Computational Methods for Deciphering Genomic Structures in Prokaryotes

by

Dongsheng Che

(Under the direction of Ying Xu and Liming Cai)

Abstract

High-throughput sequencing technologies have generated huge amounts of genomic data. This wealth of genomic data provides computational biologists unprecedented opportunities to unveil the biological machinery encoded in genomes. Characterizing the structure of genomes is an important and challenging task; it is an essential step towards deciphering the networks and pathways in a biological system.

The characterization of microbial genomic structures includes: (1) identifying neighboring genes that are co-transcribed (also known as operons); (2) identifying groups of operons with evolutionary relationships (also known as uber-operons); and, (3) elucidating higher level structures that share common regulatory controls, including protein-DNA binding events and *cis*-regulatory elements among operons (also known as regulons). The primary goal of this thesis is to develop computational methods for elucidating the above three categories of genomic structures in prokaryotes.

UNIPOP, a maximum bipartite matching-based algorithm, is designed and implemented to predict operon structures of any prokaryotic genome, without relying on experimental data or training data. The prediction accuracy of UNIPOP is shown to be superior to most other operon predictors when evaluating two well-studied organisms.

The evolutionary relationships among operons are elucidated by using comparative genomic data and a maximum matching-based algorithm. The comparative study of uber-operons and regulons has shown that they are highly related, indicating the effectiveness of using uber-operons for predicting regulons.

With the availability of predicted operons, we propose an approach, phylogenetic footprinting for prokaryotes, to study *cis* regulatory motifs in the promoter regions of operons. By integrating the motif data with uber-operon data, and formulating it as a graph partitioning problem, we predicted regulons in *Escherichia coli* K12. Different sources of validation have shown that our predicted regulons were consistent with the data of known regulons, functional relatedness and expression data. More importantly, we have also derived some novel regulons which were biologically meaningful.

In summary, we predict different levels of genomic structures by developing novel graph-theoretic based algorithms and using comparative genomic analysis. Our methods are universally applicable to all sequenced microbial genomes, and outperform most of the other published methods in terms of prediction accuracy. Our prediction tools can provide assistance in understanding the machinery of gene regulation, biological networks and pathways.

INDEX WORDS: Bipartite graphs, Maximum matching, Genomic structures, Prokaryotes, Operons, Regulons, Uber-operons, Motif

Computational Methods for Deciphering Genomic Structures in Prokaryotes

by

Dongsheng Che

B.Ag., Zhejiang Forestry College, P.R. China, 1992

M.S., Beijing Forestry University, P.R. China, 1995

M.S., The University of Georgia, 2000

M.S., The University of Georgia, 2002

A Dissertation Submitted to the Graduate Faculty

of The University of Georgia in Partial Fulfillment

of the

Requirements for the Degree

Doctor of Philosophy

Athens, Georgia

2008

Computational Methods for Deciphering Genomic Structures in Prokaryotes

by

Dongsheng Che

Approved:

Major Professors: Ying Xu
Liming Cai

Committee: Khaled Rasheed
Robert W. Robinson

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
August 2008

Dedication

To my beloved, late parents.

Acknowledgments

I would like to express sincere thanks to my major professor, Dr. Ying Xu, for his friendly advice, honest criticism and endless patience throughout my program study. Without his insight and guidance, completion of this thesis would have been impossible. In addition, staying in a top-notch lab benefits me a lot in various aspects, including interacting people with different backgrounds, broadening my research vision for my future career.

I would also like to thank my co-advisor, Dr. Liming Cai, for his continuous guidance, help and support in various aspects. It is Dr. Cai introduced me to this exciting interdisciplinary field, encouraged me to explore the field and discover new problems.

I would also like to thank my other committee members, Dr. Robert Robinson and Dr. Khaled Rasheed. Dr. Robinson brought me the field of graph theory, while Dr. Rasheed brought me to the field of machine learning.

Special gratitude must go to Dr. Guojun Li. Many ideas and algorithms related to this work were proposed by Dr. Li, though he is not served in my committee.

In addition, I would like to thank various members of the Computational System Biology Lab at the University of Georgia, including Ms. Joan Yantko, Dr. Phuongan Dam, Dr. Hongwei Wu, Dr. Fenglou Mao, and the members in RNA-Informatics group, particular Dr. Russell Malmberg, Dr. Fangfang Pan, Dr. Yinglei Song and Dr. Jizhen Zhao.

The work is, in part, supported by :

- The National Science Foundation (DBI-0354771/DBI-0542119/ITR-IIS-0407204/CCF-0621700)
- A Distinguished Cancer Scholar grant from the Georgia Cancer Coalition

- The U.S. Department of Energy's BioEnergy Science Center (BESC) grant through the Office of Biological and Environmental Research

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction and Literature Review

A genome of an organism is a deoxyribonucleic acid (DNA) sequence with the building blocks of four nucleotides. Each nucleotide is made of a phosphate, a sugar, and an organic base known as A, C, G, or T. DNA usually consists of two strands of nucleotides, where the base pairs of A-T and C-G from two strands form and twisted into a double helix. The whole genome contains many segments of functional sequences, known as *genes*, which will be transcribed into messenger ribonucleic acid (mRNA).

The transcription process from DNA to mRNA is a very complicated process. RNA polymerase keeps moving on the DNA sequence until it identifies a region called '*promoter*', which is usually located on the upstream of a gene. The recognition of the promoter region initiates the transcription. Usually, the initiation of the transcription process is regulated by other factors, such as *transcription factors* (TFs) or RNA molecules. The regulated region of DNA by such factors is known as *operator*. Finally, the transcription process is terminated at the terminator region.

In prokaryotic genomes, about half genes have its own promoter, operator and terminator region, and collectively they are called *single-gene operons*. Some other genes, however, share a common promoter, operator and terminator, and they are known as *multi-gene operons* (as shown in Figure 1.1). In practice, biologists are mainly interested in which genes belong to the same operon. Therefore, when we talk about single-gene operons or multiple-gene operons in this thesis, we only mean the structural genes within an operon.

Biological experiments, such as Northern blot, or reverse transcription polymerase chain reaction (RT-PCR), have determined several hundred operons, which are mainly restricted

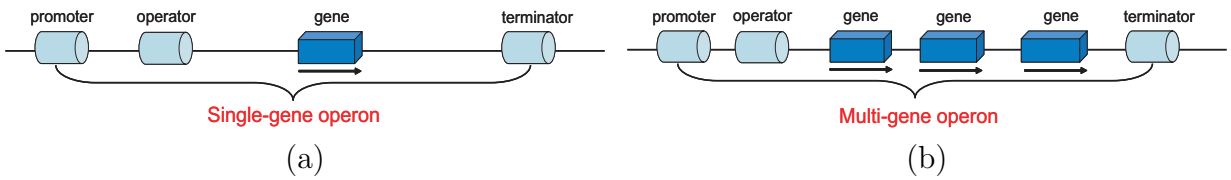


Figure 1.1: Two types of operon structures: (a) single-gene operon and (b) multiple-gene operon

in a few well-studied organisms, such as *Escherichia coli* and *Bacillus subtilis*. These experimentally verified operons can be accessed in operon databases, such as RegulonDB for *E. coli* [57], and BSORF for *B. subtilis* (`http://Bacillus.genome.jp/`).

The determination of operon structures can be used for higher order genomic structure studies. By comparing operon structures across multiple genomes, Lathe *et al.* [94] discovered that many operons were not conserved across multiple genomes, but interestingly combined operon sets were. The set of evolutionary related operons was defined as *uber-operon* by Lathe *et al.* [94].

Another higher order genomic structures constructed by operons are called a *regulon*, which is a set of operons regulated by the same TF (see Figure 1.2). As we described before, the transcription of a gene (or an operon) is usually regulated by one or more TF(s). By binding on the regulatory regions (*i.e.*, operator), usually located in the upstream sequences of genes, TFs can either induce or repress the transcription of genes.

Biological experiments have identified a few hundred TFs and their co-regulated operons. Some regulons, such as Crp-associated, contain a few hundred operons, while other regulons, such as SoxR, may contain a few operons. Currently, there are several regulon databases (*e.g.*, regulonDB, PRODORIC [113]) containing a small number of regulons from well-studied organisms, such as *E. coli* and *B. subtilis*.

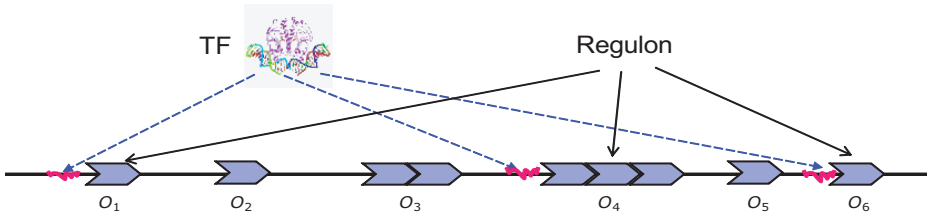


Figure 1.2: An illustrative example of a regulon

The determination of genomic structures, including operons, uber-operons and regulons, is very important. This is especially needed for new sequenced microbial genomes whose information has not been established. The determination of operons will help biologists to understand higher-level genomic structures, including uber-operon, regulon and metabolic pathway. On the other hand, the elucidation of uber-operon and regulon structures will be beneficial to the understanding the biological pathway, and the machinery of regulatory networks.

While genomic structures can be determined by experiments, this method is generally expensive and time-consuming. To date, experimentally verified genomic structures of microbes are limited in a few organisms, and they only represent a small fraction of the whole genome. It is obvious that there is a huge gap between a few known genomic structures and several hundred completely sequenced genomes ($\sim$ 700 as of June 2008). Actually, this problem will become more serious as many more sequences will be completed in the near future. To bridge this gap, computational tools have been developed for the whole genome-scale prediction of genomic structures, using the features associated with known genomic structures.

The remainder of this chapter is organized as follows. In Sections 1.1-1.3, I will briefly overview the computational approaches developed for predicting operons, uber-operons, and regulons respectively. As our research is mainly about the use of bipartite graph matching

based algorithms to elucidate microbial genomic structures, bipartite matching algorithms and related biological applications will be reviewed in Section 1.4. We conclude in Section 1.5 the outline of this thesis.

