Article ID e4653, 2009.

[34] L. S. Chen, E. O. Johnson, N. Breslau et al., "Interplay of genetic risk factors and parent monitoring in risk for nicotine dependence," *Addiction*, vol. 104, no. 10, pp. 1731–1740, 2009.

[35] N. L. Saccone, S. F. Saccone, A. L. Hinrichs et al., "Multiple distinct risk loci for nicotine dependence identified by dense coverage of the complete family of nicotinic receptor subunit (CHRN) genes," *American Journal of Medical Genetics, Part B*, vol. 150, no. 4, pp. 453–466, 2009.

[36] S. F. Saccone, A. L. Hinrichs, N. L. Saccone et al., "Cholinergic nicotinic receptor genes implicated in a nicotine dependence association study targeting 348 candidate genes with 3713 SNPs," *Human Molecular Genetics*, vol. 16, no. 1, pp. 36–49, 2007.

[37] G. R. Uhl, T. Drgon, C. Johnson et al., "Genome-wide association for smoking cessation success: participants in the Patch in Practice trial of nicotine replacement," *Pharmacogenomics*, vol. 11, no. 3, pp. 357–367, 2010.

[38] G. R. Uhl, Q. R. Liu, T. Drgon et al., "Molecular genetics of successful smoking cessation: convergent genome-wide association study results," *Archives of General Psychiatry*, vol. 65, no. 6, pp. 683–693, 2008.

[39] R. B. Weiss, T. B. Baker, D. S. Cannon et al., "A candidate gene approach identifies the CHRNA5-A3-B4 region as a risk factor for age-dependent nicotine addiction," *PLoS Genetics*, vol. 4, no. 7, Article ID e1000125, 2008.

[40] M. J. Ahn, H. H. Won, J. Lee et al. et al., "The 18p11. 22 locus is associated with never smoker non-small cell lung cancer susceptibility in Korean populations," *Human Genetics*, vol. 131, no. 3, pp. 365–372, 2012.

[41] D. M. Dick, J. Meyers, F. Aliev et al., "Evidence for genes on chromosome 2 contributing to alcohol dependence with conduct disorder and suicide attempts," *American Journal of Medical Genetics, Part B*, vol. 153, no. 6, pp. 1179–1188, 2010.

[42] P. F. Giampietro, C. McCarty, B. Mukesh et al., "The role of cigarette smoking and statins in the development of postmenopausal osteoporosis: a pilot study utilizing the marshfield clinic personalized medicine cohort," *Osteoporosis International*, vol. 21, no. 3, pp. 467–477, 2010.

[43] G. Joslyn, A. Ravindranathan, G. Brush, M. Schuckit, and R. L. White, "Human variation in alcohol response is influenced by variation in neuronal signaling genes," *Alcoholism: Clinical and Experimental Research*, vol. 34, no. 5, pp. 800–812, 2010.

[44] L. J. Zuo, J. Gelernter, C. K. Zhang et al. et al., "Genome-wide association study of alcohol dependence implicates KIAA0040 on chromosome 1q," *Neuropsychopharmacology*, vol. 37, no. 2, pp. 557–566, 2012.

[45] S. Purcell, B. Neale, K. Todd-Brown et al., "PLINK: a tool set for whole-genome association and population-based linkage analyses," *American Journal of Human Genetics*, vol. 81, no. 3, pp. 559–575, 2007.

[46] http://bioinfo1.hku.hk:13080/kggweb//.

[47] K. R. Merikangas, M. Stolar, D. E. Stevens et al., "Familial transmission of substance use disorders," *Archives of General Psychiatry*, vol. 55, no. 11, pp. 973–979, 1998.

[48] W. R. True, A. C. Heath, J. F. Scherrer et al., "Interrelationship of genetic and environmental influences on conduct disorder and alcohol and marijuana dependence symptoms," *American Journal of Medical Genetics*, vol. 88, no. 4, pp. 391–397, 1999.

[49] M. D. Li and M. Burmeister, "New insights into the genetics of addiction," *Nature Reviews Genetics*, vol. 10, no. 4, pp. 225–231, 2009.

The Scientific World Journal
Volume 2012, Article ID 435257, 12 pages
doi:10.1100/2012/435257

The ScientificWorldJOURNAL

Review Article

A Review of Integration Strategies to Support Gene Regulatory Network Construction

Hailin Chen and Vincent VanBuren

Department of Medical Physiology, Texas A&M HSC College of Medicine, Temple, TX 76504, USA

Correspondence should be addressed to Vincent VanBuren, vanburen@tamu.edu

Received 4 October 2012; Accepted 25 November 2012

Academic Editors: R. Jiang, W. Tian, J. Wan, and X. Zhao

Gene regulatory network (GRN) construction is a central task of systems biology. Integration of different data sources to infer and construct GRNs is an important consideration for the success of this effort. In this paper, we will discuss distinctive strategies of data integration for GRN construction. Basically, the process of integration of different data sources is divided into two phases: the first phase is collection of the required data and the second phase is data processing with advanced algorithms to infer the GRNs. In this paper these two phases are called "structural integration" and "analytic integration," respectively. Compared with the nonintegration strategies, the integration strategies perform quite well and have better agreement with the experimental evidence.

1. Introduction

1.1. Conventional Strategies of Building GRNs. Biological functions comprise numerous reactions at all levels of biological organization, including cells, tissues, organs, and body, and interchange with the environment. Overall, every life phenomenon found in this multilevel system is supported through many reactions interconnecting with each other to compose the orchestra of life. It is, therefore, crucial to have a systematic perspective in biomedical research. To gain an overview of such a complex system, we can visualize it in the form of a network. For instance, protein-protein interactions, metabolic reactions, and genetic regulations correspond respectively to a protein-protein interaction network (PPI), metabolic network, and gene regulatory network (GRN), which are subnetworks of the complex multi-level system. In the representation of a network, nodes typically correspond to molecules, while edges represent the relationships between nodes. In the study of biological networks, GRNs are one of the most popular models, especially in the field of development. Developmental GRNs provide important clues to elucidate the temporal and spatial dynamics of gene expression during development. The use of sea urchin and *Drosophila* has led to some of the greatest successes in studying developmental GRNs to explain complex developmental processes [1, 2]. Traditionally, the first step is to identify putative regulatory genes through genome-wide screening, such as expression microarrays, across distinct temporal and spatial states. Quantitative PCR is used afterwards to verify specific expression patterns [3]. Amazingly, the gene repertoire used in the control of development is relatively conserved across species, and thus regulatory genes can be identified by sequencing-based homology alignments [4]. As a central objective of modeling developmental GRNs is to identify the epistatic relations among these regulatory genes, the second step is to define experiments to perturb/activate the system and examine the responses via loss-of-function and gain-of-function experiments [3]. In a sea urchin GRN study, perturbation with morpholino-substituted antisense oligonucleotides (MASOs) was the main approach [5]. Rescue experiments are also an important part of this step. Finally, by assembling findings from many individual experiments, investigators may establish the developmental GRN. Validation of the established GRN can be accomplished precisely via mutagenesis of regulator binding sites for their target genes to observe the abolishment of the regulatory effect [6, 7].

Elucidation of gene regulation in the endomesoderm specification in the sea urchin and in the development of *Drosophila* embryos provides potent examples of the type

of complexities revealed by the study of GRNs. In the sea urchin embryo, *blimp1* is autorepressive when its product accumulates to high levels. At the same time, it provides a required input for *Wnt8* expression, which produces a positive feedback effect for *blimp1* via inducing *Tcf* to activate *blimp1* expression. *Wnt8* can infect the adjacent cells/territories with this circular bioinformation flow via diffusion. This flow is terminated due to *blimp1* autorepression [8]. In the early development of the *Drosophila* embryo, *Snail* repressor activates the synthesis of *Delta* ligand in the ventral mesoderm via repressing the transcription of *Tom*, an inhibitor of the *Delta*, which is called a double-negative gate. *Delta* triggers *Notch* signaling in the adjacent cells via diffusion. However, transcription of the *Notch* signaling target genes is repressed by the intraterritorial *Snail* repression in the ventral mesoderm itself. An exactly parallel mechanism causing transcriptional alternation inter-territorially is also found in the sea urchin skeletogenic mesoderm [1, 2]. Despite such accomplishments, there is still a large portion of the overall GRN in animal models that has not been defined. The laborious approach to elucidating GRNs from experiments for every node and every edge produces reliable biological information as prior knowledge to support novel findings. However, due to the complications in GRNs as discussed above, elucidating the complete GRN of complex eukaryotic organisms with respect to the whole genome would be extremely difficult using this strategy, as much time and labor are required even for just one conditional state. The strategy described above is the bottom-up approach of network construction. Computational strategies offer a top-down approach to network construction that complements what is described above.

1.2. Computational Strategies for Building GRN

1.2.1. Nonintegration Strategies. During this blooming period of biomedical research, high-content experimental data is fuelling systems biology research, such as GRN construction at the genome-wide scope. For example, expression microarrays that can detect the relative abundance of gene transcripts by comparing two or more biological samples are commonly used for GRN construction. The new approaches provide a perspective on the global molecular interactions that bridge the gap between the external signal and internal response. There are several popular algorithms being used to construct GRNs from expression data (reviewed in [9]).

In the graphical representation of GRNs, nodes typically represent genes corresponding to the transcription factor proteins or target genes, while edges represent the regulations between the transcription factors and their targets. Boolean networks describe each element as a variable with the value 0 or 1 to represent the state of the element as "off" or "on," respectively. A Boolean network $G(V, F)$ is defined by a set of nodes corresponding to genes $V = \{x_1, \ldots, x_n\}$ and a list of Boolean functions $F = (f_1, \ldots, f_n)$ describes how genes in the network change their state (on or off) from one time point to the next. The future state of an element is completely determined by the states of other elements (regulators) by means of the underlying logical Boolean functions.

Second, Bayesian networks model the biomedical network with a *directed acyclic graph*. "*Directed*" means that there are arrows to indicate causal influences, and "*acyclic*" means that causal loops are prohibited. For each element, a conditional distribution $P(\mathrm{Xv} \mid \mathrm{parents(Xv)})$ is defined through the application of the conditional probability table (CPT), where parents(Xv) denotes the variables corresponding to the regulators of this element. Thereafter, an optimization approach is applied, with the Bayesian information Criteria (BIC) optimized to infer the best fitting network model among a finite set of models.

In a third alternative, differential equations extract the network from high-throughput experimental data through taking the instantaneous concentration of each element into consideration. The instantaneous concentration of each element is completely determined by the concentration (x_n) of other elements involving a regulation function.

Differential equation modeling:

$$\frac{dx_i}{dt} = f_i(x_1, \ldots, x_n, t). \tag{1}$$

In a fourth alternative, coexpression is used to model GRNs based on co-variance analysis. However, the comparison between the covariances from datasets having different scales would be difficult. The Pearson correlation coefficient addresses this difficulty. It measures the coexpression between any two elements across a series of states resulting in the value with the range from -1 to 1, which allows networks to be established based on a certain threshold for the magnitude of the correlation.

Finally, Mutual Information (MI) offers another approach to modeling GRNs based on the probability theory. The mutual dependence of any two elements in the network is measured using MI. It is reported that MI outperforms the correlation in some studies [10, 11]. Using a reasonable threshold, networks will be accurately constructed. Context likelihood of relatedness (CLR) [10, 12], MRNet (maximum relevance/minimum redundancy network) (R package), and ARACNE (algorithm for the reconstruction of accurate cellular networks) [11, 13] are the three representative strategies of network construction applying MI. Numerous approaches to GRN construction have been developed using various combinations of the five main approaches described above.

1.2.2. Motivations for an Integration Strategy. The most popular algorithms contributing to the construction of GRNs from genomic expression data were described above. However, each of them has certain drawbacks. The Boolean algorithm assigns each variable a binary value, which could omit important information of continuous variables. Bayesian network construction is promising for representing and inferring causal relationships, but this strategy is only effective for the construction of small GRNs, due to the superexponential increase in the algorithm running time for large networks. The differential equation algorithm requires knowledge of the equation of dynamics and parameter estimation to optimize the GRN model against real data. However, deriving an appropriate equation of dynamics remains a challenge. Furthermore, solving a differential equation

system of any realistic complexity is difficult. As to the correlation and mutual information algorithms, manually setting appropriate thresholds without a principled reference poses difficulties. Strategies applying algorithms with these drawbacks are not satisfying; therefore, it motivates us to improve the computational strategies. New strategies continue to be developed against those difficulties. It is a great challenge to refurbish algorithms to improve GRN construction using genomic expression data. Improvements are difficult to obtain algorithmically; however, the integration of multiple types of genome-wide datasets with literature-based information of regulation as prior knowledge is a straightforward alternative to offer improvement. Generally, in the computational GRN construction methods mentioned above, only genomic expression data like microarray data is used to produce the desired network applying one of the algorithms described [10, 11]. Based on a straightforward intuition that more relevant information generates better confidence for making correct predictions, we are optimistic about the prospects of making improvement by data integration. We have increasing availability of genome-wide data with respect to every aspect of biology, genomic expression data, genome sequences, proteomic data, genome-wide protein-DNA binding site data [14], genomic SNPs, and high-content data collections created from various types of biological or pathological research objectives. Therefore, with reference to the literature-based information of regulation as the prior knowledge and the multiple types of genome-wide datasets available as analyzable data, an integration strategy can offer an excellent opportunity for elucidating complete GRNs.

2. Integration Strategies for Building GRNs

2.1. Sources for Integration. The past few decades were an age of rapid progress in the development of biomedical science. Numerous advanced technologies along with well-founded theories lead the way for new findings in industrial and academic biomedical research. For example, biomedical investigators have developed genomic expression by microarray, rapid genome and microbiome sequencing, proteome definition by mass spectrometry, genome-wide protein-DNA binding site definition by ChIP-seq, genomic SNP identification by SNP array, and high-content knowledge by literature mining. Overwhelmed with such impressive quantity of genome-wide achievements, we are encouraged to apply strategies to make good use of them intuitively, such as integrating them properly for GRN construction. First we need to take stock of the status of the biomedical sources that are available to us.

It is difficult to summarize all the biomedical sources as most sources are scattered in distinct research papers. We will focus our attention on databases, as they are an effective form of rearranging and storing sources for specific objectives. Nucleic Acids Research (NAR) summarizes the biomedical database status each year (Figure 1) (http://nar.oxfordjournals.org/). Here is a table (Table 1) summarizing some genome-wide databases popular in the research of systems biology.

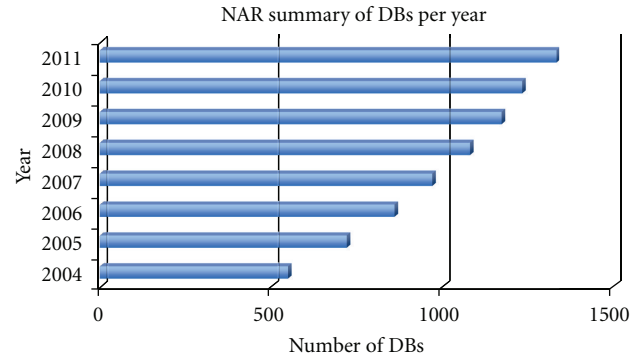


Figure 1: This is the database (DB) summary from NAR database issues. Each bar represents the total number of databases identified by NAR that year.

Table 1: Prominent databases.

Category	Databases
Metabolic pathways	KEGG, ENZYME
Signaling pathways	KEGG, WikiPathways
Protein-protein interactions	BIND, STRING
Transcription factor binding motifs	JASPAR, TRANSFAC
Genetic interaction networks	BIND, BioGRID
Gene expression	GEO, ArrayExpress
Sequences	UCSC Genome Browrer
Protein-compound interactions	DrugBank, STITCH, ResNet, CLiBE
Gene-disease associations	OMIM

2.2. Structural Integration. A large number of genome-wide sources are available that have not been fully leveraged to infer novel GRNs. Before entering into a discussion of the analytic algorithms for integrating multiple genome-wide datasets for GRNs construction, we must first address the challenge of extracting the desired datasets from the ocean of biomedical sources. Structural integration retrieves desired datasets from multiple heterogeneous sources to facilitate querying the data for further analytic integration. There are many sophisticated approaches being used for structurally integrating target datasets through programmatic extraction and recombination. Overall, these approaches to structural integration can be divided into three general categories: warehouse integration, mediator-based integration, and navigational integration [15].

Before discussing the approaches to structural integration in the following Sections 2.2.1–2.2.3, we will finish this section with a discussion of some key defining characteristics of structural integration.

Variety of Data. This describes the typical data that can be integrated and includes high-throughput datasets, molecular structures, molecular interactions, molecular pathways, Gene Ontology annotation, and disease characteristics, hence *vertical integration* is the aggregation of *semantically*

similar data from multiple heterogeneous sources, while *horizontal integration* is the composition of *semantically complementary* data from multiple heterogeneous sources [15].

Heterogeneity of Descriptive Terms. Semantics is the study of the relation between form and meaning. Each source of data or knowledge may refer to the same semantic concept or field with its own descriptive term or identifier, which can lead to a semantic confusion between the many sources. Conversely, some sources may use the same term to refer to the different semantic concepts. Semantic mapping is indispensable in order to match descriptive terms or identifiers among multiple heterogeneous sources or between the sources and the objective integrated datasets.

Heterogeneity of Naming and Identity. One major hurdle in current data integration efforts is the issue of naming and identity such that a variety of aliases (e.g., synonyms for gene symbol) exist for many genes, proteins, and keywords. Alias mapping through lookups is critical for retrieving desired data from multiple heterogeneous sources.

2.2.1. Warehouse Integration. Warehouse integration arranges desired datasets from multiple sources into a local warehouse (e.g., a local database) before querying, through loading the required data from distinct sources and converting them into standard formats before being stored locally. Relying less on the Internet connectivity to access data limits the impact of various problems such as access restrictions, network bottlenecks, low response times, and the occasional unavailability of sources. Moreover, using local warehouses allows for improved accuracy, efficiency, and flexibility for the subsequent query as it is performed locally. However, this integration has an important drawback of the overall system maintenance. It is expensive to have the warehouse updated regularly to reflect those modifications of heterogeneous external sources [15, 16]. Furthermore, since the data retrieved and stored in the warehouse will eventually be converted into the warehouse-specific format every time the warehouse is updated, the semantic structure of the warehouse database may need to be reformatted often.

NCBI, the UCSC Genome Browser [17], and EMBL-EBI (http://www.ebi.ac.uk/) are three representative data warehouses. Given the appeal of these resources, efforts are increasingly made to improve the warehouse strategy against its drawbacks. The GeNS platform is one of the efforts to improve the efficiency of database maintenance. GeNS is a biological data integration platform for warehouse integration [18]. Representative databases were selected to cover a broad area of biomedical research when constructing the GeNS database. This warehouse accommodated the data from EMBL-EBI, UniPort (Swissport and TrEMBL), ExPASy (PROSITE and ENZYME), NCBI (Entrez, Taxonomy, Pubmed, RefSeq, GeneBank, and OMIM), Biomart, ArrayExpress, InterPro, Gene Ontology, KEGG (genes, pathways, orthology and drugs), and PharmGKB (genes, drugs, and diseases). A loader application responsible for converting the corresponding data from each source database into the format compatible with GeNS schema was designed to coordinate tasks such as alias mapping. In order to overcome the difficulty of maintenance, a general schema and a specific schema were both developed in GeNS. To physically store the data, a general model (general schema) that certified the framework of the database was used, while supporting this general model with a concrete meta model (specific schema) where all the entities and relations from a specific contributing database were specified locally [18]. Therefore, the addition/modification of databases into this warehouse needs modification in the meta model only, rather than in the general model.

2.2.2. Mediator-Based Integration. Mediator-based integration retrieves desired datasets from multiple heterogeneous sources at the time of querying through query translation, as opposed to the data translation that is manifested at the time of database creation in warehouse integration [15, 16]. The mediator, or core of the query translation, is an interface responsible for reformulating a query given by the user into the queries accommodating the local schemas of the underlying data sources via a single mediated schema defined by the mediator-based integration platform. Therefore, a mapping is required in the mediated schema to capture the semantic relation or the identity alias' relation between the sources and the given query, which thus allows the query made by a user to be translated via the mediator into the appropriate queries onto the individual sources. This correspondence mapping is a crucial step in creating the mediator, as it will influence the query reformulation and the addition of new sources to or the removal of the old sources from the integrated system.

There are two main approaches for establishing the mediator, global-as-view (GAV) and local-as-view (LAV) [15, 16]. The GAV has the mediator that translates the given queries directly into the formats of the source queries. The LAV has the format of query in every source defined into the common format of mediation, which is defined by the mediator via a wrapper. Therefore, each local source needs a wrapper component that exports a view of the local data into a common format of mediation via mediated schema. Since the mediator-based integration retrieves data at the run-time of querying, the problems such as access restriction, network bottlenecks, low response time, and the occasional unavailability of sources may occur. However, since the queries are performed in the real-time fashion, there is no special need of system maintenance via manually updating the databases. More specifically, LAV makes it very simple to add or to remove sources, while for GAV the addition or removal of sources is much more difficult, as it requires a modification of the mediated schema on the correspondence mapping.

The mediator approach is a very popular approach of data integration. Platforms like K2, TAMBIS, Discovery-Link, and BACIIS are all designed based on this approach. In the Discovery-Link platform (http://www.redbooks.ibm.com/abstracts/sg246290.html/), the source-specific wrapper symbolizes its data sources for further integration.

2.2.3. Navigational Integration. To extract the desired datasets, navigational integration follows the workflow in which the query outputs from a source are redirected as the query inputs to the next resource until the requested information is reached [15, 16]. It resembles the nature of the web in the context of increasing number of data sources, and it, therefore, frees users from manually browsing several web pages or data sources in order to obtain the desired datasets. However, the drawbacks of the navigational integration are similar to those of the mediator-based integration, such as access restrictions, network bottlenecks, low response times, and the occasional unavailability of sources. Additionally, the time and effort required to build the correspondence mapping are still costly.

Examples of this approach are Entrez and DiseaseCard databases. DiseaseCard [19] is a web-based collaborative service that aims to comprehensively integrate genetic and medical information, including the information of rare genetic diseases.

2.2.4. Choosing a Method for Structural Integration. Here is a brief comparison (Table 2) that summarizes the features of different structural integration approaches of extracting desired datasets from the ocean of biomedical sources.

The main purpose of the structural integration in most cases is to compile all available information for specific objectives to prepare for arbitrary analytic integration according to the user interest.

An ideal integration schema should have the following characteristics.

(1) Efficient. It can optimize the time that users need to finish the query. One of the recent ideas is to build semantic webs.

(2) Easy to maintain.

(3) Stable.

(4) System performance metrics. It is critical for an integration system to study source statistics in order to refine the query plans and improve the overall functionality and performance of the system. The essential statistics that should be learned are the coverage of sources, the average response time, the cost, and the overlap between sources [15].

(5) High quality. The data integrated are extracted from various heterogeneous sources, having different degrees of quality. For example, compared with the old data, new data from improved technologies may have better quality; also, compared with computationally predicted data, the experimental data is expected to have better quality. Quality varies within heterogeneous data sources, and some effort to account for these differences should be considered in the data integration strategies.

(6) Automated. the disciplines of operational optimization and machine learning should be applied for an effective automation program.

2.3. Analytic Integration. Along with the desired datasets extracted from multiple heterogeneous sources through structural integration, analytic integration is performed to infer GRNs via data integration algorithms applied to the desired datasets. The integration algorithm is, therefore, an essential ingredient for optimizing GRN construction. In contrast with the algorithms described in the above section, the integration algorithm needs to be capable of dealing with multiple types of data simultaneously. As a result, heterogeneous data should merge smoothly regardless of the differences in data types. As we discussed previously in Section 1, many types of genome-wide datasets could contribute to GRN construction. In the following discussion of the analytic integration for GRN construction from multiple types of genome-wide datasets or with reference to prior knowledge, there are three main schemas to consider: naïve Bayesian applications, supervised learning, and network topology applications. Each of these schemas represents a distinct approach to analytic integration, yet each can be applied to multiple categories of hypothesis inference, such as transcriptional regulation, protein-protein interaction, and gene-disease association.

2.3.1. Naïve Bayesian Applications. The Bayesian schema applying the naïve Bayesian is specified in the biological context: if association of two molecules occurs across multiple heterogeneous sources, there is an increased likelihood that they have a strong connection that may, for example, include a productive regulation or an indispensable physical interaction. Therefore, the functional importance of the pairwise connection is evaluated through its incidence across the multiple sources. And many types of genome-wide datasets, such as genomic expression and phylogenetic profiles, will contribute to the perceived functional importance of the pairwise connections in the genome-wide scope. Therefore, a scoring system is then applied to evaluate the functional importance of the pairwise connections in the genome-wide scope to gain insight about the confidence of the inferred GRNs or PPIs. Two successful examples with naïve Bayesian applications are described below.

The STRING web application was designed to infer the PPI via integrating multiple types of genome-wide datasets. It was primarily constructed from the integration of three genome-wide datasets, including phylogenetic profiles, a database of transcription units, and a database of gene-fusion events [20–24]. Phylogenetic profiles are derived from the evolutionary tree. During evolution, functionally linked proteins tend to be either preserved or eliminated in new species simultaneously. This property of correlated evolution is leveraged in the STRING database by characterizing each protein via its phylogenetic profile that records the presence or absence of an orthologous protein in every known genome. Those proteins having matching profiles have a strong tendency to be functionally linked. Transcriptional units (operons) are extracted from a number of genomes through identifying the conserved gene clusters. The protein products of the genes in transcriptional units are hypothesized to be functionally linked with each other. Gene-fusion

Table 2: Properties of distinctive structural integration approaches.

Approach	Maintenance	System stability	Effectiveness
Warehouse	Difficult, costly	Stable	Poor
Mediator-based	Easy for LAV	Depends on source availability, accessibility, traffic	Fair
Navigational	Easy	Depends on source availability, accessibility, traffic	Good

events can be interpreted by the example that the interacting proteins GyrA and GyrB subunits of *E. coli* DNA gyrase are orthologs of a single fused chain (topoisomerase II) in yeast; thus, the similarities of GyrA and GyrB to some segment of topoisomerase II might be used to predict their functional interaction in *E. coli*. STRING was developed as a multi-dimensional integration interface by combining its three original components (phylogenetic profiles, transcription units, and gene fusions) together with genomic expression and genome-wide dataset of protein-protein interaction discovered via text mining from PubMed abstract, and so forth. Putative protein-protein interaction of the PPI can be evaluated with the confidence score of functional association between two proteins across those genome-wide datasets. Different datasets are weighted differently for their respective contribution to the confidence score. In the STRING project, a weight was assigned to each dataset by benchmarking the performance of the prediction in this dataset against a common reference set of trusted knowledge. The developers chose the functional grouping of proteins maintained at KEGG (Kyoto Encyclopedia of Genes and Genomes) as the common reference set. The benchmark weight of each dataset in STRING corresponded to the probability of finding the linked proteins that were predicted in this dataset within the same KEGG pathway. In the equation of the confidence score, the confidence score is taken as S, the weight of each dataset is taken as S_i, and i is the number of qualfied datasets with incidence of the pairwise connection. Therefore, the confidence score of the putative protein-protein interaction is evaluated through qualifying the naive Bayesian probability of the incidence of the corresponding protein connection across those multiple datasets under the assumption of independence of the various datasets. Larger confidence scores indicate higher confidence in a functional protein-protein association:

$$S = 1 - \prod_i (1 - S_i), \tag{2}$$

where S_i is the weight assigned to each dataset over the common reference set.

Figure 2 shows an example result of a STRING query (http://STRING-db.org/) of the protein-protein interactions seeded by Gata4, a well-known transcription factor in cardiac development.

The confidence score for each putative protein-protein interaction is, therefore, a Bayesian-probability-like score supported by several types of genome-wide datasets. Putative PPI thus follows the evaluation in the genome-wide scope to gain confidence.

Another approach applies the Bayesian schema to rationally extend the ribosome biogenesis pathway in yeast [25]. Li et al. constructed a computational predictor for inferring the ribosome biogenesis genes by integrating multiple heterogeneous datasets into a probabilistic model. This model employed a naive Bayesian probabilistic scoring system to integrate the multiple genome-wide datasets, including genomic expression, a genome-wide dataset of protein-protein interactions derived from literature curation, a genome-wide dataset of high-throughput yeast two-hybrid assays, a genome-wide dataset of affinity purification coupled with mass spectrometry, a genomic interaction dataset, and *in silico* genome-wide interaction datasets into a network (Figure 3). The plausibility that a putative yeast gene belongs to the ribosome biogenesis pathway was evaluated by calculating the naive Bayesian probability of the incidence of its association with the known ribosome biogenesis genes in the pathway. The ROC plot from cross-validation was employed to check the effectiveness of this schema (Figure 3). The top-scoring 212 genes were manually selected for the further experimental validation.

Bayesian schemas that apply the naive Bayesian probability are a powerful approach for analytic integration. Their application in the improves network construction in the examples given by evaluating the putative network with multiple genome-wide datasets integrated to calculate the confidence score. This schema always outperforms non-integration strategies. For example, the application of the Bayesian schema in an algorithm called MAGIC, as compared with the expression-based clustering methods, predicted more true positives than clustering methods did relative to the number of false positives [26].

2.3.2. Supervised Learning. Supervised learning assumes that partial information is known for predictor variables and outcomes, and this partial information is leveraged to make deeper inferences of the target hypothesis. The known information is taken as the prior knowledge. Supervised approaches in statistics have been developed to make new inferences with the prior knowledge of the study objective to be integrated with the other relevant datasets. The accuracy of inferences regarding network topology is positively correlated with the amount of accurate prior knowledge. In contrast, unsupervised approaches have the problem that they are more likely to predict associations that are unreliable. The supervised learning schema can make inferences less error-prone. One analytic integration approach uses supervised learning to integrate the prior knowledge of the PPI with the other relevant genome-wide datasets to improve the effectiveness of PPI construction.

Kato et al. developed a schema for supervised learning of yeast PPI using known protein-protein interactions as a prior

Predicted functional partners of Gata4	Neighborhood	Gene fusion	Cooccurence	Homology	Coexpression	Experiments	Databases	TextMining	Combined_score
NKX2-5						✓		✓	0.998
Zfpm2						✓		✓	0.986
Gip							✓	✓	0.982
Tbx20								✓	0.979
Id2						✓		✓	0.943
Lhcgr								✓	0.939
Tbx5								✓	0.929
Lhx9								✓	0.915
Hey2								✓	0.914
Edn1								✓	0.912

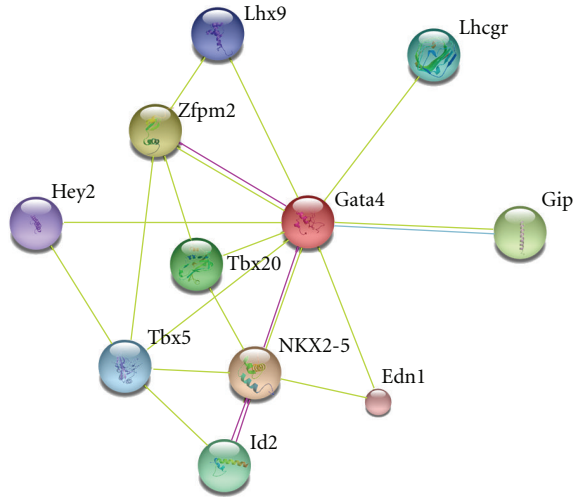


Figure 2: STRING search results for Gata4 from different sources. The network figure is the protein-protein interaction image from the search results (*Mus musculus*). Higher scores indicate greater confidence in the putative interaction. Here the highest confidence is given to NKX2-5 as an interactive partner of Gata4, as this is supported with experimental evidence.

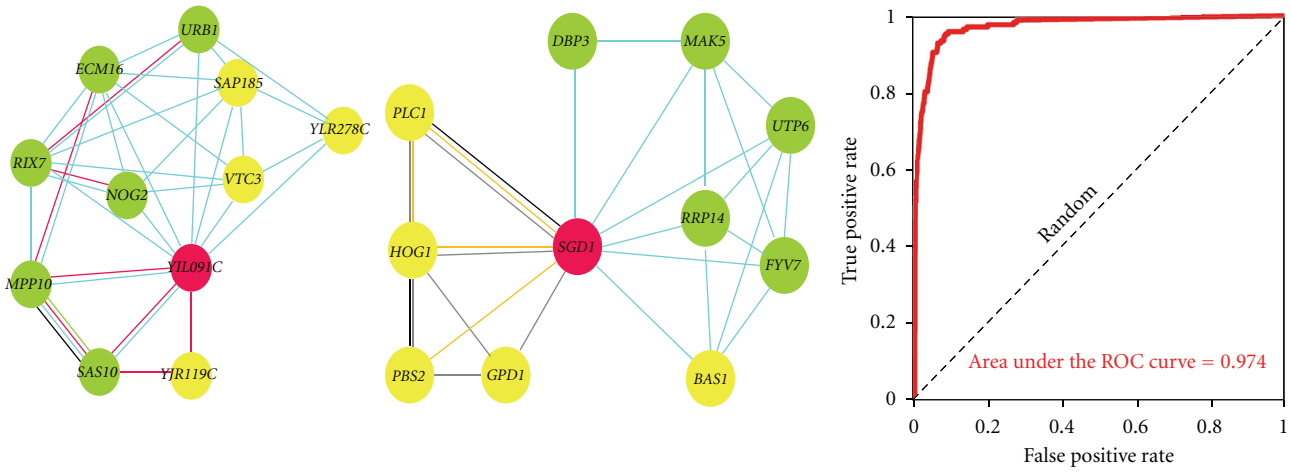


Figure 3: Predicted ribosome biogenesis genes are labeled as red nodes. Green nodes are the known ribosome biogenesis genes, and yellow nodes are genes that are not related to the ribosome biogenesis. Edge color indicates coexpression (light blue), affinity purification (red), yeast two-hybrid assay (green), genetic interaction (yellow), cocitation (gray), and literature curation (black). The ROC curve shows cross-validated recovery of the known ribosome biogenesis genes based on their network connectivity to one another. (This open-access figure was reproduced from Li et al., [25].)

knowledge to be integrated with the other relevant genome-wide datasets [27]. In this supervised network construction, a kernel matrix is applied as the basis of the integration. A kernel matrix is a matrix of similarity, and edges in the kernel-based network are assigned to the connected nodes whose kernel values (similarity) are above a certain threshold δ. The kernel matrix representation is an appropriate method for supervised PPI construction, as the network construction problem boils down to the problem of inferring an integrated kernel matrix of pairwise protein connections from combining the known yeast protein-protein interactions with the other relevant genome-wide datasets. Here, Kato et al. generated 3 main steps of the yeast PPI construction applying supervised learning.

Step 1. They translated the prior knowledge (known part of yeast PPI) into the kernel matrix by diffusion kernels. Diffusion kernels are functions for processing the network structure to mine the underlying relationships between nodes in the kernel matrix. However, this resulted in a regional kernel matrix of pairwise protein connections given a genome-wide scope because only the pairwise kernel values (the intensity of pairwise protein associations) of the proteins that were in the known part of PPI could be reconstructed.

The regional kernel matrix could approximately recover the known PPI when the appropriate threshold δ of the kernel value was applied.

Step 2. A genome-wide dataset (e.g., genomic expression) with the same objective can be used to establish a new kernel matrix. Kato et al. took multiple types of genome-wide datasets into consideration for the PPI construction. They combined these new generated kernel matrices, each of which was calculated from a particular genome-wide dataset, such as genomic expression and genome-wide phylogenetic profiles, into a combined kernel matrix of pairwise protein associations in yeast.

Step 3. They integrated the combined kernel matrix with the regional kernel matrix of the known part of yeast PPI to infer the integrated kernel matrix of pairwise protein connections that offered the pairwise kernel values in the genome-wide scope to be able to qualify the PPI edges via comparing the kernel values against the threshold δ.

Accuracy of edge prediction was measured by a 10-fold cross-validation. With setting the parameter of the degree of kernel diffusion to 3.0 when translating the known PPI into the kernel matrix by diffusion kernels, the ROC score was 0.929 for the inferred yeast PPI.

The supervised learning improves the PPI construction via integrating the experimentally-proven evidence of the study objective as the supervisor into the analysis of the other relevant genome-wide datasets. Thus the GRNs construction can also apply the schema of the supervised learning via having the known transcription factor-target gene regulations as the prior knowledge to be integrated with the other expression-relevant genome-wide datasets. One study compared supervised methods with unsupervised methods for GRN construction and found that the supervised methods are more reliable than the unsupervised ones [28].

2.3.3. Network Topology Applications. In recent decades, a large amount of experimental evidence about biological networks has been collected, and this was coupled with progress in elucidating the network topological features. Approaches that have contributed to these strides in network biology include scale-free networks, small world networks, adaptive motifs, feed-back motifs, "AND" and "OR" logic motifs, and modular networks. Therefore, a systematic effort utilizing the network topological features will be required and will benefit the effectiveness of network construction. Modularity is one of the most accepted network topological features of GRNs. The modularity of GRNs can be represented by gene module members that are co-regulated via shared transcription factors combinatorially binding their promoters. Therefore, members in such gene modules manifest coexpression. Genomic expression and the genome-wide transcription factor-DNA binding sites are thus, integrated into GRN construction by identifying coexpressed genes with conserved TF binding sites in their promoters [29, 30]. Two examples applying this integration schema to the inference of GRNs are discussed below.

GRAM is an algorithm for discovering GRNs by incorporating information from transcription factor (TF) binding motifs, genome sequence, and genomic expression [31]. Regulatory relationships are effectively identified by genome-wide location analysis of DNA-binding TFs via blasting the corresponding TF binding motifs against promoter sequences to infer the binding sites at the genome-wide scope. However, location analysis may infer potential physical interactions between TFs and DNA at the genome-wide scope but may not necessarily identify functional bindings. Integrating the location analysis with genomic expression, GRAM employs an effective and exhaustive strategy for GRN construction. It searches over all the possible combinations of TFs indicated by location analysis. When the binding sites are in close proximity, the corresponding TFs are defined to be in combination. A TF's combinations are used to identify its regulating gene set members that have common combinations of TFs binding their promoters as defined by location analysis. From the complete gene set, a subset is generated by members that have highly correlated expression in the expression dataset. The subset is taken as the "seed" of a gene module. Then GRAM revisits the genomic expression to add more genes having relatively high correlated expression with the "seed" into the gene module using less strict criteria (Figure 4). GRAM allows genes to belong to more than one module. Regulation is, therefore, inferred between the co-expression module and its TFs combination to foster GRN construction.

In the GRAM project, this schema was applied to the TF binding motif data of 106 TFs and over 500 microarray expression experiments in *Saccharomyces cerevisiae*. The GRN was reconstructed via identification of modules. Gene modules were also identified as groups of genes annotated with similar pathways. Identified gene modules were controlled by more than one TF, which was the evidence for inferring the TFs' interactions (protein-protein interactions). GRAM can assign different regulators to genes with similar expression patterns, which cannot be accomplished using the expression clustering methods alone. Moreover, by applying the enrichment test of specific DNA binding motifs, genes in the discovered modules are more likely to be coregulated when compared with the set of genes obtained using genomic location analysis alone.

Another application of this integration schema in GRNs construction was developed by Segal et al. [32]. Those authors designed an algorithm integrating a *Saccharomyces cerevisiae* genomic expression dataset with the genome-wide TF binding sites that were inferred via searching the corresponding TF binding motifs in the genome-wide scope. In their framework, a regulatory module was a set of genes that were regulated in concert by a shared regulation program. A regulation program specified the expression of the genes in the module as a function of the expression of a small set of regulators (Figure 5). After the enrichment test of TF binding motifs to the regulatory module, novel regulations were predicted between the TFs corresponding to the overrepresented binding motifs and the regulatory module to foster the GRN construction. Segal et al. found in many regulatory modules that the TFs corresponding to

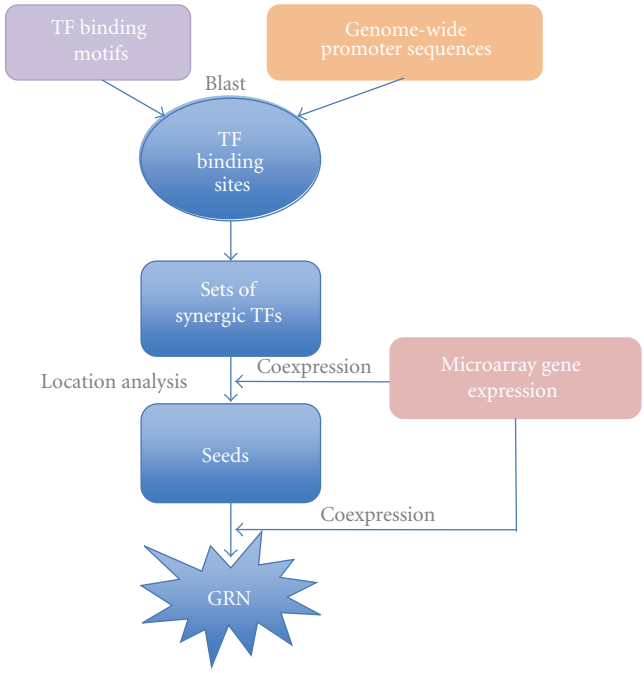


Figure 4: Workflow of the GRAM algorithm. Known TF binding motifs are blasted against the promoter sequences in the genome-wide scope to infer the corresponding TF binding sites. A set of synergic TFs is identified when the TFs' binding sites are close to each other. Regulated gene sets are defined by the corresponding sets of synergic TFs through location analysis. A "seed" of a gene module is selected from the regulated gene set based on the highly correlated expression. Then GRAM revisits the genomic expression to add more genes with closely correlated expression with the "seed" into the gene module of the "seed." The GRN construction is fostered by the established regulations between the coexpression gene modules and their corresponding sets of synergic TFs.

the overrepresented binding motifs of the module matched the known regulators of the genes in that module quite well.