1.1 Operon Prediction

The operon prediction problem is simply considered to be the partitioning of a genome into gene clusters, where all genes within the cluster share a promter, operator and terminator, or it can be simply treated as a classification problem, *i.e.*, determining whether an adjacent gene pair belongs to the same operon or not.

1.1.1 Feature Selection

Initial attempts to operon prediction were focused on identifying boundaries of operons, *i.e.*, promoter and terminator regions. In general, it is difficult to characterize promoter and terminator. The problem becomes much more complicated when the existence of internal promoter and terminator of some operons is taken into consideration. Recently, de Hoon *et al* [42] has found that the terminators of the phylum of Firmicutes consists of an inverted repeats followed by a stretch of thymine residues, which can be used to predict operons very accurately. Janga *et al.* [75] found that the oligonucleotide signatures of promoter regions are different from the upstream regions in the middle of operons. The application of trinucleotide signatures in operon prediction also shows a fairly high prediction in *E. coli*.

One of the most effective features for operon prediction was discovered by Salgado *et al.* [135]. They found that adjacent gene pairs within an operon (also known as *operonic gene pairs*) tend to have shorter intergenic distance, while gene pairs from two consecutive operons (also known as *non-operonic gene pairs*) tend to have longer distances. Due to the effectiveness of this feature, most of later operon predictors have incorporated the inter-genic distance information in their programs.

With the availability of multiple genome sequences, comparative genomic analysis seems to be also useful in predicting operons. Ermolaeva *et al.* [50] found that some neighbor genes are fully conserved, including both gene list and gene order, across multiple genomes. However, using the feature of full conservation to predict operon seems to have a very low sensitivity, as this scenario represents only a small fraction of operon conservation across multiple genomes.

Experimental data, such as microarray data or metabolic pathway data, may also be useful for operon prediction. For example, Sabatti *et al.* [134] used microarray expression data to evaluate operons. They found that adjacent operon gene pairs had higher gene expression correlation than non-operonic gene pairs. This is because the genes within an operon are co-transcribed, and thus share the same time-course or condition expression patterns. Zheng *et al.* [171] used metabolic pathway data to predict operons, but the coverage is usually low as reported in [19].

Various other features have been used for the inference of operons. For example, the more functional related of a pair of genes, measured by Cluster of Orthologous Genes (COG) [151], or Gene Ontology (GO) [4], the more probable the pair from the same operon [30, 126].

1.1.2 Prediction Methods

Numerous computational methods have been applied for operon prediction using various discovered features. We can roughly group them into two major categories, machine-learning based and statistical based approaches. The basic framework for machine-learning based approaches is to construct models using training set (*i.e.*, known operon data), and then to predict the remaining unknown dataset based on the trained models. The statistical based approaches, on the other hand, are to establish statistical models for evaluating and predict operons. The major approaches have been listed as follows.

- Hidden Markov Model (HMMs)

Inspired by the highly accurate prediction of genes using HMMs, Yada *et al.* [168] constructed HMMs based on known promoters, ribosomal binding sites, coding regions and terminators of *E. coli*. The prediction accuracy of this approach, however, was only 60%, mainly caused by the poor characterization of promoter and terminator.

- Naïve Bayesian

  Several independent groups have used this approach by incorporating multiple features. Operon Finding Software (OFS) [11] used intergenic distance, function relatedness (measured by GO) and conservation of gene cluster. Price *et al.* [126] used the features of intergenic distance, codon usage, gene neighborhood information and functional similarity (*i.e.*, COG). Each feature of their naïve Bayesian models is assumed to conditionally independent. The integrative score based on individual scores is used to evaluate whether a gene pair to be operonic or non-operonic.

- Bayesian Network

  Bockhorst *et al.* [13] trained the Basyesian network model using multiple features, including operon length, codon usage, gene spacing, microarray expression data, promoter and terminator. Unlike the naive Bayesian model, this model captures the dependencies among features.

- Neural Network

  Joint Prediction of Operons (JPOP) [30] is a neural network-based operon predictor, which uses inter-genic distance, COG function and phylogenetic profiles as input nodes. Related to this approach, Tran *et al.* [155] designed a neural network architecture which takes the prediction scores of gene pairs from three operon predictors (JPOP [30], OFS [163] and VIMSS [126]). The performance of this neural network model was better than any of other three predictors.

- Decision Tree

This approach proved to have high prediction accuracy when the decision-tree model trained from an individual organism is applied to the same organism [25, 40].

- Log-likelihood

  The likelihood of an operonic gene pair is derived by comparing the feature value of known operon gene pairs and those of non-operonic pairs. It seems to very effective to use this simple statistical model by relying on one feature, such as intergenic distance [135], and oligonucleotide signatures of promoter regions and operon junctions [75].

- Genetic Algorithm

  The main idea of the fuzzy guided genetic algorithm-based approach [73] is to encode the whole genome into an array of integers, where each integer represents a gene and each integer value labels the operon ID. For example, an individual of '11222' represents the genome with five genes, with the first two genes belong to the first operon and the remaining three belong to the second operon. The fitness function is evaluated based on four features: inter-genic distance, metabolic pathway data, gene order conservation across multiple genomes, and functional similarity.

- Graph matching-based

  Edwards *et al.* [46] formulated the operon identification problem by using a relaxed version of gene cluster conservation across multiple genomes. The idea is to seek conserved gene cluster of a target genome by identifying unordered groups of homologous genes from multiple reference genomes, with the assumption that gene clusters are in the same strand. A maximum bipartite matching based algorithm has been developed.

Most of these computational methods have been tested to perform well in terms of prediction accuracy. However, since experimentally verified operons were restricted only in a few organisms, *i.e.*, *E. coli* and *B. subtilis*, their predicting power are not clear when they are applied to other organisms. As we know, many features such as experimental data (*e.g.*,

microarray data), metabolic pathway, and functional assignment may not be available for newly sequenced genomes. Thus, those programs that need these features cannot be used to predict operons for these new sequenced genomes. Furthermore, some feature values, such as inter-genic distance distribution versus inter-operonic distance distribution, are genome-specific, resulting in substantial performance reduction when trained on one genome (*e.g.*, *E. coli*) and tested on another (*e.g.*, *B. subtilis*).

1.2 Uber-operon Prediction

Several computational approached have been developed since Lathe *et al.* [94] proposed the concept of uber-operon. Lathe *et al.* [94] was the first group to design an algorithm for predicting uber-operons through detecting conserved unions of operons across multiple bacterial genomes. Assuming that the orthologous gene relationships across are given, the algorithm starts with one gene and its orthologs chosen from a number of genomes, and determines the conserved gene neighbors of these orthologous genes. Gene neighbors are treated as conserved if more than three genomes have such neighborhood relationship. The algorithm repeats the process of identification of orthologous genes and their conserved neighbors until no new conserved neighbors are found.

Rogozin *et al.* [130] formulated the original problem as to be searching maximal trail in a digraph, by taking advantage of gene neighborhood conservation across multiple genomes. Briefly, orthologugous genes in compared genomes are first extracted from the COG database, and conserved gene pairs are obtained with the assumption that two genes are in the same direction and separated by at most two genes in at least three genomes, which are used for constructing a digraph. They developed a heuristic algorithm for searching the maximal trail.

Martin *et al.* [104] defined an uber-operon as a maximal set of operons across two genomes that share common homologous genes. The problem was formulated as to find connected components in a graph $G = (V, E, W)$, where V is a set of vertices representing genes that

has at least one homolog in the other genome, E is a set of edges representing gene pairs in the same operon, and W is a set of edge weights representing the occurrence of gene pairs in the same operon. Like other approaches, the orthologous relationships were also obtained from COG database. They developed a method called 'Hierarchical Union of Genes from Operons' (HUGO) to derive uber-operons.

Nebulon [74] is similar to HUGO in that both methods treat genes within an operon have direct relationships. Nebulon considers a pair of genes within an operon to be an *internal link*, and their ortholgs of two genes from another genome but within an operon is an *external link*. By setting up a threshold and including the internal and external linked genes above the threshold value iteratively, Nebulon could recover uber-operons.

There are a number of issues remains to be addressed for uber-operon prediction. First, the orthologous gene relationship used in these methods was mainly obtained from the COG database. Several studies [166, 27] have shown that the orthologous relationships can be better characterized for prokaryotes when considering operon structures. Secondly, since there are no existing experimental uber-operon data, it is very hard to asses these predicted uber-operons.

1.3 Regulon Prediction

1.3.1 Scanning Method

In the scanning method, some experimental TFBSs are aligned and used to construct position weight matrix (PWM) as shown in Table 1. The frequency matrix is first used to record the occurrence of base A, C, G or T in the alignment, and a corresponding PWM is calculated based on various formulas, such as information contents used in [67]. This PWM is used to scan the whole genome to identify more regulon members associated with the TF. Each possible sequence with the motif width in the genome is calculated based on the PWM. For instance, the score of the sequence of 'AGGTG' is 1.2+1.0+1.0+0.6+0.6 = 4.4. Those upstream sequences whose scores greater than a threshold are considered to be binding sites,

Table 1.1: Motif representation using position weight matrix (PWM)

(a)Alignment

	1	2	3	4	5
S1	A	A	G	A	C
S2	A	G	G	C	G
S3	A	G	G	T	T
S4	A	G	T	T	G

(b)Frequency matrix

	1	2	3	4	5
A	4	1	0	1	0
C	0	0	0	1	1
G	0	3	3	0	2
T	0	0	1	2	1

(b)Weight matrix

	1	2	3	4	5
A	1.2	0.0	-1.6	0.0	-1.6
C	-1.6	-1.6	-1.6	0.0	0.0
G	-1.6	1.0	1.0	-1.6	0.6
T	-1.6	-1.6	0.0	0.6	0.0

and all genes (or operons) corresponding to those binding sites are considered to be a regulon (see Figure 1.2).

This method has been applied in identifying more regulon members in several studies. For example, Tan *et al.* [148] used the scanning approach to identify new members of CRP and FNR regulons based on a number of known TFBSs. Similarly, Su *et al* [146] searched the literature to collect TFBSs and construct a PWM for the NcR regulon, and used the PMW to predict NcR regulon members.

The general problem of this scanning method, however, is the high false positives because of the nature of short motif length or degenerated pattern for many motifs. This can be avoided by introducing more constraints for specific problems. For example, Su *et al.* [146] found that the false positive rate could be reduced 40 folds by incorporating the binding site information of σ^{70} in their scanning method.