Applying the modularity feature in GRNs construction via integrating genomic expression with genome-wide TF binding sites improves the quality of network construction. However, only limited information has been elucidated about the GRN topological features. The schema with GRN topology applied is expected to perform more compellingly with increasing knowledge of those features in GRN.

2.3.4. Choosing a Method of Analytic Integration. The Bayesian application schema for the naive Bayesian probability theorem is well accepted in most scientific fields. The naive Bayesian integrates multiple types of relevant genome-wide datasets into a scoring system that produces a confidence score for the inferred network (e.g., PPI and GRN). However, there is an important caveat with this approach: it is rational to apply the naive Bayesian theorem only when the situation satisfies the basic assumption that each type of source dataset is independent of any other. Therefore, under this assumption, there is no dependency between any two types. However, in reality some datasets have known dependencies. For example, in the case of STRING, the datasets of experiments, databases, and text mining are not completely independent of each other. The method of evaluating individual weight is also a controversial part of this schema. In the case of STRING, KEGG is used as the standard for calculating the weights. However, KEGG is an incomplete database in the genome-wide scope, and it is actually constructed from various experiments, databases, and text-mining resources, so it is necessarily dependent on those resources. It is, therefore, not a good standard, as it is biased—giving high weights to its own resources while giving low weights to the others. This may promote its accuracy but limit its predictive power. Hence, naïve Bayesian applications in GRN construction may be affected by those limitations.

Supervised learning integrates prior knowledge of the study objective with the other relevant genome-wide datasets to learn the networks (e.g., PPI, GRN). However, the quality of its prediction varies with the quantity of the prior knowledge. Also, when multiple datasets are involved, weighting each dataset properly is still problematic. If we employ the nonweighting integration approach to make the primary prediction of the unknown part before it is trained by the prior knowledge, we may have better quality on the overall prediction even when the quantity of the prior knowledge is relatively small.

The schema of network topology is a compelling strategy of GRN construction via integrating genomic expression with genome-wide TF binding sites. It associates the two sources through the modularity feature to connect the gene co-expression with the conserved TF binding sites on their promoters. However, as mentioned in the two examples, the TF binding sites are inferred from the corresponding TF binding motifs via a genome-wide blast. It will be improved when the CHIP-seq datasets regarding different TFs are employed instead to generate the genome-wide TF binding sites. It is a developing schema that keeps step with the development of our knowledge of network topological features.

The schemas of supervised learning and network topology application may be described as advanced forms of the schema of Bayesian application, progressing from the naive to evidence-based logic. These approaches use principled and logical integration of datasets rather than integration only. Along with the increased experimentally proven knowledge about regulatory relationships, the schema of network topology application can be combined with supervised learning to gain increased confidence in the inferred GRNs. Overall, a positive-feedback effect that contributes to better GRNs helps to develop our knowledge of additional GRN topological features, while the more topological features provide more or better clues for GRNs' construction. The PPI could be embedded into the GRN to assess the TFs' combinatory regulations.

3. Summary and Future Directions

GRN construction via integration of multiple types of genome-wide datasets or via literature-based information

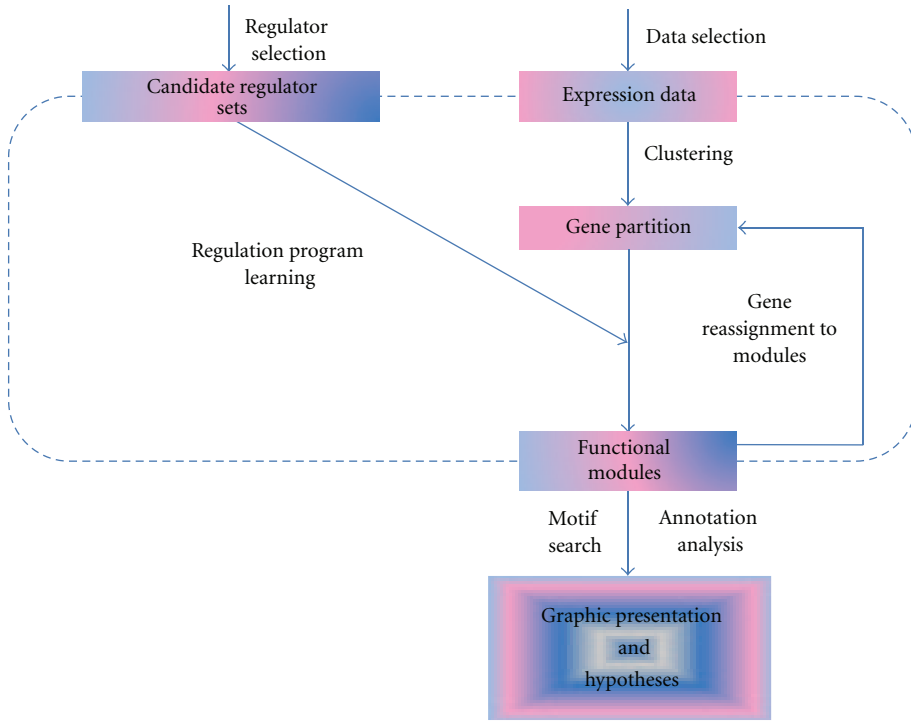


Figure 5: Workflow of the algorithm by Segal et al. This is an iterative procedure with application of the expectation maximization (EM) algorithm. In the maximization step (M step), genes are partitioned into modules that result from previous clustering upon genomic expression data and the best regulation program is learned for each module. In the E step, the best regulation programs corresponding are compared with each gene module to determine the optimal predictor (the optimal predictive regulation program). The module corresponding to the best predictor is selected and genes are reassigned to this module. The regulatory program learning stops on convergence. Secondly, TFs are associated with the regulatory module via an enrichment test of their corresponding binding motifs to the module.

about regulation as the prior knowledge partly avoids or overcomes the drawbacks of the nonintegration strategies. Along with the continuous increase in the availability of new data sources, new opportunities emerge for us to use integration strategies to construct GRNs. There are two main categories of integration strategies: *structural integrations* for extracting and recombining the required datasets and *analytic integrations* for processing the queried datasets to infer GRNs. There are three main types of structural integration: *warehouse integration* naively aggregates the required datasets into local storage before data querying, *mediator-based integration* establishes a mediator application to retrieve the required datasets via reformatting the user's query into the formats of queries in local data sources at the time of data processing, and the *navigational integration* follows the chain of data querying at the time of data processing via using the query outputs of one step in the process as query inputs in a next step. In a subsequent analytic integration, the schema of Bayesian applications use the naive Bayesian probability to integrate multiple types of genome-wide datasets into a scoring system to compute a confidence score for inferred GRNs. Supervised learning integrates the prior knowledge of the study objective with the other relevant genome-wide datasets to learn the GRNs. And the schema of network topology applications integrate genomic expression with genome-wide TF binding sites through the modularity feature to connect gene co-expression with conserved TF binding sites in their promoters to foster the GRNs construction. Overall, the integration strategies perform well and reliably as compared to the non-integration strategies. Structural integration and analytic integration take central roles in the overall integration strategy of GRN construction.

Recently, cooperation of traditional experimental approaches with computational approaches has energized biomedical research. These new approaches offer the ability to computationally infer novel hypotheses from prior knowledge and relevant datasets to guide experimentation by setting research priorities. A salient example of this successful cooperation defines how to rationally extend the ribosome biogenesis pathway in yeast [25]. After revealing 212 candidates from the Bayesian applied integration analysis of multiple relevant genome-wide datasets, experiments were employed to validate their findings. Li et al. identified 15 previously unreported ribosome biogenesis genes (TIF4631, SUN66, YDL063C, JIL5, TOP1, SGD1, BCP1, YOR287C, BUD22, YIL091C, YOR006C/TSR3, YOL022C/TSR4, SAC3, NEW1, and FUN1). Segal et al. used a similar workflow to validate the GRN construction [32]. Therefore, GRNs inferred from the analysis with multiple types of integrated datasets offer a sophisticated atlas for setting research priorities.

Abbreviations

BIC: Bayesian Information Criteria
EMBL: European Molecular Biology Laboratory
GO: Gene ontology
GRN: Gene regulatory network
KEGG: Kyoto Encyclopedia of Genes and Genomes
NAR: Nucleic Acids Research
NCBI: National Center for Biotechnology Information
PCR: Polymerase chain reaction
PPI: Protein-protein interaction
SNP: Single-nucleotide polymorphism.

Glossary

Bayesian information criterion (BIC): In statistics, it is a criterion for model selection among a finite set of models
Cis-regulatory motif: A nucleotide pattern that is widespread and has a biological significance for regulatory factor binding
ChIP-seq: A technology that combines chromotin immunoprecipitation (ChIP) with massive sequencing to identify the binding sites of DNA-associated proteins on a genomic scale
Cross-validation: A technique for assessing how the results of a statistical analysis will generalize to an independent data set
Epistatic: In genetics, it is the phenomenon where the effects of one gene are modified by one or several other genes
Gene ontology: A controlled vocabulary for annotating genes and gene products
Gene regulatory network: A network that summarizes gene regulatory influences in a biological process
In silico: ALatin expression used to mean "performed on computer" or "computer simulation"
Mesoderm: In all bilaterian animals, the mesoderm is one of the three primary germ cell layers in the early embryo
Operon: In genetics, an operon is a functioning unit of genomic DNA containing a cluster of genes under the control of a single regulatory signal or promoter
Phylogenetic profile: Also called phylogenetic tree, is a branching diagram or "tree" showing the inferred evolutionary relationships among various biological species or other entities based upon similarities and differences in their physical and/or genetic characteristics
Schema: A representation of a plan, theory, or data structure, normally expressed as an outline or model
Semantic: Relating to meaning language or logic; in a biological context, this usually refers to the meaning of specifically defined annotations, concepts, or logical relationships between biological entities
Wrapper: A computer program that translates one format of data to another or a computer program that simplifies user interactions with a more complex program.

Acknowledgments

The authors thank David C. Zawieja, Jerry Trzeciakowski, and Xu Peng for their thoughtful comments on the paper.

References

[1] E. H. Davidson and M. S. Levine, "Properties of developmental gene regulatory networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 51, pp. 20063–20066, 2008.
[2] M. Levine and E. H. Davidson, "Gene regulatory networks for development," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 14, pp. 4936–4942, 2005.
[3] E. Li and E. H. Davidson, "Building developmental gene regulatory networks," *Birth Defects Research Part C*, vol. 87, no. 2, pp. 123–130, 2009.
[4] D. H. Erwin and E. H. Davidson, "The last common bilaterian ancestor," *Development*, vol. 129, no. 13, pp. 3021–3032, 2002.
[5] E. H. Davidson, J. P. Rast, P. Oliveri et al., "A genomic regulatory network for development," *Science*, vol. 295, no. 5560, pp. 1669–1678, 2002.
[6] D. Calva, F. S. Dahdaleh, G. Woodfield et al., "Discovery of SMAD4 promoters, transcription factor binding sites and deletions in juvenile polyposis patients," *Nucleic Acids Research*, vol. 39, no. 13, pp. 5369–5378, 2011.
[7] K. S. Zaret, J. K. Liu, and C. M. DiPersio, "Site-directed mutagenesis reveals a liver transcription factor essential for the albumin transcriptional enhancer," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 87, no. 14, pp. 5469–5473, 1990.
[8] J. Smith, C. Theodoris, and E. H. Davidson, "A gene regulatory network subcircuit drives a dynamic pattern of gene expression," *Science*, vol. 318, no. 5851, pp. 794–797, 2007.
[9] H. de Jong, "Modeling and simulation of genetic regulatory systems: a literature review," *Journal of Computational Biology*, vol. 9, no. 1, pp. 67–103, 2002.

[10] J. J. Faith, B. Hayete, J. T. Thaden et al., "Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles," *PLoS Biology*, vol. 5, no. 1, article e8, 2007.

[11] A. A. Margolin, I. Nemenman, K. Basso et al., "ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context," *BMC Bioinformatics*, vol. 7, supplement 1, article S7, 2006.

[12] A. Madar, A. Greenfield, E. Vanden-Eijnden, and R. Bonneau, "DREAM3: network inference using dynamic context likelihood of relatedness and the inferelator," *PloS One*, vol. 5, no. 3, article e9803, 2010.

[13] P. Zoppoli, S. Morganella, and M. Ceccarelli, "TimeDelay-ARACNE: reverse engineering of gene networks from time-course data by an information theoretic approach," *BMC Bioinformatics*, vol. 11, article 154, 2010.

[14] M. B. Gerstein, A. Kundaje, M. Hariharan et al., "Architecture of the human regulatory network derived from ENCODE data," *Nature*, vol. 489, no. 7414, pp. 91–100, 2012.

[15] T. Hernandez and S. Kambhampati, "Integration of biological sources: current systems and challenges ahead," *Sigmod Record*, vol. 33, no. 3, pp. 51–60, 2004.

[16] L. D. Stein, "Integrating biological databases," *Nature Reviews Genetics*, vol. 4, no. 5, pp. 337–345, 2003.

[17] P. A. Fujita, B. Rhead, A. S. Zweig et al., "The UCSC Genome Browser database: update 2011," *Nucleic Acids Research*, vol. 39, database issue, pp. D876–D882, 2011.

[18] J. Arrais, J. E. Pereira, J. Fernandes, and J. L. Oliveira, "GeNS: a biological data integration platform," *Proceedings of World Academy of Science, Engineering and Technology*, vol. 58, pp. 850–855, 2009.

[19] G. S. Dias, J. L. Oliveira, J. Vicente, and F. Martin-Sanchez, "Integrating medical and genomic data: a successful example for rare diseases," *Studies in Health Technology and Informatics*, vol. 124, pp. 125–130, 2006.

[20] L. J. Jensen, M. Kuhn, M. Stark et al., "STRING 8—a global view on proteins and their functional interactions in 630 organisms," *Nucleic Acids Research*, vol. 37, database issue, pp. D412–D416, 2009.

[21] B. Snel, G. Lehmann, P. Bork, and M. A. Huynen, "STRING: a web-server to retrieve and display the repeatedly occurring neighbourhood of a gene," *Nucleic Acids Research*, vol. 28, no. 18, pp. 3442–3444, 2000.

[22] C. von Mering, M. Huynen, D. Jaeggi, S. Schmidt, P. Bork, and B. Snel, "STRING: a database of predicted functional associations between proteins," *Nucleic Acids Research*, vol. 31, no. 1, pp. 258–261, 2003.

[23] C. von Mering, L. J. Jensen, M. Kuhn et al., "STRING 7—recent developments in the integration and prediction of protein interactions," *Nucleic Acids Research*, vol. 35, database issue, pp. D358–D362, 2007.

[24] C. von Mering, L. J. Jensen, B. Snel et al., "STRING: known and predicted protein-protein associations, integrated and transferred across organisms," *Nucleic Acids Research*, vol. 33, database issue, pp. D433–D437, 2005.

[25] Z. Li, I. Lee, E. Moradi, N. J. Hung, A. W. Johnson, and E. M. Marcotte, "Rational extension of the ribosome biogenesis pathway using network-guided genetics," *PLoS Biology*, vol. 7, no. 10, article e1000213, 2009.

[26] O. G. Troyanskaya, K. Dolinski, A. B. Owen, R. B. Altman, and D. Botstein, "A Bayesian framework for combining heterogeneous data sources for gene function prediction (in Saccharomyces cerevisiae)," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 100, no. 14, pp. 8348–8353, 2003.

[27] T. Kato, K. Tsuda, and K. Asai, "Selective integration of multiple biological data for supervised network inference," *Bioinformatics*, vol. 21, no. 10, pp. 2488–2495, 2005.

[28] L. Cerulo, C. Elkan, and M. Ceccarelli, "Learning gene regulatory networks from only positive and unlabeled data," *BMC Bioinformatics*, vol. 11, article 228, 2010.

[29] G. Pavesi, P. Mereghetti, G. Mauri, and G. Pesole, "Weeder Web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes," *Nucleic Acids Research*, vol. 32, web server issue, pp. W199–W203, 2004.

[30] G. Pavesi and G. Pesole, "Using Weeder for the discovery of conserved transcription factor binding sites," *Current Protocols in Bioinformatics*, chapter 2:unit 2.11, 2006.

[31] Z. Bar-Joseph, G. K. Gerber, T. I. Lee et al., "Computational discovery of gene modules and regulatory networks," *Nature Biotechnology*, vol. 21, no. 11, pp. 1337–1342, 2003.

[32] E. Segal, M. Shapira, A. Regev et al., "Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data," *Nature Genetics*, vol. 34, no. 2, pp. 166–176, 2003.

The Scientific World Journal
Volume 2012, Article ID 560101, 19 pages
doi:10.1100/2012/560101

The ScientificWorldJOURNAL

Research Article

In Silico Evolution of Gene Cooption in Pattern-Forming Gene Networks

Alexander V. Spirov,[1,2] Marat A. Sabirov,[2] and David M. Holloway[3]

[1] *Computer Science and CEWIT, SUNY Stony Brook, 1500 Stony Brook Road, Stony Brook, NY 11794, USA*
[2] *Laboratory of Evolutionary Modeling, The Sechenov Institute of Evolutionary Physiology and Biochemistry, Thorez Prospect 44, Saint Petersburg 2194223, Russia*
[3] *Mathematics Department, British Columbia Institute of Technology, 3700 Willingdon Avenue, Burnaby, BC, Canada V5G 3H2*

Correspondence should be addressed to Alexander V. Spirov, alexander.spirov@gmail.com

Received 29 September 2012; Accepted 13 November 2012

Academic Editors: R. Jiang, W. Tian, J. Wan, and X. Zhao

Gene recruitment or cooption occurs when a gene, which may be part of an existing gene regulatory network (GRN), comes under the control of a new regulatory system. Such re-arrangement of pre-existing networks is likely more common for increasing genomic complexity than the creation of new genes. Using evolutionary computations (EC), we investigate how co-option affects the evolvability, outgrowth and robustness of GRNs. We use a data-driven model of insect segmentation, for the fruit fly Drosophila, and evaluate fitness by robustness to maternal variability—a major constraint in biological development. We compare two mechanisms of gene cooption: a simpler one with gene Introduction and Withdrawal operators; and one in which GRN elements can be altered by transposon infection. Starting from a minimal 2-gene network, insufficient for fitting the Drosophila gene expression patterns, we find a general trend of coopting available genes into the GRN, in order to better fit the data. With the transposon mechanism, we find co-evolutionary oscillations between genes and their transposons. These oscillations may offer a new technique in EC for overcoming premature convergence. Finally, we comment on how a differential equations (in contrast to Boolean) approach is necessary for addressing realistic continuous variation in biochemical parameters.

1. Introduction

In the pregenomic era, it was a common assumption that complex organisms, with complex body plans and tissue types, had higher genetic complexity than simpler organisms; that unique features corresponded to unique genes. As more and more organisms have had their genomes sequenced, it has become apparent that there are enormous genetic similarities between organisms as diverse as vertebrates, corals and mollusks (see review in [1]). Even in a gene-counting sense, there is little correlation with organismal complexity [2], humans have somewhat more genes than fruit flies or nematodes, but less than pufferfish, cress, or rice [3–5]. If it is not novel genes, what, then, is the source of organismal diversity? It has become increasingly apparent that evolution acts chiefly on gene regulation, on the mechanisms by which a particular gene is expressed or repressed. In the genome, only a small percentage of DNA is involved in coding proteins. It has been realized that the far greater proportion of non-protein-coding DNA (formerly called "junk DNA") is of critical importance in gene regulation, containing, for example, binding sites for enzymes which activate or inhibit expression of particular genes [6].

The diverse organismal forms we see are generated during the process of development, proceeding from sexual or asexual reproduction to the adult form. Development depends critically on the regulated expression of genes at the correct times and places in order to create an organism's anatomy and physiology. While DNA is commonly referred to as the unique "blueprint" of an organism, current understanding suggests that DNA is far from a catalogue associating specific genes for specific tissue types; rather the genome (especially in eukaryotes) tends to code for relatively few, multifunctional, proteins (~20,000 in humans), along with

the markers which enable the genes for these proteins to be regulated in very complex ways. It is this regulation that enables genes to be expressed at the correct times and places. (See special feature on "Describing biology's dark matter" in the September 6, 2012, issue of *Nature*).

In evolutionary terms, this means that a picture is emerging that species (or novel methods for specifying tissues and developmental sequences) do not generally arise due to generation of novel genes. Rather, evolution tends to act on the regulatory sequences of the DNA. Insertion of new regulatory sequences can transfer transcriptional control of a pre-existing gene to other members of the genome [8, 9], and lead to novel patterns of gene expression [10–13]. Existing genes can become new regulators for other pre-existing genes. In developmental pathways, in which networks of genes interact to form particular tissues, co-option (also known as recruitment) of genes from other networks can result in novel dependencies between tissue types, or in new properties of a particular tissue [4, 5].

There are numerous documented cases now of the co-option of genes from one developmental stage to another. For instance, in fruit flies it has been shown how regulatory binding sites in the *yellow* gene were added evolutionarily to control pigmentation patterns in the wing [15]; in sea urchins co-option and optimization of a sequence adjacent to the *spec2a* gene have been elucidated [16]; in brain evolution, the genes involved in vertebrate neural crest cell migration and the midbrain/hindbrain boundary were present in the ancestral chordate—they were coopted into these new roles with the evolution of vertebrates [17]. See also [18, 19]. Indeed, it is commonly thought that early in metazoan evolution, gene networks specifying developmental events may have consisted of no more than two or three interacting genes. Over time, these were augmented by incorporating new genes and integrating originally distinct pathways [8]. In the not so distant past, evolutionary-development research focused on finding phylum-specific genes for phylum-specific features; this has more recently been challenged by evidence that the evolution of body plans proceeds by the changes in gene regulatory circuitries more than by gain or loss of genes [20–22]. Such considerations have led to the view that biological "evolution cannot be fully understood without understanding the evolution of developmental programmes" [23], and such concepts as *developmental reprogramming* [8, 24–26] have been developed to describe the processes lying between mutation and selection at the organismal level (i.e., from an altered gene product (protein) to a new phenotype). Reprogramming should be considered as an evolutionary mechanism because some ontogenetic changes may be promoted by existing developmental mechanisms while others are prevented [23, 27, 28]. It is likely that developmental constraints are powerful factors in the direction of evolutionary change [1, 23, 27, 28].

When considering the evolution of developmental programs, one needs to ask what the constraints are; that is, if change occurs in a gene regulatory network, by what measures is the new program tested with respect to its fitness? Development needs to be robust to numerous factors, such as environmental temperature, egg size, dosage of maternal regulatory molecules, intrinsic noise in gene expression, and variability in cell geometry and cell order. Developmental networks must optimize fitness to all of these challenges (and more). Strong genetic or environmental perturbations can induce increased phenotypic variance [29–32]. Waddington introduced the concept of canalization to describe how wild type (normal) development buffers against such perturbations, such that the developmental program tends very strongly to achieve a well-defined end result, despite perturbations which may cause some diversity in the paths that reach that end point. Quantitative experiments are beginning to demonstrate canalization in developmental sequences (diverse trajectories to a precise end point, e.g., [33, 34]). However, there are still large unknowns regarding what specifically makes given networks robust, and exactly how such networks have evolved. Since developmental events generally involve very complex gene network dynamics, frequently in concert with cell-cell communication and tissue mechanics, computational modeling is a key tool in understanding not only how such processes operate (for instance, to generate spatial domains of gene expression), but what in their dynamics confers robustness to diverse perturbations. In addition to studying gene network function, computation can allow us to test how networks arise through evolution. In concert with experimental data, this can address specific questions regarding how evolution operates on gene regulation, and how network evolution contributes to developmental robustness.

Early segmentation of the invertebrate body plan has long been very popular for studying the specifics of both developmental mechanisms and evolution [35, 36]. As reviewed in [36] it appears "that throughout evolution there was a parallel co-option of gene regulatory networks that had conserved ancestral roles in determining body axes and in elongating the anterior-posterior (AP) axis. Inherent properties in some of these networks made them easily recruitable for generating repeated patterns and for determining segmental boundaries. Phyla where this process happened (arthropods, annelids, and chordates) are among the most successful in the animal kingdom, as the modular nature of the segmental body organization allowed them to diverge and radiate into a bewildering array of variations on a common theme."

For instance, the Notch and *wnt* pathways have ancient roles in axis elongation. Discovered and most intensively studied in the fruit fly *Drosophila melanogaster*, these genes began forming periodic spatial patterns somewhere along the lineage to arthropods. These periodic patterns have now come to underlie segmentation in numerous phyla. Some genes such as *engrailed* (*en*) had a primitive role in neural patterning (which is also segmental) and appear to have been coopted to body axis segmentation (a classic example of boundary formation in *Drosophila* involves *en* and the *wnt* pathway). In fact, a number of the segmentation genes appear in both neurogenesis and segmentation [37, 38], including the *hunchback* (*hb*) and *Kruppel* (*Kr*) gap genes covered in more detail below [39–41]—it may be the nervous system that provides a large reservoir of useful components that have already been tested in gene networks. *Even-skipped* (*eve*), a gene upstream of *en* (in *Drosophila* development)

was involved in axial elongation and became coopted into segmentation, see [36]. *Caudal* (*cad*) is involved in axis elongation in many invertebrates, but was shown to have evolved a role in generating segmental periodicity in the centipede *Strigamia maritima* [42, 43]. This role for *cad* is clearly derived, but its recruitment to this role would have been facilitated by it already being expressed in the segmenting tissue at the correct time during development.

In insects, two distinct modes of segmenting the body have evolved. In primitive insects, such as the grasshopper, the short-germ band mode lays out body segments sequentially. Many more highly derived insects, such as flies, use the long germ band mode to establish all body segments simultaneously. This simultaneous mechanism must act quickly during development; it has been proposed that it evolved by co-option of new genes to the short-germ band mechanism, in order to maintain accurate regulation of patterned gene transcription over the whole embryo in a condensed time frame [8]. This complex task appeared to be solved by evolution in a short geological span at the sacrifice of, as a minimum, doubling the number of genes in the segmentation network. As doubling occurs, genes from other gene ensembles are often recruited into the network.

As well as the wealth of comparative information on the evolution of segmentation, the molecular biology and genetics of development are extremely well characterized in insects, particularly *Drosophila*. This has allowed for the development of very detailed quantitative models of the developmental mechanisms involved in segmentation. These models can be used to study both the functioning of the segmentation network, and how it evolved (in concert with comparative data between species); see [33, 34, 39, 44–48].

Segmentation in *Drosophila* is temporally hierarchical. In this paper, we focus on the earliest stages, the maternal, and gap gene patterning. Maternal factors (mRNAs and proteins) form monotonic concentration gradients along the AP axis. These products are transcriptional regulators, and their first targets are the zygotic gap genes, which form broad expression domains. We work with a gene circuit model of 4 trunk gap genes (adapted from [33, 34])—(*hb*; *Kr*; *giant*, *gt*; *knirps*, *kni*) under the control of the maternal Bicoid (Bcd) gradient. See the HOX pro Web resource [49, 50] for a catalogue of the known regulatory elements for the trunk gap gene ensemble. We model the central part of embryo only, from 34% to 82% egg length (%EL, head to tail), where contributions from terminal regulatory networks can reasonably be neglected (see [51, 52]). Models of this small network have been fit in detail to *Drosophila* expression data and serve as a starting point for exploring how this network may have evolved and what this can say in general about evolutionary mechanisms.

Our chief focus in this project is to study how gene recruitment affects network structure and dynamics, and what this implies for developmental robustness. The 4-gap gene model provides a small very well characterized network for investigating this. In this study, we specifically focus on robustness to maternal perturbations. That is, we test to what degree the gap gene expression patterns are robust to variability in the maternal Bcd gradient. In earlier work, we optimized gap gene models for robustness to naturally occurring levels of Bcd variability [33, 34]. In the current project, we use these models as a starting point for evolutionary computations to study gene recruitment. We can ask if the current *Drosophila* 4-gap network is optimal for this type of robustness, or whether recruitment of additional genes could increase robustness. We can also study how the 4-gene network may have evolved; for example, by starting with 2-gene models, we can study how these might have recruited genes into the current network. With computation, we can study many aspects of such a process. Does recruitment increase robustness? If so, what types of genes are recruited, that is, in what ways do they connect into the original network; what types of expression patterns might they have (do they recapitulate known patterns or form novel ones)? How fast does recruitment occur? (I.e., what is the relation between recruitment and evolvability?) However, at the same time that we want to address such questions regarding the evolution of development, many of the basics of how recruitment occurs are poorly understood. For instance, some studies assume that recruitment occurs occasionally, by chance, and is then subject to evolution; others believe that there are special evolutionary mechanisms for recruitment. By trying different recruitment scenarios, a computational approach can address such questions as do different means of recruitment lead to more robust networks, or larger networks, or faster evolving networks?

A large diversity of methods have been developed in recent years to model evolution. These range from techniques inspired by biological evolution but used for diverse optimization problems (e.g., in engineering), such as Genetic Algorithms (GA) and evolutionary computations (EC) generally [53], to techniques which have been developed specifically for studying the mechanisms of biological evolution (*in silico* evolution). A new research program in evolutionary systems biology is beginning to arise through the fusion of systems biology, network theory, and evolutionary theory. Within this, a number of groups have developed computational approaches for the evolution of gene regulatory networks (GRNs), evolving populations of individuals represented by dynamically modeled transcriptional regulatory networks. Some examples include work on the evolution of robustness and evolvability [7, 54–63], work on the mechanisms of genetic assimilation [64]; study of the role of network topology [65], and computational investigations into gene duplication and subfunctionalization [66]. Wagner's model in particular has helped elucidate why mutants often show a release of genetic variation that is cryptic in the wild type (Waddington's canalization), and how adaptive evolution of robustness occurs in genetic networks of a given topology [7, 54, 56, 60, 63]. Variants of this model have proven useful for studying the evolution of modularity in gene circuits [67] and the evolution of new gene activity patterns [61, 68, 69]. Also see [70–73].

In this work, we develop EC techniques for studying mechanisms of gene co-option. Following earlier work [71], one approach is to add Gene Introduction and Gene Withdrawal operators to a standard GA algorithm (running repeated cycles of mutation, selection, and reproduction). These give a probability to adding (or subtracting) a gene

to a given network (a random co-option event), along with its attendant connectivities to the other network genes. This approach is at the network level (i.e., with genes as the fundamental units, or network nodes). We have also developed approaches at the next level of detail, studying the evolution of regulation at the DNA level. At this level, we can begin to characterize the dynamics of particular mechanisms of regulatory evolution. For instance, we have improved GA optimization speeds by developing crossover operators inspired by the mechanisms of retroviral recombination [75, 76]. Here, we describe a technique to model genetic change due to transposons (also see [77–83]).

A great deal of the so-called junk DNA is comprised of intermediate repeats of DNA elements that are able to move (or transpose) throughout the genome. Transposons are ubiquitous and may comprise up to 45% of an organism's genome. Transposons jump between different parts of a genome to propagate themselves, and these events are usually to the detriment of their host [6]. Many transposons have a unique DNA site that acts as a forwarding address and directs the transposon to a complementary DNA site in its host genome [6]. It has been estimated that 80% of spontaneous mutations are caused by transposons [6]. Changes include the creation of novel genes, the alteration of gene expression in development, and the induction of major genomic rearrangements [84–86]. Transposable elements are ubiquitous among contemporary organisms and have probably existed since the dawn of life. Transposons can be viewed as parasites that have coevolved with their hosts, over time introducing useful variations into host genomes. Transposons are natural tools for genetic engineering [87]. Since transposons are likely to be active players in the rewiring of preestablished regulatory networks, we are interested in characterizing the dynamics of transposon-induced evolution of GRNs. We can imagine that transposons may be more effective agents of change than local random mutations, since transposons can deliver large sequences of DNA (whole genes or regulatory regions). By better understanding a major mechanism of biological evolution we may be better able to use it for optimization problems or for directed evolution experiments (e.g., see [88, 89]). Transposons also allow us to study the "arms-race" aspect of the coevolution of the host GRN and the "parasitic" transposons.

Our approach uses continuous PDE models of the evolving gap GRN (i.e., gene product concentrations and regulator strengths are represented as real numbers). Prior publications in GRN evolution have tended to use discrete models (e.g., Boolean approaches in which a gene is "on" or "off"). We are able to move beyond the knock-out mutations and/or abstract environmental stochasticity used in discrete models and address continuous variation in gene products, such as continuous variation in the maternal gradients. Through this, we can ask questions such as the following: how far can any particular component in the wild type be varied before an alternate phenotype is accessed; and with continuous variation, are the observed transitions between phenotypes continuous or discrete?

In this publication we consider four scenarios for gene cooption (Figure 1). Two of these are at the network level: *static determination of recruits*, in which an evolutionary search has a fixed number of potential genes to recruit from the population (i.e., all individuals have the same number of genes—two obligatory initial genes and N-2 recruits—and this stays constant during the evolutionary search); *dynamic addition of recruits*, in which the Gene Introduction and Withdrawal operators are used and produce gradual changes in population-average number of additional genes in the evolved population. The static determination provides a baseline against which to understand the effect of dynamic recruitment. At the more detailed level of DNA regulation, we consider two mechanisms for cooption, via transposons and transposition operators. These are as follows: *static transposon tests*, in which all individuals in a population keep the same transposon at the same location (in terms of the discussion below, Figure 2, this is the W^{a0} column of the W matrix) and the transposon permanently forces evolution (by keeping the W^{a0} elements predetermined); *dynamic evolutionary forcing by transposons*, in which transposon and transposition operators gradually enlarge the population-average length of transposons in the evolved population. In this case the evolutionary pressure by transposons rises with evolutionary time.

We test these four mechanisms for their ability to co-opt genes into existing developmental GRNs to alter function. We use the specific case of maternal gradient reading in the *Drosophila* segmentation network, for which we can calibrate network dynamics against quantitative data. Within this system, fit GRNs are those which create precisely positioned gap gene domains despite variability in the maternal gradient. Robustness to maternal factor variability is a key feature of developing systems and has spurred a great deal of interest in the biology community with respect to how embryos might achieve this [90–95]. Through EC, we are aiming to characterize what some of the key factors might have been in evolving GRNs with such robustness.

In simulations we can alter and evolve thousands of GRNs, a subset of which may be robust to maternal variability. Analysis of these solutions allows us to compare the efficiency of different co-option mechanisms (e.g., random mutation versus transposons), and whether particular mechanisms may favor particular types of recruits, in terms of, say, their expression patterns or network connectivity and how they affect network behavior. Discussion in this area has predicted that growth of GRNs via co-option should cause both structural (genes duplicating existing ones) and functional (development of compensatory pathways) redundancy [96]; this has been observed in a number of organisms [96]. Such redundancy is likely to affect characteristics such as evolvability (ability of the network to change) or robustness to perturbations and variability during development. Our simulations offer a direct way of characterizing such interrelations.

2. Methods and Approaches

The mutual inhibition of the 4 trunk gap genes (*hb*, *Kr*, *kni*, *gt*) plays a major role in establishing their expression

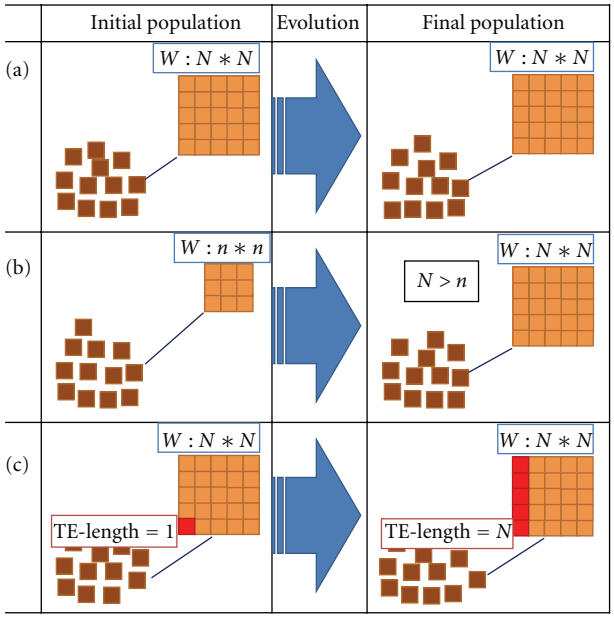


Figure 1: Three ways to introduce gene recruits for cooption in W matrix-based models (gene-gene interaction, see Figure 2 below) of GRN evolution *in silico*. (a) Static determination of recruits. Initial population has fixed numbers of obligatory (2) and additional (recruited) genes. Only the obligatory genes are used to fit the data. The number of the recruits does not change in a given GRN or its descendants during evolution. The W matrix dimension here is $N \times N$. (b) Dynamic addition of recruits. A Gene Introduction operator adds new genes to the GRNs during the evolutionary search. Gene Introduction can be complemented by a Gene Withdrawal operator. (c) Dynamic evolutionary forcing by transposons. A set of transposon and transposition operators forces evolutionary search via a restriction of evolutionary space (e.g., by zeroing the elements of the W^{a0} column of the W matrix, Figure 2). With evolutionary time, transposons (transposable elements, TE) form one-dimensional clusters of length N (TE-length = N).

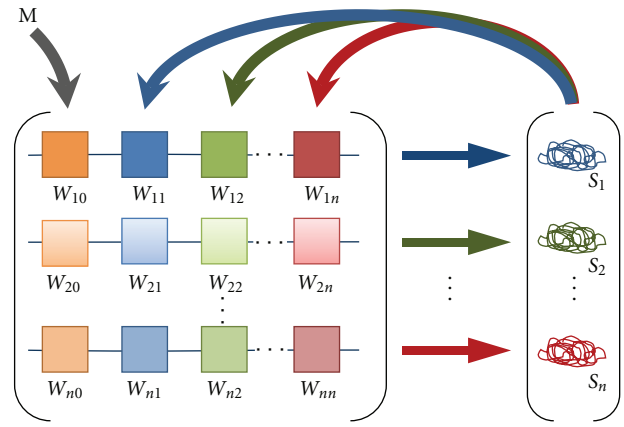


Figure 2: The gene-gene interaction matrix, a core element of the coarse-grained modeling of gene regulatory networks. Each gene (horizontal arrow) is regulated by the products of other genes via upstream enhancer elements (boxes). The strength and direction of regulation (depicted as differently colored saturation levels) are a function of both the regulatory element and the abundance of its corresponding gene product. The left-most column W^{i0} corresponds to the regulatory elements for the action of the morphogen, M = Bicoid (external factor for the GRN). The genotype is represented as the matrix, W, of the regulatory interactions, and the phenotype is the vector, $\hat{S}$, of the gene product levels at equilibrium. Modified after [7].

domains. Models with all 4 genes capture most of the features of gap expression domains. However, the dynamics have also been broken down and studied in terms of mutually inhibitory pairs, such as Kr-gt and kni-hb (e.g., [39–41, 97]). While some subnetworks of the 4 genes can recapitulate major features of the trunk pattern—for instance, addition of Kr to a Bcd-hb subnetwork confers robustness of hb pattern to Bcd variability [71]—other combinations will not. For instance, a Bcd-Kr-kni subnetwork is not sufficient to form gap patterns. We will use this feature to study the role of co-option in the gap network. By making only Kr and kni obligatory in the starting networks, we can create a tendency for the network to coopt additional genes in order to meet the criteria of forming normal gap patterns (as evaluated by fitting model output to experimental Kr and kni patterns). We can think of the obligatory Kr and kni genes as an "ancestral" network, which evolution needs to enlarge in order to solve the problem of simultaneous gap patterning.