1.3.2 De Novo Method

It remains to be a challenging to use de novo methods for predicting regulons at the whole genome scale. Nevertheless, several approaches have been developed to tackle this very challenging but important problem. The general strategy of the regulon prediction problem is divided into two steps: 1). Predicting all TFBSs (or motifs) for all operons using motif-

finding programs; 2). Grouping all motifs into clusters based on the similarity of motifs. The followings briefly review computational strategies for identifying and clustering motifs.

Motif Discovery

The motif discovery problem is very fundamental in computational biology, and it has been studied for a decade. Sequences must be collected before the prediction of motifs. As TFBSs are usually located in the upstream region of genes (or operons), the sequences of these regions are collected. For prokaryotes, it is usually adequate to collect up to several hundred upstream nucleotides. Multiple sequences are needed for all motif discovery programs, and they are assumed to contain conserved motifs bound by one TF (or several), and embedded in background sequences. To obtain multiple sequences, biological experiments, such as microarray profile, or chromatin immunpreciptation [129] need to be employed. This way of collecting sequence data is known as 'multiple genes, same species', which is expensive and labor-intensive. On the other hand, with the growth of genomic sequences, collecting sequences from 'same gene, multiple species' seems to be attractive, which can be done by finding orthologous genes across multiple genomes. This strategy is also known as 'phylogenetic footprinting'.

There are roughly two major categories to find motifs among the multiple sequences: consesus-based and PWM-based. The general procedure for identifying consensus sequences is to enumerate all possible motif patterns, and check to see which ones are truly enriched in the whole sequences, possibly employing some statistical analysis. Several programs, such as Oligodyad [156], Weeder [123] and YMF [142], use this strategy. The main drawback is that the time complexity is exponential, and enumerating all longer consensus motifs becomes impossible.

Most motif-finding program use PWM formulation. The general procedure for this type of programs is to randomly initiate the motif matrix and then refine it iteratively. The

major categories include: greedy-based (*i.e.*, CONSENSUS [67]), expectation maximization-based (*i.e.*, MEME [7]), Gibbs-sampling based (*i.e.*, AlignACE [133], Bioprospector [100], MotifSampler [153]). Unlike consensus sequence-based approaches, these programs run fast. However, they cannot guarantee to identify global optimal results.

While numerous methods have been developed, the benchmark evaluation of several dataset has indicated that no single approach performs better others in all cases. The possible reasons is that motifs are usually short, ranging from 6 to 30 nucleotides, and they are usually variable, making it difficult to asses whatever how sophisticated the model is. In addition, many programs need the parameter of motif width, which is usually known *a priori*.

Recent studies have shown that reevaluating predicted motifs from different programs might increase the predicting power [72]. In addition, taking care of sequence dataset, such as reducing the length of sequences, may help to increase prediction accuracy.

Motif Clustering

To cluster those similar motifs, motif similarity measurement should be established first. CompareACE [133] used Pearson correlation coefficient of nucleotide frequencies to measure the similarity of two motifs. Wang *et al* [161] introduced the concept of 'Average Log Likelihood Ratio' (ALLR) to measure the similarity of any pair of motif profile.

Wang *et al* [161] then employed a maximum clique-finding strategy to group similar motifs by ALLR, and discovered 296 statistically significant motifs in *Saccharomyces cerevisiae*, covering more than 90% known motifs.

Qin [127] developed a Bayesian motif clustering (BMC) algorithm for clustering. The algorithm starts with a random partition of motifs. For each iteration, one motif is selected to be reassigned to a new cluster given the current partition of other motifs and the probability of the motif into a new cluster.

Related to the BMC approach, Jensen *et al* [77] used Bayesian hierarchical clustering model to cluster motifs, and developed a program called 'PHYLOCLUS'. PHYLOCLUS allows variable motif width in motif finding, and considers both single and two-block motifs.

1.4 Bipartite Matching and Biological Applications

A *bipartite graph* is a graph $G = (U, V, E)$ such that for any edge $uv \in E$, u and v must from two kinds of node sets, *i.e.*, $u \in U$ and $v \in V$. A *bipartite graph matching* M is a subset of E such that no two edges share a common node. A *maximum cardinality bipartite matching* is a bipartite matching with a maximum number of edges. A *maximum weighted bipartite matching* is a bipartite matching such that the sum of the weights of the edges in the matching is maximum. A *maximum cardinality bipartite matching* is a special case of maximum weighted bipartite matching, where the weights of all edges are set to 1. A *maximum weighted maximum cardinality bipartite matching* is a maximum cardinality bipartite matching with maximum weight.

An important concept related to matching problems is an *augmenting path*. An edge is *matched* if it is in matching M and *unmatched* otherwise. A vertex is *matched* is adjacent to an edge in M and *free* otherwise. An *alternating path* is a simple path with matched and unmatched edges alternated. An *augmenting path* is an alternating path which starts and ends with free vertices. The flipping of unmatched and matched edges in an augmenting path leads to a better matching. Berge [9] and Norman and Rabin [118] proved an important theorem associated with the augmenting path, *i.e.*, a matching M in a graph G is a maximum matching if and only if there is no augmenting path with respect to M in G. This theorem laid the foundation for most current matching algorithms.

1.4.1 Bipartite Matching Algorithms

One simple way to find a maximum cardinality bipartite matching is to take advantage of the Ford-Fulkerson method [36]. By reformulating this matching problem into flow network

problem, we can easily see that it can be done in time $O(mn)$. Many other algorithms are based on the theorem of Berge [9]. The general procedure starts with an empty matching M, keeps finding augmenting paths and flipping the edges until there is no further augmenting path. Finding an augmenting path is easily done by running a breadth first search [36] or depth first search [36]. By searching for a maximal set of vertex-disjoint shortest augmenting paths and then augmenting them simultaneously in each iteration, Hopcroft and Karp's algorithm can be implemented in time $O(m)$ [70].

The first algorithm for maximum weighted bipartite matching was the $O(n^3)$ Hungarian method [89]. It basically follows the primal-dual paradigm of linear programming algorithms. For sparse graphs, Fredman and Tarjan [55] used Fibonacci Heaps to reduce the time to $O(m(mlogm + m))$. Assuming the input weights are integer values ranging from zero to a constant C, Gabow and Tarjan [56] used a cost scaling approach and blocking flow techniques to improve the running time to $O(mlog(nC))$. Goldberg and Kennedy [61] used global price updates and a push-relabel implementation to developed an algorithm with the same running time. In addition, Kao *et al.* [81] proposed a decomposition approach to bridge the gap between the best known time complexity of a *maximum cardinality bipartite matching* and *maximum weighted bipartite matching*.

LEDA [110] provided an algorithm for *maximum weighted maximum cardinality bipartite matching*. The basic idea is to convert the original problem into the problem of finding *maximum weighted bipartite matching*, with each weight $w(e)$ in the original problem re-weighted as $w(e) + \sum w(e)$.

1.4.2 Biological Applications

Finding maximum bipartite matching has many applications in the bioinformatic field. It has been used for predicting protein-protein interaction network [28], identifying relationships of TFs and binding sites [149], and predicting operons of prokaryotic genomes [46].

Chen and Yuan [28] built the yeast protein-protein interaction network based on the dataset of proteomics and microarray. They used Floyd-Warshall algorithm to find all shortest paths in their protein network. For any connected protein pair in the network, a corresponding bipartite graph was constructed. The construction of bipartite graphs was used to represent the non-redundant edge-betweeness of that connected protein pair, which is subsequently used for separating protein networks.

Tan *et al.* [149] applied the maximum weighted bipartite matching approach to find the connection between transcription factors and their DNA motifs. Transcription factors (TFs) represent one side of the vertices, while motifs represent another side of vertices. The weights between any pair of TF and motif are evaluated based on three types of independent information (*i.e.*, distance constraint between a TF and its closest binding site, phylogenetic correlation, and specificity of TFs on binding sites).

Edwards *et al.* [46] applied the maximum weighted maximum cardinality matching into operon map prediction of microbial genomes. The basic idea is to identifying conserved gene clusters by calculating the maximum weighted maximum cardinality matching. The genes of one genome represent vertices of one side, while the genes of another genome represent vertices of another side. The homologous gene relationships from two genomes represent the edges, with their similarity scores representing their weights. They used the LEAD package [110] to identify operon structures.

With the explosion of fully sequenced genomic data, bipartite graph matching-based algorithms will be more widely used. Comparative genomic analysis using graph matching-based approaches will surely reveal much meaningful information, including our research work on the characterization of prokaryotic genomic structures.

1.5 Thesis Outline

This thesis will outline my research on predicting three levels of prokaryotic genomic structures (*i.e.*, operons, uber-operons and regulons) by using comparative genomic data and developing bipartite graph matching based algorithms.

The remainder of this thesis is organized as follows. In Chapter 2, the main problems of the current operon predictors will be stated. I will then present our operon predictor using our bipartite graph matching based algorithm. I evaluate our operon predictor using known operon dataset.

In Chapter 3, I will introduce our novel algorithm for predicting uber-operon. By comparing gene functions of our predicted uber-operons with those of known metabolic pathway and regulons, we have shown the intrinsic relationship between uber-operons and metabolic pathways, or regulons.

In Chapter 4, I will propose a new approach to phylogenetic footprinting in prokaryotes. The advantage of using conserved operon approach will be addressed, and the comparison of our approach with previous methods will be presented. In the following Chapter 5, I will present a computational framework for regulon prediction, which incorporates this new approach and predicted uber-operons. Evaluation of our framework for regulon prediction will also be addressed.

I will conclude this dissertation with the summary of our work and contributions in Chapter 6. The challenging issues related to the elucidation of microbial genomic structures, as well as the future work, will also be discussed.

Chapter 2

A Universal Operon Predictor for Prokaryotic Genomes[1]

[1]G. Li*, D. Che*, and Y. Xu. To appear in*Journal of Bioinformatics and Computational Biology*, 2008 (6)6. *Co-first author

2.1 Introduction

The operon structure is one of the features unique to prokaryotic organisms, although a few eukaryotic organisms, such as *Caenorhabditis elegans*, do have operon-like structures [11]. An operon is defined as a set of genes that are arranged in tandem and are co-transcribed as a unit, which share a common pair of promoter and terminator. As the pool of the sequenced prokaryotic genomes is expanding at an exponential rate (as of October 2007, 584 prokaryotic genomes have been sequenced), accurate identification of operons in a sequenced genome is becoming an urgent issue, simply to keep up with the world-wide sequencing efforts of prokaryotic genomes.