2.1. Regulation Matrix-Based Modeling of the GRN. The network is represented at the coarse-grained "gene circuit" level [98]; the dynamics of each gene product (protein) a in each nucleus i (1 nucleus ~1%EL in distance) is given by a system of *number of proteins* times *number of nuclei* ODEs (Ordinary Differential Equations) of the form

$$\frac{\partial v_i^a}{\partial t} = R_a g(u^a) + D_a \Delta v_i^a - \lambda_a v_i^a. \quad (1)$$

The main terms on the right hand side of (1) represent protein synthesis (R_a), diffusion (D_a, Δ) and decay (λ_a). $g(u^a)$ is a sigmoid regulation-expression function. For values u^a below $-1.5 g(u^a)$ rapidly approaches zero and above 1.5 approaches unity. u^a is given by $u^a = \sum_b W^{ab} v_i^b + h^a$. The genetic interconnectivity matrix, W^{ab}, is the key component describing the gene-gene connections and their strengths (Figure 2). The W^{ab} elements represent the activation of gene a by the product of gene b (with concentration v_i^b) if positive, repression if negative, and no interaction if close to zero. h^a represents regulatory input from ubiquitous factors.

2.2. Experimental Data for Fitting. We fit our model results to data from a large-scale project we were engaged in to collect, process, and analyze the expression of the *Drosophila* segmentation genes [91, 94, 99]. This FlyEx dataset is now available publicly [14]. In this paper, we use expression data from mid nuclear cleavage cycle 14 (prior to full cellularization), the developmental stage during which segmentation patterns become mature. Figure 3 shows an example of this data for the 6 gene products in our model (maternal proteins Bcd and Cad and the 4 gaps). Models (in this publication) are evaluated by the quality of their fit to the Kr, kni data (Figure 3(a)).

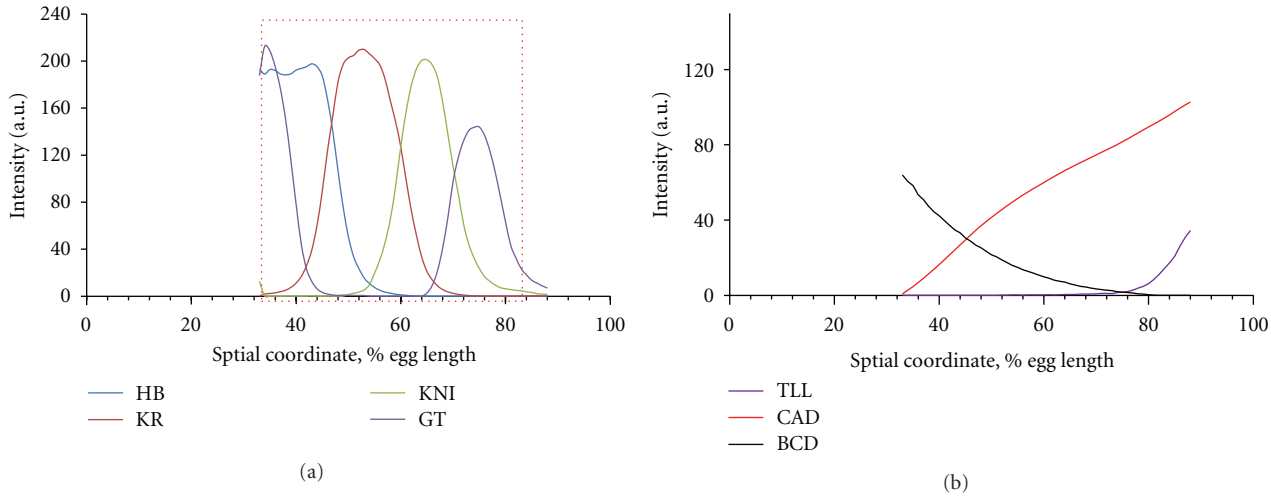


Figure 3: Biological data used to fit ODE model by GA. Integrated gene expression profiles for mid nuclear cleavage cycle 14 A. Vertical axis, relative protein concentration (proportional to intensity); horizontal axis, relative position along the anteroposterior (AP) embryo axis (0% is the anterior pole). Data from the FlyEx database [14]. (a) Protein profiles for four trunk gap genes (giant; gt, hunchback, hb; Kruppel, Kr; & knirps, kni). The red rectangle marks the part of the AP axis modeled in this publication. Models are fit to the Kr and kni data. (b) Profiles of two proteins which are external inputs in our simulations (the primary morphogen Bicoid, Bcd, and the transcription factor Caudal, Cad).

Within the framework described in Section 2.1 (1), we model gap gene cross-regulation and their control by up to two nongap transcription factors: the primary morphogen Bicoid (Bcd) (Sections 3.1–3.5: results); and the transcription factor Caudal (Cad) (Section 3.6: results).

2.3. Evolutionary Computations to Simulate Evolution of GRNs. The set of ODEs (1) representing the gap GRN was solved numerically by Euler's method [100]. A cost function was calculated from the difference of the output model gene product concentrations and the corresponding experimental concentrations:

$$E = \sum_{b} \left(v_{i\,\text{model}}^{a} - v_{i\,\text{data}}^{a} \right)^{2}. \quad (2)$$

Evolutionary computations (EC) were run on the elements of the interaction matrix W^{ab} to minimize the cost function E. The other model parameters, m^a, h^a, R_a, D^a, and λ_a, were found in preliminary runs and then used as fixed parameters. EC followed the general scheme of population dynamics (common to both GA [101] and general simulations of biological evolution), with repeated cycles of mutation, selection, and reproduction. Following the standard GA approach, the program generated a population of floating-point chromosomes, one chromosome for each gene a. The value of a given floating-point array a (chromosome a) at index b corresponds to a W^{ab} value.

Initial chromosome values were generated at random. The program then calculated the v_i by (1) and scored each chromosome set (Wmatrix) by the cost function E (2). An average score was then calculated for all the chromosome sets run. Chromosome sets with worse-than-average scores were replaced by randomly chosen chromosome sets with better-than-average scores. A portion (40%) of the chromosomes were then selected to reproduce, undergoing the standard operations of mutation and crossover (defined below), changing one or more of the W^{ab} values. The complete cycle of ODE solution, scoring, replacement of below-average chromosome sets, and mutation and crossover was repeated until the E score converged below a set threshold, typically 50–100 generations. (In case convergence did not occur, all computations were stopped by EvalSum = 1,000,000 evaluations.)

In GA, mutation is a genetic operator used to maintain genetic diversity from one generation of a population of chromosomes to the next, analogous to biological mutation. Point mutation in GA involves a probability that a W^{ab} value on a chromosome will be changed from its original state (comparable to changing a nucleotide in biological point mutation). Upon mutation, a W element is updated according to $[W^{ab}] = [W^{ab}] \pm \ln(\text{Random(Power)})$, where Power = $1,000,000$.

GA crossover is a genetic operator used to vary chromosomes from one generation to the next, by swapping strings of values between chromosomes, analogous to crossover in biological reproduction. We use one-point crossover in this study, in which a point on a parent chromosome is selected, then all data beyond that point is swapped between two parent chromosomes.

The model is implemented in Delphi (Windows) and Free Pascal (Linux) and available from the authors upon request.

2.3.1. Introduction and Withdrawal of New Genes. As a first way of modeling dynamic recruitment of genes to the gap network, we introduce new GA operators for Gene Introduction and Gene Withdrawal. Gene Introduction adds a new gene to the network at a rate of 5–10% per generation

(depending on the simulation). Specifically, this adds a new row and column to the W^{ab} matrix (Figure 4), which can be then be operated on by mutation and crossover. To balance this process and control the number of genes in the network, Gene Withdrawal removes a row and column from the W^{ab} matrix (at a rate of 2–10% per generation, depending on the simulation). Gene Withdrawal does not operate if the network is minimal ($N = 2$ genes). Since Gene Introduction simply adds a gene to the network, which then adapts, it does not distinguish between the biological cases of a new gene arising by duplication or of an existing gene being recruited from another network [1, 3]. Two parameters control the Introduction and Withdrawal procedure. The ToRecruiting-Proc parameter defines what part of population (from 0 to 1) will be subjected to the procedure. Another parameter, WithdrAdd (from 0 to 1) specifies the probability that a given solution (W matrix) will be subjected to Gene Withdrawal or Gene Introduction. (If WithdrAdd = 0 then all solutions will go through Gene Introduction only; If WithdrAdd = 1 then all solutions will go through Gene Withdrawal only.)

2.3.2. Involvement of the Recruited Gene in the Functioning of the GRN. To quantify how much added genes affect network fitness, we used the following procedure: the model solution in a given generation was evaluated according to (2); then, for each additional gene (above the obligatory 2), fit to the data was evaluated with the W elements for that gene zeroed out. In cases where this produced a drop in fitness score (2) of more than 10% (a threshold determined in preliminary runs) compared to the full GRN, we kept the added genes as recruits to the GRN. We further filtered the most functionally significant recruits by use of a 33% threshold. Results are presented below for both threshold levels.

2.3.3. Evaluation of GRN Robustness. GRN solutions with E scores below threshold represent good fits to the gene expression data (Figure 3(a)). That is, these GRNs solve the problem of forming gap expression domains. In addition to this, though, we want to test the robustness of these gap solutions to maternal variability. To do this, we took each good solution and tested its robustness to Bcd variability. We perturbed Bcd from Figure 3(b) according to

$$[bcd] = [bcd] \pm [bcd] * \text{Random}\,(0.2) \tag{3}$$

(i.e., the Bcd profile varied within limits of ±20%). We reran the GRN with the perturbed Bcd values and compared these against the unperturbed result according to

$$E' = \sum_{b} \left(v^{a}_{i\ \text{perturbed}} - v^{a}_{i\ \text{unperturb}} \right)^{2}. \tag{4}$$

This measure was calculated for 100 Bcd perturbations for each GRN, and the results averaged for a measure of the GRN's robustness (e.g., Figure 12).

2.3.4. Artificial Transposons for GA. The above Gene Introduction operator does not model the mechanism by which genes are incorporated into the genome. To begin addressing this, we have developed a model for transposon dynamics in our simulations. We define an artificial transposon as a marked block of the host's code. The mark is transmittable from host to host. For this, we use double-string chromosomes, in which the main string (floating point) is used for the host codes, while an additional string (binary) is used for the transposon marks (1 denotes a transposon mark; 0 is unmarked):

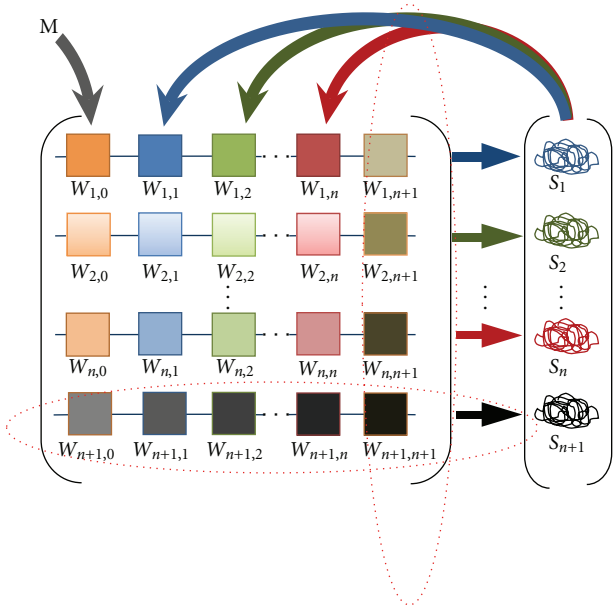


FIGURE 4: Gene Introduction adds a new $(n + 1)$ column (right-most) and row (lowest) to the W interaction matrix. Gene Withdrawal eliminates the right-most column and bottom row.

$$\begin{aligned} &\text{the additional string: } 1 \quad 0 \quad 0 \quad \cdots \quad 0 \\ &\text{the main string: } a_1 \quad a_2 \quad a_3 \quad \cdots \quad a_n, \end{aligned} \tag{5}$$

where a_i are host code floating-point values (only the a_1 element is transposon marked in this example).

2.3.5. Artificial Transposons as Mutators. As with biological transposition, an artificial transposon tends to be deleterious to the host. To see how this affects the W interaction matrix, consider the following: let the first-row, first-column element of $W_{A \leftarrow M}$ be infected by a transposon (Figure 5(a), highlighted). The transposon's deleterious action is then implemented by decreasing the value of the infected host element. Specifically, we halve the $W_{A \leftarrow M}$ value in each generation. This quickly drops the element value to near zero. In this manner, the transposon effectively cuts the $A \leftarrow M$ regulatory connection.

2.3.6. Spread of Artificial Transposons. Transposons tend to form clusters in host chromosomes. We simulated this feature by spreading transposon infection by at most one element per generation. In this operation a transposon can mark the $W^{j \leftarrow i}$ element above it as a new transposon. (Figures 5(b) and 5(c)).

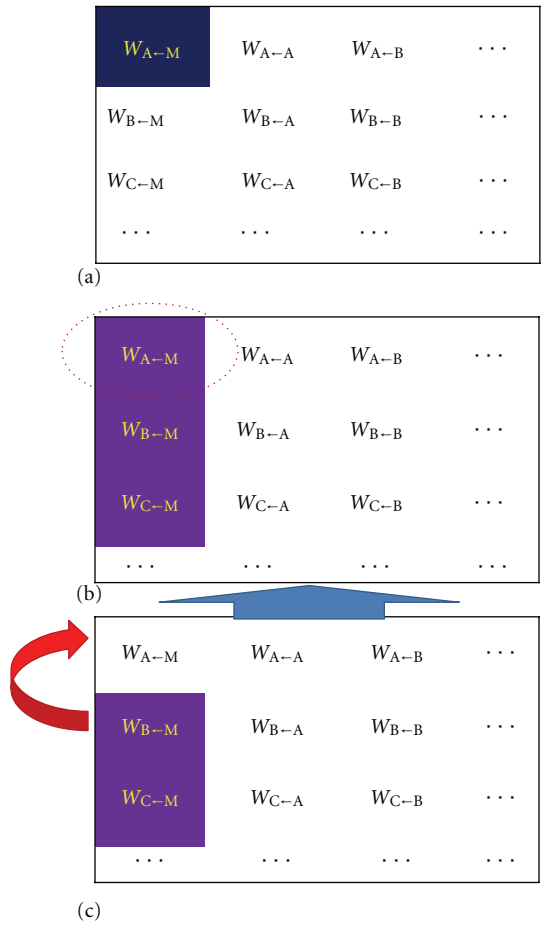


Figure 5: Artificial transposons (a) and their mechanism of spreading (b-c).

2.3.7. Transmission of Transposons. We used fixed transposon coordinates to transmit transposons from host to host. (Whole transposons were never moved along the chromosomes.) The two-place transmission operator was implemented as follows: first, a pair of hosts was chosen at random; then a chromosome from either host was scanned for transposon marks. If a transposon was found, it was replicated in the partner chromosome, regardless of the original string character in the target chromosome. Copying only occurred if the secondary strings had transposon marks.

3. Results

In general, we found recruitment of new genes to a preexisting GRN to be typical; we saw this trend in all of the evolutionary computational designs implemented (Figure 1). Recruited genes can be uniform or spatially patterned. These patterns can either recapitulate patterns of existing network genes or introduce patterns novel to the model network (but like patterns seen in the full biological gap network).

3.1. Parameter Optimization for the Evolutionary Computations. In order to compare results between co-option mechanisms, we established standardized settings for a number of the computational parameters. We have found that the most important parameters for fast evolutionary searches are population volume, Popul; reproduction rate (quota of population to be replaced by offspring in each generation), Reprod and the mutation and crossover rates (Mut and Cross, resp.). In preliminary tests, we found a reasonable value of Reprod to be 0.4 (i.e., in each generation 40% of the population is subjected to reproduction by a truncation strategy). The most effective population volume depends on the number of genes recruited: for 4 genes, best results were at Popul = 8000; for the 8 genes, best results were for Popul = 12,000 – 16,000 (Figure 6(a)). For mutation, we tested Mut values of 40%, 32%, 24%, and 16% (with Popul = 8,000, Reprod = 40%; EvalSum = 1,000,000; Power = 1,000,000, Cross = 2%; and 4 recruited genes); see Figure 6(b). Best results were achieved for Mut = 40% and 32%. For crossover rate, we tested Cross = 2%, 4%, 10%, and 20% (with Popul = 8,000; Reprod = 0.4; Mut = 20%; EvalSum = 1,000,000; Power = 1,000,000; and 4 recruited genes); see Figure 6(c).

3.2. Static Recruitment: GRN Evolutionary Complexification without External Forces. We ran gene co-option computations with 3 different mechanisms (see Figure 1 and Sections 3.3 and 3.4). All scenarios start from the obligatory Kr-kni system and add genes to improve the fit of the computations to the experimental expression data for Kr and kni. As a control, our simplest scenario is to have all additional (nonobligatory) genes already in the W matrix. Genes are neither introduced nor withdrawn from the matrix, but the W^{ab} elements of the nonobligatory genes adapt over the course of the computations in order to improve the fit to the data. We expected the additional genes to become increasingly incorporated (necessary) in the pattern-forming network, since Kr-kni alone is insufficient for proper patterning. We considered cases from two to eight additional genes (W matrices of dimension 4 to 10). These computations allowed us to estimate some key evolutionary parameters for comparison with the directed (forced) evolution cases below (Sections 3.3 and 3.4) such as the following: (a) how fast can evolution find the desired Kr and kni patterns (evolvability), (b) how many genes on average are recruited to the networks (and what proportion of these recruits are highly involved in the network), and (c) how robust are the networks to Bcd variability.

The results on evolvability are shown in Figure 7 for W matrices of dimension of 4, 6, 8, 10, 12, and 14. (We show a minimum of two additional genes, since preliminary runs indicated that a single additional gene was not sufficient to improve the Kr-kni fit.) First, we see that all solutions with good fitness scores tend to recruit all available additional genes, and that the functional involvement of the recruits in the GRN is very high (even in this simple static scenario), with a negligible difference between the 10% and 33% selection thresholds. Second, evolvability slightly but steadily improves (networks become more evolvable) with the dimension of W (the number of available recruits).

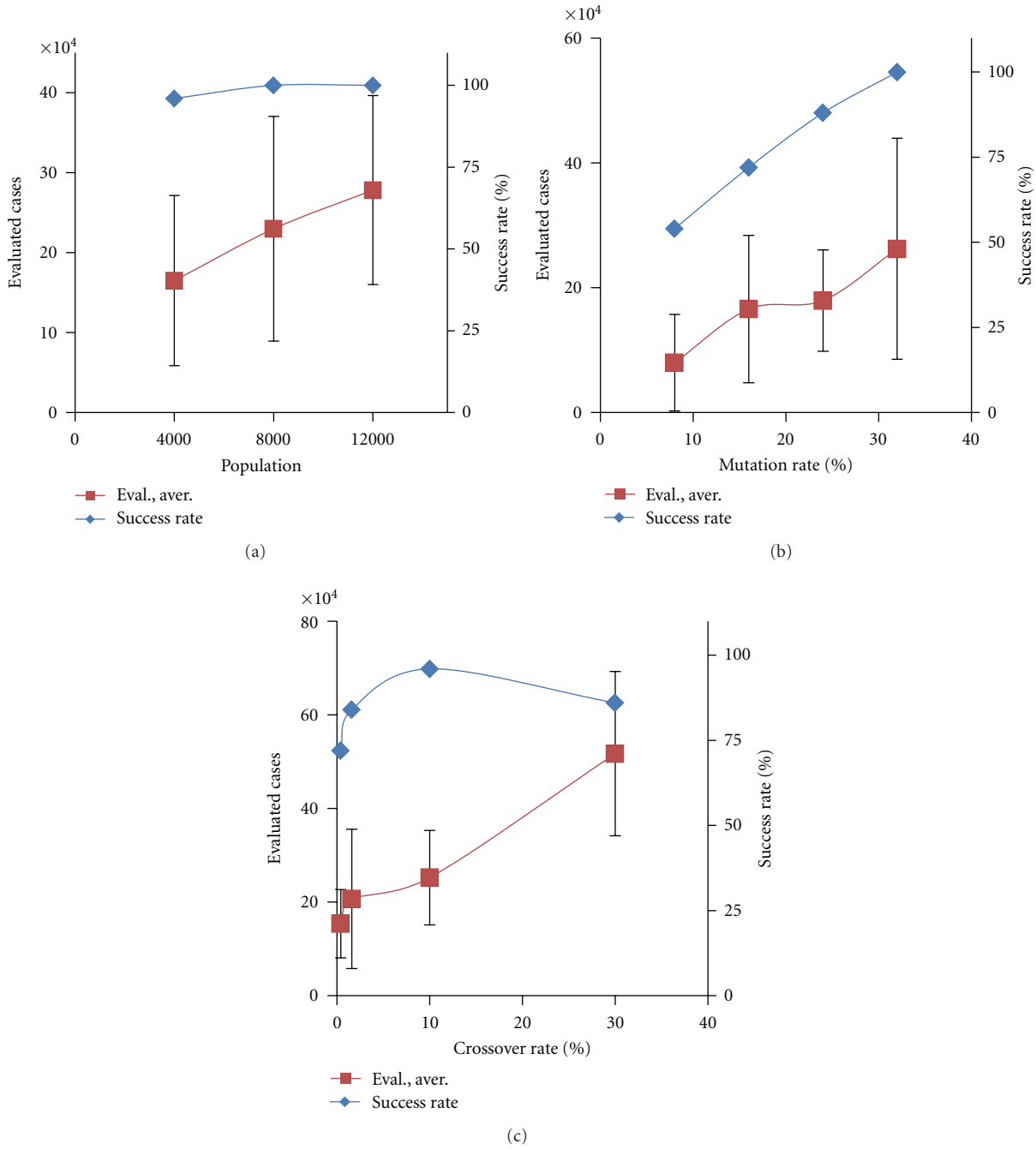


Figure 6: Dependence of evolutionary search on several key computational parameters. (a) Dependence on population volume (Popul = 4,000; 8,000; 12,000). (Other parameters were as follows: Reprod = 40%; Mut = 38%; EvalSum = 1,000,000; Cross = 2%; Power = 1,000,000.) Success rate is the percentage of runs achieving the desired fitness level. (b) The dependence on mutation rate (Mut = 40%, 32%, 24%, 16%). (c) The dependence on crossover rate (Cross = 0.5%, 2%, 10%, 30%). (50 runs for each point).

3.3. Dynamic Addition of Genes: Introduction and Withdrawal Operators. Building on our previous work [71], we have found that addition of new genes during an evolutionary search (enlargement of the W matrix, followed by adaptation of the W^{ab} elements) is an effective way to enlarge networks with the desired functionality. With static recruitment (Section 3.2), we found that GRNs tend to incorporate all available genes in a functional manner. With dynamic control, we similarly find that networks tend to incorporate genes and enlarge. A simple explanation may be that new recruits create an implicit pressure on the network new genes arrive with zero W^{ab} values, and it is far more likely for these W^{ab} values to evolve away from zero, incorporating the new gene into the functioning of the network, than to maintain the W^{ab} at their initial zero values. Functional incorporation of genes into the network (in the dynamic scenario) should

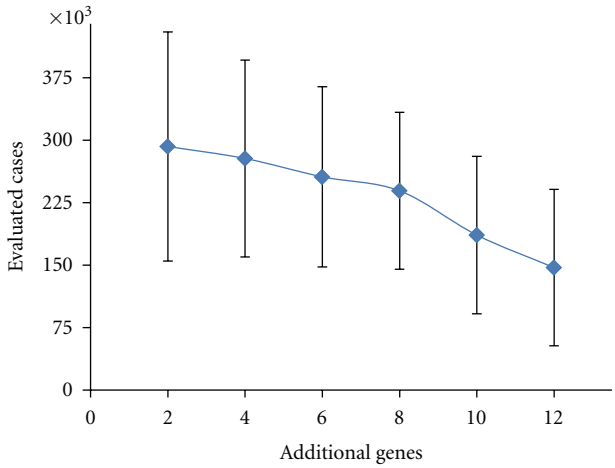


Figure 7: The dependence of evolvability (speed of evolutionary search) on the number of genes recruited to the initial population (the two obligatory genes). Results are shown for 2, 4, 6, 8, 10, or 12 additional genes. Average evolvability (lower values indicate faster evolution) is plotted against the number of recruits in the W matrix.

depend on the rate of the Introduction operator. To test this, we can tune the rate of the Withdrawal operator to control the net Introduction rate. Recruitment should be favored when the Withdrawal/Introduction ratio is low (Introduction probability $\gg$ Withdrawal probability; the implicit pressure is high) and reduced when the ratio is large.

The tendency towards recruitment can still occur for Withdrawal/Introduction ratios more than 50% (more Withdrawal cases than Introduction cases), due to the effect of random mutations (which can add genes). With the mutation rates used in our simulations, recruitment begins to shut off for ratios less than 1/3 (Introduction probability = 25%; Withdrawal probability = 75%). By tuning these parameters, we can characterize their effect on network outgrowth (addition of genes) and evolvability (Figure 8).

We see that higher relative Withdrawal rates slow the rate of network outgrowth (Figure 8(a)), and also decrease the success rate (Figure 8(b)). The decrease in network size due to the Withdrawal rate does not appear to be affected by the co-option threshold (Figure 8(c)).

3.4. Forced Evolution by Artificial Transposons. Here we begin to model a biological mechanism for the introduction of genes to the network, by the action of transposons. As introduced in Section 2.3.5, our mechanism for artificial transposition can effectively shut down the input into the GRN from a particular regulator. As illustrated in Figure 5, for example, transposon infection can cut out influence of the maternal regulator (Bcd). Transposon infection, computationally, is a means of restricting the search space of the evolutionary problem.

3.4.1. Static Transposons. As a control for studying dynamic transposon infection, we have run a series of computations with statically "knocked out" regulators, that is, W^{ab} in which an entire column is zeroed or held constant. We have run a series of computations in which the first column (for maternal regulation) is held constant while the rest of the matrix is free to evolve. Table 1 summarizes these, giving the constant W^{a0} values run for each matrix dimension (number of genes) run.

Table 1: Elements of W^{a0} kept constant in the static transposon tests.

Matrix dimension	W^{10}	W^{20}	W^{30}	W^{40}	W^{50}	W^{60}	W^{70}	W^{80}	W^{90}	W^{90}
4	0	33	67	100						
6	0	20	40	60	80	100				
8	0	14	29	43	57	71	86	100		
10	0	11	22	33	44	56	67	78	89	100

The static transposon tests are generally slower to evolve than the static matrix case (Section 3.2), but faster than the dynamic matrix scenario (Section 3.3). Figure 9 shows that static transposon tests do show the same trend of increasing evolvability with increasing network size.

3.4.2. Dynamic Transposons. Here, we consider dynamic introduction of transposons, as illustrated in Figure 5. For simplicity, we initiated these computations with all members of the population invaded by transposons. Transposon dynamics are controlled by 2 parameters: TE growth, which controls the rate at which transposons can spread in a column of the W matrix (to a maximum length set by transposon length = 2 + (1 or 2); TE action, which controls how fast W^{ab} elements decay given transposon infection.

In general, we find that the efficacy of the evolutionary search in this scenario is comparable to the cases of fixed W matrix (Figure 7) or static transposons (Figure 9) (all simulations compared at GRN dimension of 10). Higher and lower TE action values may produce higher evolvability; midrange TE action may slow evolution (Figure 10(b)). As TE action is increased, the transposon length more quickly achieves the maximum length (Figures 10(a) and 10(c)). Similar results were seen for TE growth. As in Sections 3.2 and 3.3, the best scoring GRNs tend to coopt all available genes (make the largest possible GRN).

3.5. Spatial Patterns of Coopted Genes. We found gene recruitment and functional incorporation into the GRNs to be quite general, regardless of the Gene Introduction mechanism. What sort of functionality do these recruited genes take in the network? We found recruited genes produced sophisticated spatial patterns with subdomains, influencing the spatial patterning of the obligatory two genes of the network (Kr & kni). Figure 11 shows representative examples of such networks.

These patterns are reminiscent of the mature patterns of *Drosophila* gap genes and demonstrate how recruitment could supply new gap genes for an evolving segmentation network (as in the transition from short to long germ band mechanisms discussed in the Introduction).

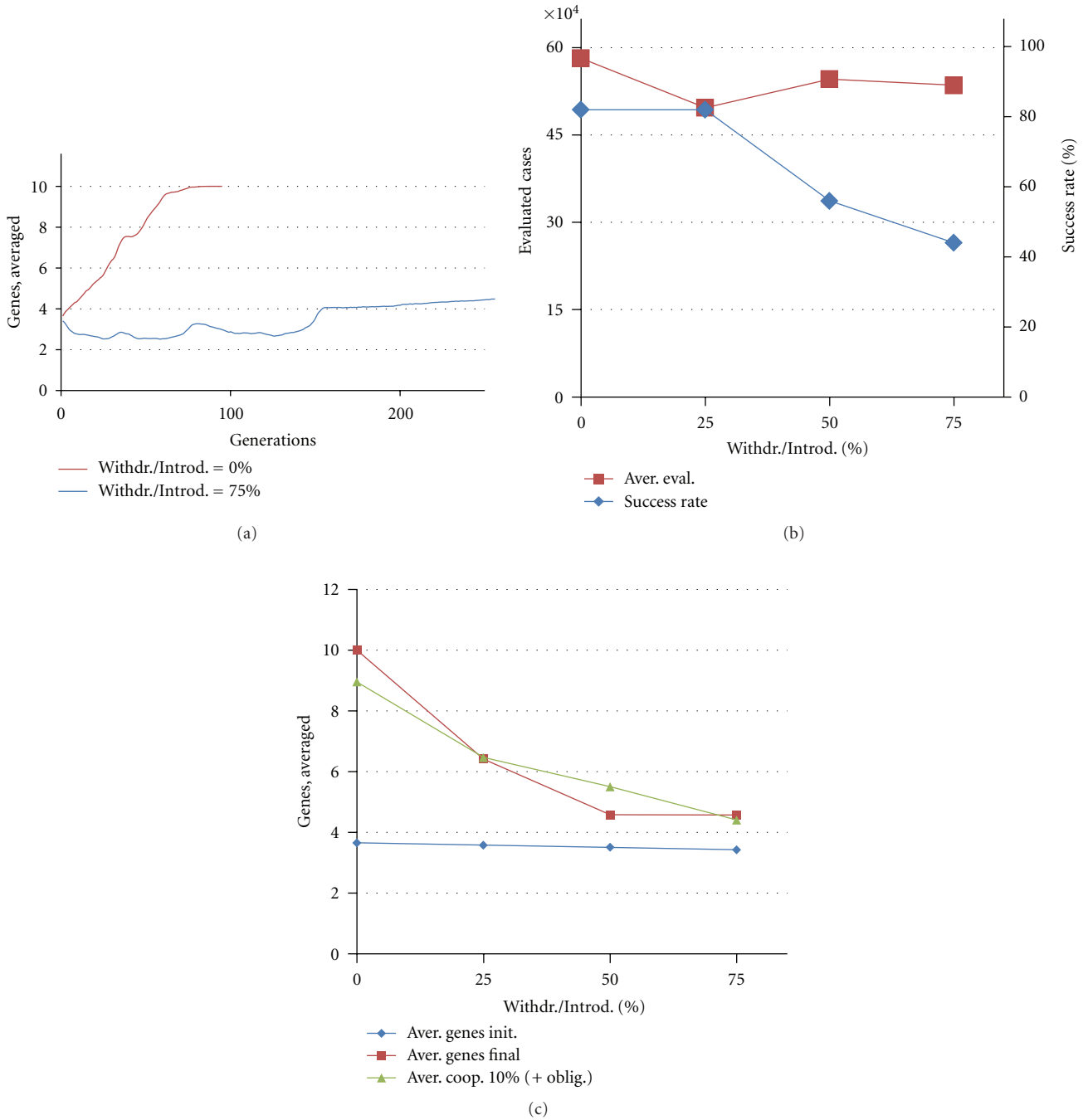


Figure 8: Effects of the Gene Withdrawal-to-Introduction ratio. Results are averaged over 50 runs for each Withdrawal/Introduction value. (a) Recruitment—growth of the number of genes recruited (population average) during computational evolution without Withdrawal (0.00) and with high relative Withdrawal rate (0.75). (b) Evolvability—Evaluated cases and success rate against the Withdrawal/Introduction rate. (c) Growth of networks during evolutionary search at different Withdrawal/Introduction rates: population average of genes in initial GRNs, population average of genes in final GRNs, and average functionally coopted recruits (10% threshold). Nearly all recruits are functional. Other parameters are as follows: Popul = 8000; Reprod = 40%; potential recruits = 8; Mut = 32%; Cross = 8%; EvalSum = 1000000.

Since the initial (Kr-kni) model networks lack the hb and gt regulators found in the real *Drosophila* 4-gene network, we were very curious whether the evolutionary computations might recapitulate hb- and gt-like patterns and functions. Indeed, the patterns of the coopted genes are usually reminiscent of anterior and posterior hb or gt domains (sometimes in reverse orientation). It could be that the evolutionary search is tending to fill in the missing gap patterns to generate the structure of the real, complete gap network (and better fit the real expression data).

Our simulations show that outgrowth of 2-gene subnetworks via recruitment leads to co-option of genes which

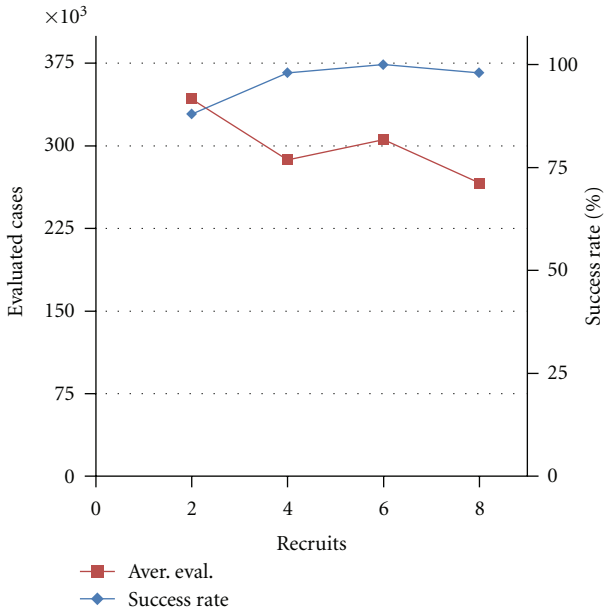


FIGURE 9: Static transposon tests. (a) Evolvability improves (and evolution speeds up) with number of genes in network. Other parameters are as follows: Popul = 8000; Reprod = 40%; Recrs = 2,4,6,8 (maximum number of recruits); Mut = 32%; Cross = 8%; EvalSum = 1000000.

do recapitulate the patterns of real gap genes (i.e., gap genes which are part of the real segmentation network but are originally missing in the simulations). Our simulations, therefore, are an indication of how the gap gene network may have evolved to solve the particular problem of simultaneously forming properly positioned expression domains. In particular, our simulations may indicate how insect segmentation GRNs may have evolved from the primitive short germ mode to the derived long germ mode.

3.6. Gene Networks under the Control of Two Gradients. We found that the GRN solutions described in Section 3.5 (in which the only external, nongap gradient was Bcd) were not robust to Bcd variability. This is in contrast with our observations from a similar computational evolution project [71], in which Bcd-robust solutions were found in a few percent of all good solutions (those that fit the expression data). The main differences are that the earlier project considered (i) *hb* and *Kr* as the pair of obligatory genes, and (ii) two maternal morphogenetic gradients as external inputs (Bcd and Cad). As noted above, some of the Bcd-only networks did recruit posterior-anterior gradients, perhaps to compensate a missing essential feature of the biological network. To determine the effect of the posterior gradient, we added Cad to the model (see Figure 3(b)). Computationally, this two-gradient version (Bcd-Cad) of the model behaved very similarly to the Bcd-only model. We will therefore focus here on the characteristics of the resultant GRNs.

3.6.1. Robustness of the Gene Networks with Two Gradients. Addition of Cad to the network resulted in a number of the solutions displaying robustness to Bcd variability (Figure 12). Some of these showed higher robustness than is observed for real *Drosophila* segmentation genes (Figure 12(a); c.f. [33]). However, many good or even very good solutions (according to the fitness score for matching experimental expression patterns) can show no robustness to Bcd variability. These non-robust solutions can give Kr and kni variability as high as that for Bcd (Figure 12(b)); that is, Bcd variability is directly transmitted to its downstream targets, in contradiction to the observed severalfold reduction in variability seen in the data [94]. We see little correlation between goodness-of-fit to the expression data and robustness to Bcd variability; best-fit solutions can span from highly robust (Figure 12(a)), capable of filtering out Bcd variability nearly completely, to solutions unable to filter variability at all (Figure 12(b)).

It has been experimentally established that the position of each domain border of each gap gene pattern is under the control of different combinations of regulatory inputs from the other members of the segmentation network. The 2 obligatory genes (Kr and kni) in the model have two borders (anterior and posterior) each. Even for good-scoring solutions, there are cases where the *kni* border positions are robust but the Kr borders are less robust (even non-robust, Figure 12(d)). In many cases, the anterior Kr border is less robust than the posterior one (Figure 12(c)).

Our results indicate that robustness of the Kr-kni pattern depends on external gradients from both ends of the embryo, as provided by Bcd and Cad. We find that robustness can evolve relatively independently at each border. Hence, the positional error for each border can be relatively independent. This implies that whatever the mechanism of robustness for boundary precision, this may need to be evolved and established for each boundary, especially for systems such as *Drosophila* segmentation in which the combination of regulators controlling each boundary is unique.

4. Discussion

The main conclusion of this work is that GRN evolution tends to coopt all available genes. Network enlargement and functional redundancy of gene-gene connections do not prevent the cooption of new functional genes. With our Gene Introduction and Gene Withdrawal operators, we could directly investigate the effect of these rates on network outgrowth. If random mutation is also operating, Withdrawal rates can be significantly higher than Introduction rates before network outgrowth is halted. These findings are in agreement with the natural tendency towards gene recruitment found biologically (see Section 1).

Our modeling may offer insights into the evolution of insect segmentation. Our obligatory 2-gene network may have parallels to the short-germ mode of segmentation. The 2-gene model is not initially sufficient to fit the long-germ *Drosophila* data, but recruitment of additional genes can produce good fits to the long-germ mode. Introduction of a new gene often does not appear to directly increase the fitness. However, Withdrawal of the gene, after evolution to a good-scoring solution, can greatly reduce fitness, showing that it

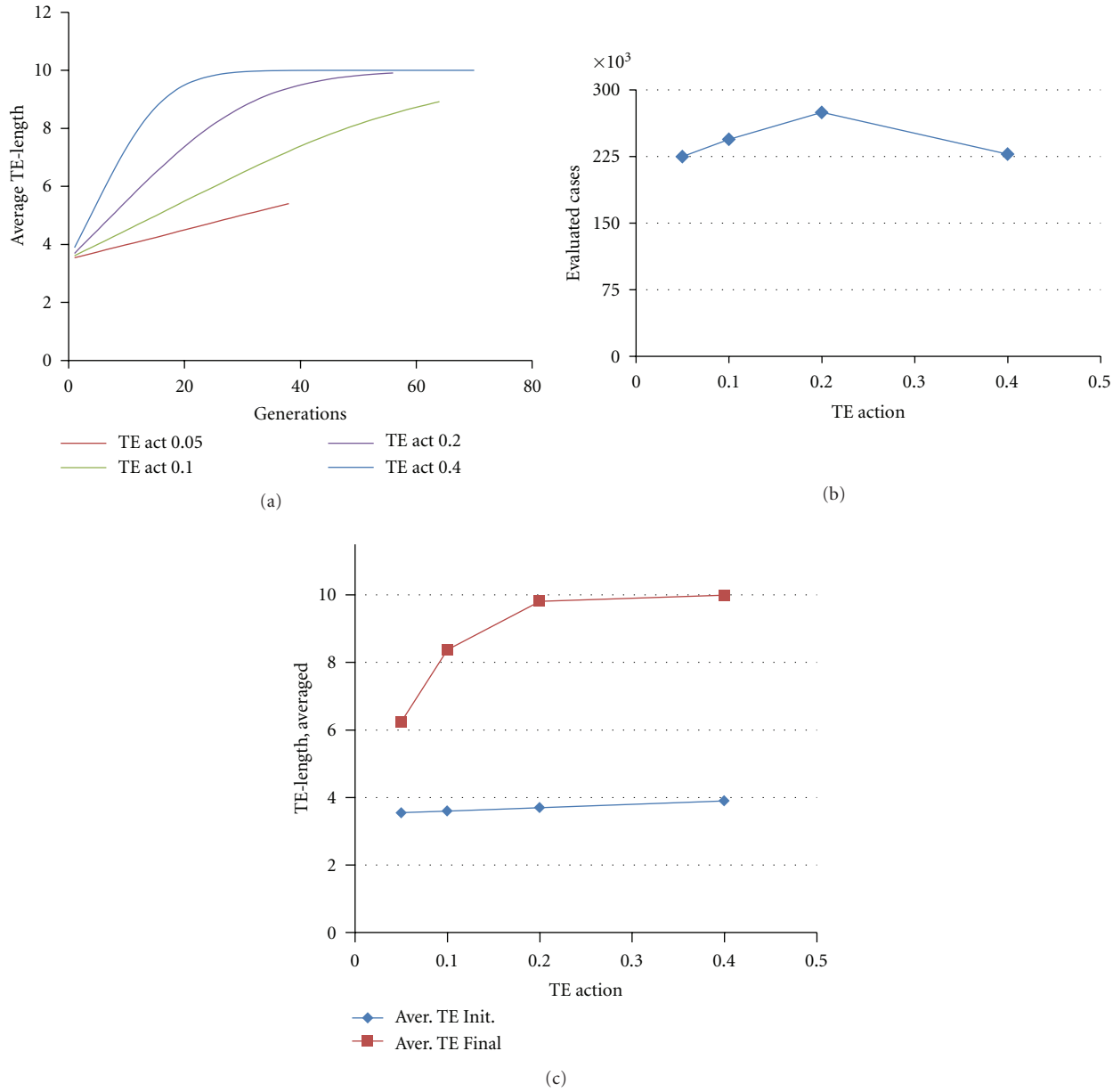


Figure 10: Dynamic transposons. (a) Examples of the growth of transposons (population average) during computational evolution, for different values of the transposon/transposition operators TE growth and TE action. (b) The dependence of evolvability on transposon activity. (c) Sustainable growth of transposon length during evolutionary search at different activity levels of the transposons. Other parameters are as follows: Popul = 8000; Reprod = 40%; Recrs = 8 (maximum number of recruits); Mut = 32%; Cross = 8%; EvalSum = 1000000; TE growth = TE action.

has acquired functionality in the network. Added genes do not generally provide structural redundancy, in which they "back up" a particular existing gene; rather, recruitment of a gene tends to alter the interactions in the original network.

4.1. Redundancy and Robustness of Gene Networks. A notable feature of the early segmentation GRN is that it is under the control of not one but several maternally supplied gradients of transcription factors. For the core of the early GRN—the trunk gap genes—one should consider not only the primary morphogenetic gradient of Bcd, but also the maternal Hb and Cad gradients, and the terminal gradients (see review in [41]). We believe our evolutionary computations can shed some light on the functionality of this apparent redundancy of the biological gradients. In particular, we found that addition of the posterior Cad gradient was necessary in the present Kr-kni (obligatory) model to produce robustness to Bcd variability, in contrast to our earlier findings with a hb-Kr model [71]. This indicates that while particular 2-gene subnetworks may have evolved with robustness to Bcd variability (in agreement with recent theoretical work, [102]), other 2-gene pairs may require additional gradient input to form patterns that are robust to variability. Our present

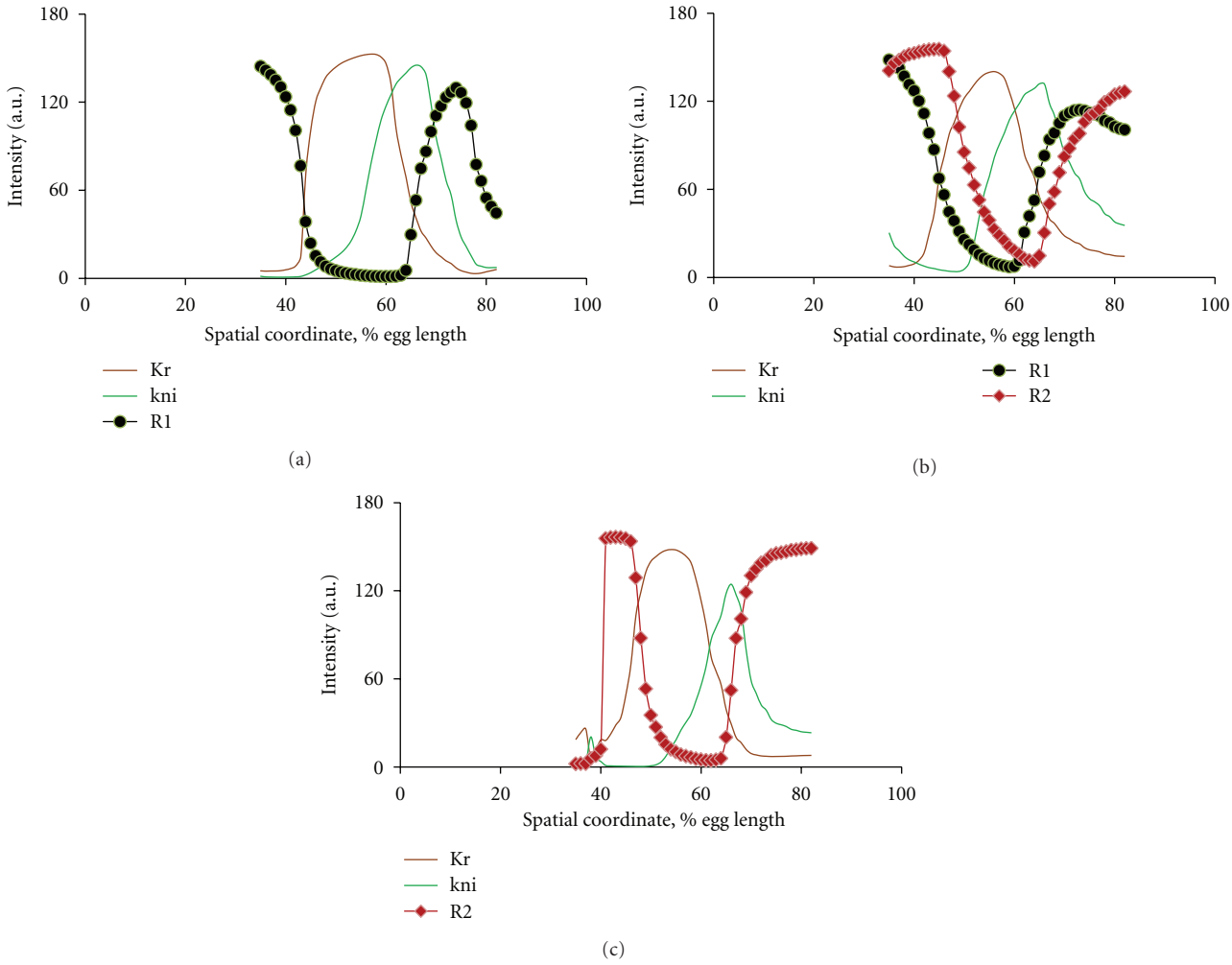


Figure 11: Representative examples of 2-gene models with recruits. (a) Recruit pattern is similar to gt or hb (black; c.f. Figure 3(a)). (b) Recruit patterns are similar to hb and gt (black and red; Cf. Figure 3(a)). (c) Recruit pattern looks like gt in reverse orientation (red; Cf. Figure 3(a)).

results begin to characterize these variations in robustness between gene pairs, and the role multiple gradients may play in creating robustness in the complete *Drosophila* long-germ segmentation network.

With the Bcd-Cad model, robustness to Bcd variability can take a variety of forms, from all Kr and kni borders being very precise to cases in which particular borders show much different robustness than others. We feel this reflects the biological nature of the problem, in which different borders are under different regulatory factors. By comparing the present results to our prior work on the hb-Kr module, and extending our approach to investigate other gap pairs and robustness to variability in other gradients, such as Cad and maternal Hb, our modeling can offer insight into the ways in which these factors interact to confer local spatial precision, and insight into how these interactions evolved. For example, solutions which fit the data and are robust (in this and our prior work) tend to be found much less frequently than solutions which simply fit the data. Evolutionarily, solutions, for example to long-germ segmentation, may have evolved readily, but search for solutions with robustness to variability may take much longer. This frequency of robust solutions will be explored more fully in future work.

4.2. Forced Evolution by Artificial Transposons. The method of forced GRN evolution by artificial transposons is described in further detail in [83]. Together with the present work, we are gaining insights into some of the diverse features of the coevolution of GRNs and their transposons.

For example, preliminary computations making the 4 core gap genes (*gt*, *hb*, *Kr*, and *kni*) obligatory and limiting the number of potentially recruited genes to 1 (R1) show parallels between GRN-transposon coevolution and host-parasite (or predator-prey) dynamics. Figure 13 shows a time course of these dynamics. Given an initial population of GRNs ($GRN_{ini} = GRN_4$), a primary invasion of the initial transposons ($TE_{ini} = TE_4$, i.e., transposons of length = 4) spreads through the initial population. TE_4 infection gradually reduces the GRN_4 fitness score. Transposons in this case

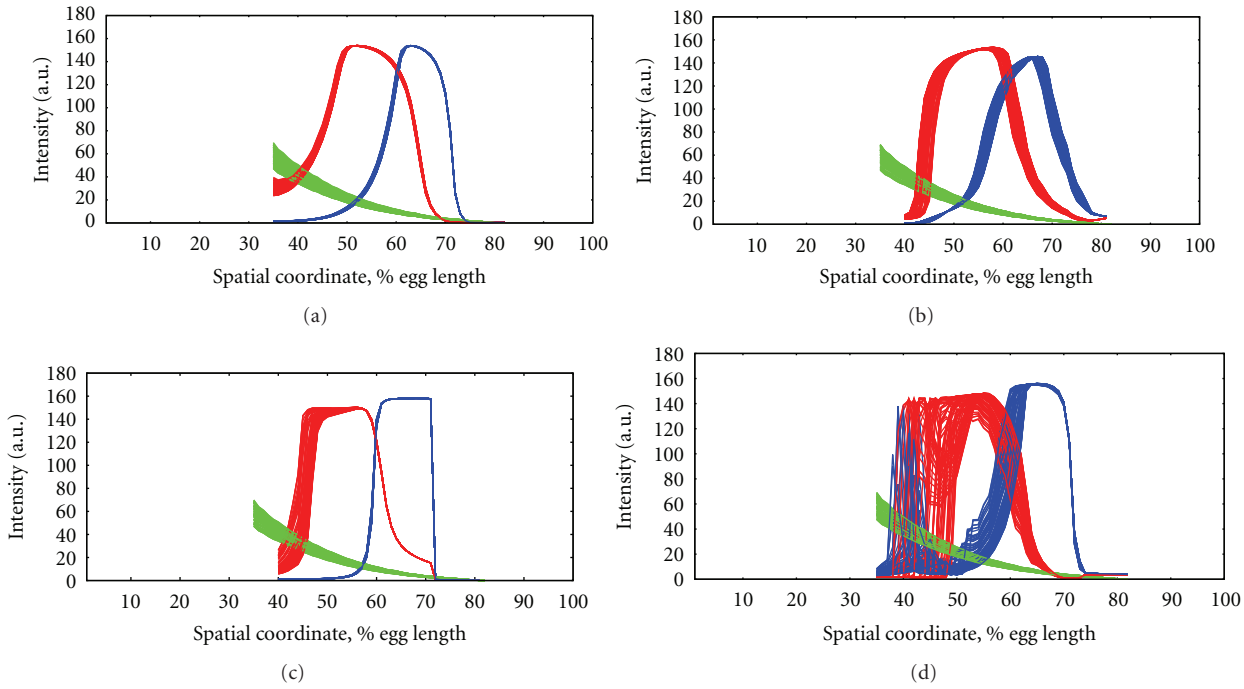


FIGURE 12: 2-gradient model (Bcd-Cad) shows robustness to Bcd variability. Bcd—green; Kr—red; *kni*—blue. (a) Highly robust Kr and kni. (b) Nonrobust Kr and kni. (c) All borders are robust, except for anterior Kr. (d) Severe nonrobustness, especially for Kr, probably caused by bifurcations between GRN basins of attraction (multistability; c.f. [74]).

are defined as growing and transmitting. As a result of the selection pressure on GRN_4, recruitment allows R1 networks (which cannot be infected by TE_4) to become prevalent in the population. However, due to transposon growth, transposons of length 5 (TE_5) soon appear and begin to infect R1-GRNs. The infection gradually decreases the R1-GRN scores while increasing the prevalence of TE_5 (Figure 13, early times). The decreasing proportion of TE_4 in the population makes GRN_4 relatively fit again, and the TE_5-infected R1-GRNs begin to be eliminated by selection and replaced by GRN_4s (which are defined in the model to be steadily supplied from an "external reservoir"). In this way, the population becomes rejuvenated and free of TE_5. The prevalence of GRN_4, however, makes the population susceptible to TE_4 infection again; the cycle repeats, and we observe oscillations in the abundance of the GRN and TE species (Figure 13). Such coevolutionary oscillations are wellknown from host-parasite or predator-prey dynamics. Of interest for future work is the nature of the irregularity in the oscillations, for example understanding why R1-GRN and TE_5 start in-phase and gradually settle into an out-of-phase relation (Figure 13; e.g., do the initial dynamics point to a pool of evolved TE_5 "waiting" for the host R1-GRN to evolve, with subsequent dynamics more tightly codependent?).

We believe that these scenarios of GRN-transposon coevolution could be used as a new tool in forced GRN computational evolution. Specifically, it is a promising mechanism for gentle and indirect forced GRN evolution. The observed oscillations could be useful in overcoming the very general problem of premature convergence in evolutionary searches.

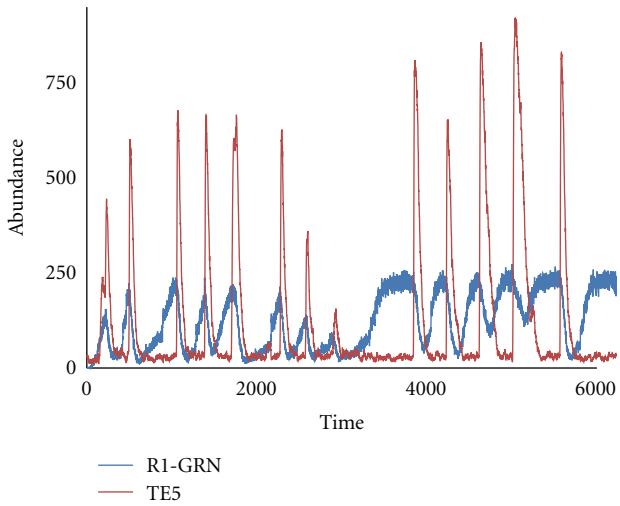


FIGURE 13: Host-transposon coevolution. Oscillatory dynamics of the different "strains" of GRNs and transposons under the restriction of possible gene recruits to 1 (R1). The initial 4-gene network (GRNini = GRN_4) abundance oscillates with the one-recruit network (4+1 genes, or R1-GRN). The oscillations are accompanied by short bursts of TE_5 transposon abundance.

4.3. Discrete versus Continuous Approaches (Boolean versus ODE/PDE Models). A great deal of the work on evolutionary computations of GRNs has been done with discrete-value Boolean approaches, in which genes are either "on" or "off." While these approaches can be fast and lead to general con-

clusions on evolutionary dynamics [60], they can be insufficient for addressing real biochemical networks, where use of continuous differential equations may be more appropriate. For GRNs reverse-engineered to experimental data, evidence suggests that continuous models are more faithful to known interactions than Boolean models. For example, for AP segmentation in *Drosophila*, Perkins et al. [103] compared two discrete logical models with two continuous reaction-diffusion (RD) models and found both RD models fit the data better than the logical models.

Another caution is that the evolutionary landscape of GRNs can be quite different depending on whether a discrete or continuous approach is used. Using a discrete approach, Ciliberti with coauthors [60] suggested that the collection of GRNs which create a particular phenotype (e.g., expression pattern) form a neutral basin in the fitness landscape, such that drift within the basin allows for a neutral means of sampling different phenotypic variations (at the "borders" of the basin). However, this discrete approach does not address the natural continuous variation of gene-gene interaction parameters (due, e.g., to tuning of enzymatic co-factors or complex coregulation by multiple transcription factors). Our evolutionary searches indicate that very small differences in these parameters can produce very different phenotypes (e.g., robust versus non-robust to maternal variability). Our results suggest that the achievement of robust GRNs in a continuous evolutionary search can be quite rare, and that such solutions can be quite isolated, reflecting a complex fitness landscape which is far from neutral. Continuous descriptions are needed to capture the size and complexity of the genotype space. Such complexity is also indicated by theoretical studies of continuous-GRN parameter spaces showing multistability (e.g., [104]).

In addition to a more complex description of the evolutionary landscape, modeling at the PDE level in this work has allowed us to specifically investigate the continuous variation of the Bcd gradient, for testing robustness of the GRNs to maternal variability; as well as allowing us to model the effect of transposons as a gradual zeroing of network interactions, rather than as discrete knockouts. The qualitative differences between the discrete and continuous approaches, and the different questions that can be asked with each, warrant careful consideration when developing models or analyzing results.

Acknowledgments

This work was supported by the Joint NSF/NIGMS BioMath Program, 1-R01-GM072022, and the National Institutes of Health, 2-R56-GM072022-06 and 2-R01-GM072022.

References

[1] M. Sanetra, G. Begemann, M. B. Becker, and A. Meyer, "Conservation and co-option in developmental programmes: the importance of homology relationships," *Frontiers in Zoology*, vol. 2, article 15, 2005.

[2] D. Duboule and A. S. Wilkins, "The evolution of 'bricolage'," *Trends in Genetics*, vol. 14, no. 2, pp. 54–59, 1998.

[3] J. R. True and S. B. Carroll, "Gene co-option in physiological and morphological evolution," *Annual Review of Cell and Developmental Biology*, vol. 18, pp. 53–80, 2002.

[4] S. B. Carroll, J. K. Grenier, and S. D. Weatherbee, *From DNA of Diversity: Molecular Genetics and the Evolution of Animal Design*, Blackwell, Malden, Mass, USA, 2001.

[5] S. B. Carroll, "Evolution at two levels: on genes and form," *PLoS Biology*, vol. 3, no. 7, article e245, 2005.

[6] W. Makalowski, "SINEs as a genomic scrap yard," in *The Impact of Short Interspersed Elements (SINEs) on the Host Genome*, J. R. Maraia, Ed., Chapter 5, R.G. Landes Company, Austin, Tex, USA, 1995.

[7] M. L. Siegal and A. Bergman, "Waddington's canalization revisited: developmental stability and evolution," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 16, pp. 10528–10532, 2002.

[8] A. S. Wilkins, *The Evolution of Developmental Pathways*, Sinauer Associates, Sunderland, Mass, USA, 2002.

[9] E. H. Davidson, *Genomic Regulatory Systems: Development and Evolution*, Academic Press, San Diego, Calif, USA, 2001.

[10] W. Arthur, "The concept of developmental reprogramming and the quest for an inclusive theory of evolutionary mechanisms," *Evolution and Development*, vol. 2, no. 1, pp. 49–57, 2000.

[11] E. Abouheif, "Developmental genetics and homology: a hierarchical approach," *Trends in Ecology and Evolution*, vol. 12, no. 10, pp. 405–408, 1997.

[12] E. Abouheif, M. Akam, W. J. Dickinson et al., "Homology and developmental genes," *Trends Genet*, vol. 13, pp. 432–433, 1997.

[13] M. Lynch and A. Force, "The probability of duplicate gene preservation by subfunctionalization," *Genetics*, vol. 154, no. 1, pp. 459–473, 2000.

[14] E. Poustelnikova, A. Pisarev, M. Blagov, M. Samsonova, and J. Reinitz, "A database for management of gene expression data in situ," *Bioinformatics*, vol. 20, no. 14, pp. 2212–2221, 2004.

[15] N. Gompel, B. Prud'homme, P. J. Wittkopp, V. A. Kassner, and S. B. Carroll, "Chance caught on the wing: cis-regulatory evolution and the origin of pigment patterns in Drosophila," *Nature*, vol. 433, no. 7025, pp. 481–487, 2005.

[16] S. Dayal, T. Kiyama, J. T. Villinski, N. Zhang, S. Liang, and W. H. Klein, "Creation of cis-regulatory elements during sea urchin evolution by co-option and optimization of a repetitive sequence adjacent to the spec2a gene," *Developmental Biology*, vol. 273, no. 2, pp. 436–453, 2004.

[17] L. Z. Holland and S. Short, "Gene duplication, co-option and recruitment during the origin of the vertebrate brain from the invertebrate chordate brain," *Brain, Behavior and Evolution*, vol. 72, no. 2, pp. 91–105, 2008.

[18] K. Kawasaki, T. Suzuki, and K. M. Weiss, "Genetic basis for the evolution of vertebrate mineralized tissue," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 31, pp. 11356–11361, 2004.

[19] W. Wang, J. F. Grimmer, T. R. Van De Water, and T. Lufkin, "Hmx2 and Hmx3 homeobox genes direct development of the murine inner ear and hypothalamus and can be functionally replaced by Drosophila Hmx," *Developmental Cell*, vol. 7, no. 3, pp. 439–453, 2004.

[20] K. J. Peterson and E. H. Davidson, "Regulatory evolution and the origin of the bilaterians," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 97, no. 9, pp. 4430–4433, 2000.

[21] M. Levine and R. Tjian, "Transcription regulation and animal diversity," *Nature*, vol. 424, no. 6945, pp. 147–151, 2003.

[22] S. B. Carroll, "Endless forms: the evolution of gene regulation and morphological diversity," *Cell*, vol. 101, no. 6, pp. 577–580, 2000.

[23] R. A. Raff, "Evo-devo: the evolution of a new discipline," *Nature Reviews*, vol. 1, no. 1, pp. 74–79, 2000.

[24] R. A. Raff, *The Shape of Life: Genes, Development and the Evolution of Animal Form*, Chicago University Press, Chicago, Ill, USA, 1996.

[25] E. H. Davidson, *Genomic Regulatory Systems. Development and Evolution*, Academic Press, San Diego, Calif, USA, 2001.

[26] W. Arthur, "The emerging conceptual framework of evolutionary developmental biology," *Nature*, vol. 415, no. 6873, pp. 757–764, 2002.

[27] W. Arthur, "The concept of developmental reprogramming and the quest for an inclusive theory of evolutionary mechanisms," *Evolution and Development*, vol. 2, no. 1, pp. 49–57, 2000.

[28] W. Arthur, "Developmental drive: an important determinant of the direction of phenotypic evolution," *Evolution and Development*, vol. 3, no. 4, pp. 271–278, 2001.

[29] C. H. Waddington, "Canalization of development and the inheritance of acquired characters," *Nature*, vol. 150, no. 3811, pp. 563–565, 1942.

[30] C. H. Waddington, "Genetic assimilation of an acquired character," *Evolution*, vol. 7, pp. 118–126, 1953.

[31] C. H. Waddington, "Genetic assimilation of the bithorax phenotype," *Evolution*, vol. 10, pp. 1–13, 1956.

[32] A. McLaren, "Too late for the midwife toad: stress, variability and Hsp90," *Trends in Genetics*, vol. 15, no. 5, pp. 169–171, 1999.

[33] Manu, S. Surkova, A. V. Spirov et al., "Canalization of gene expression in the Drosophila blastoderm by gap gene cross regulation," *PLoS Biology*, vol. 7, no. 3, article e1000049, 2009.

[34] Manu, S. Surkova, A. V. Spirov et al., "Canalization of gene expression and domain shifts in the drosophila blastoderm by dynamical attractors," *PLoS Computational Biology*, vol. 5, no. 3, 2009.

[35] W. Arthur, T. Jowett, and A. Panchen, "Segments, limbs, homology, and co-option," *Evolution and Development*, vol. 1, no. 2, pp. 74–76, 1999.

[36] A. D. Chipman, "Parallel evolution of segmentation by co-option of ancestral gene regulatory networks," *BioEssays*, vol. 32, no. 1, pp. 60–70, 2010.

[37] N. H. Patel, E. E. Ball, and C. S. Goodman, "Changing role of even-skipped during the evolution of insect pattern formation," *Nature*, vol. 357, no. 6376, pp. 339–342, 1992.

[38] N. H. Patel, "Developmental evolution: insights from studies of insect segmentation," *Science*, vol. 266, no. 5185, pp. 581–590, 1994.

[39] J. Jaeger, S. Surkova, M. Blagov et al., "Dynamic control of positional information in the early Drosophila embryo," *Nature*, vol. 430, no. 6997, pp. 368–371, 2004.

[40] D. E. Clyde, M. S. G. Corado, X. Wu, A. Paré, D. Papatsenko, and S. Small, "A self-organizing system of repressor gradients establishes segmental oomplexity in Drosophila," *Nature*, vol. 426, no. 6968, pp. 849–853, 2003.

[41] J. Jaeger, "The gap gene network," *Cellular and Molecular Life Sciences*, vol. 68, no. 2, pp. 243–274, 2011.

[42] A. D. Chipman, W. Arthur, and M. Akam, "A double segment periodicity underlies segment generation in centipede development," *Current Biology*, vol. 14, no. 14, pp. 1250–1255, 2004.

[43] A. D. Chipman and M. Akam, "The segmentation cascade in the centipede Strigamia maritima: involvement of the Notch pathway and pair-rule gene homologues," *Developmental Biology*, vol. 319, no. 1, pp. 160–169, 2008.

[44] B. C. Goodwin and S. A. Kauffman, "Spatial harmonics and pattern specification in early Drosophila development. Part I. Bifurcation sequences and gene expression," *Journal of Theoretical Biology*, vol. 144, no. 3, pp. 303–319, 1990.

[45] J. Reinitz and D. H. Sharp, "Mechanism of eve stripe formation," *Mechanisms of Development*, vol. 49, no. 1-2, pp. 133–158, 1995.

[46] A. Hunding, S. A. Kauffman, and B. C. Goodwin, "Drosophila segmentation: supercomputer simulation of prepattern hierarchy," *Journal of Theoretical Biology*, vol. 145, no. 3, pp. 369–384, 1990.

[47] Z. Burstein, "A network model of developmental gene hierarchy," *Journal of Theoretical Biology*, vol. 174, no. 1, pp. 1–11, 1995.

[48] L. Sánchez and D. Thieffry, "A logical analysis of the Drosophila gap-gene system," *Journal of Theoretical Biology*, vol. 211, no. 2, pp. 115–141, 2001.

[49] A. V. Spirov, T. Bowler, and J. Reinitz, "HOX Pro: a specialized database for clusters and networks of homeobox genes," *Nucleic Acids Research*, vol. 28, no. 1, pp. 337–340, 2000.

[50] A. V. Spirov, M. Borovsky, and O. A. Spirova, "HOX Pro DB: the functional genomics of hox ensembles," *Nucleic Acids Research*, vol. 30, no. 1, pp. 351–353, 2002.

[51] W. X. Li, "Functions and mechanisms of receptor tyrosine kinase Torso signaling: lessons from Drosophila embryonic terminal development," *Developmental Dynamics*, vol. 232, no. 3, pp. 656–672, 2005.

[52] Y. Kim, Z. Paroush, K. Nairz, E. Hafen, G. Jiménez, and S. Y. Shvartsman, "Substrate-dependent control of MAPK phosphorylation in vivo," *Molecular Systems Biology*, vol. 7, article 467, 2011.

[53] K. A. De Jong, *Evolutionary Computation: A Unified Approach*, MIT Press, Cambridge, Mass, USA, 2006.

[54] A. Wagner, "Does evolutionary plasticity evolve?" *Evolution*, vol. 50, no. 3, pp. 1008–1023, 1996.

[55] I. Salazar-Ciudad, J. Garcia-Fernández, and R. V. Solé, "Gene networks capable of pattern formation: from induction to reaction-diffusion," *Journal of Theoretical Biology*, vol. 205, no. 4, pp. 587–603, 2000.

[56] A. Bergman and M. L. Siegal, "Evolutionary capacitance as a general feature of complex gene networks," *Nature*, vol. 424, no. 6948, pp. 549–552, 2003.

[57] R. V. Solé, P. Fernández, and S. A. Kauffman, "Adaptive walks in a gene network model of morphogenesis: insights into the Cambrian explosion," *International Journal of Developmental Biology*, vol. 47, no. 7-8, pp. 685–693, 2003.

[58] R. B. R. Azevedo, R. Lohaus, S. Srinivasan, K. K. Dang, and C. L. Burch, "Sexual reproduction selects for robustness and negative epistasis in artificial gene networks," *Nature*, vol. 440, no. 7080, pp. 87–90, 2006.

[59] E. Huerta-Sanchez and R. Durrett, "Wagner's canalization model," *Theoretical Population Biology*, vol. 71, no. 2, pp. 121–130, 2007.

[60] S. Ciliberti, O. C. Martin, and A. Wagner, "Robustness can evolve gradually in complex regulatory gene networks with

varying topology," *PLoS Computational Biology*, vol. 3, no. 2, article e15, 2007.

[61] S. Ciliberti, O. C. Martin, and A. Wagner, "Innovation and robustness in complex regulatory gene networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 34, pp. 13591–13596, 2007.

[62] T. MacCarthy and A. Bergman, "Coevolution of robustness, epistasis, and recombination favors asexual reproduction," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 31, pp. 12801–12806, 2007.

[63] O. C. Martin and A. Wagner, "Multifunctionality and robustness trade-offs in model genetic circuits," *Biophysical Journal*, vol. 94, no. 8, pp. 2927–2937, 2008.

[64] J. Masel, "Genetic assimilation can occur in the absence of selection for the assimilating phenotype, suggesting a role for the canalization heuristic," *Journal of Evolutionary Biology*, vol. 17, no. 5, pp. 1106–1110, 2004.

[65] M. L. Siegal, D. E. L. Promislow, and A. Bergman, "Functional and evolutionary inference in gene networks: does topology matter?" *Genetica*, vol. 129, no. 1, pp. 83–103, 2007.

[66] T. MacCarthy and A. Bergman, "The limits of subfunctionalization," *BMC Evolutionary Biology*, vol. 7, no. 1, article 213, 2007.

[67] C. Espinosa-Soto and A. Wagner, "Specialization can drive the evolution of modularity," *PLoS Computational Biology*, vol. 6, no. 3, 2010.

[68] T. Kimbrell and R. D. Holt, "Canalization breakdown and evolution in a source-sink system," *American Naturalist*, vol. 169, no. 3, pp. 370–382, 2007.

[69] J. Draghi and G. P. Wagner, "The evolutionary dynamics of evolvability in a gene network model," *Journal of Evolutionary Biology*, vol. 22, no. 3, pp. 599–611, 2009.

[70] A. Spirov and D. Holloway, "Evolution in silico of genes with multiple regulatory modules on the example of the Drosophila segmentation gene hunchback," in *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pp. 244–251, San Diego, Calif, USA, May 2012.

[71] J. F. Knabe, K. Wegner, C. L. Nehaniv, and M. J. Schilstra, "Genetic algorithms and their application to in silico evolution of genetic regulatory networks," in *Computational Biology*, D. Fenyo, Ed., pp. 297–321, Humana Press, 2010.

[72] A. V. Spirov and D. M. Holloway, "Design of a dynamic model of genes with multiple autonomous regulatory modules by evolutionary computations," in *proceedings of the 10th International Conference on Computational Science 2010 (ICCS '10)*, pp. 1005–1014, June 2010.

[73] P. François, V. Hakim, and E. D. Siggia, "Deriving structure from evolution: metazoan segmentation," *Molecular Systems Biology*, vol. 3, article 154, 2007.

[74] V. V. Gursky, L. Panok, E. M. Myasnikova et al., "Mechanisms of gap gene expression canalization in the Drosophila blastoderm," *BMC Systems Biology*, vol. 5, no. 1, article 118, 2011.

[75] A. V. Spirov and D. M. Holloway, "Retroviral genetic algorithms: implementation with tags and validation against benchmark functions," in *Proceedings of the International Conference on Evolutionary Computation Theory and Applications (ECTA-FCTA) (IJCCI '11)*, vol. 2011, pp. 233–238, 2011.

[76] A. V. Spirov and D. M. Holloway, "New approaches to designing genes by evolution in the computer," in *Real-World Applications of Genetic Algorithms*, O. Roeva, Ed., pp. 235–260, InTech Press, 2012.

[77] A. V. Spirov, "Self-assemblage of gene networks in evolution via recruiting of new netters," in *Lecture Notes in Computer Science*, vol. 1141, pp. 91–100, 1996.

[78] A. V. Spirov and M. G. Samsonova, "Strategy of co-evolution of transposons and host genome: application to evolutionary computations," in *Proceedings of the Third Nordic Workshop on Genetic Algorithms and their Applications*, pp. 71–82, Helsinki University, 1997.

[79] A. V. Spirov and A. S. Kadyrov, "Transposon element technique applied to GA-based John Muir's trail test," in *High-Performance Computing and Networking*, pp. 925–928, 1998.

[80] A. V. Spirov and A. B. Kazansky, "Jumping genes-mutators can raise efficacy of evolutionary search," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '02)*, pp. 561–568, Morgan Kaufmann Publishers, San Francisco, Calif, USA, 2002.

[81] A. V. Spirov and A. B. Kazansky, "The usage of artificial transposons for the protection of already found building blocks: the tests with royal road functions," in *Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI '02)*, pp. 75–80, International Institute of Informatics and Systemics, Orlando, Fla, USA, 2002.

[82] A. V. Spirov, A. B. Kazansky, L. Zamdborg et al., "Forced-evolution in silico by artificial transposonsand their genetic operators: the John Muir Ant problem," http://arxiv.org/abs/0910.5542.

[83] D. M. Holloway, A. B. Kazansky, and A. V. Spirov, "Complexification of gene networks by co-evolution of genomes and genomic parasites," in *Proceedings of the 4th International Conference on Evolutionary Computation Theory and Applications (ECTA '12)*, Barcelona, Spain, October 2012.

[84] E. R. Lozovskaya, D. L. Hartl, and D. A. Petrov, "Genomic regulation of transposable elements in Drosophila," *Current Opinion in Genetics and Development*, vol. 5, no. 6, pp. 768–773, 1995.

[85] M. R. Wallace, L. B. Andersen, A. M. Saulino, P. E. Gregory, T. W. Glover, and F. S. Collins, "A de novo Alu insertion results in neurofibromatosis type 1," *Nature*, vol. 353, no. 6347, pp. 864–866, 1991.

[86] L. Girard and M. Freeling, "Regulatory changes as a consequence of transposon insertion," *Developmental Genetics*, vol. 25, no. 4, pp. 291–296, 1999.

[87] C. C. King, "Modular transposition and the dynamical structure of eukaryote regulatory evolution," *Genetica*, vol. 86, no. 1-3, pp. 127–142, 1992.

[88] E. L. Haseltine and F. H. Arnold, "Synthetic gene circuits: design with directed evolution," *Annual Review of Biophysics and Biomolecular Structure*, vol. 36, pp. 1–19, 2007.

[89] Y. Yokobayashi, R. Weiss, and F. H. Arnold, "Directed evolution of a genetic circuit," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 26, pp. 16587–16591, 2002.

[90] B. Houchmandzadeh, E. Wieschaus, and S. Leibler, "Establishment of developmental precision and proportions in the early Drosophila embryo," *Nature*, vol. 415, no. 6873, pp. 798–802, 2002.

[91] D. M. Holloway, J. Reinitz, A. Spirov, and C. E. Vanario-Alonso, "Sharp borders from fuzzy gradients," *Trends in Genetics*, vol. 18, no. 8, pp. 385–387, 2002.

[92] A. V. Spirov and D. M. Holloway, "Making the body plan: precision in the genetic hierarchy of Drosophila embryo

segmentation," *In Silico Biology*, vol. 3, no. 1-2, pp. 89–100, 2003.

[93] B. Houchmandzadeh, E. Wieschaus, and S. Leibler, "Precise domain specification in the developing Drosophila embryo," *Physical Review E*, vol. 72, no. 6, Article ID 061920, 2005.

[94] D. M. Holloway, L. G. Harrison, D. Kosman, C. E. Vanario-Alonso, and A. V. Spirov, "Analysis of pattern precision shows that Drosophila segmentation develops substantial independence from gradients of maternal gene products," *Developmental Dynamics*, vol. 235, no. 11, pp. 2949–2960, 2006.

[95] T. Gregor, D. W. Tank, E. F. Wieschaus, and W. Bialek, "Probing the Limits to Positional Information," *Cell*, vol. 130, no. 1, pp. 153–164, 2007.

[96] A. Wagner, "Distributed robustness versus redundancy as causes of mutational robustness," *BioEssays*, vol. 27, no. 2, pp. 176–188, 2005.

[97] R. Kraut and M. Levine, "Spatial regulation of the gap giant during Drosophila development," *Development*, vol. 111, no. 2, pp. 601–609, 1991.

[98] E. Mjolsness, D. H. Sharp, and J. Reinitz, "A connectionist model of development," *Journal of Theoretical Biology*, vol. 152, no. 4, pp. 429–453, 1991.

[99] S. Surkova, D. Kosman, K. Kozlov et al., "Characterization of the Drosophila segment determination morphome," *Developmental Biology*, vol. 313, no. 2, pp. 844–862, 2008.

[100] E. Poustelnikova, A. Pisarev, M. Blagov, M. Samsonova, and J. Reinitz, "A database for management of gene expression data in situ," *Bioinformatics*, vol. 20, pp. 2212–2221, 2004.

[101] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes*, Cambridge University Press, Cambridge, UK, 1988.

[102] H. Hardway, B. Mukhopadhyay, T. Burke, T. J. Hitchman, and R. Forman, "Modeling the precision and robustness of Hunchback border during Drosophila embryonic development," *Journal of Theoretical Biology*, vol. 254, no. 2, pp. 390–399, 2008.

[103] T. J. Perkins, J. Jaeger, J. Reinitz, and L. Glass, "Reverse engineering the gap gene network of Drosophila melanogaster," *PLoS Computational Biology*, vol. 2, no. 5, pp. 417–428, 2006.

[104] C. B. Muratov and S. Y. Shvartsman, "An asymptotic study of the inductive pattern formation mechanism in Drosophila egg development," *Physica D*, vol. 186, no. 1-2, pp. 93–108, 2003.

The Scientific World Journal
Volume 2012, Article ID 130491, 10 pages
doi:10.1100/2012/130491

The ScientificWorldJOURNAL

Research Article

Network Analysis of Functional Genomics Data: Application to Avian Sex-Biased Gene Expression

Oliver Frings,[1,2] Judith E. Mank,[3] Andrey Alexeyenko,[1,4] and Erik L. L. Sonnhammer[1,2,5]

[1] *Stockholm Bioinformatics Centre, Science for Life Laboratory, Box 1031, SE-171 21 Solna, Sweden*
[2] *Department of Biochemistry and Biophysics, Stockholm University, SE-106 91 Stockholm, Sweden*
[3] *Department of Genetics, Evolution and the Environment, University College London, WC1E 6BT, UK*
[4] *School of Biotechnology, Royal Institute of Technology, SE-171 65 Solna, Sweden*
[5] *Swedish eScience Research Center, SE-100 44 Stockholm, Sweden*

Correspondence should be addressed to Erik L. L. Sonnhammer, erik.sonnhammer@sbc.su.se

Received 25 September 2012; Accepted 25 November 2012

Academic Editors: R. Jiang, W. Tian, J. Wan, and X. Zhao

Gene expression analysis is often used to investigate the molecular and functional underpinnings of a phenotype. However, differential expression of individual genes is limited in that it does not consider how the genes interact with each other in networks. To address this shortcoming we propose a number of network-based analyses that give additional functional insights into the studied process. These were applied to a dataset of sex-specific gene expression in the chicken gonad and brain at different developmental stages. We first constructed a global chicken interaction network. Combining the network with the expression data showed that most sex-biased genes tend to have lower network connectivity, that is, act within local network environments, although some interesting exceptions were found. Genes of the same sex bias were generally more strongly connected with each other than expected. We further studied the fates of duplicated sex-biased genes and found that there is a significant trend to keep the same pattern of sex bias after duplication. We also identified sex-biased modules in the network, which reveal pathways or complexes involved in sex-specific processes. Altogether, this work integrates evolutionary genomics with systems biology in a novel way, offering new insights into the modular nature of sex-biased genes.

1. Introduction

Although primary sex determining genes are responsible for the initial sex determining cues in the gonad, most of the heritable differences in morphology, behavior, and life history between males and females are the result of different expression levels of genes present in both sexes [1, 2]. Sex-biased genes, which comprise up to 50% of metazoan transcriptomes [3–7], are the product of sexually antagonistic selection for different male and female optima [8, 9]. This antagonism is resolved with the emergence of sex-specific transcriptional regulatory elements that decouple expression between the sexes, thereby allowing separate female and male phenotypes to emerge from a shared genome. Sex-biased genes behave according to the evolutionary predictions for sexually selected and sexually antagonistic traits [10–18], and the study of sex-biased gene expression is emerging as a method to connect sex-specific selection pressures, which act on the whole organism, to the encoding loci.