Various genomic features have been found to be associated with operon structures, and have been used for prediction of operons. One of the most effective ones is the intergenic distance [135] as it is found that intergenic distances within an operon tend to be shorter than inter-operonic ones. In addition, Ermolaeva *et al.* [50] found that gene cluster conservation across genomes represents another useful feature for operon prediction. Specifically, they found that if a pair of adjacent genes on the same strand has their orthologues also adjacent in another genome, they are likely to belong to the same operon. Recently, Janga *et al.* [75] found that the oligonucleotide signatures of the promoter regions of operons are different from those of the intergenic regions within operons. Other features such as signals of promoters and terminators [30, 38], functional relatedness of genes measured by clusters of orthologous groups (COG) and Gene Ontology (GO) [126, 31], codon usages [13, 126], plus microarray gene expression data [134] and metabolic pathways [171, 73] have also been reported to be useful for operon prediction.

Utilizing these or some of these features, numerous computational techniques have been developed to predict operons. We can generally group the computational techniques into two categories, unsupervised and supervised classification methods. Unsupervised classification methods do not require known operons in advance to train the classifiers. For example, Westover *et al.* [163] developed a naïve Bayesian method called 'OFS' for operon prediction by

utilizing three features: intergenic distance, common function annotation and conservation of gene clusters. This method achieves a sensitivity of 88% and specificity of 80% for its operon prediction in *E. coli*. Similarly, Price *et al.* [126] applied a Bayesian approach called 'VIMSS' for operon prediction, using intergenic distance, codon usage, gene neighborhood and functional similarity (measured using COG). Its prediction accuracies on *E. coli* K12 and *B. subtilis* 168 are 85% and 83%, respectively. Supervised classification methods, on the other hand, rely on the training data of known operons to construct a model for operon prediction. Methods that fall into this category include neural network (NN) [30, 155], Bayesian network (BN) [38, 13], Hidden Markov Models (HMM) [168], and decision tree [40, 25] based approaches. For example, Chen *et al.* [31] developed a neural network-based method, called 'JPOP', which uses intergenic distance, COG function and phylogenetic profiles for operon prediction. The program achieves an overall accuracy of 83.8% in *E. coli* K12. Tran *et al.* [155] used the prediction results from three popular operon predictors (JPOP, OFS and VIMSS) to train a neural network, and showed that the approach could reach the prediction accuracy of around 90% in both *E. coli* K12 and *B. subtilis* 168.

While these methods and others have been effective in predicting operons of the two well studied organisms, *E. coli* and *B. subtilis*, they generally do not generalize well to other organisms. The problem is that some features such as functional annotations (*e.g.*, COG), or experimental data (*e.g.*, microarray data) may not be available for newly sequenced genomes, and some feature values, such as inter-genic distance distribution versus inter-operonic distance distribution, are genome-specific, resulting in substantial performance reduction when trained on one genome (*e.g.*, *E. coli*) and tested on another (*e.g.*, *B. subtilis*) as numerous groups, including our group, have noticed this issue [40]. This raises a serious problem as currently only two bacterial organisms, *E. coli* K12 and *B. subtilis* 168, have large numbers of experimentally validated operons, making it challenging to derive accurately operon information for genomes other than those two plus possibly their closely related genomes. To address this problem, Edward *et al.* [46] developed a "universally" applicable approach

to operon prediction based on conserved genomic context information. Unfortunately, this method suffers from the problem of low prediction sensitivity.

We present here a universal and accurate method, called UNIPOP (UNIversal Prediction Of oPerons), for operon prediction for prokaryotic genomes. We predict a set of tentative operons in the target genome based on each of the reference genomes. Then each pair of adjacent genes in the target genome was scored in terms of the number of occurrences of the gene pair in the same tentative operon predicted based on different reference genomes. We then predict the final operons based on the adjacent gene pairs that have scores above some pre-determined threshold. Experimental results showed that our method achieved both high prediction sensitivity and specificity, and is applicable to all prokaryotes.

2.2 Materials and Methods

2.2.1 Data source and pre-processing

The annotated complete genomes were downloaded from NCBI GenBank database (`http://www.ncbi.nlm.nih.gov/genomes/lproks.cgi`). To obtain homologous genes for each gene in our target genome, we have carried out a homologous gene mapping for each target genome against the other prokaryotic genomes using BLAST. We consider a pair of genes from two different genomes to be homologous if both their reciprocal BLAST e-values are $< 10^{-6}$.

Experimentally confirmed operon dataset of *E. coli* K12 were downloaded from regulonDB database [136]. Operons of other three species were extracted from operon database (ODB) (`http://odb.kuicr.kyoto-u.ac.jp/`) [119], in which operons of *B. subtilis* 168 were originally obtained from transcriptional maps stored in BSORF (`http://Bacillus.genome.jp/`), and the operon data of *Agrobacterium tumefaciens* C58 UWash and *Pseudomonas aeruginosa* PA01 were collected through searching the literature.

We also downloaded microarray (kinetic) gene expression data of nine organisms, namely *B. subtilis*, *Campylobacter jejuni*, *E. coli*, *Francisella tularensis*, *Helicobacter pylori*, *Strepto-*

coccus pneumoniae, *Streptomyces coelicolor, Synechocystis* PCC 6803 and *Vibrio cholera*e, from the Stanford MicroArray Database (`http://genome-www5.stanford.edu`) [62].

2.2.2 Algorithm design

Our operon prediction algorithm uses two key parameters, *maximum allowed distance* (*MAD*) and *threshold of supporting evidence* (*TSE*). *MAD* is used to guarantee to choose conserved gene clusters with the characteristics of short intergenic distances, while *TSE* is used to guarantee the predicted operon structures to be statistically reliable while relying on multiple reference genomes. Our algorithm consists of two main stages: 1) partition of a target genome into tentative operons based on the identified conserved gene clusters compared against a set of reference genomes; and 2) generation of final operon prediction using a voting scheme.

Stage I: At this stage, the algorithm iterates through N (specified) reference genomes to produce N sets of tentative operon prediction (called *operon maps*) for the target genome. For each reference genome and the target genome, we produce a tentative operon map as follows.

1) Creation of graphs: Define a bipartite graph H as follows. Each gene in the two genomes (the target genome and one reference genome) is represented as a vertex, and two vertices are connected by an edge if and only if they represent two homologous genes in different genomes defined by reciprocal BLAST (see the first subsection of Materials and Methods). We also define an auxiliary (dynamic) graph $D = (U, V, E)$ as follows, where U is the vertex set consisting of each gene and each non-coding nucleotide in the reference genome and V is defined similarly for the target genome. We consider that the vertices in U and V are numbered from 1 to $|U|$ and $|V|$, respectively. The edge set E of D is initially defined as the edge set of H. Recall that a directon in a genome is a list of consecutive genes on the same strand of DNA that are not interrupted by genes on the opposite strand [135]. Since all genes in an operon should be in the same directon, our refinement process of the tentative operon

predictions is done without violating this property. A matching of the graph D is a set of edges, none of which share a common vertex, and a maximum matching is a matching with the most number of edges possible [36]. The basic idea of our algorithm is to repeatedly find conserved gene clusters (not necessarily maintaining the same gene order) between the target and the reference genome until each gene cluster could not be further expanded (based on the rule explained below).

2) Refinement of tentative operon prediction iteratively: Let $W < min(|U|, |V|)$ be a positive integer, called *MAD*. The algorithm first finds the feasible positions i, j in the two vertex list U and V such that the maximum matching for the subgraph of D induced by the vertex set $((i, ..., i + W - 1), (j, ..., j + W - 1))$ has the largest cardinality that is at least 2, among all possible i's and j's, where a position i is called *feasible* if all genes within $[i, ..., i+W-1]$ are in the same directon. If such positions are found, we update the D graph by contracting all vertices in $[i, ..., i + W - 1]$ into one vertex; so do we on the vertices in $[j, ..., j+W-1]$. Similarly, the edges between the vertices of $[i, ..., i+W-1]$ and the vertices of $[j, ..., j + W - 1]$ are contracted into one edge. At this time, we delete all edges incident with the contracted edge. Fig. 2.1A-B illustrates the updating process of the dynamic graph G in one iteration. The procedure continues until no further contraction can be made. We predict genes in each contracted vertex as an operon in reference to this current reference genome.

We should point out that, the contracted vertex from previous iterations can form a new vertex set with its neighbors in later iterations, as long as the requirement of *MAD* is satisfied (Fig. 2.1A-B). Therefore, our algorithm can guarantee to find operons with large sizes, probably in several steps instead of one. In addition, *MAD* has the following properties: In the beginning of the procedure, a chosen vertex set contains many genes separated by non-coding nucleotides, whose length is also known as intergnic distance. At this point, *MAD* can cover several short intergenic distances. In the later procedure, the qualified vertices may only contain two vertices (either with the gene type or the type contracted from previous

iterations) separated by vertices of non-coding nucleotides. Thus, *MAD* can only cover one long intergenic distance.

Stage II: We now score each pair of adjacent genes in the target genome based on the N tentative operon maps generated at Stage I: we count the total number of occurrences for any pair of adjacent gene pair belonging to the same operon in some tentative operon maps, denoted as $S(\mathrm{g}_i, \mathrm{g}_{i+1})$. We then use a pre-determined threshold to make the final operon prediction: for each pair of adjacent genes, we consider it to be in the same operon in the target genome if $S(\mathrm{g}_i, \mathrm{g}_{i+1})$ is not less than a threshold, which we call *TSE*, For each such gene pair, we call it an *operonic gene pair* in the target genome; otherwise a *non-operonic gene pair*. We consider those consecutive operonic gene pairs to be an operon. Fig. 2.1C illustrates a simple example of an operon map, containing three operons of (1, 2, 3, 4), 5, and (6, 7, 8, 9, 10).