This connection between sex-biased genes and sexually dimorphic traits offers a way to study the complex interactions between the phenotype to the underlying genome. Most studies of sex-biased gene expression treat individual genes as independent units, ignoring correlated expression that results from the interactive nature of genetic pathways and networks. This simplification compresses the multidimensional nature of the transcriptome. However, because many sexually dimorphic phenotypes are complex amalgams of numerous genes [5, 19, 20], we require a way to study the interactions of the genes underlying them if we wish to understand the constraints acting on these traits and how they respond to selection. In addition to this complexity, many genes contribute to more than one phenotype, pathway, or subnetwork. This pleiotropy is likely an important factor in the evolution of sex-biased gene expression which may ameliorate intralocus sexual conflict acting on a given gene [9].

For genes with high levels of pleiotropy, the many functions of a single locus result in strong evolutionary constraints hindering change due to selection pressure for any single function [21]. This is important for studies of sex-biased genes, as sex-biased gene expression patterns resulting from sexually antagonistic selection for any single function may be detrimental in other functionalities [22]. This would suggest that genes with many pathway connections, though not necessarily less likely to experience sexually antagonistic selection, are less likely to resolve that antagonism through sex-biased expression, as this could result in detrimental effects in other phenotypes encoded by the same loci [23]. More simply stated, the resolution of sexually antagonistic selection may be more common for genes with fewer network interactions. This prediction suggests that (1) pleiotropic genes may contain relatively high levels of unresolved sexual conflict and (2) sexually dimorphic phenotypes are more often encoded by genes with few other functions. This has important implications for evolutionary models of sexual selection which typically assume single functionalities and simple inheritance patterns.

Here we test the relationship between network interaction and sex-biased gene expression with a newly developed gene interaction atlas of the chicken. Previously, we have shown that sex-biased expression is prevalent in chicken [23] and that sex-biased genes in chicken exhibit evolutionary patterns consistent with sexual selection and sexual conflict [16, 18, 24]. In this analysis, we created a functional coupling network from data integration [25] of chicken and incorporated sex-biased expression data into it in order to analyze the connectivity of sex-biased and unbiased genes in both the gonad and soma. Overall, our goal was to better understand the relationship between sexually dimorphic phenotypes, the sexually antagonistic selection pressures shaping them, and the genes encoding them.

2. Materials and Methods

2.1. Network. The chicken network was generated using the FunCoup framework [25, 26]. This framework reconstructs global large-scale networks of functional coupling by Bayesian integration of diverse high-throughput datasets. More specifically raw scores of various types of functional coupling are turned into probabilistic estimates that are then integrated across different types of data and model organisms. The different types of evidence comprised: protein-protein interactions, mRNA coexpression, subcellular colocalization, phylogenetic profile similarity, cotargeting by either miRNA or transcription factors, protein co-expression, and domain-domain interactions. The integration of data from different sources enabled more comprehensive network reconstruction with higher quality. Furthermore, data from other eukaryotic species were transferred via orthologs. Ortholog assignments for cross-species mapping were obtained from the InParanoid database [27]. Signaling and metabolic pathways from KEGG as well as both pathway types combined were used as gold standard for Bayesian training. The network has consequently three different kinds of links: metabolic, signaling, and combined. The network was predicted using seven chicken-specific microarray expression datasets (see Table S1 in Supplementary Materials available online at doi:10.1100/2012/130491), phylogenetic profile similarity across eukaryotes, and information transferred from other species via orthologs. The use of ortholog transfer was of special importance in this case, as it allows us to overcome the lack of chicken-specific interaction data.

2.2. Microarray Expression Datasets. The network was studied in the context of sex bias under different conditions. We used three different Affymetrix chicken expression datasets from the embryonic gonad, the adult gonad, and the adult brain (previously described in Mank et al. [16], Mank and Ellegren [28], and Mank et al. [24]). Each tissue/time-point array hybridization was based on three replicate nonoverlapping within-sex pools of 3–5 individual samples from male and female embryonic and adult chickens. All datasets were normalized using the MAS5 algorithm from the Affy Bioconductor package.

2.3. Differential Gene Expression. There are several different ways to define differential gene expression. Traditionally genes that are meant to be over- or underrepresented in one condition compared to a second condition have been identified by fold-change. Although this method is still widely used, it might be biased in multiple ways. A high fold-change can be caused by a single flawed sample or by negligible differences in expression level just above the detection limit. In other words it ignores if the differences in expression change are statistically significant or not. Different methods have been proposed to assess the significance of changes in gene expression. The Student's t-test and Welch test are commonly used to estimate the significance of differential gene expression. However, the reliability of those methods strongly depends on the sample variance and the number of samples for each condition. Besides numerous statistical packages have been developed that account for differential gene expression, for example, SAM, EBAM, and so forth.

It also has been widely recognized that using different methods might result in rather distinct sets of differential expressed genes. We approached the problem by using the R MWT-package to determine significant differential gene expression [29]. The MWT method is essentially a moderated Welch test that aims to circumvent the problem of a low sample number by pooling the variance over the whole probe set. To adjust for multiple testing, all P values related to differential expression were corrected using the Benjamini-Hochberg method [30] that is rendered into false discovery rate (FDR) values.

2.4. Network Randomization. To determine the significance of the level of connectivity between a predefined set of genes and a second set (or itself) we used the CrossTalkZ network randomization package (http://sonnhammer.sbc.su.se/download/software/CrossTalkZ/). The method compares the number of observed connections between two gene sets to the number of connections in a randomized version of the

network. In the course of network randomization, links between genes are swapped so that the original connectivity of a gene is conserved. The randomization was repeated 100 times, and all results were averaged. For each gene set a number of statistics were calculated including a z-score, a P value, and a Benjamini-Hochberg corrected FDR value.

2.5. Functional Gene Modules. To identify gene modules that are relevant to different developmental stages and sexes we compiled for each condition networks of male or female-biased genes separately. In addition these networks contained other genes strongly connected to those sex-biased genes. We used the hypergeometric test to identify such genes, and genes with a Bonferroni corrected P value of less than 10% were included in such networks.

A large number of network clustering techniques exist to infer modules, but it is not obvious which ones are most robust, that is, perform well under many different circumstances. From a benchmark study of 8 popular methods we selected the two overall top performing methods, MGclus (http://sonnhammer.sbc.su.se/download/software/MGclus/) and MCL [31]. The latter was used with an inflation parameter of 3.5. The significance of the derived modules was evaluated by comparing the number of enriched GO terms per module to the expected number of enriched GO terms given a set of genes of that size. The expected number of GO terms per module was estimated by 500 times randomly picking n genes from the parental subnetwork, where n equals the number of genes for a module. Based on the distributions of the expected numbers of enriched GO terms, a z-score was calculated for enrichment of GO terms per clustering.

3. Results

3.1. The Chicken Network. With the FunCoup tool and dataset collection, we derived a global chicken gene interaction network. FunCoup can be used to determine confidence values regarding the value of observed functional coupling links, and the chicken network has roughly 1.8 million links at a confidence cutoff (c) > 0.02 and about 58,000 at $c > 0.75$ (Table 1). The network was trained on three different categories: metabolic, signaling, and both metabolic and signaling combined. In the following we used a $c > 0.25$ as it represents a reasonable tradeoff between accuracy and coverage.

The proteins with the highest connectivity are mainly related to fundamental cellular processes such as protein synthesis and degradation, translation, and transcription (Table S2). Many of them are involved in multiple processes. The most connected protein in our chicken network is the RA-related nuclear protein (RAN). Due to its various functions in nuclear transport and cell cycle regulation, it acts as a major hub with a host of other proteins. Interestingly, RAN is highly differentially expressed between male and female chicken (i.e., sex biased) in the gonad (FDR $P < 10^{-4}$ in the adult), which is actually less common for hubs as we show in the following.

3.2. Sex Bias Depends on Network Connectivity. Is there a dependency between sex bias and network connectivity? To answer this question, we first grouped the genes in three sex bias categories: male biased, female biased, and unbiased. For this we used the MWT statistic of differential expression with an FDR P value cutoff of 0.1. This was done for all four tissue/stage conditions: the embryonic and adult gonad and brain. The number of sex-biased genes in the network for each category is shown in Table 2. Remarkably, the embryonic brain contained almost no sex-biased genes and was therefore left out of this analysis. The adult brain had more sex-biased genes, but these still represented only 3% of the genes in the network. In contrast, the gonad abounded with sex-biased genes in the network: 43% in the embryo and 82% in the adult.

Sex-biased hub genes were thus frequent in the gonad, but not in the brain, and this may be due to the fact that the male and female gonads have extensive sex-specific functions, while the brain consists of many different tissues of which only small fractions of our microarray samples may be affected by the sex. The sex-specific expression signal in the brain will therefore be diluted by the nonaffected tissues until it is no longer statistically significant. Finer-scale analysis of specific brain tissues might reveal more dimorphism in gene expression, particularly those regions related to vocalization differences between male and female birds [32] or reproductive behavior [33].

We calculated Spearman's rank correlation coefficient between FDR values from differential expression analysis and node degree (i.e., the number of connections a gene has in the network), for each tissue/stage combination. As can be seen in Table 3, all but one of the sex-biased categories in the gonads had a significant positive correlation at FDR $P < 0.1$, indicating a tendency for fewer network connections as sex-bias increased. The exception was male biased genes in the adult gonad, but when lowering the cutoff to 0.001 these gave a weak but significant positive correlation ($r = 0.1, P < 0.05$). Unbiased genes were not significantly correlated with connectivity, nor were the brain genes, as may be expected given the dilution problem of brain expression mentioned above. This trend also held true when using fold-change as a measure for sex-bias. In other words, sex biased pathways seemed to generally affect local components of the network, except for the ones overexpressed in the male adult gonad, which tends to act more often in global components.

3.3. Sex-Biased Hub Genes. From the previous section it is clear that the level of sex was a function of both by tissue/stage condition as well as connectivity. We demonstrated that low connectivity genes tend to be more sex biased than high connectivity genes, yet some hub genes have strong sex bias. To focus on such sex-biased hubs, we first ranked each gene according to sex bias or connectivity separately and then reranked them according to the sum of both ranks. The highest ranked genes thus represent the most sex-biased hub genes. Table S3 shows the twenty top ranked sex-biased hubs in each condition.

Table 1: Number of links (first number) and unique genes (second number) at different FBS (final Bayesian score) cutoffs in FunCoup, where c is the corresponding confidence value of functional coupling.

	Metabolic	Signaling	Combined	Total
FBS > 3 ($c > 0.02$)	1375931/10555	601101/10569	1152763/10549	1809810/11311
FBS > 5.9 ($c > 0.25$)	171490/5520	33934/4861	124616/5383	199120/6748
FBS > 7 ($c > 0.5$)	89818/4132	13401/2990	62707/3885	100887/4902
FBS > 8 ($c > 0.75$)	52285/3175	6365/1869	35821/2930	57690/3673

Table 2: Number of sex-biased and unbiased genes separated by the cut-off FDR < 0.1 according to MWT.

	Male	Female	Unbiased
Embryonic gonad	1304	1128	3244
Adult gonad	1934	2693	1049
Embryonic brain	0	2	5675
Adult brain	87	92	5497

Table 3: Sex-biased genes tend not to be hubs. This is evidenced by Spearman's correlation coefficient between differential expression (measured as MWT FDR) and network connectivity which was significantly positive in most cases. Sex-biased genes were separated from unbiased genes using a cutoff of FDR = 0.1. The first number is the correlation, and in brackets is the corresponding P value. Significant correlations ($P < 0.01$) are marked in bold.

	Male	Female	Unbiased
Embryonic, gonad	**0.12 (7.98e − 06)**	**0.16 (1.55e − 07)**	−0.03 (0.12)
Adult, gonad	−0.04 (0.10)	**0.06 (9.91e − 04)**	−0.003 (0.93)
Adult, brain	0.12 (0.29)	−0.06 (0.60)	0.03 (0.02)

Among the highly connected hubs in Table S3, tubulin alpha-3e (TUBA3E), ranked first and second in the embryonic and adult gonads, has 426 links. It is female biased in the embryonic gonad but male biased in the adult gonad. Other highly connected tubulins are also in the list. This indicates that sexual differentiation and sex-specific function are partly orchestrated via sex-specific tubulin assembly. Some of the proteins in Table S3 are directly implicated in sex determination, for example, the testis-specific tubulin alpha-2 (TUBAL2, connectivity 94), the meiotic recombination SPO11 (connectivity 37), or the NASP the nuclear autoantigenic sperm protein (connectivity 97). A major hub in the embryonic as well as the adult gonad is CDK3, cell division protein kinase 3 (connectivity 189). CDK3 is further linked to the KEGG pathways oocyte meiosis as well as progesterone-mediated oocyte maturation. Intriguingly, CDK3 was strongly female biased in the embryonic gonad but strongly male biased in the adult gonad. Overall this suggests that a major difference between the sexes results from a complex interplay between components of the cell division and development systems.

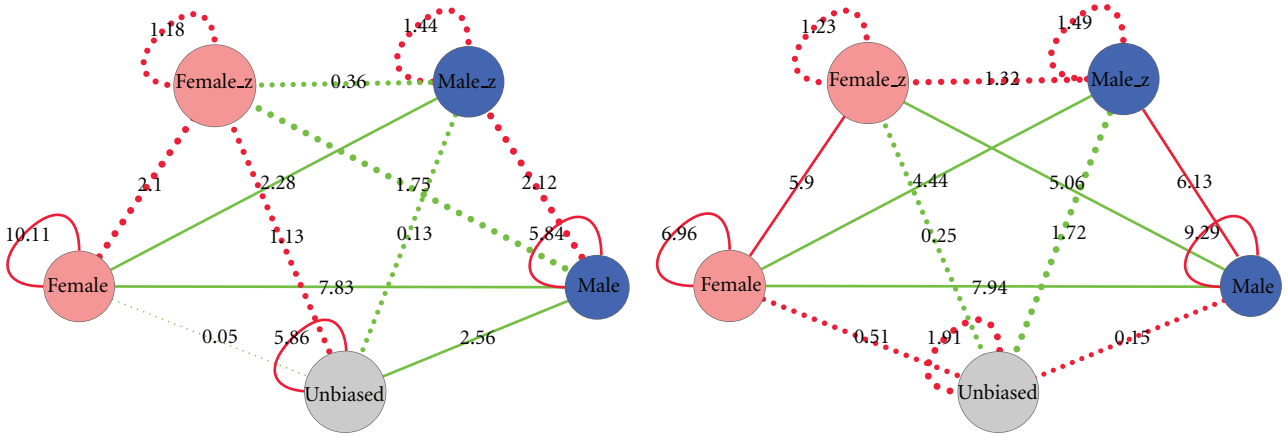
3.4. Interconnectivity of Sex-Biased Genes. To answer the question if sex-biased genes are stronger connected to genes of the same bias, we compared the connection frequencies between the different categories to what is expected by chance alone. For topology-preserving randomization of the network we used the CrossTalkZ program (see Section 2) and performed 100 randomization runs. The results for the embryonic and adult gonads are shown in Figure 1.

In the gonad we found genes of the same sex bias (e.g., male versus male) to be more frequently connected to each other than to genes of a different sex bias or unbiased genes. It is striking that both in the embryonic and adult gonads, male-biased genes have significantly fewer connections to female-biased genes than expected by chance. In the brain, we did not observe a significant crosstalk between male- or female-biased genes, probably due to the dilution problem mentioned above. Separate female- and male-specific networks are thus common throughout the chicken network in the gonad, and these sex-specific networks function to encode dimorphic processes in this tissue.

Sex-biased genes on the Z-chromosome are shown as separate nodes in Figure 1. The Z chromosome had to be treated separately from the autosomes due to the lack of complete dosage compensation in birds which results in a pervasive male bias for nondosage sensitive genes [34]. Sex-biased genes on the Z-chromosome are shown as separate nodes in Figure 1. Genes of the same sex bias located on the Z-chromosome were more connected to each other than expected by chance and were significantly enriched in links to genes of the same sex bias on other chromosomes. Connections between female and male genes on the Z-chromosome were about as frequent as expected, but there were significantly fewer connections than expected between whole-genome male and Z-chromosome female-biased genes and vice versa. These results show that the reconstructed chicken network is largely made up of male-specific and female-specific modules.

Table 4: Results of the inparalog group analysis showing the number of groups in the various categories. In total we found 69 groups with at least two inparalogs in chicken. However, only 59 could be processed since expression data were not available for all genes in ten of the groups. The number in brackets in the mixed cluster field is the number of groups that contain both male- and female-biased genes. A significant difference between the observed number of inparalogs and what is expected by chance is indicated by $^*(P < 0.05)$, + indicates a number higher than expected by chance, and − a number lower than expected.

	Gonad adult	Gonad embryo	Brain adult
All-male	15+*	7+*	0
All-female	18+*	8+*	0
All-unbiased	6+*	25	58+
Mixed	20 (5)−*	19 (1)−*	1 (0)−



(a) Embryonic gonad (b) Adult gonad

Figure 1: Network of crosstalk, that is, enrichment or depletion of links, between sex-biased and unbiased genes. Positive crosstalk (i.e., enrichment of links) is shown in red and depletion in green. Solid lines indicate significant crosstalk with FDR < 0.05. Edge width and label show the z-score of the crosstalk analysis.

3.5. Duplicated Sex-Biased Genes. Gene duplication is a mechanism for creating new functions, and such a functional niche could be associated with a particular sex bias. Previous work has shown that duplicates of unbiased genes often develop sex-biased expression [35]. However, it is not yet clear if sex-biased genes that were recently duplicated tend to maintain the same pattern of expression bias. To answer this question, we restricted the analysis to orthologs. Orthologous genes are known to retain identical or closely related biological function more often than other types of homologs [36–39]. Two genes in one species are considered as inparalogs with respect to another species if the gene duplication occurred after the respective speciation event. In order to clarify if inparalog genes in chicken would more often have the same sex bias or are biased towards the opposite sex, we selected all inparalogs between chicken and human from the InParanoid database [27]. An ortholog group was only analyzed if it had at least two alternative inparalogs in chicken and if expression data were available. Roughly half of the groups could thus be analyzed. The group was then evaluated for differentially expressed genes using a FDR cutoff of 10%.

The results of this analysis can be seen in Table 4 (and Table S4). In the gonad, the numbers of male- and female-biased groups were similar, while in the brain none of the groups were biased towards one of the sexes. A big fraction of the groups is however a mix between sex-biased and unbiased genes. Remarkably, five ortholog groups in the adult gonad contained both male- and female-biased genes. An example of such a group contained female-biased glutathione S-transferase 2 (GSTA2) and male-biased glutathione S-transferase 3 (GSTA3). These inparalogs were connected to each other in the FunCoup network as well as to a set of other sex-biased genes (see Figure 2). However, 75% of GSTA2 links and 48% of GSTA3 links were not in common. At a cutoff of 0.5 only 2 links were shared between the two genes. It thus represents a likely example of subfunctionalization driven by sex differentiation.

To evaluate significance of these findings we compared the obtained numbers of inparalogs with the same bias to the distribution expected by chance (see Table S4). To this end, we randomly sampled genes of each ortholog group from the complete expression dataset. This procedure was repeated 1000 times, and the obtained numbers of groups with the same sex bias were compared to the observed values. For both the embryonic and adult gonadsthe original number of inparalogs with the same sex bias significantly exceeded the number of what would be expected by chance alone ($P < 0.05$). In the brain however there was no clear trend. We conclude that inparalogs that emerged after the mammal-bird speciation generally preserved sex bias, although a few exceptions exist.

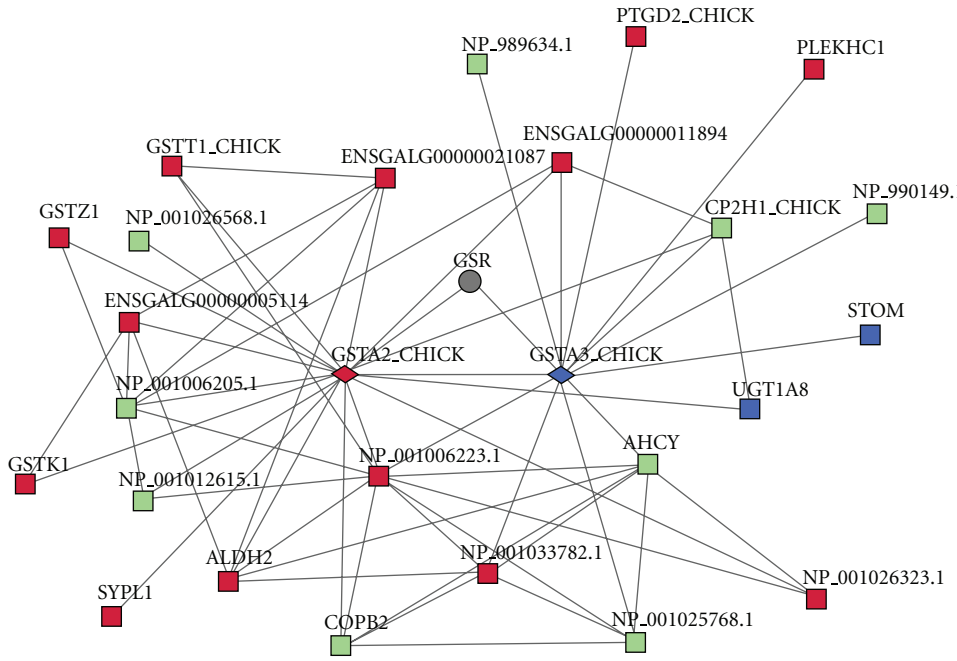


Figure 2: Example of sex-differentiation-driven subfunctionalization. The chicken genes GSTA2 and GSTA3 (glutathione S-transferases 2 and 3; shown as diamonds) originate from a duplication that happened after the divergence from human, making them inparalogs. GSTA2 is male biased, but GSTA3 is female biased in the adult gonad. Their interaction partners in the chicken network are shown with sex bias. Male-biased genes are shown in blue, female-biased in red, unbiased in green, and unknown in grey.

3.6. Sex-Biased Network Modules. Network modules, or clusters, can be useful to find groups of functionally related genes. Such modules may represent parts of pathways or complexes that can be discerned as cliques of genes that are strongly linked to each other in the network. To identify functional modules of sex-biased genes, we calculated for each condition a set of male- or female-biased modules. We derived a network of sex-biased genes as well as genes which were strongly connected to them for each condition (see Section 2). Different clustering methods have different advantages and disadvantages and might as well result in relatively different sets of modules. We used two different methods to derive functional modules, MCL and MGclus. In the following we contrast the results of both methods as well as discuss the significance of the derived modules based on a few selected examples.

MCL is a global clustering approach that simulates random walks in the underlying interaction network. MGclus tries to identify clusters of strongly mutually linked genes using a scoring function that additionally accounts for shared neighbors. Thus nodes in the same cluster are thus likely to share a large fraction of shared neighbors, which increases cohesiveness within the cluster.

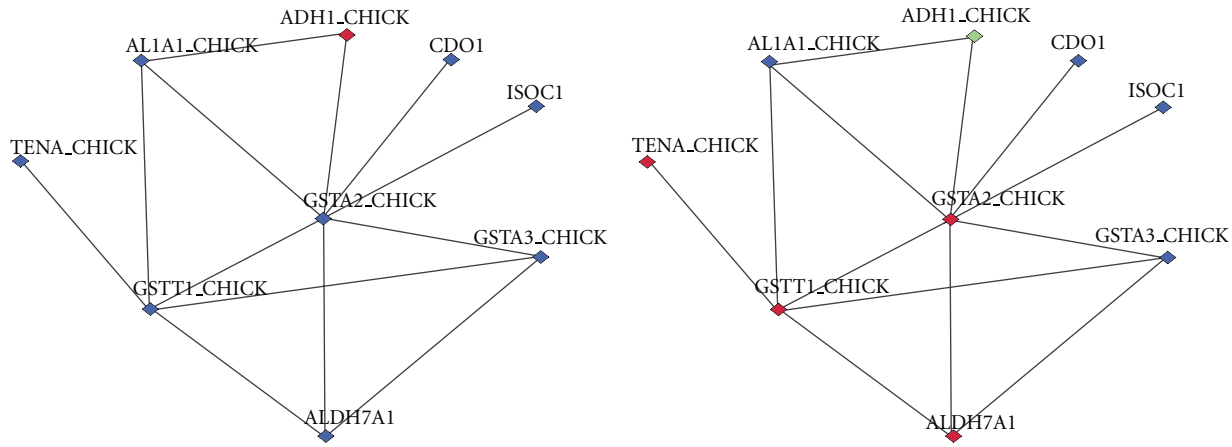
The overall outcomes of the MGclus and MCL clusterings are shown in Table 5. In all of the cases the clusters were significant, that is, had at least one enriched GO term, which was assigned to more than one gene in the cluster. Further, in all cases except for the adult brain, all clusters had on average significantly more enriched GO terms than random modules of the same size. The adult brain might however have too few sex-biased genes to see this. It is also worth noting that the MGclus clusters had on average more enriched GO terms than the MCL clusters.

How different are MCL and MGclus clusters? The overlap strongly depended on the size of the input network. While the overlap was notable for smaller networks (e.g., the embryonic gonad or brain), it was limited for larger networks. To illustrate the overlap between the different clusterings we calculated UPGMA trees based on the fraction of the intersection relative to the union (Jaccard index) of genes in MCL and MGclus clusters. The same was done for enriched GO terms. For the male adult gonad, a few MGclus and MCL clusters overlap to a high degree, but most of them do not have a counterpart with more than 30% overlap (Figure S2). On the other hand, for the male embryonic gonad, which had much fewer differential expressed genes, most of them find a counterpart with more than 60% overlap (Figure S3), indicating that these clusters are relatively reliable. Unsurprisingly, gene and GO term overlap trees were very similar.

One module identified from the embryonic gonad contained eight male-biased genes and one female-biased gene (Figure 3(a)). The female-biased gene was included because it was significantly enriched in connections to the male-biased genes. This module was functionally related to cell growth and development. It contained eight enzymes with biosynthetic functions and one extracellular matrix protein Tenascin (TENA_CHICK) which is important in tissue development. Two of the enzymes (AL1A1_CHICK, ADH1_CHICK) are important for retinoic acid (RA) synthesis [40]. RA is known to be crucial for embryonic development, growth, and reproduction. Four of the genes

Table 5: Number of MGclus and MCL clusters, number of clusters with significant GO term enrichment, and the level of significant GO term enrichment compared to random. A z-score above 2 corresponds to a significance level of $P < 0.05$.

	Clusters	Significant	Avg. sig. terms	Random avg. sig. terms
		MGclus		
Gonad embryo male	6	6	43.50	18.72
Gonad adult male	24	24	20.58	9.56
Brain adult male	3	3	16.67	13.01
Gonad embryo female	6	6	31.17	16.09
Gonad adult female	22	22	62.23	23.58
Brain adult female	3	3	25.67	20.94
		MCL		
Gonad embryo male	5	5	30.80	16.87
Gonad adult male	31	29	18.52	6.92
Brain adult male	6	6	9.00	6.94
Gonad embryo female	3	3	16.67	10.85
Gonad adult female	64	64	27.97	10.62
Brain adult female	3	3	24.33	16.42



(a) Embryonic gonad (b) Adult gonad

Figure 3: Example of sex bias switching between developmental stages. Shown is an MGclus cluster colored according to sex bias in the embryonic (a) and adult (b) gonads. Male-biased genes are shown in blue, female-biased in red, and unbiased in green.

change sex bias and become female-biased in the adult gonad (Figure 3(b)), indicating that this module switches its function during development depending on the sex.

4. Discussion

By analyzing sex bias within the chicken gene network, we have been able to deduce several network properties pertaining to sexual dimorphism that gives new biological insights. Our analysis suggests that network hub genes tend not to be sex biased, although with some interesting exceptions. This suggests that most sex-biased genes tend to act within local network environments, and relatively few of them interact on a more global scale. This is consistent with recent studies that show that pleiotropy, as measured by expression breadth, tends to constrain the evolution of sex-biased expression [41, 42]. This analysis extends the measure of pleiotropy to network connectivity, with broadly consistent results.

We also investigated the propensity of sex-biased genes to form network modules in several ways. First, we noted that genes of the same sex bias tend to be more connected to each other than expected. Second, recently duplicated genes, which are similar in biochemical function, tend to have the same sex bias. Finally, a set of sex-biased modules were extracted from the network, and these showed unexpected functional homogeneity. These observations support a network structure that embodies sex-biased network modules. The implication of this is that the mechanisms underlying sex-specific development can be organized according to these modules, which simplifies the study and understanding of this complex system.

This work provides the first integrated, multidimensional analysis of the network structure underlying sex-biased gene expression and, as such, offers a more realistic link between sex-biased gene expression and sexually dimorphic phenotypes. Our analysis suggests, that rather than operating as distinct entities, genes of the same sex bias often group

together in network modules, potentially due to shared regulatory elements or hierarchical pathway structures. This has several evolutionary genetic implications. First, it suggests that when many genes act in concert to encode sexually dimorphic phenotypes, they may be controlled by a shared regulatory apparatus. This collective regulatory control could then be exploited by emergent sexual dimorphisms, resulting in associated phenotypic differences [43]. It also suggests that single- or oligolocus models of sexual selection evolution (e.g., [44, 45]) are appropriate for some sexually dimorphic traits, even when transcriptome analysis reveals that gene expression of those phenotypes differs for many genes between the sexes. Although genes do not operate as independent units but are rather tethered in modules in a complex network of interactions, they however often work in concerted regulatory patterns. Therefore, our analysis somewhat paradoxically suggests that the control of complex sexual dimorphism may be ultimately attributable to relatively few key regulators.

Sex chromosomes often exhibit a nonrandom distribution of sex-biased genes associated with masculinizing or feminizing selection [46, 47]. Additionally, female heterogametic sex chromosomes, including those exhibited by birds, are also predicted to be particularly associated with the evolution of certain types of sexually selected traits [45, 48, 49]. Our analysis is consistent with these predictions. The crosstalk observed in the adult gonad between sex-biased genes on the Z chromosome and sex-biased genes on the autosomes suggests that the Z chromosome, which contains a relatively modest proportion of the total avian coding content, may play a disproportionately large role in the regulation of sex-biased genes.

Previous work has shown a nonrandom distribution of sex-biased genes on the avian Z chromosome [50–52], with more male-biased and fewer female-biased genes on the Z chromosome than would be expected by chance alone. However this issue is complicated by the incomplete dosage compensation observed on the avian Z chromosome. Studies in a range of bird species have shown a persistent male bias on the Z chromosome due to the fact that males have two copies of every locus and females just one [34, 53, 54]. It has therefore been difficult to disentangle the effects of masculinizing selection for gene expression from incomplete dosage compensation [18]. Our analysis does not suffer from this type of conflation, as the crosstalk enrichment takes the relative abundances of different biases into account. This should minimize any effects of incomplete dosage compensation on our network.

In conclusion, our results suggest that network approaches to the study of sex-biased gene expression can offer new insights into the programming and genetic basis of sexual differentiation. Current transcriptome profiling produces massive datasets measuring relative gene expression, but this approach alone results in the false perception that each locus is independent of all others. Gene network approaches such as the one described here make it possible to consider a more multidimensional and integrated view of genome regulation which is particularly insightful for complex phenotypes.

Acknowledgments

O. Frings was supported by a grant from the Swedish Research Council. J. E. Mank is supported by the BBSRC and ERC (Grant AGREEMENT 260233).

References

[1] H. Ellegren and J. Parsch, "The evolution of sex-biased genes and sex-biased gene expression," *Nature Reviews Genetics*, vol. 8, no. 9, pp. 689–698, 2007.

[2] J. E. Mank, "Sex chromosomes and the evolution of sexual dimorphism: lessons from the genome," *The American Naturalist*, vol. 173, no. 2, pp. 141–150, 2009.

[3] J. M. Ranz, C. I. Castillo-Davis, C. D. Meiklejohn, and D. L. Hartl, "Sex-dependent gene expression and evolution of the Drosophila transcriptome," *Science*, vol. 300, no. 5626, pp. 1742–1745, 2003.

[4] P. Khaitovich, I. Hellmann, W. Enard et al., "Evolution: parallel patterns of evolution in the genomes and transcriptomes of humans and chimpanzees," *Science*, vol. 309, no. 5742, pp. 1850–1854, 2005.

[5] X. Yang, E. E. Schadt, S. Wang et al., "Tissue-specific expression and regulation of sexually dimorphic genes in mice," *Genome Research*, vol. 16, no. 8, pp. 995–1004, 2006.

[6] Y. Zhang, D. Sturgill, M. Parisi, S. Kumar, and B. Oliver, "Constraint and turnover in sex-biased gene expression in the genus Drosophila," *Nature*, vol. 450, no. 7167, pp. 233–237, 2007.

[7] B. Reinius, P. Saetre, J. A. Leonard et al., "An evolutionarily conserved sexual signature in the primate brain," *PLoS Genetics*, vol. 4, no. 6, Article ID e1000100, 2008.

[8] T. Connallon and L. L. Knowles, "Intergenomic conflict revealed by patterns of sex-biased gene expression," *Trends in Genetics*, vol. 21, no. 9, pp. 495–499, 2005.

[9] J. E. Mank and H. Ellegren, "Are sex-biased genes more dispensable?" *Biology Letters*, vol. 5, no. 3, pp. 409–412, 2009.

[10] C. D. Meiklejohn, J. Parsch, J. M. Ranz, and D. L. Hartl, "Rapid evolution of male-biased gene expression in Drosophila," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 100, no. 17, pp. 9894–9899, 2003.

[11] M. Parisi, R. Nuttall, D. Naiman et al., "Paucity of genes on the Drosophila X chromosome showing male-biased expression," *Science*, vol. 299, no. 5607, pp. 697–700, 2003.

[12] P. P. Khil, N. A. Smirnova, P. J. Romanienko, and R. D. Camerini-Otero, "The mouse X chromosome is enriched for sex-biased genes not subject to selection by meiotic sex chromosome inactivation," *Nature Genetics*, vol. 36, no. 6, pp. 642–646, 2004.

[13] V. Reinke, I. S. Gil, S. Ward, and K. Kazmer, "Genome-wide germline-enriched and sex-biased expression profiles in Caenorhabditis elegans," *Development*, vol. 131, no. 2, pp. 311–323, 2004.

[14] Z. Zhang, T. M. Hambuch, and J. Parsch, "Molecular evolution of sex-biased genes in Drosophila," *Molecular Biology and Evolution*, vol. 21, no. 11, pp. 2130–2139, 2004.

[15] M. Pröschel, Z. Zhang, and J. Parsch, "Widespread adaptive evolution of Drosophila genes with sex-biased expression," *Genetics*, vol. 174, no. 2, pp. 893–900, 2006.

[16] J. E. Mank, L. Hultin-Rosenberg, E. Axelsson, and H. Ellegren, "Rapid evolution of female-biased, but not male-biased, genes expressed in the avian brain," *Molecular Biology and Evolution*, vol. 24, no. 12, pp. 2698–2706, 2007.

[17] D. Sturgill, Y. Zhang, M. Parisi, and B. Oliver, "Demasculinization of X chromosomes in the Drosophila genus," *Nature*, vol. 450, no. 7167, pp. 238–241, 2007.

[18] J. E. Mank and H. Ellegren, "Sex-linkage of sexually antagonistic genes is predicted by female, but not male, effects in birds," *Evolution*, vol. 63, no. 6, pp. 1464–1472, 2009.

[19] A. L. Ducrest, L. Keller, and A. Roulin, "Pleiotropy in the melanocortin system, coloration and behavioural syndromes," *Trends in Ecology & Evolution*, vol. 23, no. 9, pp. 502–510, 2008.

[20] D. Wright, S. Kerje, H. Brändström et al., "The genetic architecture of a female sexual ornament," *Evolution*, vol. 62, no. 1, pp. 86–98, 2008.

[21] J. F. Ayroles, M. A. Carbone, E. A. Stone et al., "Systems genetics of complex traits in Drosophila melanogaster," *Nature Genetics*, vol. 41, no. 3, pp. 299–307, 2009.

[22] M. J. Fitzpatrick, "Pleiotropy and the genomic location of sexually selected genes," *The American Naturalist*, vol. 163, no. 6, pp. 800–808, 2004.

[23] J. E. Mank, L. Hultin-Rosenberg, M. T. Webster, and H. Ellegren, "The unique genomic properties of sex-biased genes: insights from avian microarray data," *BMC Genomics*, vol. 9, article 148, 2008.

[24] J. E. Mank, K. Nam, B. Brunström, and H. Ellegren, "Ontogenetic complexity of sexual dimorphism and sex-specific selection," *Molecular Biology and Evolution*, vol. 27, no. 7, pp. 1570–1578, 2010.

[25] A. Alexeyenko and E. L. L. Sonnhammer, "Global networks of functional coupling in eukaryotes from comprehensive data integration," *Genome Research*, vol. 19, no. 6, pp. 1107–1116, 2009.

[26] A. Alexeyenko, T. Schmitt, A. Tjarnberg, D. Guala, and O. Frings, "Comparative interactomics with Funcoup 2.0," *Nucleic Acids Research*, vol. 40, no. 1, pp. D821–D828, 2012.

[27] G. Östlund, T. Schmitt, K. Forslund et al., "Inparanoid 7: new algorithms and tools for eukaryotic orthology analysis," *Nucleic Acids Research*, vol. 38, no. 1, Article ID gkp931, pp. D196–D203, 2009.

[28] J. E. Mank and H. Ellegren, "All dosage compensation is local: gene-by-gene regulation of sex-biased expression on the chicken Z chromosome," *Heredity*, vol. 102, no. 3, pp. 312–320, 2009.

[29] M. Demissie, B. Mascialino, S. Calza, and Y. Pawitan, "Unequal group variances in microarray data analyses," *Bioinformatics*, vol. 24, no. 9, pp. 1168–1174, 2008.

[30] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: a practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society. Series B*, vol. 57, no. 1, pp. 289–300, 1995.

[31] A. J. Enright, S. Van Dongen, and C. A. Ouzounis, "An efficient algorithm for large-scale detection of protein families," *Nucleic Acids Research*, vol. 30, no. 7, pp. 1575–1584, 2002.

[32] C. V. Mello, D. S. Vicario, and D. F. Clayton, "Song presentation induces gene expression in the songbird forebrain," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 89, no. 15, pp. 6818–6822, 1992.

[33] D. S. Manoli, M. Foss, A. Villella, B. J. Taylor, J. C. Hall, and B. S. Baker, "Male-specific fruitless specifies the neural substrates of Drosophila courtship behaviour," *Nature*, vol. 436, no. 7049, pp. 395–400, 2005.

[34] Y. Itoh, E. Melamed, X. Yang et al., "Dosage compensation is less effective in birds than in mammals," *Journal of Biology*, vol. 6, no. 1, article 2, 2007.

[35] M. Gallach and E. Betrán, "Intralocus sexual conflict resolved through gene duplication," *Trends in Ecology & Evolution*, vol. 26, no. 5, pp. 222–228, 2011.

[36] M. E. Peterson, F. Chen, J. G. Saven, D. S. Roos, P. C. Babbitt, and A. Sali, "Evolutionary constraints on structural similarity in orthologs and paralogs," *Protein Science*, vol. 18, no. 6, pp. 1306–1315, 2009.

[37] A. Henricson, K. Forslund, and E. L. L. Sonnhammer, "Orthology confers intron position conservation," *BMC Genomics*, vol. 11, no. 1, article 412, 2010.

[38] J. Huerta-Cepas, J. Dopazo, M. A. Huynen, and T. Gabaldón, "Evidence for short-time divergence and long-time conservation of tissue-specific expression after gene duplication," *Briefings in Bioinformatics*, vol. 12, no. 5, pp. 442–448, 2011.

[39] S. Movahedi, Y. van De Peer, and K. Vandepoele, "Comparative network analysis reveals that tissue specificity and gene function are important factors influencing the mode of expression evolution in arabidopsis and rice," *Plant Physiology*, vol. 156, no. 3, pp. 1316–1330, 2011.

[40] A. Molotkov and G. Duester, "Genetic evidence that retinaldehyde dehydrogenase Raldh1 (Aldh1a1) functions downstream of alcohol dehydrogenase Adh1 in metabolism of retinol to retinoic acid," *Journal of Biological Chemistry*, vol. 278, no. 38, pp. 36085–36090, 2003.

[41] J. E. Mank, L. Hultin-Rosenberg, M. Zwahlen, and H. Ellegren, "Pleiotropic constraint hampers the resolution of sexual antagonism in vertebrate gene expression," *The American Naturalist*, vol. 171, no. 1, pp. 35–43, 2008.

[42] R. P. Meisel, "Towards a more nuanced understanding of the relationship between sex-biased gene expression and rates of protein-coding sequence evolution," *Molecular Biology and Evolution*, vol. 28, no. 6, pp. 1893–1900, 2011.

[43] S. Fukamachi, M. Kinoshita, K. Aizawa, S. Oda, A. Meyer, and H. Mitani, "Dual control by a single gene of secondary sexual characters and mating preferences in medaka," *BMC Biology*, vol. 7, article 1741, p. 64, 2009.

[44] R. Lande and S. J. Arnold, "Evolution of mating preference and sexual dimorphism," *Journal of Theoretical Biology*, vol. 117, no. 4, pp. 651–664, 1985.

[45] M. Kirkpatrick and D. W. Hall, "Sexual selection and sex linkage," *Evolution*, vol. 58, no. 4, pp. 683–691, 2004.

[46] B. Vicoso and B. Charlesworth, "Evolution on the X chromosome: unusual patterns and processes," *Nature Reviews Genetics*, vol. 7, no. 8, pp. 645–653, 2006.

[47] D. Bachtrog, M. Kirkpatrick, J. E. Mank, S. F. McDaniel, J. C. Pires, and W. Rice, "Are all sex chromosomes created equal?" *Trends in Genetics*, vol. 27, no. 9, pp. 350–357, 2011.

[48] W. R. Rice, "Sex chromosomes and the evolution of sexual dimorphism," *Evolution*, vol. 38, no. 4, pp. 735–742, 1984.

[49] A. Y. K. Albert and S. P. Otto, "Evolution: sexual selection can resolve sex-linked sexual antagonism," *Science*, vol. 310, no. 5745, pp. 119–121, 2005.

[50] R. Storchová and P. Divina, "Nonrandom representation of sex-biased genes on chicken Z chromosome," *Journal of Molecular Evolution*, vol. 63, no. 5, pp. 676–681, 2006.

[51] V. B. Kaiser, M. Van Tuinen, and H. Ellegren, "Insertion events of CR1 retrotransposable elements elucidate the phylogenetic branching order in galliform birds," *Molecular Biology and Evolution*, vol. 24, no. 1, pp. 338–347, 2007.

[52] H. Ellegren, "Emergence of male-biased genes on the chicken Z-chromosome: sex-chromosome contrasts between male and female heterogametic systems," *Genome Research*, vol. 21, no. 12, pp. 2082–2086, 2011.

[53] S. Naurin, B. Hansson, D. Hasselquist, Y.-H. Kim, and S. Bensch, "The sex-biased brain: sexual dimorphism in gene expression in two species of songbirds," *BMC Genomics*, vol. 12, article 37, 2011.

[54] J. B. W. Wolf and J. Bryk, "General lack of global dosage compensation in ZZ/ZW systems? Broadening the perspective with RNA-seq," *BMC Genomics*, vol. 12, article 91, 2011.

The Scientific World Journal
Volume 2012, Article ID 753430, 6 pages
doi:10.1100/2012/753430

The ScientificWorldJOURNAL

Research Article

Gene Expression Network Reconstruction by LEP Method Using Microarray Data

Na You, Peng Mou, Ting Qiu, Qiang Kou, Huaijin Zhu, Yuexi Chen, and Xueqin Wang

School of Mathematics & Computational Science, Sun Yat-Sen University, Guangzhou, Guangdong 510275, China

Correspondence should be addressed to Xueqin Wang, wangxq88@mail.sysu.edu.cn

Received 28 September 2012; Accepted 25 November 2012

Academic Editors: R. Jiang, W. Tian, J. Wan, and X. Zhao

Gene expression network reconstruction using microarray data is widely studied aiming to investigate the behavior of a gene cluster simultaneously. Under the Gaussian assumption, the conditional dependence between genes in the network is fully described by the partial correlation coefficient matrix. Due to the high dimensionality and sparsity, we utilize the LEP method to estimate it in this paper. Compared to the existing methods, the LEP reaches the highest PPV with the sensitivity controlled at the satisfactory level. A set of gene expression data from the HapMap project is analyzed for illustration.

1. Introduction

Genes on the chromosomes behave interactively controlling the gene expression profiles of a cluster of genes, and their own expressions are in turn regulated by a bundle of genes. Exploring the gene expression regulatory network is essentially important to understand the progress of complex diseases, find the causal genes, and develop new drugs. In the past decades, the development of microarray technology allows us to measure the expression levels of tens of thousands of genes simultaneously, providing an opportunity to study the complex relationships among genes. In order to reconstruct the gene expression network, for any two particular genes, the conditional independence given all other genes needs to be investigated.

Because of the convenience of describing the interactions among variables, the graphical models become a common choice to study the relationships between variables, including but not limited to Boolean network [1], Bayesian network [2–4], autoregression model [5], and graphical Gaussian model [6]. However, the statistical inference on the independence is not easy. Under the Gaussian assumption, the independence is identical to being uncorrelated, and the conditional dependence between variables is able to be represented by the partial correlation coefficient matrix. When the number of observations n is equal or greater than the number of variables p, [7] mentioned two ways to estimate the partial correlation coefficient matrix in the graphical Gaussian model. If $n < p$, neither of these two ways is applicable due to the singular matrix.

As a typical high-dimensional data, there are usually not many available chips, while a great number of genes are included in the microarray data analysis. Fortunately, more and more studies [8–10] showed that the gene expression network is sparse, which means, for a particular gene, it only interacts with a few other genes. This fact implies that the majority entries of the partial correlation coefficient matrix are zero. To efficiently explore the sparsity and identify non-zero entries, the penalized linear regression is established where the sum of squared residuals (SSR) plus a penalty term is minimized, and has been widely used to estimate the sparse partial correlation coefficient matrix to reconstruct the gene expression network using microarray data [7, 11].

The most pioneering penalized linear regression method, the least absolute shrinkage and selection operator (LASSO) proposed by [12], utilizes the L_1 penalty to shrink the estimate which is close to zero from non-zero to zero, but it shrinks the estimates for parameters farther away from zero more severely, leading to a substantial bias. The authors in [13] indicated that LASSO may cause a bias even in a simple regression and suggested the smoothly clipped absolute deviation (SCAD) method, where a nonconcave penalty term

with desirable statistical properties, such as unbiasedness, sparsity, and continuity, was introduced. However, the SCAD penalty is not smooth, resulting in the optimization problem being complicated. Upon this, [14] proposed the Laplace error penalty (LEP) method with a penalty which is unbiased, sparse, continuous, and almost smooth.

In this paper, we will apply the LEP method to reconstruct the gene expression network, and compare it to LASSO and SCAD in the performance of estimating the partial correlation coefficient matrix. The paper is structured as follows. In Section 2, the LASSO, SCAD, and LEP methods will be briefly described. In Section 3, we will report the results of simulations and a real data analysis. A short discussion is given in Section 4.

2. Methods

The graphical Gaussian model, or GGM for abbreviation, is an undirected graphical model. Let $\mathbf{X} = (X_1, \ldots, X_p)'$ indicate a p-dimensional random variable, subject to the multivariate normal distribution $N(\boldsymbol{\mu}, \Sigma)$, where $\boldsymbol{\mu}$ is the mean vector and Σ is the variance-covariance matrix. Given n samples from $N(\boldsymbol{\mu}, \Sigma)$, $(x_{ij})_{p\times n}$, the partial correlation coefficient matrix $(\rho_{ij})_{p\times p}$, which reflects the conditional dependence between different components of $\mathbf{X}$, could be estimated by $\hat{\rho}_{ij} = \text{sign}(\hat{\beta}_{ij})\sqrt{\hat{\beta}_{ij}\hat{\beta}_{ji}}$, where $\hat{\beta}_{ij}$ is the estimator for β_{ij} in the linear regression model

$$X_{ij} = \sum_{1\le k\ne i\le p} \beta_{kj} X_{kj} + \epsilon_{ij}, \quad i = 1, 2, \ldots, p;\ j = 1, 2, \ldots, n, \tag{1}$$

ϵ_{ij}, $i = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, n$, are independent and identically distributed, and independent of $\mathbf{X}$, and $\text{sign}(x)$ is an indicator function, being -1, 0, or 1 when x is smaller, equal, or greater than 0, respectively. For the "small N large P" problem, instead of the classical least square optimization, the objective function

$$\sum_{i=1}^{p}\sum_{j=1}^{n}\left(X_{ij} - \sum_{k\ne i}\beta_{kj}X_{kj}\right)^2 + \sum_{i=1}^{p}\sum_{1\le j\ne i\le N} p_\lambda\left(\beta_{ij}\right) \tag{2}$$

is minimized to get the estimator for β_{ij}, $\hat{\beta}_{ij}$, where $p_\lambda(\cdot)$ indicates a penalty function on the parameters. The formula $p_\lambda(\cdot)$ is essentially important. It not only determines the way to shrink the estimators, but also directly affects the complexity of the optimization algorithm. A good penalty function should have several desirable statistical properties, unbiasedness, sparsity, continuity [13], and smoothness [14].

The LASSO, proposed by [12], has the penalty $p_\lambda(\beta) = \lambda|\beta|$. Although it succeeded in many applications of variable selection, it shrinks the estimates of larger parameters more significantly than that of the smaller parameters, causing a substantial bias. The SCAD penalty function, suggested by [13], has the derivative $p'_\lambda(\beta) = \lambda\{I(|\beta| \le \lambda) + (\lambda\alpha - |\beta|)_+ I(|\beta| > \lambda)/(\lambda(\alpha - 1))\}$. Beside the sparsity and continuity, it gains the unbiasedness but loses the smoothness. The SCAD penalty is made of piecewise quadratic splines, making the optimization of (2) complicated. To overcome this problem, Wen et al. [14] proposed the LEP with penalty term

$$p_\lambda(\beta) = \lambda\left(1 - \exp\left(-\frac{|\beta|}{\kappa}\right)\right), \tag{3}$$

where λ and $\kappa > 0$ are two tuning parameters.

The LEP penalty not only satisfies the unbiasedness, sparsity, and continuity, but also is an almost smooth function. It emphasizes the smoothness and complexity, since the smooth function is more stable, and the complexity of optimization algorithm highly depends on the complexity of $p_\lambda(\cdot)$, which determines whether the proposed method could be widely applied, especially in the high-dimensional data situations. In order to solve the optimization problem, [14] extended the block coordinate gradient descent (BCGD) algorithm [15] and provided a faster computing algorithm, as will be shown in the simulation studies. For the details of the LEP method and the optimization algorithm, please refer to [14].

3. Results

3.1. Simulations. Suppose there are n microarray chips and p genes, then $n \times p$ equations with $p \times (p - 1)$ parameters are involved in (1). When p is fixed, increasing/decreasing n would increase/decrease the number of equations but the number of parameters would remain the same. In this case, the penalized linear regression, including LEP, LASSO and SCAD, performed as expected that is, their estimates became more or less accurate as n became larger or smaller (results not shown here). Therefore, in the following simulations, we fixed $n = 120$ and only varied $p = 10$ or 20.

In order to fully evaluate the performances of LEP, LASSO, and SCAD in different situations, four scenarios were set up. In each scenario, a covariance matrix Σ of size $p \times p$ was generated, and n random vectors of dimension p were sampled from the multivariate normal distribution $N(0, \Sigma)$ independently. The partial correlation coefficient matrix was then estimated from the sampled data. We fixed Σ and made 100 repetitions in each scenario to get the average of the estimates for fair comparison. In the first two scenarios $p = 10$, and $p = 20$ in scenario 3 and 4. Two data generating procedures used in [11] were employed to generate the covariance matrix Σ. In scenario 1 and 3, the (i, j)-element of Σ, $\sigma_{ij} = \exp(-a|s_i - s_j|)$, where $a = 2$ and $s_1 < s_2 < \cdots < s_p$ were generated by setting $s_i - s_{i-1}$, following a uniform distribution $U(0.5, 1)$. In scenario 2 and 4, a sparse precision matrix Ω was generated as proposed in [16], and $\Sigma = \Omega^{-1}$.

The partial correlation coefficient matrix was estimated by LEP, LASSO or SCAD, respectively, in each scenario. To evaluate the performances of different methods, the sensitivity which is the fraction of "true non-zero and also estimated non-zero parameters" to "true non-zero parameters" and PPV which is the fraction of "true non-zero and also estimated non-zero parameters" to "estimated non-zero parameters" were calculated. Furthermore,

Table 1: The sensitivity, PPV, and F_1 values in four scenarios of the simulation studies.

	PPV	Sensitivity	F_1	SSE	Time (s)
Scenario 1: $P = 10$					
LEP	0.934	0.708	0.805	1075.313	0.121
LASSO	0.602	0.908	0.724	1053.937	2.184
SCAD	0.891	0.727	0.801	1070.202	0.594
Scenario 2: $P = 10$					
LEP	0.926	0.826	0.873	1185.403	0.259
LASSO	0.833	0.916	0.873	1215.098	2.507
SCAD	0.932	0.828	0.877	1191.352	0.933
Scenario 3: $P = 20$					
LEP	0.778	0.707	0.741	2112.376	2.431
LASSO	0.467	0.868	0.607	2100.014	16.976
SCAD	0.693	0.741	0.716	2089.248	6.247
Scenario 4: $P = 20$					
LEP	0.831	0.834	0.832	2351.997	2.763
LASSO	0.667	0.910	0.770	2380.255	23.792
SCAD	0.735	0.852	0.789	2298.897	7.193

the F_1 score $= 2 \cdot \text{sensitivity} \cdot \text{PPV}/(\text{sensitivity} + \text{PPV})$ was also presented. The results in four scenarios were listed in Table 1.

As the number of genes p increases, the number of parameters to be estimated increases rapidly. Due to the sparsity of partial correlation coefficient matrix, the number of true zero parameters increases much more than that of true non-zero parameters, causing the chance of estimating a zero parameter to be non-zero increases more than that of estimating a non-zero parameter to be zero. As presented in Table 1, although the sensitivity of LEP did not change significantly as p increasing from 10 to 20, its PPV reduced obviously from ∼90% in scenario 1 and 2 to ∼80% in scenario 3 and 4. The LASSO and SCAD showed similarly. Note that beside the penalty term, the performances of different methods also depend on the true value of covariance matrix Σ, which was generated at the beginning of each scenario.

Across all the scenarios, although LASSO reached the highest sensitivity, its PPV was far lower than that of SCAD and LEP, which means that LASSO could identify more gene regulatory relationships, but there might be many false positives. Among these three methods, LEP achieved the highest PPV with its sensitivity controlled at similar level to that of SCAD. Its F_1 score also reached the highest value in scenario 1, 3, and 4. More importantly, using the algorithm proposed by [14], LEP was the fastest, whose computation time was almost 1/18, 1/10, 1/7 and 1/9 of LASSO and 1/5, 1/4, 1/3, and 1/3 of SCAD in four scenarios, respectively.

For intuitive illustration, we also plotted the relative frequency matrix for each method in each scenario, where the (i, j)-element indicates the relative frequency of non-zero estimates among 100 repetitions. The darker the color is, the higher the frequency of non-zero estimates is. The true partial correlation coefficient matrix was shown in the first panel of each row in Figure 1. From Figure 1, we can see that the color of LASSO is significantly darker than others, especially the truth, which means that LASSO estimated many true zero parameters to be non-zero, resulting in

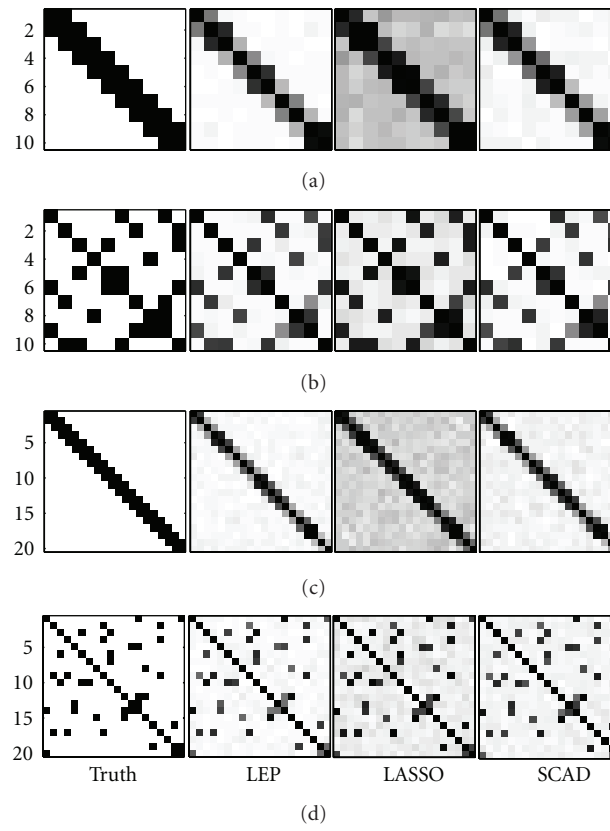


Figure 1: The relative frequency matrices in four scenarios of the simulation studies. The first, second, third and forth rows correspond to scenario 1, 2, 3 and 4, respectively.

Table 2: The number of edges connected with each gene in GSE6536 data example.

No.	Gene	LEP	LASSO	SCAD	Gene functions
1	ABCF1	0	9	1	ATP-binding cassette
2	EIF3D	0	8	1	Eukaryotic translation initiation factor 3
3	SRP14	0	8	1	Signal recognition particle 14 kDa (homologous Alu RNA-binding protein)
4	RPL28	0	8	0	Ribosomal protein L28
5	EIF3F	0	8	0	Eukaryotic translation initiation factor 3
6	CYP2A6	3	7	3	Cytochrome P450
7	RPL35	0	7	2	Ribosomal protein L35
8	GDI2	0	7	1	GDP dissociation inhibitor 2
9	RPL11	0	7	1	Ribosomal protein L11
10	GAS6	3	6	3	Growth arrest-specific 6
11	DAD1	1	6	2	Defender against cell death 1
12	RPL21	0	6	1	Ribosomal protein L21
13	EPHB3	3	5	3	EPH receptor B3
14	MMP14	2	5	4	Matrix metallopeptidase 14 (membrane inserted)
15	ESRRA	2	5	3	Estrogen-related receptor alpha
16	PRPF8*	2	5	0	PRP8 pre-mRNA processing factor 8 homolog (*S. cerevisiae*)
17	HSPA6	1	5	2	Heat shock 70 kDa protein 6 (HSP70B′)
18	PARK7	1	5	2	Parkinson protein 7
19	TARDBP	0	5	4	TAR DNA-binding protein
20	SEPT2	0	5	0	Septin 2
21	DDR1*	3	4	0	Discoidin domain receptor tyrosine kinase 1
22	TRADD	1	4	2	TNFRSF1A-associated via death domain
23	EIF4G2	0	4	1	Eukaryotic translation initiation factor 4 gamma
24	CAPNS1	0	4	0	Calpain, small subunit
25	PLD1	1	3	2	Phospholipase D1
26	UBA7*	2	2	0	Ubiquitin-like modifier activating enzyme 7
27	CYP2E1	0	2	1	Cytochrome P450
28	FNTB	1	1	1	Farnesyltransferase
29	GUCA1A	0	1	1	Guanylate cyclase activator 1A (retina)
30	CCL5	0	0	0	Chemokine (C-C motif) ligand 5

*Indicates LEP exclusive genes.

many false positives. Comparing to LASSO, the SCAD plot became much closer to the truth and LEP made a further improvement upon the SCAD plot.

3.2. A Real Data Example. In this section, the publicly available gene expression dataset GSE6536 (http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE6536) was investigated. There are gene expression values of 47,294 human transcripts from 270 HapMap individual samples [17], including 30 Caucasian trios of northern and western European background (CEU), 30 Yoruba trios from Ibadan, Nigeria (YRI), 45 unrelated individuals from Beijing, China (CHB), and 45 unrelated individuals from Tokyo, Japan (JPT). After the microarray data were log2-transformed and background corrected, within and across the population normalized, the gene expression values were saved in a matrix for further analysis, which also could be downloaded from the website mentioned above.

Frommlet et al. [18] listed 44 genes which are significantly differentially expressed across individual samples. Since the platform Sentrix Human-6 Expression BeadChip used in this experiment was publicly available in 2005, so far the gene annotation database has been updated a lot. Of those 44 genes, 14 either were found to be pseudogenes or have been removed as a result of standard genome annotation processing and therefore were excluded from our following analysis. The partial correlation coefficient matrix of the rest 30 genes were estimated using 270 samples data. In the graphical model, one non-zero estimate off diagonal in the partial correlation coefficient matrix corresponds to one edge connecting different genes on the graph. The number of edges connected with each gene was listed in Table 2.

As shown in Table 2, LASSO identified 76 edges between 30 genes, SCAD found 21 and LEP reported 13. The LASSO recognized that almost all of the genes interacted with others and identified much more edges than SCAD or LEP. To compare the performances of different methods, we only

focus on the important genes which carry the most or secondly most number of edges. For LASSO, there is 1 such important gene with 9 edges and 4 with 8 edges each. The SCAD found 2 important genes with 4 edges each and 4 with 3 edges each. The LEP identified 4 with 3 edges each and 4 with 2 edges each.

For the important genes recognized by LASSO, none of them were taken to be important by SCAD or LEP. According to the gene functions described in Table 2, although these genes have very important functions, they usually accomplish these functions together with many others genes, and once they could not be normally expressed, these functions could be completed by other genes, then this would not significantly affect the gene expression in the network. On the contrary, the genes identified by SCAD or LEP usually have unique gene function, which could not be recovered by other genes once they are expressed abnormally, resulting in the irregular expression in the network.

Beside those 5 common genes identified by both of LEP and SCAD, LEP found 3 more exclusive genes and SCAD found 1 more exclusive gene. Those 3 LEP exclusive genes not only play a key role in the cellular mechanism, but also have very close relationships with other genes. Among them, PRPF8 (gene 16) is a component of both U2- and U12-dependent spliceosomes, which removes the vast majority of introns (more than 99%) in mammals [19, 20]. DDR1 (gene 21), one of the receptor tyrosine kinases, is important in the communication between the cells and their microenvironments and gets involved in many cellular activities, like growth, differentiation, and metabolism [21]. UBA7 (gene 26), widely expressed in a variety of cell types, belongs to the ubiquitin conjugation pathway, which is of fundamental and central importance [22]. However, the SCAD exclusive gene, TARDBP (gene 19), although plays an important role in modulating HIV-1 gene expression; it only represses transcription from the HIV-1 long terminal repeat, no other transcription from other promoters [23]. Due to this fact, it should not interact heavily with other genes, as the LEP concluded.

4. Discussion

In this paper, we applied the LEP method to estimate the partial correlation coefficient matrix to reconstruct the gene expression network. Comparing to the existing methods, for example, LASSO and SCAD, LEP reached the highest PPV, and its sensitivity was controlled at the similar level as SCAD. As seen from the relative frequency matrix plot in the simulation studies, LEP showed the superiority in exploring the sparsity of the partial correlation coefficient matrix.

There are two tuning parameters in the LEP penalty function. We used the EBIC criteria [24] to select the approximate values for parameters. But as seen from the simulation results (not shown here), any combination of κ and λ which satisfy some certain function relation would return very close estimation results. Therefore, we only need to vary one of κ and λ and keep the other a constant for parameter choosing. As in the real data analysis, we set $\kappa = 0.01$ and vary λ.

Acknowledgments

This work is partially supported by the Major State Basic Research Development Program (2012CB517900), NSFC (11001280), NDFGP (10151027501000066), RFDP (20090171110017), and the Fundamental Research Funds for the Central Universities.

References

[1] P. D'Haeseleer, S. Liang, and R. Somogyi, "Genetic network inference: from co-expression clustering to reverse engineering," *Bioinformatics*, vol. 16, no. 8, pp. 707–726, 2000.

[2] N. Friedman, M. Linial, I. Nachman, and D. Pe'er, "Using bayesian networks to analyze expression data," *Journal of Computational Biology*, vol. 7, no. 3-4, pp. 601–620, 2000.

[3] E. Segal, M. Shapira, A. Regev et al., "Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data," *Nature Genetics*, vol. 34, no. 2, pp. 166–176, 2003.

[4] N. Friedman, "Inferring cellular networks using probabilistic graphical models," *Science*, vol. 303, no. 5659, pp. 799–805, 2004.

[5] R. Opgen-Rhein and K. Strimmer, "Learning causal networks from systems biology time course data: an effective model selection procedure for the vector autoregressive process," *BMC Bioinformatics*, vol. 8, supplement 2, article S3, 2007.

[6] J. Schäfer and K. Strimmer, "An empirical Bayes approach to inferring large-scale gene association networks," *Bioinformatics*, vol. 21, no. 6, pp. 754–764, 2005.

[7] N. Krämer, J. Schäfer, and A. L. Boulesteix, "Regularized estimation of large-scale gene association networks using graphical Gaussian models," *BMC Bioinformatics*, vol. 10, no. 1, article 384, 2009.

[8] T. I. Lee, N. J. Rinaldi, F. Robert et al., "Transcriptional regulatory networks in Saccharomyces cerevisiae," *Science*, vol. 298, no. 5594, pp. 799–804, 2002.

[9] S. Ma, Q. Gong, and H. J. Bohnert, "An Arabidopsis gene network based on the graphical Gaussian model," *Genome Research*, vol. 17, no. 11, pp. 1614–1625, 2007.

[10] R. D. Leclerc, "Survival of the sparsest: robust gene networks are parsimonious," *Molecular Systems Biology*, vol. 4, article 213, 2008.

[11] J. Fan, Y. Feng, and Y. Wu, "Network exploration via the adaptive lasso and scad penalties," *The Annals of Applied Statistics*, vol. 3, no. 2, pp. 521–541, 2009.

[12] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society B*, vol. 85, no. 1, pp. 267–288, 1996.

[13] J. Fan and R. Li, "Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties," *Journal of the American Statistical Association*, vol. 96, no. 456, pp. 1348–1360, 2001.

[14] C. Wen, S. Wang, and X. Wang, "Ultra-high dimensional variable selection based on global minimizers of penalized least squares," Tech. Rep., Sun Yat-Sen University, 2012.

[15] P. Tseng and S. Yun, "A coordinate gradient descent method for nonsmooth separable minimization," *Mathematical Programming*, vol. 117, no. 1-2, pp. 387–423, 2009.

[16] H. Li and J. Gui, "Gradient directed regularization for sparse Gaussian concentration graphs, with applications to inference of genetic networks," *Biostatistics*, vol. 7, no. 2, pp. 302–317, 2006.

[17] The International HapMap Consortium, "A haplotype map of the human genome," *Nature*, vol. 437, no. 7063, pp. 1299–1320, 2005.

[18] F. Frommlet, F. Ruhaltinger, P. Twarog, and M. Bogdan, "A model selection approach to genome wide association studies," 2010, http://arxiv.org/abs/1010.0124.

[19] H. R. Luo, G. A. Moreau, N. Levin, and M. J. Moore, "The human Prp8 protein is a component of both U2- and U12-dependent spliceosomes," *RNA*, vol. 5, no. 7, pp. 893–908, 1999.

[20] P. A. Sharp and C. B. Burge, "Classification of introns: U2-type or U12-type," *Cell*, vol. 91, no. 7, pp. 875–879, 1997.

[21] A. N. Shelling, R. Butler, T. Jones, S. H. Laval, J. M. Boyle, and T. S. Ganesan, "Localization of an epithelial-specific receptor kinase (EDDR1) to chromosome 6q16," *Genomics*, vol. 25, no. 2, pp. 584–587, 1995.

[22] K. Kok, R. Hofstra, A. Pilz et al., "A gene in the chromosomal region 3p21 with greatly reduced expression in lung cancer is similar to the gene for ubiquitin-activating enzyme," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 90, no. 13, pp. 6071–6075, 1993.

[23] S. H. Ignatius Ou, W. U. Foon, D. Harrich, L. F. García-Martínez, and R. B. Gaynor, "Cloning and characterization of a novel cellular protein, tdp-43, that binds to human immunodeficiency virus type 1 tar dna sequence motifs," *Journal of Virology*, vol. 69, no. 6, pp. 3584–3596, 1995.

[24] J. Chen and Z. Chen, "Extended Bayesian information criteria for model selection with large model spaces," *Biometrika*, vol. 95, no. 3, pp. 759–771, 2008.

The Scientific World Journal
Volume 2012, Article ID 541786, 11 pages
doi:10.1100/2012/541786

The ScientificWorldJOURNAL

Review Article

Molecular Mechanisms and Function Prediction of Long Noncoding RNA

Handong Ma, Yun Hao, Xinran Dong, Qingtian Gong, Jingqi Chen, Jifeng Zhang, and Weidong Tian

Institute of Biostatistics, School of Life Science, Fudan University, 220 Handan Road, Shanghai 2004333, China

Correspondence should be addressed to Weidong Tian, weidong.tian@fudan.edu.cn

Received 30 October 2012; Accepted 21 November 2012

Academic Editors: G. P. Chrousos and T. Kino

The central dogma of gene expression considers RNA as the carrier of genetic information from DNA to protein. However, it has become more and more clear that RNA plays more important roles than simply being the information carrier. Recently, whole genome transcriptomic analyses have identified large numbers of dynamically expressed long noncoding RNAs (lncRNAs), many of which are involved in a variety of biological functions. Even so, the functions and molecular mechanisms of most lncRNAs still remain elusive. Therefore, it is necessary to develop computational methods to predict the function of lncRNAs in order to accelerate the study of lncRNAs. Here, we review the recent progress in the identification of lncRNAs, the molecular functions and mechanisms of lncRNAs, and the computational methods for predicting the function of lncRNAs.

1. Introduction

Proteins and related protein-coding genes have been the main subject of biological studies for years. However, with the development of RNA sequencing technology and computational methods for assembling the transcriptome, it has become clear that besides protein-coding genes much of the mammalian genome is transcribed, and many noncoding RNA (ncRNA) transcripts tend to play important roles in a variety of biological processes. Understanding the function of ncRNAs has become one of the most important goals of modern biological studies [1–3]. ncRNAs can be classified into several distinct subclasses, including processed small RNAs [4], promoter-associated RNAs [5], and functional long noncoding RNAs (lncRNAs) [6]. The term of lncRNA was introduced to distinguish the special class of ncRNA from well-known small regulatory RNAs (i.e. miRNAs and siRNAs). lncRNAs are generally longer than 200 nucleotides [3, 7, 8]. Recent studies have shown that lncRNAs may act as important *cis*- or *trans*-regulators in various biological processes. Mutations in lncRNAs are related with a wide range of diseases, especially cancers and neurodegenerative diseases. Even so, the functions and molecular mechanisms of most lncRNAs are unknown. Though several computational methods have been developed to predict the functions of lncRNAs, it still remains a challenging task, partly owing to the lack of conservation in both the sequence and secondary structures of lncRNAs [9–11]. In this paper, we will summarize the recent progresses and challenges in the identification, molecular mechanism, and function prediction of lncRNAs.

2. Definition and Classification of lncRNA

The definition of lncRNA is based on two criteria, the size and the lack of protein-coding potential. In this paper, lncRNA refers to nonprotein-coding RNA longer than 200 nt [7, 10–12], which distinguishes it them from mRNA and small regulatory RNA in a relatively satisfying way [11, 13]. Depending on their relationships with the nearest protein-coding genes, lncRNAs can be classified in three different ways [12, 14, 15]: (1) *sense or antisense:* lncRNAs that are located on the same strand or the opposite strand of the nearest protein-coding genes [16]; (2) *divergent or convergent:* lncRNAs that are transcribed in the divergent or convergent orientation compared to that of the nearest protein-coding genes [12]; (3) *intronic or intergenic:* lncRNAs that locate

inside the introns of a protein-coding gene, or in the interval regions between two protein-coding genes [12, 17].

3. Identification of lncRNA

To identify lncRNAs, the first step is to obtain all transcripts including ncRNAs and mRNAs in cells, and then to distinguish lncRNAs from mRNAs and other types of ncRNAs. Traditional technologies, such as microarray, focus on the identification of protein-coding RNA transcripts. New technologies, such as RNA-Seq, are not limited to the identification of protein-coding RNA transcripts, and have led to the discovery of many novel ncRNA transcripts. The discrimination between lncRNAs and other small regulatory ncRNAs depends on their length. However, the length information alone is not enough to separate lncRNAs from mRNAs, and other criteria are needed for this purpose. Below, we will first briefly introduce new technologies in identifying RNA transcripts, especially ncRNA transcripts. Then, we will review current methods to distinguish lncRNAs from mRNAs.

3.1. Experimental Methods in Identifying lncRNA

Microarray. Traditional microarray technologies use predefined probes to determine the expression level of mRNA transcripts and are not appropriate to identify lncRNAs. However, it has been found that a few previously defined mRNAs or some probe sequences actually are lncRNAs; thus, former microarray datasets can be reannotated to study the expression of lncRNAs [60]. With more and more lncRNAs discovered, new probes specific for lncRNAs can be designed. For example, Babak et al. designed probes from conserved intergenic and intragenic region to identify potential ncRNA transcripts [61]. However, microarray is not sensitive enough to detect RNA transcripts with low-expression level. Thus the use of microarray to identify lncRNAs is limited due to the low expression level of many lncRNAs.

SAGE and EST. SAGE (serial analysis of gene expression) technology produces large numbers of short sequence tags and is capable of identifying both known and unknown transcripts. SAGE has been used and proved to be an efficient approach in studying lncRNAs. For example, Gibb et al. compiled 272 human SAGE libraries. By passing over 24 million tags they were able to generate lncRNA expression profiles in human normal and cancer tissues [62]. Lee et al. also used SAGE to identify potential lncRNA candidates in male germ cell [63]. However, SAGE is much more expensive than microarray, therefore is not widely employed in large-scale studies. EST (expressed sequence tag) is a short subsequence of cDNA, and is generated from one-shot sequencing of cDNA clone. The public database now contains over 72.6 million EST (GeneBank 2011), making it possible to discover novel transcripts. For example, Furuno et al. clustered EST to find functional and novel lncRNAs in mammalian [64]. Huang et al. used the public bovine-specific EST database to reconstruct transcript assemblies, and find transcripts in intergenic regions that are likely putative lncRNAs [65].

RNA-Seq. With the development of next generation sequencing (NGS) technologies, RNA-Seq (also named whole transcriptome shotgun sequencing) has been widely used for novel transcripts discovery and gene expression analysis. Compared to traditional microarray technology, RNA-Seq has many advantages in studying gene expression. It is more sensitive in detecting less-abundant transcripts, and identifying novel alternative splicing isoforms and novel ncRNA transcripts. The basic workflow for lncRNA identification using RNA-Seq is shown in Figure 1. RNA-Seq is currently the most widely used technology in identifying lncRNAs. For example, Li et al. applied RNA-Seq to identify lncRNAs during chicken muscle development [66]. Nam and Bartel integrated RNA-Seq, poly (A)-site, and ribosome mapping information to obtain lncRNAs in *C. elegans* [16]. Pauli et al. performed RNA-Seq experiments at eight stages during zebrafish early development, and identified 1133 noncoding multiexonic transcripts [67]. Prensner et al. used RNA-Seq to study lncRNA in human prostate cancer from 102 prostate tissues and cell lines, and concluded that lncRNAs may be used for cancer subtype classification [68].

RNA-IP. RNA-IP (RNA-immunoprecipitation) is a new method developed to identify lncRNA that interacts with specific protein. Antibodies of the protein are first used to isolate lncRNA-protein complexes. Then, cDNA library is constructed followed by deep sequencing of interacting lncRNAs. Using RNA-IP, Zhao et al. discovered a 1.6-kb lncRNA within Xist that interacts with PRC2 [69].

Chromatin Signature-Based Approach. The above-mentioned methods target on RNA transcripts directly. In contrast, chromatin signature-based approach uses chromatin signatures, such as H3K4me3 (the marker of active promoters) and H3K36me3 (the marker of transcribed region), to study actively transcribed genes including lncRNAs. In this approach, ChIP-Seq is used to generate genome-wide profiles of chromatin signatures [70], and the transcribed regions are mapped in the genome, where lncRNAs are determined and studied. For example, Guttman et al. identified 1,600 large multiexonic lncRNAs that are regulated by key transcription factors such as p53 and NFkB [71]. The advantage of this approach is its directness in investigating the mechanisms that regulate lncRNA expression.

3.2. Computational Methods in Identifying lncRNA

ORF Length Strategy. Unlike protein-coding genes, the start codons and termination codons in lncRNAs tend to distribute randomly. As a result, the ORF length of lncRNAs can hardly extend to over 100 from a probabilistic point of view. Based on this principle, one way to discriminate lncRNAs from mRNAs is by ORF length. For example, the FANTOM project used a maximum ORF length cutoff of 100 codons to differentiate noncoding RNAs from mRNAs [72]. However, some lncRNAs are known to have ORFs longer than 100 codons, while some protein coding genes have fewer than 100 amino acids, such as RCI2A gene in *Arabidopsis* which encodes a protein of 54 amino acids

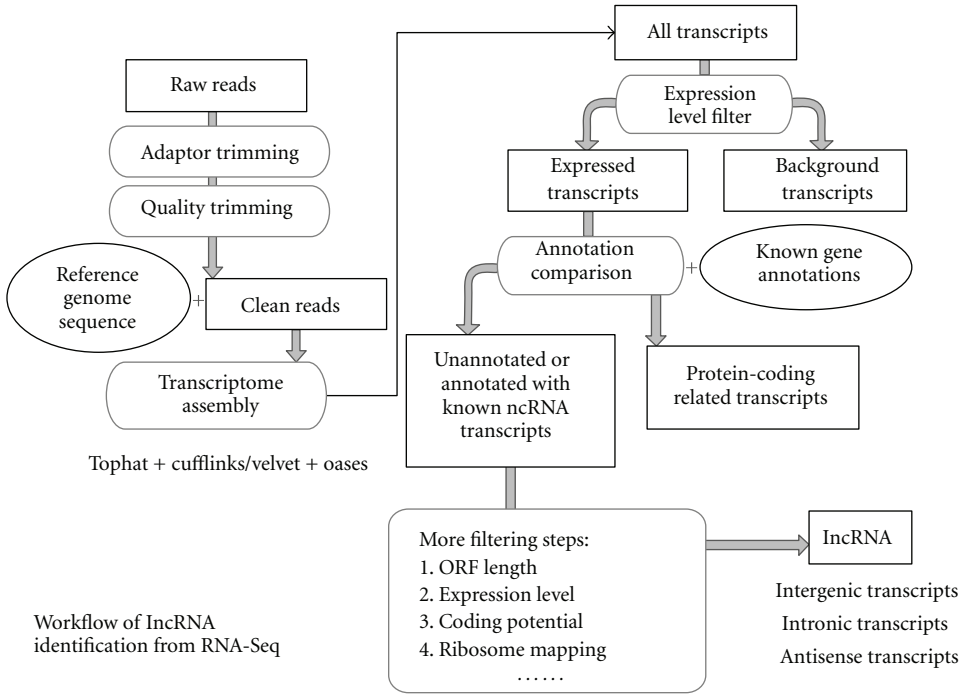


Figure 1: Workflow of lncRNA identification from RNA-Seq.

[73]. Thus, this approach may cause misclassification. To overcome the drawbacks of methods based on ORF length, Jia et al. utilize a comparative genomics method to refine ncRNA candidates. They defined the RNA sequences as ncRNAs only if the cDNAs have no homologous proteins longer than 30 amino acids across the mammalian genomes [7]. However, this method relies largely on the completeness of the databases. Therefore, deficiency in protein coding annotation may cause misclassification of lncRNAs as well.

Sequence and Secondary Structure Conservation Strategy. Compared to protein coding genes, noncoding genes are generally less conservative, meaning they are more inclined to mutate [21, 67]. Thus, measuring the coding potential is considered a way of identifying lncRNAs. Codon Substitution Frequency (CSF) is one of the criteria. For example, Guttman et al. used the maximum CSF score to assess the coding potential of a RNA sequence [71]. Clamp et al. and Lin et al. further combined CSF with reading frame conservation (RFC) to discriminate lncRNAs from mRNAs [74, 75]. Other similar methods include PhyloCSF use a phylogenetic framework to build two phylogenetic codon models that can distinguish coding from noncoding regions [76]. RNAcode combines amino acid substitution with gap patterns to assess the coding potential [77]. There are also methods that explore the conservation of RNA secondary structures to identify lncRNAs, including programs QRNA [78], RNAz [79], and EvoFOLD [80]. However, this approach is limited by lack of common conserved secondary structures specific for lncRNAs.

Machine Learning Strategies. Owing to the complex identities of lncRNAs, recently an increasing number of machine learning-based methods have been developed to integrate various sources of data to distinguish lncRNAs from mRNAs. Table 1 summarizes the machine learning methods and the features used to train the model for identifying lncRNAs. For instance, CONC utilizes a series of protein features such as amino acid composition, secondary structure, and peptide length, to train a SVM model that distinguishes lncRNAs from mRNAs [18]. CPC (Coding Potential Calculator) also uses SVM for modeling and extracting sequence features and the comparative genomics features to assess the coding potential of transcripts [19, 20]. Lu et al. developed a machine learning method that integrates GC content, DNA conservation, and expression information to predict lncRNAs in *C. elegans* [21].