2.2.3 Remark

At the first glance the performance of the algorithm depends on the two parameters of our program, *MAD* and *TSE*. Our analyses on the prediction results, however, show that the prediction results are stable in terms of these two parameters, and the program performs well for *MAD* to be ~ 200 and *TSE* to be 2 or 3 for most of the sequenced prokaryotic organisms. It should be noted that using one reference genome only is obviously not enough to derive all operons in the target genome. At one extreme, if our target genome has 100% sequence identity with the only reference genome, the prediction of our algorithm is only determined by *MAD*, similar to those prediction algorithms that rely on intergenic distances for operon prediction. At the other extreme, if the two genomes are evolutionarily too distant, there might be too few homologous gene pairs between them, and thus only very few operons could be detected. Hence, we have used multiple reference genomes to predict operons for the target genome. In general, the more reference genomes we use the better prediction results our algorithm should provide. Thus, we have picked one representative strain of each

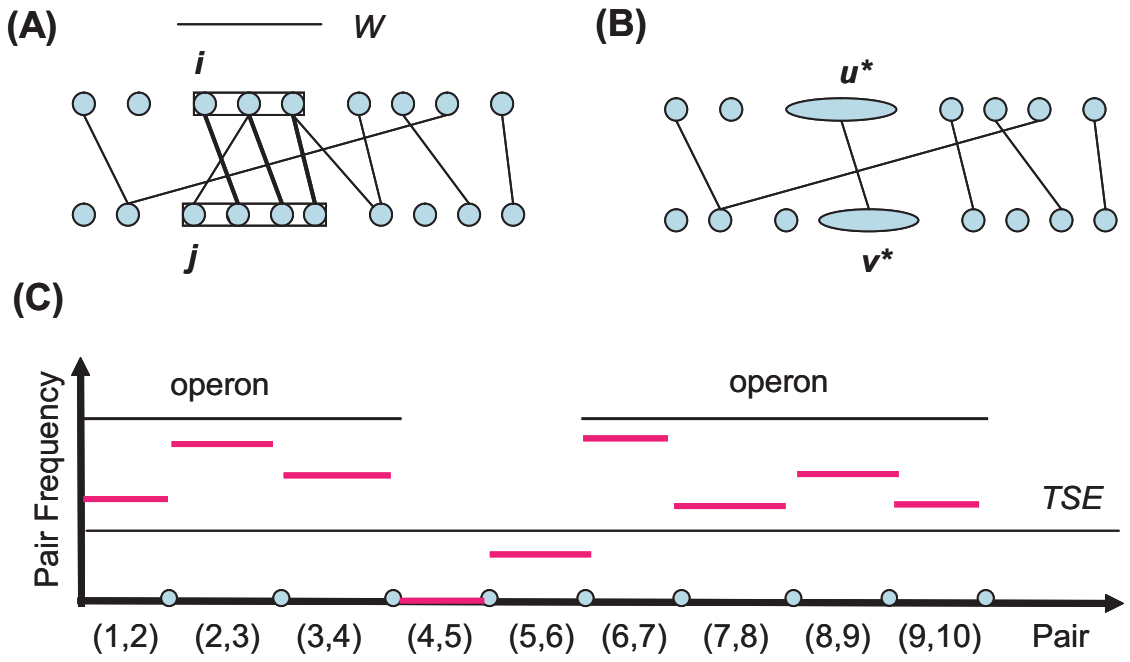


Figure 2.1: An illustration of the algorithm design in UNIPOP

(A) Dynamic graph D in stage 1. For simplicity, only vertices designated as genes are shown here, all non-coding nucleotides between genes are omitted. The top parts of vertices are from the target genome, while the bottom parts are from the reference genome. The edges represent homogolous gene relationships from the two genomes. A subgraph *S(i, j)* of graph D can be constructed as follows: Starting with gene i (not necessary vertex i as we omit the non-coding nucleotides here) in the target genome, we keep include vertices into the list until the number of vertices is greater than W, or the directon requirement is violated. We can choose vertices starting gene j in the other side in a similar way. The edge relationship within these vertices will be kept. All possible subgraphs are constructed, and corresponding maximum matchings are calculated. The subgraph whose maximum matching is the largest and greater than 1 is chosen for updating graph D. (B) The updated dynamic graph after one iteration in stage 1. All vertices from each side of the selected bipartite graph are contracted into one vertex, denoted as $u*$ and $v*$. This process continues until there is no qualified subgraph. (C) The construction of a operon map in stage 2. For any adjacent genes whose pair frequency equal or greater than *TSE*, we consider it to be an operonic gene pair.

species in all sequenced prokaryotic genomes to avoid the over-representation issue of closely related genomes, and at the same time, to cover as many conserved gene clusters for operon elucidation as possible. This simple selection strategy is applicable for operon prediction to any sequenced genome. In our current work, 274 reference genomes are used.

2.3 Results

2.3.1 Prediction accuracy analysis in four species

To evaluate our method for operon prediction, we have tested it on four organisms with known operon data, including *E. coli* K12, *B. subtilis* 168, *A. tumefaciens* C58 UWash and *P. aeruginosa* PA01.

Two key parameters were used in our algorithm, *MAD* and *TSE*. *MAD* is related to the intergenic distance among conserved gene clusters between two genomes, while *TSE* is related to the number of conserved operons in all comparative genomes. By setting the *MAD* (W) in the range of 50 to 400 bp, and the *TSE* (T) = 1 - 8, we ran our program and predicted operons for each of the four organisms. To evaluate the prediction performance, we record the number of true positives (TP), which is the number of correctly predicted operonic gene pairs among all known operonic gene pairs, and the number of false positives (FP), which is the number of incorrectly predicted operonic gene pairs in known non-operonic gene pairs. Similarly, we count the number of true negatives (TN), the number of correctly predicted non-operonic gene pairs in known non-operonic gene pairs, and the number of false negative (FN), the number of incorrectly predicted non-operonic gene pairs in known operonic gene pairs. Like in other similar studies, we define the prediction sensitivity (Sen), specificity (Spe) and accuracy (Acc) as $TP/(TP+FN)$, $TN/(TN+FP)$ and $(TP+TN)/(TP+FN+TN+FP)$, respectively.

Figure 2.2 shows three-dimensional (3D) views of prediction accuracies in terms of the two parameters *MAD* and *TSE*, on the four organisms. Different colors represent different prediction accuracies, with dark red for the highest accuracy and blue for the lowest accuracy.

Figure 2.2A shows the surface curve for the prediction accuracy in *E. coli* K12. As we can see, the highest accuracies were achieved in the area where W was $\sim$ 200, and T was 2 or 3. The highest accuracy obtained was 93.3% with $W = 200$ and $T = 2$. For *B. subtilis* 168, we have a similar surface pattern with that of *E. coli* as shown in Figure 2.2B. The highest prediction accuracy, however, was 83.0%, about 10% lower than that of *E. coli*. The highest prediction accuracies for *Agrobacterium tumefaciens* C58 UWash and *Pseudomonas aeruginosa* PA01 were 90.0% and 92.5% (Figure 2.2C and 2.2D), respectively. Unlike those of *E. coli* and *B. subtilis*, the surface areas for *A. tumefaciens* and *P. aeruginosa* were not as smooth. We believe that this was mainly caused by the small dataset of known operons for these two organisms. However, the areas that gave the highest accuracies were still similar to those of the other two genomes. Interestingly, the maximum allowed distance 200 used in our program was the same as for the adjacency gene constraint used in Ermolaeva *et al.*'s method [50]. Our prediction results, as well as other reports, indicate that $MAD = \sim 200$ with a small *TSE* value at 2-3 is generally applicable to most of the prokaryotic organisms. Therefore, a fixed parameter setting (*i.e.*, W=200 and $T = 2$) was used for predicting operon structures of 365 genomes.

2.3.2 Validation of operon prediction in other organisms

Besides those experimentally validated operons collected from operon databases, we have also searched the literature to obtain experimentally validated operons in *Mycobacterium tuberculosis* and *Staphylococcus aureus*. For *M. tuberculosis*, most of the experimentally verified operons that we can find are in good agreement with our operon prediction. For example, our program predicted an operon consisting of *Rv3134c-devR-devS*, which were reported to be co-transcribed [6]. We have found both the virulence operon (*Rv0986-Rv0987-Rv0988*) and *pst* operon accurately as reported in [132, 14]. In a few cases, our program predicted known operons partially correct. For example, the *mceI* operon was reported to contain 13 genes encompassing *Rv0166* to *Rv0178* [23](25), while the operon we predicted contains 12

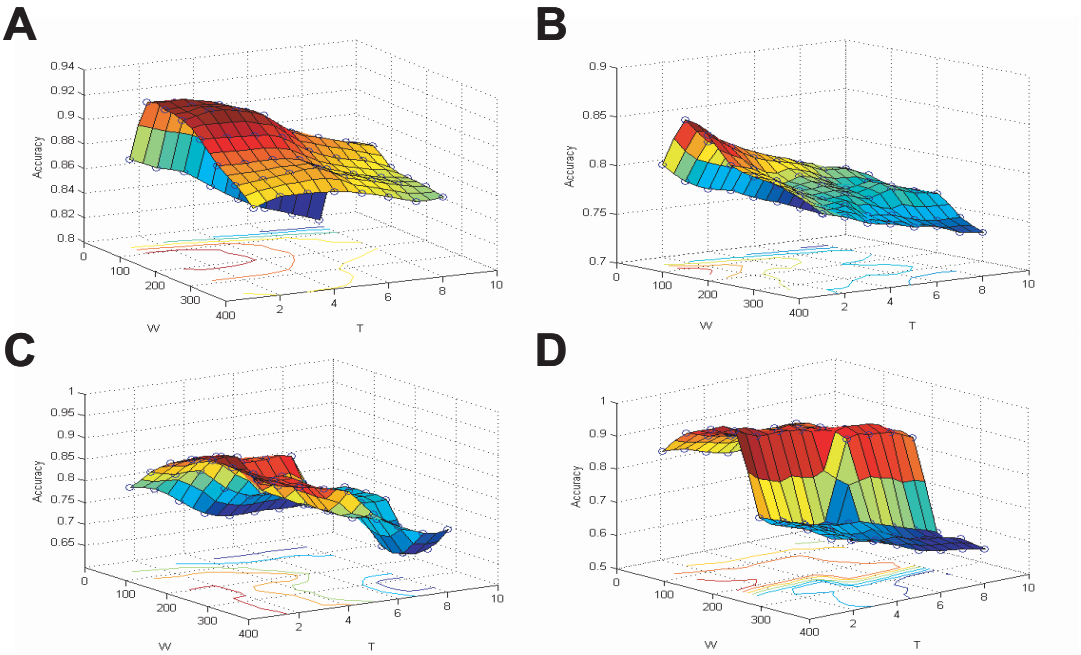


Figure 2.2: Operon prediction accuracy in four specices

(A). *E. coli* K12; (B). *B. subtilis* 168; (C). *A. tumeficiens* UWash and (D). *P. aeruginosa* PA01.

genes, missing the gene *Rv0166*. Our further analysis showed that the intergenic distance between *Rv0166* and *Rv0167* is 204, slightly larger than *MAD* used in our program, thus leading to exclusion of *Rv0166* from the predicted *mcel* operon.