Although the above-described methods have shown their effectiveness in identifying lncRNAs, exceptional cases still remain. For instance, whether an RNA transcript is translated or not may be changeable during the course of evolution. As an example, *Xist*, a well-known lncRNA, evolves from a protein-coding gene [81]. Besides, some genes are bifunctional, and both the coding and noncoding isoforms exist. The steroid receptor RNA activator (SRA) was characterized as a noncoding RNA previously but the coding product was detected later [82]. Such ambiguity will be clarified when more about lncRNAs are known.

4. lncRNA Function

lncRNAs have once been thought as the "dark matter" of the genome, because of our limited knowledge about their functions [83]. With more studies about lncRNAs conducted, it has become clear that lncRNAs have many specific functional features, and are likely to be involved in many diverse

Table 1: Machine-learning methods for identifying lncRNAs.

Method	Features	Algorithm	References
CONC	Peptide length Amino acid composition Hydrophobicity Secondary structure content Percentage of residues exposed to solvent Sequence compositional entropy Number of homologs obtained by PSI-BLAST Alignment entropy	SVM	[18]
CPC	ORF prediction quality Number of homologs obtained by BLASTX Alignment quality Segment distribution	SVM	[19, 20]
Lu et al.	RNA-seq experiments Tilling arrays poly-A + RNA-seq experiments poly-A + tilling arrays GC content DNA conservation Predicted protein sequence conservation Predicted secondary structure free energy Predicted secondary structure conservation	Naïve Bayes Bayes Net Decision Tree Random Forest Logistic Regression SVM	[21]

biological processes in cells. Rather than "dark matter," they may act as necessary functional parts in the genome. These functional features include but are not limited to (i) lncRNAs have conserved splice junctions and introns [84]; (ii) the expression patterns of lncRNAs are tissue- and cell-specific [12, 67]; (iii) the altered expression of lncRNAs can be found in neurodegeneration, cancer, and other diseases [9, 10]; (iv) lncRNAs are associated with particular chromatin signatures that are indicative of actively transcribed genes [11, 85]. Below, we will briefly summarize the cellular functions of lncRNAs and molecular mechanisms of their functions.

4.1. Cellular Functions of lncRNA. With thousands of lncRNAs identified in mammals and other vertebrates [16], a few lncRNAs have been extensively studied, which have shed light on their possible functions. Firstly, lncRNAs are involved in various epigenetic regulations through recruitment of chromatin remodeling complexes to specific genomic loci, such as Xist, Air, and Kcnq1ot1 [22, 43]. Secondly, lncRNAs can regulate gene expression by interacting with protein partners in biological processes like protein synthesis, imprinting (Kcnq1ot1, Air), cell cycle control (TERRA), alternative splicing (MALAT1), and chromatin structure regulation (DNMT3b, PANDA) [9, 10, 38, 71, 85–89]. Thirdly, lncRNAs are involved in enhancer-regulating gene activation (eRNAs), in which cases they may interact directly with distal genomic regions [90]. Fourthly, some lncRNAs serve as interacting partners or precursors for short regulatory ncRNAs [91]. For example, microRNAs (miRNAs) can be generated through sequential cleavage of lncRNAs, while Piwi-interacting RNAs (piRNAs) can be produced by processing a single lncRNA transcript [88].

Recent studies have shown the expression of lncRNA is tissue specific. Loewer et al. studied the expression of lncRNA in global remodeling of the epigenome and during reprogramming of somatic cells to induce pluripotent stem cells (iPSCs). They found some lncRNAs have cell-type specific expression pattern [26, 92]. Loss-of-function studies on most intergenic lncRNAs expressed in mouse embryonic stem (ES) cells revealed that knockdown of intergenic lncRNAs has major consequences on gene expression patterns, which are comparable to the effects of knockdown of well-known ES cell regulators [93]. This indicated that lncRNAs might play important roles in regulating developmental process. The ENCODE project analyzed the tissue-specific expression of lncRNAs in 31 cell types, and found that many lncRNAs have brain-specific expression pattern [9, 12]. There are increasing lines of evidences that link dysregulations of lncRNAs to diverse human diseases ranging from neuron diseases to cancer [9, 10], suggesting that the involvement of lncRNAs in human diseases can be far more prevalent than previously thought [94].

4.2. Molecular Mechanisms of lncRNA. The precise mechanism of how lncRNAs function still remains largely unknown. Currently, there are several hypothesis about it, including (1) RNA:DNA:DNA triplex (*trans*-); (2) RNA:DNA hybrid; (3) RNA:RNA hybrid of lncRNA with a nascent transcript; (4) RNA-protein interaction (*cis*-/*trans*-). Although only (1), (2), and (4) have been experimentally demonstrated so far [14], it is generally thought that lncRNAs may function through the interaction with its partners, such as DNA, RNA, or protein, and serve the following roles: signal, decoy, scaffold, and guide [11, 14]. Table 2 lists lncRNAs that use different mechanisms when carrying out their functions. Below, we give examples for the above-mentioned mechanisms.

Signal. Some lncRNAs have been reported to respond to diverse stimuli, hinting they may act as molecular signals [12, 24, 25, 27, 35]. For example, lncRNAs can act as markers for imprinting (Air and Kcnq1ot1), X inactivation (Xist), and silencing (COOLAIR). ChIP-Seq studies showed that the gene-activating enhancers produce lncRNA transcripts (eRNAs) [29, 95], and their expression level positively correlates with that of nearby genes, indicating a possible role in regulating mRNA synthesis. This is supported by a recent Loss-of-Function study that found the knockdown of 7 out of 12 lncRNAs affects expression of their cognate neighboring genes [8].

Decoy. lncRNA can function as molecular decoy to negatively regulate an effector. Gas5 contains a hairpin sequence motif that resembles the DNA-binding site of the glucocorticoid receptor [31]. It can serve as a decoy to release the receptor from DNA to prevent transcription of metabolic genes [14]. Another example is the telomeric repeat-containing RNA (TERRA). It interacts with the telomerase protein through a repeat sequence complementary to the template sequence of telomerase RNA [11, 34].

Guide. Upon interaction with the target molecular, lncRNA may have the ability to guide it into the proper position either in *cis* (on neighboring genes) or in *trans* (on distantly located genes). The newly found eRNAs appear to exert their effects in *cis* by binding to specific enhancers and actively engaged in regulating mRNA synthesis [11, 29]. HOTAIR and HOTTIP are transcribed within the human *HOX* clusters, and serve as signals of anatomic positions by expressing in cells that have distal and posterior positional identities; they both require the interacting partners to be properly localized to the site of action [6]. In this process, chromosomal looping of the 5′ end of *HOXA* brings HOTTIP into the spatial proximity of multiple *HOXA* genes, enforcing the maintenance of H3K4me3 and gene activation [14]. This long-range gene activation mechanism suggests that chromosome looping plays a central role in delivering lncRNA to its site of action [11, 45].

Scaffold. Recent studies found that several lncRNAs have the capacity to bind more than two protein partners, where the lncRNAs serve as adaptors to form the functional protein complexes. The telomerase RNA TERC (TERRA) is a classic example of RNA scaffold, and is essential for telomerase function. *HOTAIR* binds the polycomb complex PRC2 to exert its "signal" function. A recent study found that the 3,700 nt of HOTAIR also interact with a second complex consisting of LSD1, CoREST, and REST to antagonize gene activation, further emphasizing its important role as the scaffold of the functional complex [11, 51].

Cis- and Trans-Action of lncRNAs. lncRNAs can be classified as *cis*- or *trans*-regulators depending on whether it exerts its function on a neighboring gene on the same allele from which it is transcribed [96]. It was considered that many lncRNAs act as *cis*-regulators, as the expression of lncRNA is significantly correlated with their neighboring protein-coding genes [97, 98]. However, recent studies have questioned that the positive correlation between lncRNAs and their neighboring genes may be due to shared upstream regulation (such as, lincRNA-*p21* [24] and lincRNA-*Sox2* [6]), positional correlation (such as, HOTAIR [6]), transcriptional "ripple effects" [98], and indirect regulation of neighboring genes, instead of the effects of *cis*-regulation. This was supported by the fact that knock down of different number of lncRNAs had little effect on the expression of neighboring genes [96]. In general, it has been accepted that some lncRNAs are *cis*-regulators [99, 100], while the vast majority may function as *trans*-regulators [6, 11, 93]. Recently, some *cis*-regulating lncRNAs were found to have the capacity to act in *trans* [33, 101, 102], highlighting the complexity of lncRNAs.

Although substantial research progresses have been made since the discovery of lncRNAs, it still remains a challenge to understand the functions of lncRNAs. One reason is, unlike protein-coding genes whose mutations may result in severely obvious phenotypes, mutations in lncRNAs often do not cause significant phenotypes [85]. It is likely that lncRNAs may function at specific stage of development process or under specific conditions, and thus condition-specific studies of lncRNAs' phenotypes may be necessary. With more omics data about lncRNAs accumulating, computational prediction of the function of lncRNAs can help to design experiments to accelerate the understanding of lncRNAs.

5. lncRNA Database

The current lncRNA databases are summarized in Table 3. lncRNAdb is an integrated database specific for lncRNAs, including annotation, sequence, structural, species, and function categories of lncRNAs [55]. NONCODE is a database about ncRNAs that have been experimentally confirmed. It covers almost all published 73,272 lncRNAs in human and mouse; it also includes expression profiles of lncRNAs and their potential functions predicted from Coding-Noncoding coexpression network (see below) [56]. LNCipedia is another integrated lncRNA database, which includes 21,488 annotated human lncRNAs. It contains lncRNAs information about the coding potential, secondary structure, and microRNA binding sites [57]. fRNAdb and NRED are databases for ncRNAs including lncRNAs [58, 59].

Table 2: Function classification of lncRNAs.

Archetype	lncRNA name	Length	Target	Function	*cis*-/*trans*-	References
Signal	KCNQ1ot1, Air, Xist	91 kb, 108 kb, ~17 kb	G9a, PRC, YY1	Transcriptional silencing of multiple genes; X inactivation (XCI)	*cis*-	[11, 14, 22, 23]
	HOTAIR, Frigidair, HOTTIP,	2.2 kb, N.A., 3.7 kb	LSD1-CoREST	Signals of anatomic position,	*trans*-	[6, 11, 14]
	lincRNA-p21, PANDA	3 kb; 1.5 kb	hnRNP-K	p53 targets in response to DNA damage	*trans*-	[14, 24, 25]
	lincRNA-RoR	2.6 kb	Oct4, Sox2, Nanog	Pluripotency-associated	N.A.[b]	[11, 26]
	COOLAIR, COLDAIR	Multiple spliced: 400 bp/750 bp; ~1.1 kb	FLC, PRC2	Combinatorial transcriptional regulation	N.A.	[27, 28]
	eRNA	Various sizes	MLL-WDR5, TFs[a]	Promotes mRNA synthesis	*cis*-	[29, 30]
	Gas5	~7 kb	Glucocorticoid receptor	Represses the glucocorticoid receptor	N.A.	[31]
	1/2-sbsRNAs	N.A.[c]	SMD	Formation of STAU1 binding sites	N.A.	[32]
Decoys	DHFR-Minor	7.3, 5.0, 1.4, and 0.8 kb	TFIIB	Inhibits assembly of the preinitiation complex	N.A.	[33]
	TERRA	Various sizes	Telomerase	Regulation and protection of chromosome ends	N.A.	[34]
	PANDA	1.5 kb	NF-YA	Inhibits expression of apoptotic genes	*trans*-	[35]
	PTENP1	~3.9 kb	PTEN	Sequestration of miRNAs	N.A.	[36, 37]
	MALAT1	~7 kb	SR splicing factors	Alters pattern of alternative splicing	N.A.	[38, 39]
Guides	Xist	~17 kb	PRC2, YY1	Inactives X chromosome	*cis*-	[14, 40–42]
	Air, COLDAIR	108 kb,	G9a, PRC2	Silences transcription, affects histone acetylation and methylation states	*cis*-	[28, 43, 44]
	HOTTIP	~3.8 kb	MLL-WDR5	Chromosomal looping, chromatin modifications	*cis*- (looping)	[11, 45]
	HOTAIR	2.2 kb	LSD1-CoREST	Alters and regulates epigenetic states	*trans*-	[14, 46, 47]
	Jpx	Multiple isoforms	polycomb complex[a]	Activation of Xist RNA on the inactive X	*trans*-	[11, 48]
	lincRNA-p21	3 kb	hnRNP-K[a]	p53 targets in response to DNA damage	*trans*-	[11, 24]
Scaffold	TERC	Various sizes	TERT	Telomerase catalytic activity	*trans*-	[49, 50]
	HOTAIR	2.2 kb	PRC2, LSD1, CoREST, REST	Demethylates histone H3 on K4 to antagonize gene activation	*trans*-	[46, 51]
	ANRIL	Multiple spliced: 3.9 kb/34.8 kb	PRC1, PRC2	Contributes to the functions of both PRC1 and PRC2 proteins	*trans*-	[52, 53]
	Alpha Satellite Repeat LncRNA	N.A.	SUMO-HP1	Molecular scaffold for the targeting and local accumulation of HP1	N.A.	[11, 54]

[a] Not yet understood.
[b] Not clearly referred as *cis*-action.
[c] No length data available in all six databases listed in Table 3.

Table 3: List of lncRNA databases.

Tools	Source	Description	Reference
lncRNAdb	http://www.lncrnadb.org/	Contain comprehensive list of lncRNAs in eukaryotes, and mRNAs with regulatory roles	[55]
NONCODE	http://noncode.org/	Integrative annotation of noncoding RNA (73,372 lncRNAs)	[56]
LNCipedia	http://www.lncipedia.org/	21 488 annotated human lncRNA transcripts with secondary structure information, protein coding potential, and microRNA binding sites	[57]
fRNAdb	http://www.ncrna.org/frnadb/	A large collection of noncoding transcripts including annotated/unannotated sequences from H-inv database, NONCODE, and RNAdb	[58]
NRED	http://jsm-research.imb.uq.edu.au/nred/cgi-bin/ncrnadb.pl/	Noncoding RNA Expression Database	[59]

The above databases provide great convenience for further analysis and applications of lncRNAs.

6. Function Prediction of lncRNA

Computational prediction of lncRNA functions is still at its early development stage. Unlike protein-coding genes whose sequence motifs are indicative of their function, lncRNA sequences are usually not conserved and do not contain conserved sequence motifs [103, 104]. The secondary structures of lncRNA are also not conserved [105]. Thus, it is difficult to infer the function of lncRNAs based on their sequences or secondary structures alone. Since current knowledge suggests that lncRNAs function by regulating or interacting with its partner molecular, current methods focus on exploring the relationships between lncRNAs and protein-coding genes or miRNAs. Below, we will describe several current approaches for predicting the functions of lncRNAs.

6.1. Comparative Genomics Approach. Although most lncRNAs are not conserved, there are lncRNAs that are conserved across species, indicating their essential functions. Amit et al. identified 78 lncRNAs transcripts conserved in both human and mouse, and found 70 are either located within or close (<1000 nt distance) to a coding gene that is also conserved in the two genomes [106]. They assumed these lncRNAs might have close functional relationships with the nearby coding genes. However, this approach is limited because of the poor conservation of lncRNAs and cannot be applied at genome scale.

6.2. Coexpression with Coding Genes Approach. Many studied lncRNAs play important regulatory roles, and it is likely that lncRNAs regulating a specific biological process may be coexpressed with the genes involved in the same process. Thus, identifying coding genes that are coexpressed with lncRNAs may help to infer the function of lncRNAs. Based on this assumption, Guttman et al. developed a coexpression based method to predict lncRNAs functions at genome scale [71]. For each lncRNA, they ranked coding genes based on their coexpression level with the lncRNAs, and then performed a Gene Set Enrichment Analysis (GSEA) for the top-ranked genes to identify enriched functional terms corresponding to the lncRNAs. Out of 150 lncRNAs subjected for experimental validation, 85 exhibited the predicted functions, proving the effectiveness of using the coexpressed coding genes to infer the function of lncRNAs from their coexpressed coding genes. According to their predictions, lncRNAs participate in a rather wide range of biological processes such as cell proliferation, development, and immune surveillance. Andrea et al. employed a similar approach to predict the function of lncRNAs during zebrafish embryogenesis [67].

Liao et al. furthered the coexpression idea by constructing a coding-noncoding (CNC) gene coexpression network [107]. In contrast to the GSEA method that collects coding genes coexpressed for each lncRNA, the CNC method considers not only the coexpression between lncRNAs and coding genes, but also within lncRNAs group and coding gene group. When predicting the function of lncRNAs, the CNC method employs two different approaches: the hub-based and the network-module-based. In the hub-based approach, functions are assigned to each lncRNA according to the functional enrichment of its neighboring genes. In the network-module-based approach, Markov cluster algorithm (MCL) is used to identify coexpressed functional module in the CNC network; then functions of the module are transferred to the lncRNAs inside the module. Liao et al. applied the CNC method to annotate the functions of 340 mouse lncRNAs, and found these lncRNAs function mainly in organ or tissue development, cellular transport, and metabolic processes.

6.3. Interaction with miRNAs and Proteins Approach. Recent analysis found that lncRNAs share a synergism with miRNA in the regulatory network [108, 109]. It is likely that some lncRNAs function by binding miRNA. Therefore, identifying well-established miRNAs that bind lncRNAs may help to infer the function of lncRNAs. Jeggari et al. developed an algorithm named miRcode that predicts putative microRNA binding sites in lncRNAs using criteria such as seed complementarity and evolutionary conservation [110]. Jalali et al. constructed a genome-wide network of validated RNA mediated interactions, and uncovered previously unknown

mediatory roles of lncRNA between miRNA and mRNA (Saakshi Jalali, arXiv preprint). Besides the interaction with miRNA, the interaction of lncRNAs with proteins can also be explored to predict their functions. Bellucci et al. developed a method called "catRAPID" that correlates lncRNAs with proteins by evaluating their interaction potential using physicochemical characteristics, including secondary structure, hydrogen bonding, van der Waals, and so forth [111]. However, unlike the coexpression based approach, the above two approaches were successful in only a number of lncRNAs, partly because the mechanism of how lncRNAs interact with miRNAs and proteins still remains unclear.

6.4. Challenges. Computational prediction of lncRNA functions is still at its primary stage. As the sequence and secondary structure of lncRNAs are generally not conserved, function prediction of lncRNAs mainly relies on their relationships with other moleculars, such as protein coding genes, miRNAs, and proteins. However, the molecular mechanism of how lncRNA function by interacting with other molecular remains largely unknown, making it difficult to develop computational methods to precisely predict the functions of lncRNAs. On the other hand, there are currently only a small number of lncRNAs whose functions are well understood, which makes it difficult to validate and optimize computational algorithms for predicting lncRNA functions. Finally, unlike protein-coding genes that have systematic functional annotation systems, there lacks an annotation system for lncRNA functions, making it difficult to evaluate computational algorithms for function prediction. Nevertheless, the success of predicting lncRNAs using the coexpression based approach has shown promises. With more functional genomics data about lncRNAs available in the near future, more powerful and accurate methods will be developed to help decipher the functions of lncRNAs.

7. Perspectives

It has been widely accepted that lncRNAs play important functional roles in cell, though the molecular mechanism of how lncRNAs function remains to be unraveled. In this paper, we have described several currently proposed models about the molecular mechanism of lncRNA functions. One commonality about these models is that lncRNAs function through the interaction with other molecular, including DNA, RNA, and proteins. Given the abundance of lncRNAs in genome, it is likely that the interaction between lncRNAs and other moleculars may be specific. This thus raises the possibility of developing novel methods to target certain lncRNA for gene-specific regulation. However, phenotypic studies of lncRNAs suggested that knockdown of many lncRNAs does not result in obvious phenotypes, making it difficult to understand their functions. Computational prediction of lncRNAs can provide hypothesis about the functions of lncRNAs, and help to design experiments to test them under specific conditions. Yet, it remains a significant challenge to develop effective methods to accurately infer the lncRNA functions, owing to the lack of detailed information about the molecular mechanisms of lncRNAs. In order to develop powerful computational methods, more studies about the derivation of lncRNAs, the molecular mechanism of lncRNAs and tissue-specific, or development-specific expression about lncRNAs are necessary.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant no. 31071113).

References

[1] P. Carninci, T. Kasukawa, S. Katayama et al., "The transcriptional landscape of the mammalian genome," *Science*, vol. 309, pp. 1559–1563, 2005.

[2] E. Birney, J. A. Stamatoyannopoulos, A. Dutta et al., "Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project," *Nature*, vol. 447, pp. 799–816, 2007.

[3] P. Kapranov, J. Cheng, S. Dike et al., "RNA maps reveal new RNA classes and a possible function for pervasive transcription," *Science*, vol. 316, no. 5830, pp. 1484–1488, 2007.

[4] J. E. Wilusz, S. M. Freier, and D. L. Spector, "3′ end processing of a long nuclear-retained noncoding RNA yields a tRNA-like cytoplasmic RNA," *Cell*, vol. 135, no. 5, pp. 919–932, 2008.

[5] A. C. Seila, J. M. Calabrese, S. S. Levine et al., "Divergent transcription from active promoters," *Science*, vol. 322, no. 5909, pp. 1849–1851, 2008.

[6] J. L. Rinn, M. Kertesz, J. K. Wang et al., "Functional demarcation of active and silent chromatin domains in human HOX loci by noncoding RNAs," *Cell*, vol. 129, no. 7, pp. 1311–1323, 2007.

[7] H. Jia, M. Osak, G. K. Bogu, L. W. Stanton, R. Johnson, and L. Lipovich, "Genome-wide computational identification and manual annotation of human long noncoding RNA genes," *RNA*, vol. 16, no. 8, pp. 1478–1487, 2010.

[8] U. A. Ørom, T. Derrien, M. Beringer et al., "Long noncoding RNAs with enhancer-like function in human cells," *Cell*, vol. 143, no. 1, pp. 46–58, 2010.

[9] I. A. Qureshi, J. S. Mattick, and M. F. Mehler, "Long noncoding RNAs in nervous system function and disease," *Brain Research*, vol. 1338, no. C, pp. 20–35, 2010.

[10] O. Wapinski and H. Y. Chang, "Long noncoding RNAs and human disease," *Trends in Cell Biology*, vol. 21, no. 6, pp. 354–361, 2011.

[11] K. C. Wang and H. Y. Chang, "Molecular mechanisms of long noncoding RNAs," *Molecular Cell*, vol. 43, pp. 904–914, 2011.

[12] T. Derrien, R. Johnson, G. Bussotti, A. Tanzer, S. Djebali et al., "The GENCODE v7 catalog of human long noncoding RNAs: analysis of their gene structure, evolution, and expression," *Genome Research*, vol. 22, pp. 1775–1789, 2012.

[13] M. E. Dinger, K. C. Pang, T. R. Mercer, and J. S. Mattick, "Differentiating protein-coding and noncoding RNA: challenges and ambiguities," *PLoS Computational Biology*, vol. 4, no. 11, Article ID e1000176, 2008.

[14] J. L. Rinn and H. Y. Chang, "Genome regulation by long noncoding RNAs," *Annual Review of Biochemistry*, vol. 81, pp. 145–166, 2012.

[15] C. P. Ponting, P. L. Oliver, and W. Reik, "Evolution and functions of long noncoding RNAs," *Cell*, vol. 136, no. 4, pp. 629–641, 2009.

[16] J.-W. Nam and D. P. Bartel, "Long noncoding RNAs in *C. elegans*," *Genome Research*, vol. 22, no. 12, pp. 2529–2540, 2012.

[17] M. C. Tsai, R. C. Spitale, and H. Y. Chang, "Long intergenic noncoding RNAs: new links in cancer progression," *Cancer Research*, vol. 71, no. 1, pp. 3–7, 2011.

[18] J. Liu, J. Gough, and B. Rost, "Distinguishing protein-coding from non-coding RNAs through support vector machines," *PLoS genetics*, vol. 2, no. 4, article no. e29, 2006.

[19] S. F. Altschul, T. L. Madden, A. A. Schäffer et al., "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs," *Nucleic Acids Research*, vol. 25, no. 17, pp. 3389–3402, 1997.

[20] L. Kong, Y. Zhang, Z. Q. Ye et al., "CPC: assess the protein-coding potential of transcripts using sequence features and support vector machine," *Nucleic Acids Research*, vol. 35, pp. W345–W349, 2007.

[21] Z. J. Lu, K. Y. Yip, G. Wang et al., "Prediction and characterization of noncoding RNAs in *C. elegans* by integrating conservation, secondary structure, and high-throughput sequencing and array data," *Genome Research*, vol. 21, no. 5, pp. 276–285, 2011.

[22] R. R. Pandey, T. Mondal, F. Mohammad et al., "Kcnq1ot1 antisense noncoding RNA mediates lineage-specific transcriptional silencing through chromatin-level regulation," *Molecular Cell*, vol. 32, no. 2, pp. 232–246, 2008.

[23] F. Mohammad, T. Mondal, and C. Kanduri, "Epigenetics of imprinted long noncoding RNAs," *Epigenetics*, vol. 4, no. 5, pp. 277–286, 2009.

[24] M. Huarte, M. Guttman, D. Feldser et al., "A large intergenic noncoding RNA induced by p53 mediates global gene repression in the p53 response," *Cell*, vol. 142, no. 3, pp. 409–419, 2010.

[25] T. Hung, Y. Wang, M. F. Lin et al., "Extensive and coordinated transcription of noncoding RNAs within cell-cycle promoters," *Nature Genetics*, vol. 43, no. 7, pp. 621–629, 2011.

[26] S. Loewer, M. N. Cabili, M. Guttman et al., "Large intergenic non-coding RNA-RoR modulates reprogramming of human induced pluripotent stem cells," *Nature Genetics*, vol. 42, no. 12, pp. 1113–1117, 2010.

[27] S. Swiezewski, F. Liu, A. Magusin, and C. Dean, "Cold-induced silencing by long antisense transcripts of an *Arabidopsis* Polycomb target," *Nature*, vol. 462, no. 7274, pp. 799–802, 2009.

[28] J. B. Heo and S. Sung, "Vernalization-mediated epigenetic silencing by a long intronic noncoding RNA," *Science*, vol. 331, no. 6013, pp. 76–79, 2011.

[29] T. K. Kim, M. Hemberg, J. M. Gray et al., "Widespread transcription at neuronal activity-regulated enhancers," *Nature*, vol. 465, no. 7295, pp. 182–187, 2010.

[30] D. Wang, I. Garcia-Bassets, C. Benner et al., "Reprogramming transcription by distinct classes of enhancers functionally defined by eRNA," *Nature*, vol. 474, no. 7351, pp. 390–397, 2011.

[31] T. Kino, D. E. Hurt, T. Ichijo, N. Nader, and G. P. Chrousos, "Noncoding RNA Gas5 is a growth arrest- and starvation-associated repressor of the glucocorticoid receptor," *Science Signaling*, vol. 3, no. 107, article no. ra8, 2010.

[32] C. Gong and L. E. Maquat, "LncRNAs transactivate STAU1-mediated mRNA decay by duplexing with 39 UTRs via Alu eleme," *Nature*, vol. 470, no. 7333, pp. 284–288, 2011.

[33] I. Martianov, A. Ramadass, A. Serra Barros, N. Chow, and A. Akoulitchev, "Repression of the human dihydrofolate reductase gene by a non-coding interfering transcript," *Nature*, vol. 445, no. 7128, pp. 666–670, 2007.

[34] S. Redon, P. Reichenbach, and J. Lingner, "The non-coding RNA TERRA is a natural ligand and direct inhibitor of human telomerase," *Nucleic Acids Research*, vol. 38, no. 17, Article ID gkq296, pp. 5797–5806, 2010.

[35] T. Hung and H. Y. Chang, "Long noncoding RNA in genome regulation: prospects and mechanisms," *RNA Biology*, vol. 7, no. 5, pp. 582–585, 2010.

[36] L. Poliseno, L. Salmena, J. Zhang, B. Carver, W. J. Haveman, and P. P. Pandolfi, "A coding-independent function of gene and pseudogene mRNAs regulates tumour biology," *Nature*, vol. 465, no. 7301, pp. 1033–1038, 2010.

[37] M. S. Song, A. Carracedo, L. Salmena et al., "Nuclear PTEN regulates the APC-CDH1 tumor-suppressive complex in a phosphatase-independent manner," *Cell*, vol. 144, no. 2, pp. 187–199, 2011.

[38] V. Tripathi, J. D. Ellis, Z. Shen et al., "The nuclear-retained noncoding RNA MALAT1 regulates alternative splicing by modulating SR splicing factor phosphorylation," *Molecular Cell*, vol. 39, no. 6, pp. 925–938, 2010.

[39] D. Bernard, K. V. Prasanth, V. Tripathi et al., "A long nuclear-retained non-coding RNA regulates synaptogenesis by modulating gene expression," *EMBO Journal*, vol. 29, no. 18, pp. 3082–3093, 2010.

[40] K. Plath, S. Mlynarczyk-Evans, D. A. Nusinow, and B. Panning, "Xist RNA and the mechanism of X chromosome inactivation," *Annual Review of Genetics*, vol. 36, pp. 233–278, 2002.

[41] J. T. Lee, "The X as model for RNA's niche in epigenomic regulation," *Cold Spring Harbor Perspectives in Biology*, vol. 2, no. 9, Article ID a003749, 2010.

[42] B. K. Sun, A. M. Deaton, and J. T. Lee, "A transient heterochromatic state in Xist preempts X inactivation choice without RNA stabilization," *Molecular Cell*, vol. 21, no. 5, pp. 617–628, 2006.

[43] T. Nagano, J. A. Mitchell, L. A. Sanz et al., "The Air noncoding RNA epigenetically silences transcription by targeting G9a to chromatin," *Science*, vol. 322, no. 5908, pp. 1717–1720, 2008.

[44] J. Camblong, N. Iglesias, C. Fickentscher, G. Dieppois, and F. Stutz, "Antisense RNA stabilization induces transcriptional gene silencing via histone seacetylation in *S. cerevisiae*," *Cell*, vol. 131, no. 4, pp. 706–717, 2007.

[45] K. C. Wang, Y. W. Yang, B. Liu et al., "A long noncoding RNA maintains active chromatin to coordinate homeotic gene expression," *Nature*, vol. 472, no. 7341, pp. 120–126, 2011.

[46] A. M. Khalil, M. Guttman, M. Huarte et al., "Many human large intergenic noncoding RNAs associate with chromatin-modifying complexes and affect gene expression," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 28, pp. 11667–11672, 2009.

[47] J. Zhao, T. K. Ohsumi, J. T. Kung et al., "Genome-wide identification of polycomb-associated RNAs by RIP-seq," *Molecular Cell*, vol. 40, no. 6, pp. 939–953, 2010.

[48] D. Tian, S. Sun, and J. T. Lee, "The long noncoding RNA, Jpx, Is a molecular switch for X chromosome inactivation," *Cell*, vol. 143, no. 3, pp. 390–403, 2010.

[49] K. Collins, "Physiological assembly and activity of human telomerase complexes," *Mechanisms of Ageing and Development*, vol. 129, no. 1-2, pp. 91–98, 2008.

[50] D. C. Zappulla and T. R. Cech, "Yeast telomerase RNA: a flexible scaffold for protein subunits," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 27, pp. 10024–10029, 2004.

[51] M. C. Tsai, O. Manor, Y. Wan et al., "Long noncoding RNA as modular scaffold of histone modification complexes," *Science*, vol. 329, no. 5992, pp. 689–693, 2010.

[52] Y. Kotake, T. Nakagawa, K. Kitagawa et al., "Long non-coding RNA ANRIL is required for the PRC2 recruitment to and silencing of $p15^{INK4B}$ tumor suppressor gene," *Oncogene*, vol. 30, no. 16, pp. 1956–1962, 2011.

[53] K. L. Yap, S. Li, A. M. Muñoz-Cabello et al., "Molecular interplay of the noncoding RNA *ANRIL* and methylated histone H3 lysine 27 by polycomb CBX7 in transcriptional silencing of *INK4a*," *Molecular Cell*, vol. 38, no. 5, pp. 662–674, 2010.

[54] C. Maison, D. Bailly, D. Roche et al., "SUMOylation promotes de novo targeting of HP1alpha to pericentric heterochromatin," *Nature Genetics*, vol. 43, no. 3, pp. 220–227, 2011.

[55] P. P. Amaral, M. B. Clark, D. K. Gascoigne, M. E. Dinger, and J. S. Mattick, "LncRNAdb: a reference database for long xoncoding RNAs," *Nucleic Acids Research*, vol. 39, no. 1, pp. D146–D151, 2011.

[56] D. Bu, K. Yu, S. Sun, C. Xie, G. Skogerbo et al., "NONCODE v3. 0: integrative annotation of long noncoding RNAs," *Nucleic Acids Research*, vol. 40, pp. D210–D215, 2012.

[57] P. J. Volders, K. Helsens, X. Wang, B. Menten, L. Martens et al., "LNCipedia: a database for annotated human lncRNA transcript sequences and structures," *Nucleic Acids Research*. In press.

[58] T. Kin, K. Yamada, G. Terai et al., "fRNAdb: a platform for mining/annotating functional RNA candidates from non-coding RNA sequences," *Nucleic Acids Research*, vol. 35, no. 1, pp. D145–D148, 2007.

[59] M. E. Dinger, K. C. Pang, T. R. Mercer, M. L. Crowe, S. M. Grimmond, and J. S. Mattick, "NRED: a database of long noncoding RNA expression," *Nucleic Acids Research*, vol. 37, no. 1, pp. D122–D126, 2009.

[60] S. K. Michelhaugh, L. Lipovich, J. Blythe, H. Jia, G. Kapatos, and M. J. Bannon, "Mining Affymetrix microarray data for long non-coding RNAs: altered expression in the nucleus accumbens of heroin abusers," *Journal of Neurochemistry*, vol. 116, no. 3, pp. 459–466, 2011.

[61] T. Babak, B. J. Blencowe, and T. R. Hughes, "A systematic search for new mammalian noncoding RNAs indicates little conserved intergenic transcription," *BMC Genomics*, vol. 6, article no. 14, 2005.

[62] E. A. Gibb, E. A. Vucic, K. S. Enfield, G. L. Stewart, K. M. Lonergan et al., "Human cancer long non-coding RNA transcriptomes," *PLoS One*, vol. 6, Article ID e25915, 2011.

[63] T. L. Lee, A. Xiao, and O. M. Rennert, "Identification of novel long noncoding RNA transcripts in male germ cells," *Methods in Molecular Biology*, vol. 825, pp. 105–114, 2012.

[64] M. Furuno, K. C. Pang, N. Ninomiya et al., "Clusters of internally primed transcripts reveal novel long noncoding RNAs," *PLoS Genetics*, vol. 2, no. 4, article no. e37, 2006.

[65] W. Huang, N. Long, and H. Khatib, "Genome-wide identification and initial characterization of bovine long non-coding RNAs from EST data," *Animal Genetics*, vol. 43, pp. 674–682, 2012.

[66] T. Li, S. Wang, R. Wu, X. Zhou, D. Zhu et al., "Identification of long non-protein coding RNAs in chicken skeletal muscle using next generation sequencing," *Genomics*, vol. 99, pp. 292–298, 2012.

[67] A. Pauli, E. Valen, M. F. Lin, M. Garber, N. L. Vastenhouw et al., "Systematic identification of long noncoding RNAs expressed during zebrafish embryogenesis," *Genome Research*, vol. 22, pp. 577–591, 2012.

[68] J. R. Prensner, M. K. Iyer, O. A. Balbin et al., "Transcriptome sequencing across a prostate cancer cohort identifies PCAT-1, an unannotated lincRNA implicated in disease progression," *Nature Biotechnology*, vol. 29, no. 8, pp. 742–749, 2011.

[69] J. Zhao, B. K. Sun, J. A. Erwin, J. J. Song, and J. T. Lee, "Polycomb proteins targeted by a short repeat RNA to the mouse X chromosome," *Science*, vol. 322, no. 5902, pp. 750–756, 2008.

[70] P. J. Park, "ChIP-seq: advantages and challenges of a maturing technology," *Nature Reviews Genetics*, vol. 10, no. 10, pp. 669–680, 2009.

[71] M. Guttman, I. Amit, M. Garber et al., "Chromatin signature reveals over a thousand highly conserved large non-coding RNAs in mammals," *Nature*, vol. 458, no. 7235, pp. 223–227, 2009.

[72] Y. Okazaki, M. Furuno, T. Kasukawa et al., "Analysis of the mouse transcriptome based on functional annotation of 60,770 full-length cDNAs," *Nature*, vol. 420, no. 6915, pp. 563–573, 2002.

[73] X. Yang, T. J. Tschaplinski, G. B. Hurst et al., "Discovery and annotation of small proteins using genomics, proteomics, and computational approaches," *Genome Research*, vol. 21, no. 4, pp. 634–641, 2011.

[74] M. F. Lin, J. W. Carlson, M. A. Crosby et al., "Revisiting the protein-coding gene catalog of *Drosophila melanogaster* using 12 fly genomes," *Genome Research*, vol. 17, no. 12, pp. 1823–1836, 2007.

[75] M. Clamp, B. Fry, M. Kamal et al., "Distinguishing protein-coding and noncoding genes in the human genome," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 49, pp. 19428–19433, 2007.

[76] M. F. Lin, I. Jungreis, and M. Kellis, "PhyloCSF: a comparative genomics method to distinguish protein coding and non-coding regions," *Bioinformatics*, vol. 27, no. 13, Article ID btr209, pp. i275–i282, 2011.

[77] S. Washietl, S. Findeiß, S. A. Müller et al., "RNAcode: robust discrimination of coding and noncoding regions in comparative sequence data," *RNA*, vol. 17, no. 4, pp. 578–594, 2011.

[78] E. Rivas and S. R. Eddy, "Noncoding RNA gene detection using comparative sequence analysis," *BMC Bioinformatics*, vol. 2, article no. 8, 2001.

[79] S. Washietl, I. L. Hofacker, and P. F. Stadler, "Fast and reliable prediction of noncoding RNAs," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 7, pp. 2454–2459, 2005.

[80] J. S. Pedersen, G. Bejerano, A. Siepel et al., "Identification and classification of conserved RNA secondary structures in the human genome," *PLoS Computational Biology*, vol. 2, no. 4, article no. e33, pp. 251–262, 2006.

[81] L. Duret, C. Chureau, S. Samain, J. Weissanbach, and P. Avner, "The Xist RNA gene evolved in eutherians by pseudogenization of a protein-coding gene," *Science*, vol. 312, no. 5780, pp. 1653–1655, 2006.

[82] S. Chooniedass-Kothari, E. Emberley, M. K. Hamedani et al., "The steroid receptor RNA activator is the first functional RNA encoding a protein," *FEBS Letters*, vol. 566, no. 1-3, pp. 43–47, 2004.

[83] E. D. Kim and S. Sung, "Long noncoding RNA: unveiling hidden layer of gene regulatory networks," *Trends in Plant Science*, vol. 17, pp. 16–21, 2012.

[84] M. Hiller, S. Findeiß, S. Lein et al., "Conserved introns reveal novel transcripts in *Drosophila melanogaster*," *Genome Research*, vol. 19, no. 7, pp. 1289–1300, 2009.

[85] J. S. Mattick, "The genetic signatures of noncoding RNAs," *PLoS Genetics*, vol. 5, no. 4, Article ID e1000459, 2009.

[86] E. Bernstein and C. D. Allis, "RNA meets chromatin," *Genes and Development*, vol. 19, no. 14, pp. 1635–1655, 2005.

[87] J. Whitehead, G. K. Pandey, and C. Kanduri, "Regulation of the mammalian epigenome by long noncoding RNAs," *Biochimica et Biophysica Acta*, vol. 1790, no. 9, pp. 936–947, 2009.

[88] J. E. Wilusz, H. Sunwoo, and D. L. Spector, "Long noncoding RNAs: functional surprises from the RNA world," *Genes and Development*, vol. 23, no. 13, pp. 1494–1504, 2009.

[89] M. Beltran, I. Puig, C. Peña et al., "A natural antisense transcript regulates Zeb2/Sip1 gene expression during Snail1-induced epithelial-mesenchymal transition," *Genes and Development*, vol. 22, no. 6, pp. 756–769, 2008.

[90] U. A. Ørom and R. Shiekhattar, "Noncoding RNAs and enhancers: complications of a long-distance relationship," *Trends in Genetics*, vol. 27, pp. 433–439, 2011.

[91] J. S. Mattick and I. V. Makunin, "Small regulatory RNAs in mammals," *Human Molecular Genetics*, vol. 14, no. 1, pp. R121–R132, 2005.

[92] T. Nagano and P. Fraser, "No-nonsense functions for long noncoding RNAs," *Cell*, vol. 145, no. 2, pp. 178–181, 2011.