Out of 21 *Staphylococcus aureus* operons that we collected from the literature, we correctly predicted 15, including *agrBDCA* [10], *capABCDEFGHIJKLMNOP* [121], *ctsR-yacH-yacI-clp2392* [58], *czrAB* [91], *dltABCD* [115], *femAB* [87], *fhuCBD* [20], *hsp10-hsp60* [92], *lacABCDFEG* [17], *nrdIEF* [105], *orf1-orf2-glmM* [60], *pheST* [137], *rsbUVW-sigB* [85, 90], *sirABC* [71] and *sstABCD* [112]. For the *mnhABCDEFG* operon [69], we predicted it to have an extra gene *Sav0953* adjacent to gene *mnhA*. In addition, we predicted two operons *largAG* and *lytSR* [65] as one operon. These problems were caused by the short inter-operonic distance between two operons as well as the short intergenic distances between their homol-

ogous genes in the reference genomes, thus leading to the (possibly) incorrect prediction by our method. On the other hand, our predicted *gln* operon contains two genes *glnR* and *glnA*, while the reported *gln* operon in the literature contains a third gene *pr* [144]. In two harder cases, our program did not predict the partial operons correctly, such as *splABCDEF* [128] and *tcaRAB* [15]. We found that the *spl* operon exists only in the target genome *Staphylococcus aureus* but not in any other used reference genomes. Since our program relies on reference genomes to make the operon prediction, our prediction method is intrinsically not adequate for predicting such operons. For the *tca* operon, we found that all intergenic distances of adjacent genes are more than 250, longer than *MAD* we used for operon prediction.

2.3.3 Predicted operons have high correlations with their genes expression profiles

Gene pairs within an operon should in general have highly correlated expression patterns while non-operonic gene pairs generally do not. We have used the Pearson correlation coefficient to measure the similarity between a gene pair's expression patterns. We have used the time-course microarray gene expression data for nine organisms (see Materials and Methods), calculated the Pearson correlation coefficient [155] of the predicted operonic gene pairs and that of non-operonic gene pairs for the nine organisms, and plotted the distributions of the Pearson coefficients in Figure 2.3. As we can see, the distributions of the correlation coefficients between predicted operonic and non-operonic gene pairs are substantially different for most organisms. More specifically, the percentages of operonic gene pairs with high Pearson correlation coefficients are much higher than those of non-operonic gene pairs. For example, 34.2% predicted operonic gene pairs in *E. coli* have Pearson coefficient values higher than 0.8, compared to 8.2% for non-operonic gene pairs. The distribution patterns between non-operonic and operonic gene pairs for *Campylobacter jejuni* and *Streptomyces coelicolor* are similar. The low percentage of operonic gene pairs with high Pearson correlation coefficients in *S. coelicolor* could be caused by a general downward trend in the expression level from the

first gene to the last gene in operons as reported in [93]. We suspect that the 'unequal expression' among genes within an operon might also exist in *C. jejuni*, although further investigation is needed. We also computed the Pearson correlation coefficient of known operon data in *E. coli* and *B. subtilis*. As shown in Figure 2.3A-B, the pattern trends of our predicted operons and the known operons are quite similar, indicating the general reliability of our predicted operons.

2.3.4 Comparison to other methods

We have compared our method with seven other operon prediction programs, namely DVDA [46], FGENESB (`http://www.softberry.com`), ODB [119], OFS [163], OPERON [50], JPOP [31] and VIMSS [126]. Our comparison was limited to two organisms (*i.e.*, *E. coli* and *B. subtilis*) since only these organisms have predicted operons by all methods. Considering that most of the operon predictors are parameterized and a different parameter setting may be favorable to one specific organism in terms of the prediction accuracy, we have used the predicted operons released by each predictor and measured their prediction accuracies. Table 2.1 shows the prediction sensitivity, specificity and accuracy of the eight predictors. For *E. coli* K12, our method has the highest sensitivity (95.7%) while DVDA, another graph-theoretic approach, has the lowest sensitivity (46.3%). The low sensitivity of DVDA evaluated here is very close to that reported in [46]. ODB has the highest specificity (98.3%), but it suffers from low sensitivity. In the case of *B. subtilis* 168, all the predictors have lower prediction accuracies than those on *E. coli*. This is because most of operon prediction methods, including our method, assume that adjacent genes within an operon cannot be intervened by other genes on the opposite strand of DNA. However, intervened operons, such as *yflMK* (intervened by *yflL*), *ypfSU* (intervened by *ypfT*), have been experimentally verified [42]. On the other hand, the lower prediction accuracy could be due to incorrect operon annotations in *B. subtilis* 168. Previous investigations have shown that some of the annotated non-operonic gene pairs are actually operonic gene pairs, which have

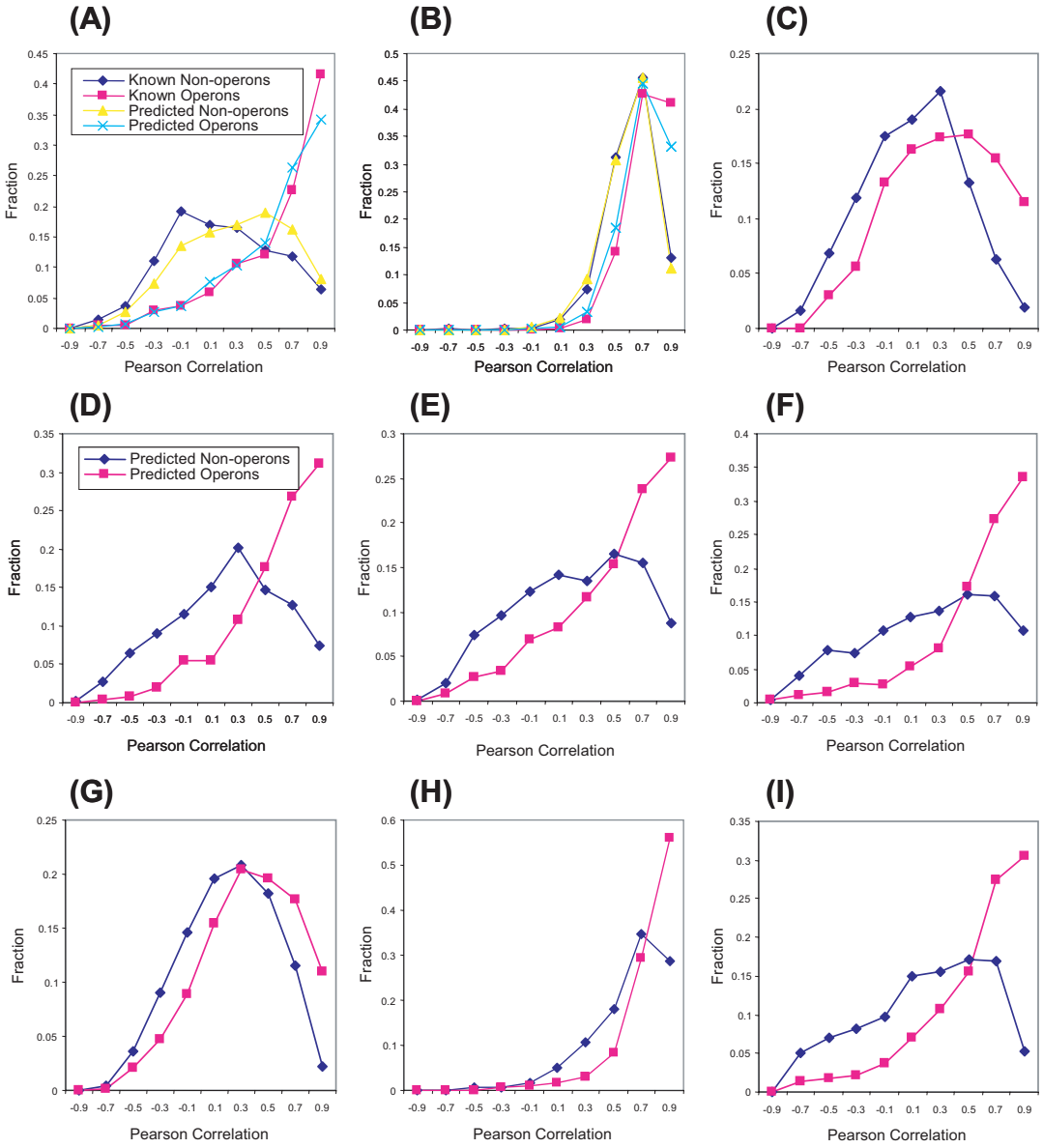


Figure 2.3: The distributions of Pearson correlation coefficients between expression profiles of gene pairs in nine organisms

(A). *B. subtilis*, (B). *E. coli*, (C). *C. jejuni* (D). *F. tularensis*, (E). *H. pylori*, (F). *S. pneumoniae*, (G). *S. coelicolor*, (H). *S. PCC* 6803 and (I). *V. cholerae*. The X-axis represents the Pearson correlation coefficients, and the Y-axis is the density function of operonic and non-operonic gene pairs.

been experimentally verified, such as *sul/folA* and *mmgE/yqiQ* that were not in the original operon list [126]. Overall, our method has achieved the highest prediction accuracy over all methods we compared.