[93] M. Guttman, J. Donaghey, B. W. Carey, M. Garber, J. K. Grenier et al., "lincRNAs act in the circuitry controlling pluripotency and differentiation," *Nature*, vol. 477, pp. 295–300, 2011.

[94] R. Johnson, "Long non-coding RNAs in Huntington's disease neurodegeneration," *Neurobiology of Disease*, vol. 46, pp. 245–254, 2012.

[95] F. De Santa, I. Barozzi, F. Mietton et al., "A large fraction of extragenic RNA Pol II transcription sites overlap enhancers," *PLoS Biology*, vol. 8, no. 5, Article ID e1000384, 2010.

[96] Z. H. Li and T. M. Rana, "Molecular mechanisms of RNA-triggered gene silencing machineries," *Accounts of Chemical Research*, vol. 45, pp. 1122–1131, 2012.

[97] J. Ponjavic, P. L. Oliver, G. Lunter, and C. P. Ponting, "Genomic and transcriptional co-localization of protein-coding and long non-coding RNA pairs in the developing brain," *PLoS Genetics*, vol. 5, no. 8, Article ID e1000617, 2009.

[98] M. Ebisuya, T. Yamamoto, M. Nakajima, and E. Nishida, "Ripples from neighbouring transcription," *Nature Cell Biology*, vol. 10, no. 9, pp. 1106–1113, 2008.

[99] C. J. Brown, A. Ballabio, J. L. Rupert et al., "A gene from the region of the human X inactivation centre is expressed exclusively from the inactive X chromosome," *Nature*, vol. 349, no. 6304, pp. 38–44, 1991.

[100] F. Sleutels, R. Zwart, and D. P. Barlow, "The non-coding Air RNA is required for silencing autosomal imprinted genes," *Nature*, vol. 415, no. 6873, pp. 810–813, 2002.

[101] J. T. Lee, "Lessons from X-chromosome inactivation: long ncRNA as guides and tethers to the epigenome," *Genes and Development*, vol. 23, no. 16, pp. 1831–1842, 2009.

[102] K. M. Schmitz, C. Mayer, A. Postepska, and I. Grummt, "Interaction of noncoding RNA with the rDNA promoter mediates recruitment of DNMT3b and silencing of rRNA genes," *Genes and Development*, vol. 24, no. 20, pp. 2264–2269, 2010.

[103] A. T. Willingham, A. P. Orth, S. Batalov et al., "Molecular biology: a strategy for probing the function of noncoding RNAs finds a repressor of NFAT," *Science*, vol. 309, no. 5740, pp. 1570–1573, 2005.

[104] T. R. Mercer, M. E. Dinger, and J. S. Mattick, "Long noncoding RNAs: insights into functions," *Nature Reviews Genetics*, vol. 10, no. 3, pp. 155–159, 2009.

[105] K. C. Pang, M. E. Dinger, T. R. Mercer et al., "Genome-wide identification of long noncoding RNAs in $CD8^+$ T cells," *Journal of Immunology*, vol. 182, no. 12, pp. 7738–7748, 2009.

[106] A. N. Khachane and P. M. Harrison, "Mining mammalian transcript data for functional long non-coding RNAs," *PLoS One*, vol. 5, no. 4, Article ID e10316, 2010.

[107] Q. Liao, C. Liu, X. Yuan et al., "Large-scale prediction of long non-coding RNA functions in a coding-non-coding gene co-expression network," *Nucleic Acids Research*, vol. 39, no. 9, pp. 3864–3878, 2011.

[108] C. Braconi, T. Kogure, N. Valeri et al., "microRNA-29 can regulate expression of the long non-coding RNA gene MEG3 in hepatocellular cancer," *Oncogene*, vol. 30, pp. 4750–4756, 2011.

[109] M. S. Ebert and P. A. Sharp, "Emerging roles for natural microRNA sponges," *Current Biology*, vol. 20, no. 19, pp. R858–R861, 2010.

[110] A. Jeggari, D. S. Marks, and E. Larsson, "miRcode: a map of putative microRNA target sites in the long non-coding transcriptome," *Bioinformatics*, vol. 28, pp. 2062–2063, 2012.

[111] M. Bellucci, F. Agostini, M. Masin, and G. G. Tartaglia, "Predicting protein associations with long noncoding RNAs," *Nature Methods*, vol. 8, no. 6, pp. 444–445, 2011.

The Scientific World Journal
Volume 2012, Article ID 957620, 8 pages
doi:10.1100/2012/957620

The ScientificWorldJOURNAL

Research Article

Network Completion Using Dynamic Programming and Least-Squares Fitting

Natsu Nakajima,[1] Takeyuki Tamura,[1] Yoshihiro Yamanishi,[2] Katsuhisa Horimoto,[3] and Tatsuya Akutsu[1]

[1] *Bioinformatics Center, Institute for Chemical Research, Kyoto University Gokasho, Uji, Kyoto 611-0011, Japan*
[2] *Division of System Cohort, Multi-scale Research Center for Medical Science, Medical Institute of Bioregulation, Kyushu University, 3-1-1 Maidashi, Higashi-ku, Fukuoka, Fukuoka 812-8582, Japan*
[3] *Computational Biology Research Center, National Institute of Advanced Industrial Science and Technology, 2-4-7 Aomi, Koto-ku, Tokyo 135-0064, Japan*

Correspondence should be addressed to Tatsuya Akutsu, takutsu@kuicr.kyoto-u.ac.jp

Received 30 August 2012; Accepted 26 September 2012

Academic Editors: W. Tian and X.-M. Zhao

We consider the problem of network completion, which is to make the minimum amount of modifications to a given network so that the resulting network is most consistent with the observed data. We employ here a certain type of differential equations as gene regulation rules in a genetic network, gene expression time series data as observed data, and deletions and additions of edges as basic modification operations. In addition, we assume that the numbers of deleted and added edges are specified. For this problem, we present a novel method using dynamic programming and least-squares fitting and show that it outputs a network with the minimum sum squared error in polynomial time if the maximum indegree of the network is bounded by a constant. We also perform computational experiments using both artificially generated and real gene expression time series data.

1. Introduction

Analysis of biological networks is one of the central research topics in computational systems biology. In particular, extensive studies have been done on inference of genetic networks using gene expression time series data, and a number of computational methods have been proposed, which include methods based on Boolean networks [1, 2], Bayesian networks [3, 4], time-delayed Bayesian networks [5], graphical Gaussian models [6–8], differential equations [9, 10], mutual information [11, 12], and linear classification [13]. However, there is not yet an established or standard method for inference of genetic networks, and thus it still remains a challenging problem.

One of the possible reasons for the difficulty of inference is that the amount of available high-quality gene expression time series data is still not enough, and thus it is intrinsically difficult to infer the correct or nearly correct network from such a small amount of data. Therefore, it is reasonable to try to develop another approach. For that purpose, we proposed an approach called network completion [14] by following Occam's razor, which is a well-known principle in scientific discovery. Network completion is, given an initial network and an observed dataset, to modify the network by the minimum amount of modifications so that the resulting network is (most) consistent with the observed data. Since we were interested in inference of signaling networks in our previous study [14], we assumed that activity levels or quantities of one or a few kinds of proteins can only be observed. Furthermore, since measurement errors were considered to be large and we were interested in theoretical analysis of computational complexity and sample complexity, we adopted the Boolean network [15] as a model of signaling networks. We proved that network completion is computationally intractable (NP-hard) even for tree-structured networks. In order to cope with this computational difficulty, we developed an integer linear programming-based method for completion of signaling pathways [16]. However, this method could not handle addition of edges because of its high computational cost.

In this paper, we propose a novel method, DPLSQ, for completing genetic networks using gene expression time series data. Different from our previous studies [14, 16], we employ a model based on differential equations and assume that expression values of all nodes can be observed. DPLSQ is a combination of least-squares fitting and dynamic programming, where least-squares fitting is used for estimating parameters in differential equations and dynamic programming is used for minimizing the sum of least-squares errors by integrating partial fitting results on individual genes under the constraint that the numbers of added and deleted edges must be equal to the specified ones. One of the important features of DPLSQ is that it can output an optimal solution (i.e., minimum squared sum) in polynomial time if the maximum indegree (i.e., the maximum number of input genes to a gene) is bounded by a constant. Although DPLSQ does not automatically find the minimum modification, it can be found by examining varying numbers of added/deleted edges, where the total number of such combinations is polynomially bounded. If a null network (i.e., a network having no edges) is given as an initial network, DPLSQ can work as an inference method for genetic networks.

In order to examine the effectiveness of DPLSQ, we perform computational experiments using artificially generated data. We also make computational comparison of DPLSQ as an inference method with other existing tools using artificial data. Furthermore, we perform computational experiments on DPLSQ using real cell cycle expression data of *Saccharomyces cerevisiae*.

2. Method

The purpose of network completion is to modify a given network with the minimum number of modifications so that the resulting network is most consistent with the observed data. In this paper, we consider additions and deletions of edges as modification operations (see Figure 1). If we begin with a network with an empty set of edges, it corresponds to network inference. Therefore, network completion includes network inference although it may not necessarily work better than the existing methods if applied to network inference.

In the following, $G(V, E)$ denotes a given network where V and E are the sets of nodes and directed edges respectively, where each node corresponds to a gene and each edge represents some direct regulation between two genes. Self loops are not allowed in E although it is possible to modify the method so that self-loops are allowed. In this paper, n denotes the number of genes (i.e., the number of nodes) and we let $V = \{v_1, \ldots, v_n\}$. For each node v_i, $e^-(v_i)$ and $\deg^-(v_i)$, respectively, denote the set of incoming edges to v_i and the number of incoming edges to v_i as defined follows:

$$e^-(v_i) = \left\{v_j \mid \left(v_j, v_i\right) \in E\right\},$$
$$\deg^-(v_i) = \left|e^-(v_i)\right|. \tag{1}$$

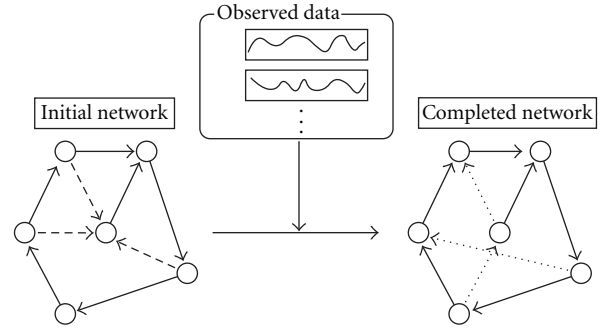


Figure 1: Network completion by addition and deletion of edges. Dashed edges and dotted edges denote deleted edges and added edges, respectively.

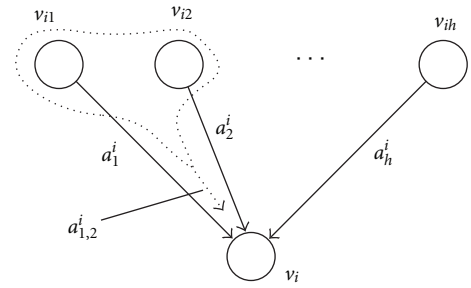


Figure 2: Dynamics model for a node.

DPLSQ consists of two parts: (i) parameter estimation and (ii) network structure inference. We employ least-squares fitting for the former part and dynamic programming for the latter part. Furthermore, there are three variants on the latter parts: (a) completion by addition of edges, (b) completion by deletion of edges, and (c) completion by addition and deletion of edges. Although the last case includes the first and second cases, we explain all of these for the sake of simplicity of explanation.

2.1. Model of Differential Equation and Estimation of Parameters. We assume that dynamics of each node v_i is determined by a differential equation:

$$\frac{dx_i}{dt} = a^i_0 + \sum_{j=1}^{h} a^i_j x_{i_j} + \sum_{j<k} a^i_{j,k} x_{i_j} x_{i_k} + b^i \omega, \tag{2}$$

where $v_{i_1}, \ldots, v_{i_h}$ are incoming nodes to v_i, x_i corresponds to the expression value of the ith gene, and ω denotes a random noise. The second and third terms of the right-hand side of the equation represent linear and nonlinear effects to node v_i, respectively (see Figure 2), where positive a^i_j or $a^i_{j,k}$ corresponds to an activation effect and negative a^i_j or $a^i_{j,k}$ corresponds to an inhibition effect.

In practice, we replace derivative by difference and ignore the noise term as follows:

$$x_i(t+1) = x_i(t) + \Delta t \left(a^i_0 + \sum_{j=1}^{h} a^i_j x_{i_j}(t) + \sum_{j<k} a^i_{j,k} x_{i_j}(t) x_{i_k}(t) \right), \tag{3}$$

where Δt denotes the time step.

We assume that time series data $y_i(t)$s, which correspond to $x_i(t)$s, are given for $t = 0, 1, \ldots, m$. Then, we can use the standard least-squares fitting method to estimate the parameters a^i_js and $a^i_{j,k}$s.

In applying the least-squares fitting method, we minimize the following objective function:

$$S^i_{i_1,i_2,\ldots,i_h} = \sum_{t=1}^{m} \left| y_i(t+1) - \left[y_i(t) + \Delta t \left(a^i_0 + \sum_{j=1}^{h} a^i_j y_{i_j}(t) + \sum_{j<k} a^i_{j,k} y_{i_j}(t) y_{i_k}(t) \right) \right] \right|^2. \tag{4}$$

2.2. Completion by Addition of Edges. In this subsection, we consider the problem of adding k edges in total so that the sum of least-squares errors is minimized.

Let $\sigma^+_{k_j,j}$ denote the minimum least-squares error when adding k_j edges to the jth node, which is formally defined by

$$\sigma^+_{k_j,j} = \min_{j_1,j_2,\ldots,j_{k_j}} S^j_{j_1,j_2,\ldots,j_{k_j}}, \tag{5}$$

where each v_{j_l} must be selected from $V - v_j - e^-(v_j)$. In order to avoid combinatorial explosion, we constrain the maximum k to be a small constant K and let $\sigma^+_{k_j,j} = +\infty$ for $k_j > K$ or $k_j + \deg^-(v_j) \geq n$. Then, the problem is stated as

$$\min_{k_1+k_2+\cdots+k_n=k} \sum_{j=1}^{n} \sigma^+_{k_j,j}. \tag{6}$$

Here, we define $D^+[k,i]$ by

$$D^+[k,i] = \min_{k_1+k_2+\cdots+k_i=k} \sum_{j=1}^{i} \sigma^+_{k_j,j}. \tag{7}$$

Then, $D^+[k,n]$ is the objective value (i.e., the minimum of the sum of least-squares errors when adding k edges).

The entries of $D^+[k,j]$ can be computed by the following dynamic programming algorithm:

$$\begin{aligned} D^+[k,1] &= \sigma^+_{k,1}, \\ D^+[k,j+1] &= \min_{k'+k''=k} \left\{ D^+[k',j] + \sigma^+_{k'',j+1} \right\}. \end{aligned} \tag{8}$$

It is to be noted that $D^+[k,n]$ is determined uniquely regardless of the ordering of nodes in the network. The correctness of this dynamic programming algorithm can be seen by

$$\begin{aligned} \min_{k_1+k_2+\cdots+k_n=k} \sum_{j=1}^{n} \sigma^+_{k_j,j} &= \min_{k'+k''=k} \left\{ \min_{k_1+k_2+\cdots+k_{n-1}=k'} \sum_{j=1}^{n-1} \sigma^+_{k_j,j} + \sigma^+_{k'',n} \right\} \\ &= \min_{k'+k''=k} D^+[k',n-1] + \sigma^+_{k'',n}. \end{aligned} \tag{9}$$

2.3. Completion by Deletion of Edges. In the above, we considered network completion by addition of edges. Here, we consider the problem of deleting h edges in total so that the sum of least-squares errors is minimized.

Let $\sigma^-_{h_j,j}$ denote the minimum least-squares error when deleting h_j edges from the set $e^-(v)$ of incoming edges to v_j. As in Section 2.2, we constrain the maximum h_j to be a small constant H and let $\sigma^-_{h_j,j} = +\infty$ if $h_j > H$ or $\deg^-(v_j) - h_j < 0$. Then, the problem is stated as

$$\min_{h_1+h_2+\cdots+h_n=h} \sum_{j=1}^{n} \sigma^-_{k_j,j}. \tag{10}$$

Here, we define $D^-[k,i]$ by

$$D^-[k,i] = \min_{k_1+k_2+\cdots+k_i=k} \sum_{j=1}^{i} \sigma^-_{k_j,j}. \tag{11}$$

Then, we can solve network completion by deletion of edges using the following dynamic programming algorithm:

$$\begin{aligned} D^-[k,1] &= \sigma^-_{k,1}, \\ D^-[k,j+1] &= \min_{k'+k''=k} \left\{ D^-[k',j] + \sigma^-_{k'',j+1} \right\}. \end{aligned} \tag{12}$$

2.4. Completion by Addition and Deletion of Edges. We can combine the above two methods into network completion by addition and deletion of edges.

Let $\sigma_{h_j,k_j,j}$ denote the minimum least-squares error when deleting h_j edges from $e^-(v_j)$ and adding k_j edges to $e^-(v_j)$ where deleted and added edges must be disjoint. We constrain the maximum h_j and k_j to be small constants H and K. We let $\sigma_{h_j,k_j,j} = +\infty$ if $h_j > H$, $k_j > K$, $k_j - h_j + \deg^-(v_j) \geq n$, or $k_j - h_j + \deg^-(v_j) < 0$ holds. Then, the problem is stated as

$$\min_{\substack{h_1+h_2+\cdots+h_n=h \\ k_1+k_2+\cdots+k_n=k}} \sum_{j=1}^{n} \sigma_{h_j,k_j,j}. \tag{13}$$

Here, we define $D[h,k,i]$ by

$$D[h,k,i] \min_{\substack{h_1+h_2+\cdots+h_i=h \\ k_1+k_2+\cdots+k_i=k}} \sum_{j=1}^{i} \sigma_{h_j,k_j,j}. \tag{14}$$

Then, we can solve network completion by addition and deletion of edges using the following dynamic programming algorithm:

$$\begin{aligned} D[h,k,1] &= \sigma_{h,k,1}, \\ D[h,k,j+1] &= \min_{\substack{h'+h''=h \\ k'+k''=k}} \left\{ D[h',k',j] + \sigma_{h'',k'',j+1} \right\}. \end{aligned} \tag{15}$$

2.5. Time Complexity Analysis. In this subsection, we analyze the time complexity of DPLSQ. Since completion by addition of edges and completion by deletion of edges are special cases

of completion by addition and deletion of edges, we focus on completion by addition and deletion of edges.

First, we analyze the time complexity required per least-squares fitting. It is known that least-squares fitting for linear systems can be done in $O(mp^2 + p^3)$ time where m is the number of data points and p is the number of parameters. Since our model has $O(n^2)$ parameters, the time complexity is $O(mn^4+n^6)$. However, if we can assume that the maximum indegree in a given network is bounded by a constant, the number of parameters is bounded by a constant, where we have already assumed that H and K are constants. In this case, the time complexity for least-squares fitting can be estimated as $O(m)$.

Next, we analyze the time complexity required for computing $\sigma_{h_j,k_j,j}$. In this computation, we need to examine combinations of deletions of h_j edges and additions of k_j edges. Since h_j and k_j are, respectively, bounded by constants H and K, the number of combinations is $O(n^{H+K})$. Therefore, the computation time required per $\sigma_{h_j,k_j,j}$ is $O(n^{H+K}(mn^4 + n^6))$ including the time for least-squares fitting. Since we need to compute $\sigma_{h_j,k_j,j}$ for $H \times K \times n$ combinations, the total time required for computation of $\sigma_{h_j,k_j,j}$s is $O(n^{H+K+1}(mn^4 + n^6))$.

Finally, we analyze the time complexity required for computing $D[h,k,i]$s. We note that the size of table $D[h,k,i]$ is $O(n^3)$, where we are assuming that h and k are $O(n)$. In order to compute the minimum value for each entry in the dynamic programming procedure, we need to examine $(H + 1)(K + 1)$ combinations, which is $O(1)$. Therefore, the computation time required for computing $D[h,k,i]$s is $O(n^3)$. Since this value is clearly smaller than the one for $\sigma_{h_j,k_j,j}$s, the total time complexity is

$$O\left(n^{H+K+1} \cdot (mn^4 + n^6)\right). \tag{16}$$

Although this value is too high, it can be significantly reduced if we can assume that the maximum degree of an input network is bounded by a constant. In this case, the least-squares fitting can be done in $O(m)$ time per execution. Furthermore, the number of combinations of deleting at most h_j edges is bounded by a constant. Therefore, the time complexity required for computing $\sigma_{h_j,k_j,j}$s is reduced to $O(mn^{K+1})$. Since the time complexity for computing $D[h,k,i]$s remains $O(n^3)$, the total time complexity is

$$O\left(mn^{K+1} + n^3\right). \tag{17}$$

This number is allowable in practice if $K \leq 2$ and n is not too large (e.g., $n \leq 100$).

3. Results

We performed computational experiments using both artificial data and real data. All experiments on DPLSQ were performed on a PC with Intel Core i7-2630QM CPU (2.00 GHz) with 8 GB RAM running under the Cygwin on Windows 7. We employed the liblsq library (http://www2.nict.go.jp/aeri/sts/stmg/K5/VSSP/install_lsq.html) for a least-squares fitting method.

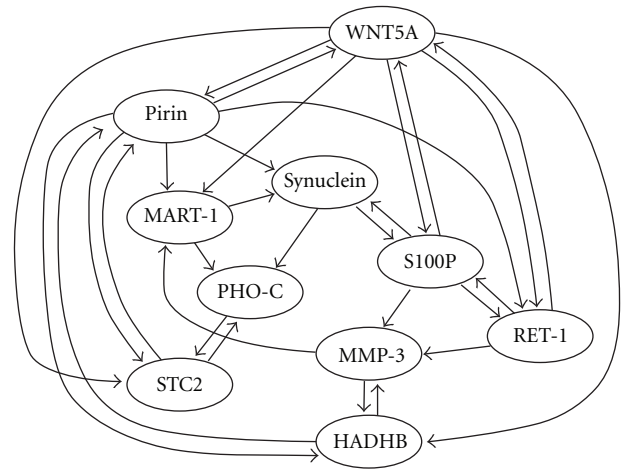


Figure 3: Structure of WNT5A network [17].

3.1. Completion Using Artificial Data. In order to evaluate the potential effectiveness of DPLSQ, we began with network completion using artificial data. To our knowledge, there is no available tool that performs the same task. Although some of the existing inference methods employ incremental modifications of networks, the number of added/deleted edges cannot be specified. Therefore, we did not compare DPLSQ for network completion with other methods (but we compared it with the existing tools for network inference).

We employed the structure of the real biological network named WNT5A (see Figure 3) [17]. For each node v_i with h input nodes, we considered the following model:

$$x_i(t+1) = x_i(t) + \Delta t\left(a_0^i + \sum_{j=1}^{h} a_j^i x_{i_j} + \sum_{j<k} a_{j,k}^i x_{i_j}(t)x_{i_k}(t) + b_i\omega\right), \tag{18}$$

where a_j^is and $a_{j,k}^i$s are constants selected uniformly at random from $[-1,1]$ and $[-0.5,0.5]$, respectively. The reason why the domain of $a_{j,k}^i$s is smaller than that for a_j^is is that non-linear terms are not considered as strong as linear terms. It should also be noted that $b_i\omega$ is a stochastic term, where b_i is a constant (we used $b_i = 0.2$ in all computational experiments) and ω is a random noise taken uniformly at random from $[-1,1]$.

For artificial generation of observed data $y_i(t)$, we used

$$y_i(t) = x_i(t) + o^i\epsilon, \tag{19}$$

where o^i is a constant denoting the level of observation errors and ϵ is a random noise taken uniformly at random from $[1,-1]$. Since the use of time series data beginning from only one set of initial values easily resulted in overfitting, we generated time series data beginning from 20 sets of initial values taken uniformly at random from $[1,-1]$, where the number of time points for each set was set to 10 and $\Delta t = 0.2$ was used as the period between the consecutive two time points. Therefore, 20 sets of time series data, each of which consisted of 10 time points, were used per trial (200 time points were used in total per trial). It is to be noted that in

our preliminary experiments, the use of too small Δt resulted in too small changes of expression values whereas the use of large Δt resulted in divergence of time series data. Therefore, after some trials, $\Delta t = 0.2$ was selected and used throughout the paper.

Under the above model, we examined several o^is as shown in Table 1. In order to examine network completion, WNT5A was modified by randomly adding h edges and deleting k edges and the resulting network was given as an initial network.

We evaluated the performance of the method in terms of the accuracy of the modified edges and the success rate. The accuracy is defined here by

$$\frac{h + k + \left|E_{\text{orig}}\right| - \left|E_{\text{orig}} \cap E_{\text{cmpl}}\right|}{h + k}, \tag{20}$$

where E_{orig} and E_{cmpl} are the sets of edges in the original network and the completed network, respectively. This value takes 1 if all deleted and added edges are correct and 0 if none of the deleted and added edges is correct. For each (h, k), we took the average accuracy over a combination of 10 parameters (a^i_js and $a^i_{j,k}$s) and 10 random modifications (i.e., addition of h edges and deletion of k edges to construct an initial network). The success rate is the frequency of the trials (among 10×10 trials) in which the original network was correctly obtained by network completion. The result is shown in Table 1. It is seen from this table that DPLSQ works well if the observation error level is small. It is also seen that the accuracies are high in the case of $h = 0$. However, no clear trend can be observed on a relationship between h, k values and the accuracies. It is reasonable because we evaluated the result in terms of the accuracy per deleted/added edge. On the other hand, it is seen that the success rate decreases considerably as h and k increase or the observation error level increases. This dependence on h and k is reasonable because the probability of having at least one wrong edge increases as the number of edges to be deleted and added increases. As for the computation time, the CPU time for each trial was within a few seconds, where we used the default values of $H = K = 3$. Although these default values were larger than h, k here, it did not cause any effects on the accuracy or the success rate. How to choose H and K is not a trivial problem. As discussed in Section 2.5, we cannot choose large H or K because of the time complexity issue. Therefore, it might be better in practice to examine several combinations of small values H and K and select the best result although how to determine the best result is left as another issue.

3.2. Inference Using Artificial Data. We also examined DPLSQ for network inference, using artificially generated time series data. In this case, we used the same network and dynamics model as previously mentioned but we let $E = \emptyset$ in the initial network. Since the method was applied to inference, we let $H = 0$, $K = 3$, and $k = 30$. It is to be noted that $\deg^-(v_i) = 3$ holds for all nodes v_i in the WNT5A network. Furthermore, in order to examine how CPU time changes as the size of the network grows, we made networks with 30 genes and 50 genes (with $k = 90$ and $k = 150$) by making 3 and 5 copies of the original networks, respectively.

Since the number of added edges was always equal to the number of edges in the original network, we evaluated the results by the average accuracy, which was defined as the ratio of the number of correctly inferred edges to the number of edges in the correct network (i.e., the number of added edges). We examined observation error levels of 0.1, 0.3, 0.5, and 0.7, for each of which we took the average over 10 trials using randomly generated different parameter values (i.e., a^i_js and $a^i_{j,k}$s), where time series data were generated as in Section 3.1. The result is shown in Table 2, where the accuracy and the average CPU time (user time + sys time) per trial are shown for each case. It is seen from the table that the accuracy is high even for large networks if the error level is not high. It is also seen that although the CPU time grows rapidly as the size of a network increases, it is still allowable for networks with 50 genes.

We also compared DPLSQ with two well-known existing tools for inference of genetic networks, ARACNE [11, 12] and GeneNet [7, 8]. The former is based on mutual information and the latter is based on graphical Gaussian models and partial correlations. Computational experiments on ARACNE were performed under the same environment as that for DPLSQ, whereas those on GeneNet were performed on a PC with Intel Core i7-2600 CPU (3.40 GHz) with 16 GB RAM running under the Cygwin on Windows 7 because of the availability of the R platform on which GeneNet works. We employed datasets that were generated in the same way as for DPLSQ and default parameter settings for both tools.

Since both tools output undirected edges along with their significance values (or their probabilities), we selected the top M edges in the output where M was the number of edges in the original network and regarded $\{v_i, v_j\}$ as a correct edge if either (v_i, v_j) or (v_j, v_i) was included in the edge set of the original network. As in Table 2, we evaluated the results by the average accuracy, that is, the ratio of the number of correctly inferred edges to the number of edges in the original network.

The result is shown in Table 3. Interestingly, both tools have similar performances. It is also interesting that the performance does not change much in each method even if the level of observation error changes. Readers may think that the accuracies shown in Table 3 are close to those by random prediction. However, these accuracies were much higher than those obtained by assigning random probabilities to edges, and thus we can mention that these tools outputted meaningful results.

It is seen from Tables 2 and 3 that the accuracies by DPLSQ are much higher than those by ARACNE and GeneNet even though both directions of edges are taken into account for ARACNE and GeneNet. However, it should be noted that time series data were generated according to the differential equation model on which DPLSQ relies. Therefore, we can only mention that DPLSQ works well if time series data are generated according to appropriate differential equation models. It is to be noted that we can use

Table 1: Result on completion of WNT5A network, where the average accuracy is shown for each case.

No. deleted edges	No. added edges		Observation error level			
			0.1	0.3	0.5	0.7
$h = 0$	$k = 1$	Accuracy	0.990	0.910	0.730	0.410
		Success rate	0.99	0.91	0.73	0.41
$h = 0$	$k = 2$	Accuracy	1.000	0.955	0.670	0.395
		Success rate	1.00	0.91	0.42	0.17
$h = 1$	$k = 0$	Accuracy	0.990	0.850	0.470	0.240
		Success rate	0.99	0.85	0.47	0.24
$h = 1$	$k = 1$	Accuracy	0.995	0.845	0.405	0.210
		Success rate	0.99	0.71	0.11	0.02
$h = 1$	$k = 2$	Accuracy	0.983	0.843	0.470	0.190
		Success rate	0.95	0.58	0.11	0.00
$h = 2$	$k = 0$	Accuracy	1.000	0.795	0.440	0.215
		Success rate	1.00	0.67	0.18	0.01
$h = 2$	$k = 1$	Accuracy	0.996	0.833	0.453	0.223
		Success rate	0.99	0.53	0.05	0.01
$h = 2$	$k = 2$	Accuracy	1.000	0.862	0.517	0.285
		Success rate	1.00	0.56	0.03	0.01

Table 2: Result on inference of WNT5A network by DPLSQ.

		Observation error level			
		0.1	0.3	0.5	0.7
$n = 10$	Accuracy	1.000	0.966	0.803	0.620
	CPU time (sec.)	0.685	0.682	0.682	0.685
$n = 30$	Accuracy	0.995	0.914	0.663	0.443
	CPU time (sec.)	66.2	66.2	66.1	65.9
$n = 50$	Accuracy	0.999	0.913	0.613	0.392
	CPU time (sec.)	534.0	534.2	533.6	533.5

Table 3: Result on inference of WNT5A network using ARACNE and GeneNet, where the accuracy is shown for each case.

	Method	Observation error level			
		0.1	0.3	0.5	0.7
$n = 10$	ARACNE	0.523	0.523	0.523	0.526
	GeneNet	0.526	0.526	0.533	0.533
$n = 30$	ARACNE	0.332	0.328	0.326	0.326
	GeneNet	0.368	0.380	0.383	0.384
$n = 50$	ARACNE	0.308	0.312	0.310	0.391
	GeneNet	0.313	0.316	0.314	0.316

other differential equation models as long as parameters can be estimated by least-squares fitting.

As for computation time, both methods were much faster than DPLSQ. Even for the case of $N = 50$, each of ARACNE and GeneNet worked in less than a few seconds per trial. Therefore, DPLSQ does not have merits on practical computation time.

3.3. Inference Using Real Data. We also examined DPLSQ for inference of genetic networks using real gene expression data. Since there is no gold standard on genetic networks and thus we cannot know the correct answers, we did not compare it with the existing methods.

We employed a part of the cell cycle network of *Saccharomyces cerevisiae* extracted from the KEGG database [18], which is shown in Figure 4. Although the detailed mechanism of the cell cycle network is still unclear, we used this network as the correct answer, which may not be true. Although each of (MCM1, YOX1, YHP1), (SWI4, SWI6), (CLN3, CDC28), (MBP1, SWI6) constitutes a protein complex, we treated them separately and ignored the interactions inside a complex because making a protein complex does not necessarily mean a regulation relationship between the corresponding genes.

As for time series data of gene expression, we employed four sets of times series data (alpha, cdc15, cdc28, elu) in [19] that were obtained by four different experiments. Since there were several missing values in the time series data, these values were filled by linear interpolation and data on some endpoint time points were discarded because of too many missing values. As a result, alpha, cdc15, cdc28, and elu datasets consist of gene expression data of 18, 24, 11, and 14 time points, respectively. In order to examine a relationship between the number of time points, and accuracy, we examined four combinations of datasets as shown in Table 4. We evaluated the performance of DPLSQ by means of the accuracy (i.e., the ratio of the number of correctly inferred edges to the number of added edges), where $K = 3$ and $k = 25$ were used. The result is shown in Table 4.

Since the total number of edges in both the original network and the inferred networks is 25 and there exist

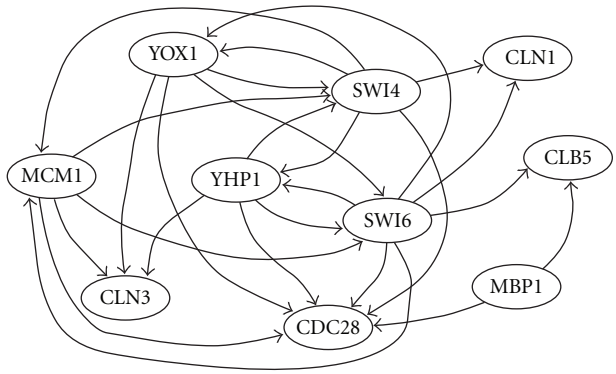


Figure 4: Structure of part of yeast cell cycle network.

$9 \times 10 = 90$ possible edges (excluding self loops), the expected number of corrected edges is roughly estimated as

$$\frac{25}{90} \times 25 = 6.944\ldots, \tag{21}$$

if 25 edges are randomly selected and added. Therefore, the result shown in Table 4 suggests that DPLSQ can do much better than random inference when appropriate datasets are provided (e.g., cdc15 only or cdc15+cdc28+alpha+elu). It is a bit strange that the accuracies for the first and last datasets are better than those for the second and third datasets because it is usually expected that adding more evidences results in more accurate networks. The reason may be that time series of cdc28 and alpha may contain larger measurement errors than those of cdc15 and elu, or some regulation rules that are hidden in Figure 4 may be activated under the conditions of cdc28 and/or alpha.

4. Conclusion

In this paper, we have proposed a network completion method, DPLSQ, using dynamic programming and least-squares fitting based on our previously proposed methodology of network completion [14]. As mentioned in Section 1, network completion is to make the minimum amount of modifications to a given network so that the resulting network is (most) consistent with the observed data. In our previous model [14], we employed the Boolean network as a model of networks and assumed that only expression or other values of one or a few nodes are observed. However, in this paper, we assumed that expression values of all nodes are observed, which correspond to gene microarray data, and regulation rules are given in the form of differential equations. The most important theoretical difference between this model and our previous model is that network completion can be done in polynomial time if the maximum indegree is bounded by a constant in this model whereas it is NP-hard in our previous model even if the maximum indegree is bounded by a constant. This difference arises not from the introduction of a least-squares fitting method but from the assumption that expression values of all nodes are observed.

It should also be noted that the optimality of the solution is not guaranteed in most of the existing methods for inference of genetic networks, whereas it is guaranteed in DPLSQ if it is applied to inference of a genetic network with a bounded maximum indegree. Of course, the objective function (i.e., minimizing the sum of squared errors) is different from existing ones, and thus this property does not necessarily mean that DPLSQ is superior to existing methods in real applications. Indeed, the result using real gene expression data in Section 3.3 does not seem to be very good. However, DPLSQ has much room for extensions. For example, least-squares fitting can be replaced by another fitting/regression method (with some regularization term) and the objective function can be replaced by another function as long as it can be computed by sum or product of some error terms. Studies on such extensions might lead to development of better network completion and/or inference methods.

Table 4: Result on inference of a yeast cell cycle network.

Experimental conditions	Accuracy
cdc15	11/25
cdc15 + cdc28	8/25
cdc15 + cdc28 + alpha	8/25
cdc15 + cdc28 + alpha + elu	11/25

Acknowledgments

T. Akutsu was partially supported by JSPS, Japan (Grants-in-Aid 22240009 and 22650045). T. Tamura was partially supported by JSPS, Japan (Grant-in-Aid for Young Scientists (B) 23700017). K. Horimoto was partially supported by the Chinese Academy of Sciences Visiting Professorship for Senior International Scientists Grant no. 2012T1S0012.

References

[1] S. Liang, S. Fuhrman, and R. Somogyi, "REVEAL, a general reverse engineering algorithm for inference of genetic network architectures," in *Proceedings of the Pacific Symposium on Biocomputing*, vol. 3, pp. 18–29, 1998.

[2] T. Akutsu, S. Miyano, and S. Kuhara, "Inferring qualitative relations in genetic networks and metabolic pathways," *Bioinformatics*, vol. 16, no. 8, pp. 727–734, 2000.

[3] N. Friedman, M. Linial, I. Nachman, and D. Pe'er, "Using Bayesian networks to analyze expression data," *Journal of Computational Biology*, vol. 7, no. 3-4, pp. 601–620, 2000.

[4] S. Imoto, S. Kim, T. Goto et al., "Bayesian network and nonparametric heteroscedastic regression for nonlinear modeling of genetic network," *Journal of Bioinformatics and Computational Biology*, vol. 1, no. 2, pp. 231–252, 2003.

[5] T. F. Liu, W. K. Sung, and A. Mittal, "Learning gene network using time-delayed Bayesian network," *International Journal on Artificial Intelligence Tools*, vol. 15, no. 3, pp. 353–370, 2006.

[6] H. Toh and K. Horimoto, "Inference of a genetic network by a combined approach of cluster analysis and graphical Gaussian modeling," *Bioinformatics*, vol. 18, no. 2, pp. 287–297, 2002.

[7] R. Opgen-Rhein and K. Strimmer, "Inferring gene dependency networks from genomic longitudinal data: a functional data approach," *RevStat*, vol. 4, no. 1, pp. 53–65, 2006.

[8] R. Opgen-Rhein and K. Strimmer, "From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data," *BMC Systems Biology*, vol. 1, article 37, 2007.

[9] P. D'Haeseleer, S. Liang, and R. Somogyi, "Genetic network inference: from co-expression clustering to reverse engineering," *Bioinformatics*, vol. 16, no. 8, pp. 707–726, 2000.

[10] Y. Wang, T. Joshi, X. S. Zhang, D. Xu, and L. Chen, "Inferring gene regulatory networks from multiple microarray datasets," *Bioinformatics*, vol. 22, no. 19, pp. 2413–2420, 2006.

[11] A. A. Margolin, K. Wang, W. K. Lim, M. Kustagi, I. Nemenman, and A. Califano, "Reverse engineering cellular networks," *Nature Protocols*, vol. 1, no. 2, pp. 662–671, 2006.

[12] A. A. Margolin, I. Nemenman, K. Basso et al., "ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context," *BMC Bioinformatics*, vol. 8, supplement 1, no. 1, article S7, 2006.

[13] S. Kimura, S. Nakayama, and M. Hatakeyama, "Genetic network inference as a series of discrimination tasks," *Bioinformatics*, vol. 25, no. 7, pp. 918–925, 2009.

[14] T. Akutsu, T. Tamura, and K. Horimoto, "Completing networks using observed data," *Lecture Notes in Artificial Intelligence*, vol. 5809, pp. 126–140, 2009.

[15] S. A. Kauffman, *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press, New York, NY, USA, 1993.

[16] T. Tamura, Y. Yamanishi, M. Tanabe et al., "Integer programming-based method for completing signaling pathways and its application to analysis of colorectal cancer," *Genome Informatics*, vol. 24, pp. 193–203, 2010.

[17] S. Kim, H. Li, E. R. Dougherty et al., "Can Markov chain models mimic biological regulation?" *Journal of Biological Systems*, vol. 10, no. 4, pp. 337–357, 2002.

[18] M. Kanehisa, S. Goto, M. Furumichi, M. Tanabe, and M. Hirakawa, "KEGG for representation and analysis of molecular networks involving diseases and drugs," *Nucleic Acids Research*, vol. 38, no. 1, Article ID gkp896, pp. D355–D360, 2009.

[19] P. T. Spellman, G. Sherlock, M. Q. Zhang et al., "Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization," *Molecular Biology of the Cell*, vol. 9, no. 12, pp. 3273–3297, 1998.