2.3.5 Operon comparison between archaea and bacteria

Out of the 365 prokaryotic genomes, for which we have made operon prediction, 28 are archaeal and 337 are bacterial. The size distribution of all the predicted operons of the two superkingdoms in terms of the number of genes is given in Fig. 2.4. On average, there is no significant difference between them although the percentage of small-sized operons in archaea is slightly higher than that of bacteria. Our analyses of operon structures, however, indicate that archaea and bacteria differ in many aspects. First of all, we found that several operons encoding ribosomal proteins are highly conserved in archaea but in bacteria, and vice versa. This is in agreement with previous studies that many ribosomal genes differ between archaeal and bacterial genomes [82, 96]. We also found that operons encoding aminoacyl-tRNA transferase and translation initiation factors are different between two the superkingdoms, indicating that the translation machinery between archaea and bacteria differs in at least three aspects. Secondly, we found that the operon structures associated with DNA replication are different. Our discovery supports previous studies that no homologues to two major bacterial operons involved in homologous recombination, *RecBCD* and *RecFOR*, were found in Archaea [3, 164], but the operon of *herA-rad50-mre11-nurA* with the same function was found in thermophilic archaea [34]. Thirdly, the operon structures of the two superkingdoms which encode for the flagellar system are different, supporting previous studies that the composition and development of archaeal flagella are different from that of bacterial flagella [8, 117].

Table 2.1: Prediction statistics and features used by each computational method on the dataset of *E. coli* K12 and *B. subtilis* 168

Method	*E. coli*			*B. subtilis*			Features used
	Sen	Spe	Acc	Sen	Spe	Acc	
DVDA	0.463	0.922	0.693	0.319	0.932	0.485	Homologous genes
FGENESB	0.772	0.972	0.85	0.721	0.904	0.771	ID, GOC, promoter and terminator
ODB	0.647	0.983	0.778	0.499	0.992	0.632	ID, pathway, microarray and GOC
OFS	0.931	0.659	0.888	0.765	0.439	0.683	ID, common gene annotation and GOC
OPERON[a]	0.66	0.871	0.742	0.531	0.892	0.629	Gene cluster conservation
JPOP	0.853	0.868	0.855	0.72	0.9	0.746	ID, COG and phylogenetic profile
VIMSS	0.884	0.827	0.876	0.764	0.871	0.78	ID, comparative features, COG and CAI
UNIPOP[b]	0.957	0.894	0.933	0.782	0.821	0.792	Homologous genes

TP, #true positives; *FP*, #false positives; *TN*, #true negatives; *FN*, #false negatives; *Sen*, defined as $TP/(TP + FN)$; Spe, defined as $TN/(TN + FP)$; Acc, defined as $(TP + TN)/(TP + FN + TN + FP)$. In the column of 'features used', ID, intergenic distance; GOC, gene order conservation; COG, clusters of othologous gene groups; CAI, codon adaptation index.

OPERON[a] does not provide predicted operon structures for *E. coli* K12 and *B. subtilis* 168. Instead, it provides confidence values for gene pairs within a directon. We simply consider two adjacent genes whose confidence value equal or greater than 60 (60 has been tested to be the suboptimal cutoff value in terms of prediction accuracy) are a predicted operonic pair.

UNIPOP[b] generates operons for all organisms, including *E. coli* K12 and *B. subtilis* 168, using a fixed parameter setting as described in the text. Thus, prediction accuracies reported here are not necessary the best ones as reported in Figure 2.2.

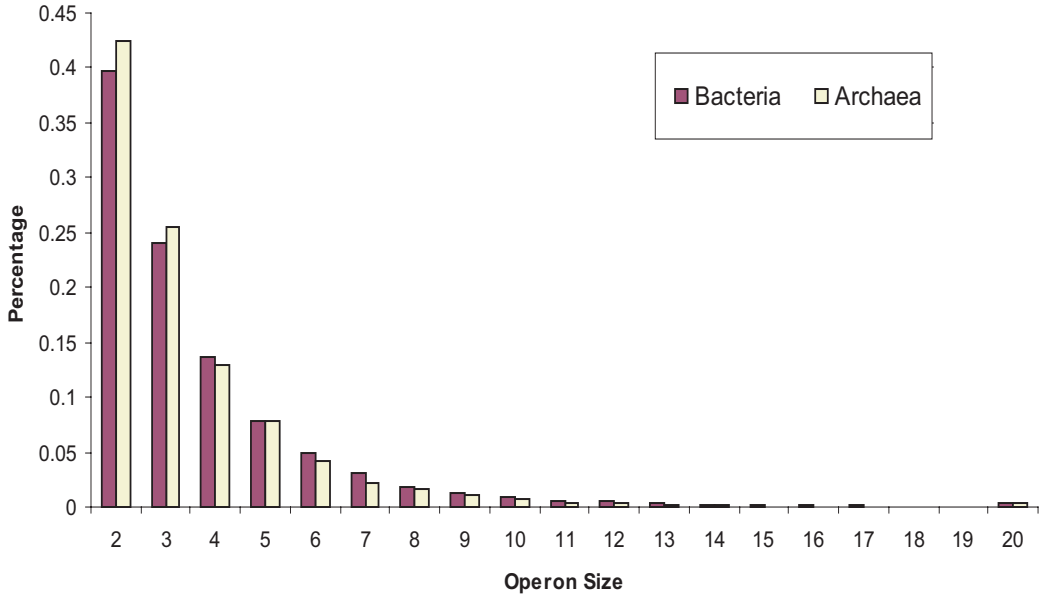


Figure 2.4: The percentage distribution of operon sizes in Archaea and Bacteria

2.4 Discussion

Theoretically, adding more genomes should help improve the prediction accuracy of our method. We should point out that using more reference genomes in our method was not the main cause of the better performance of our method over other methods we compared. Our prediction accuracy analysis on *E. coli* showed that there was only 0.5% decrease of the overall prediction accuracy when using 176 reference genomes, and 3.8% decrease when using 67 reference genomes. Either prediction accuracy was better than any other compared method, which used more than 100 reference genomes, except for DVDA (using 74 reference genomes).

The *maximum allowed distance* used in our algorithm is related to, but quite different from the intergenic distance used in most of other operon prediction methods. In those

methods, whether a pair of adjacent genes being operonic pair or not is dependent on the threshold of intergenic distance. Thus, a small change of the threshold may affect prediction accuracy dramatically. The parameter of *MAD* is used to construct bipartite graphs and identify conserved gene clusters as described in Materials and Methods. The identification of conserved gene clusters in our algorithm depends on the parameter of *MAD* and the homologous relationships simultaneously. Compared to the threshold of intergenic distance used in other methods, *MAD* is much larger ($\sim$ 200) and can be more flexible. A small change of *MAD* does not significantly affect the prediction accuracy.

The substantial difference in prediction sensitivity between our method and the other two methods OPERON [50] and DVDA [46], using somewhat similar ideas, indicates our method captures some of the key characteristics of operons more accurately than the previous methods. In the method of OPERON, the authors consider only adjacent gene pairs as candidate operonic gene pairs when their homologues are also adjacent in the reference genomes in the same order. This constraint may be too stringent since local gene order rearrangements within an operon do exist. For example, the order of L-arabinose operon is *araA-araB-araD* in *B. subtilis*, while the order is *araB-araA-araD* in *E. coli* [163]. In addition, inclusion of one or more additional genes in an operon across different genomes occurs frequently. For instance, the *glnQ-glnP-glnH* operon in *E. coli* is organized as *glnQ-glnH-glnM-glnP* in *B. subtilis* [46]. These two common scenarios haven't been taken into consideration in OPERON, while they have been taken care of in our algorithm.

Besides the directon constraint used in the approach of DVDA, our method has also incorporated a maximum allowed distance parameter MAD to help split those adjacent gene pairs with large intergenic distances into different operons. Adding the *MAD* constraint in our method becomes very effective when the size of directons in the target genome become large. Theoretically, two large-sized directons from two genomes are prone to be considered as conserved gene clusters when the directon constraint only approach is applied, because there are more possible homologous relationships between two large directons. In practice,

operon fission or fusion occurs frequently in prokaryotic genomes, and thus the conservation relationship between a large directon in one genome and several small directons in another genome because of the gene fission process cannot be correctly detected with the directon constraint only approach, but can be nicely handled when the parameter of *MAD* is incorporated.

2.5 Conclusion

We have presented a novel method for predicting operons accurately. By representing the operon prediction problem as a graph-theoretic problem, we have developed an effective algorithm for predicting candidate operon structures. We have used multiple reference genomes and a unified framework to derive operon structures that are most consistent with the conserved gene cluster information across multiple genomes. Validation on several organisms showed that our prediction method has high prediction accuracy on genomes with experimental operon data.

Our comparison with other methods on *E. coli* and *B. subtilis* showed that our method performs better than any of the existing methods. More importantly, the high accuracy of our method is not achieved by combining multiple genomic features as in most of the operon prediction programs, some of which are probably highly organism-specific. Our prediction mainly relies on the conserved adjacent homologous gene pair information, which can be easily obtained by running BLAST against the reference genomes, and two simple parameters, *MAD* and *TSE*. We found that our prediction performance is fairly stable in terms of small changes on *MAD* and *TSE*. Hence we believe that our method is universally applicable to any prokaryote with sequenced genome.

While our algorithm can detect operons common to the target genome and some of the reference genomes, it represents a challenge for the method to identify operons unique to the target genome. While we expect that this problem becomes a lesser problem as more and more genomes, particularly genomes of related organisms, get sequenced, we do plan to

deal with this problem through using some general features associated with promoter regions and/or terminators of operons as we have recently investigated in a separate work by our group [40]. We expect that this universal operon prediction program will prove to be a highly useful tool for genome annotation, particularly the ones without much experimental data.

Chapter 3

Detecting Uber-operons in Prokaryotic Genomes[1]

[1]D. Che, G. Li, F. Mao, H. Wu and Y. Xu. *Nucleic Acids Research*, 2418-2427

3.1 Introduction

The rapidly expanding pool of sequenced microbial genomes provides a very rich source of information for deciphering the hidden information encoded in a genome and the organizational structures of the encoded information. One powerful tool for decoding such information is the so-called comparative genome analysis, which attempts to derive the encoded information through directly comparing the genomes against one another. Through such comparisons, 'conserved' genomic structures at various organizational levels could possibly be detected [165, 30]. Then by linking these identified genomic structures to already well-established biological entities, one could possibly infer their biological meanings [145, 146, 39]. For situations where such links are not clearly identifiable yet, the significance of the uncovered 'conserved' genomic structures could possibly be established through statistical means. Comparative genome analyses have been used to predict operon structures, a layer of well-established genomic structure, at a whole genome scale [30, 31, 126, 163, 131]. As more powerful comparative genome analysis tools become available, we expect that more genomic structures, previously understood or new, will be revealed.

As we understand now, there are a number of well-established higher-level genomic structures beyond operons in a microbial genome, which include regulons, modulons and stimulons. A regulon [88, 159] is a group of operons which are co-regulated by the same transcriptional machinery, while a modulon [88, 159] is a group of regulons that are controlled by more global regulators and respond to more general physiological states of a cell. At an even higher level is a set of stimulons [88, 159], each of which consists of a collection of operons, regulons and/or modulons that respond to a common environmental stimulus. Each of these genomic structures generally encodes a biological pathway or a complex network (or possibly portions of a pathway/network). Hence identification and characterization of these genomic structures has direct implications in deciphering biological pathways and networks in a systematic manner, which represents one of the key tasks in the study of an organism at a systems level.

It is known that in bacteria, genes are transcribed using operons (including single-gene operons) as the basic units, while in eukaryotes genes are transcribed individually. While the exact reason for this phenomenon requires more investigation, we suspect that one possible reason might be that as organisms evolve to become more complex, they might have the tendency to use each of their genes in more biological processes, which requires the flexibility of different gene associations to efficiently handle different needs for co-transcription. This, in turn, might have led to the breakup of the fixed gene associations enforced by the (large) operon structures in the ancient and simple organisms to possibly smaller transcriptional units in more complex organisms. We have recently carried out a systematic study on the tryptophan synthesis operons. We found that these operons are fairly larger (average operon size is 6.4 for 24 archaeabacteria genomes) in some archaeabacteria while their sizes are in general smaller (average operon size is 1.4 for 17 cyanobacteria genomes) in some cyanobacteria (P. Wan, F. Mao and Y. Xu, manuscript in preparation). This observation seems to suggest that some of these operons may have experienced the fission process during the evolution. To the extreme along this discussion, all eukaryotic genomes have each of their genes individually regulated transcriptionally, *i.e.* all their 'operons' are singletons.

By identifying operons that used to be associated with some ancient organisms (*e.g.* two whole operons or parts of them belong to the same operon in an ancient organism) or other organisms in general, we may detect the footprints of operon evolution. This footprint of operon evolution might provide useful information leading to not only better understanding about genomic structures and their organization, but also possibly a new set of tools for studying biological machineries in a prokaryotic cell, just like the powerful tool that operons have provided to biological pathway prediction [39, 103, 120, 145, 146]. In this study, we focus on the identification of the footprint of a particular class of operon evolution, uber-operons, a concept introduced by Lathe *et al.* [94]. The essential idea of a uber-operon is that during evolution, larger operons might have broken into smaller operons in different ways along different evolutionary lineages. Hence by studying conservations among groups

of operons ('uber-operons') rather than individual operons, it may help to uncover the 'lost' association relationships among operons that used to work together constitutively. By the very nature of the uber-operon definition, it requires reference genomes to uncover the 'lost' association relationship among of the uber-operons. In particular, it requires the knowledge of orthologous genes across genomes. Lathe *et al.* [94] proposed an iterative procedure for identification of uber-operons, assuming that orthologous gene relationships are given, which has limited the practical value of their method. In this paper, we present a novel algorithm to simultaneously identify uber-operons in a target genome and orthologous gene relationships between the target and reference genomes. We first give a revised definition of uber-operons, which we believe is more precise and better captures the association relationship among operons as outlined above. A uber-operon, U, is a group of operons in a genome whose component operons are transcriptionally or functionally related, and U is conserved across multiple (reference) genomes in the following sense: the orthologous genes of U's genes in each of these reference genomes form a group of operons, which (approximately) contain these orthologous genes only (*i.e.* these operons approximately do not contain other genes nor miss genes). Here 'transcriptionally related' refers to that operons are transcriptionally co-regulated [84]; 'functionally related' refers to that operons include genes of the same pathway [84] or with highly similar Gene Ontology (GO) annotations [4]; and orthologous genes (or simply, orthologs) refer to isofunctional and heterospecic genes [78, 86, 125] throughout this paper. Another concept used repeatedly throughout the paper is linker genes, each of which refers to a pair of genes in a genome that each gene is in a different operon and their orthologous genes are in the same operon in a reference genome.

3.2 Materials and Methods

3.2.1 Data preparation

By selecting one complete genome in each genus, we have obtained 115 genomes from 224 complete bacterial genomes at the NCBI website (release of 03/05/2005). Operon predic-

tion results for these genomes were downloaded from `http://www.microbesonline.org/operons/`, denoted as VIMSS operons [126]. We have also applied our in-house program, JPOP [31], for operon prediction. The average operon size predicted by JPOP is slightly smaller than that of the VIMSS operons, although the two programs have similar prediction accuracy (F. Mao and Y. Xu, unpublished data). The VIMSS operons are used for our study, because their slightly larger operon size should in principle lead to lower false negative rate in linker gene identification. Since VIMSS has operon predictions for only 91 out of the 115 genomes (including *E. coli* K12), we have removed the remaining 24 genomes from further consideration.

Another dataset needed for our uber-operon prediction is the homologous genes in the reference genomes for each gene in our target genome. We have carried out a homologous gene mapping for each of the 91 genomes against the remaining 90 genomes, using BLAST search with an E-value cutoff at 10^{-3}. Both the predicted operons and the homologous genes are provided at our Uber-Operon Database: `http://csbl.bmb.uga.edu/uber`.

3.2.2 Uber-operon prediction against one reference genome

We first formulate the problem of uber-operon identification based on one reference genome, and then outline an algorithm for solving the problem. The main and fundamental difference between our algorithm and the algorithm of [94] is that we do not assume that the orthologous gene relationship is given; instead orthologous gene relationship is detected simultaneously with uber-operon prediction.

Consider a target genome G_1 and a reference genome G_2. We assume that each gene in G_1 has at most one ortholog in G_2, and vice versa. Intuitively, a uber-operon is modeled as a maximal group of transcriptionally or functionally related operons that are linked through linker genes; and there is no overlap between any two uber-operons (unlike regulons). One challenging issue in identifying uber-operons is to accurately identify orthologous genes between two genomes. Our previous study has demonstrated that existing methods,

such as BDBH [114], its variations [160] and COG [151] are not adequate for highly specific and accurate identification of orthologous genes at a large scale, since these algorithms all attempt to predict orthology based mainly on sequence similarity information, and sequence similarity information alone does not imply orthology [103]. This problem has been partially overcome by a new strategy employed in our recent work on orthologous gene mapping by using both sequence similarity and genomic structure information [103, 165]. The basic idea is as follows. If a pair of genes g_1, g_2 are in the same operon of G_1 and their homologous genes g_1' and g_2' are also in the same operon in G_2, then the probability for g_1 and g_1', and g_2 and g_2', respectively, to be orthologous is high [166]. So our uber-operon identification algorithm is to find such mappings in the context of finding uber-operons, which maximizes the overall probability for all the mapped gene pairs to be orthologous.

Formally, we define a bipartite graph $B = (U, V, E)$ for genomes G_1 and G_2 as follows. Let $U = \bigcup_{i=1}^{m} U_i$ and $V = \bigcup_{j=1}^{n} V_j$ be the two vertex sets, with $U_i = \{u_{i,s} | s = 1, 2, ..., p_i\}$ and $V_j = \{v_{j,t} | t = 1, 2, ..., q_j\}$ representing the gene list of the i^{th} operon of G_1 and the gene list of the j^{th} operon of G_2, respectively; $u_{i,s}$ and $v_{j,t}$ representing the s^{th} gene in U_i and the t^{th} gene in V_j, respectively; p_i and q_j being the numbers of genes in U_i and V_j, respectively; and m and n being the numbers of operons in G_1 and G_2, respectively. E is the edge set connecting vertices of U and V such that an edge exists if and only if the two corresponding genes are homologous defined by BLAST with an E-value cutoff 10^{-3}. A matching [12] of B is defined as a subset of E such that no two edges in the subset share a common vertex. Intuitively a matching represents a one-to-one correspondence between genes in subsets of U and V. For any matching M of B, we define a multigraph $A_M = (O, M)$, with $O = \{U_i; V_j | 1 \leq i \leq m; 1 \leq j \leq n\}$ being the vertex set and M being the edge set. It should be noted that the edge set of B and A_M are the same. In B the vertices are genes and in A_M the vertices are operons, thus there can be multiple edges between two vertices in A_M, so A_M is a multigraph. Define $c(M)$ to be the number of connected components of A_M. An uber-operon identification problem is defined as to find the maximum matching

M of B that maximizes $c(M)$. Intuitively, this formulation attempts to partition B into as highly densely linked (through homologous relationships) operons across the two genomes as possible, particularly to maximize the number of orthologous gene pairs as defined above.

For a general bipartite graph without any constraint, finding the maximum matching can be solved efficiently [12]. However, it is computationally highly challenging to solve the constrained maximum matching problem. We have proved that the uber-operon identification problem, formulated as above, is NP-hard, indicating that there is no fast and rigorous algorithm for solving this problem. So we present a heuristic algorithm for this problem. The basic idea is as follows: we first find non-overlapping individual operon pairs (no operon pairs share the same operon) across U and V that give the highest total matching size among all such operon pairs. This can be achieved by first finding one pair of operons that has highest matching size between any possible operon pairs across U and V; and then remove this pair from B and repeat this procedure on the updated B until no more operon pairs can be found. We then merge operon pairs (or operon-group pairs) into operon-group pairs if such merging can lead to the increase of the overall matching size, or more specifically the objective value. This merge operation is repeated until the objective value cannot be increased any more. The resulting operon groups in U and V are the predicted uber-operons in the two genomes, respectively. Although this heuristic algorithm does not guarantee to reach the globally optimal solution, the following property can always lead this algorithm to reach a good solution: the orthologous gene pairs in two conserved uber-operons (of two genomes) are always denser than that in two unrelated uber-operons.

We now provide a formal description of the algorithm on how to merge the connected groups. We first construct a dynamic auxiliary weighted graph $G(M) = [V(M), E(M)]$ for a given matching M, where the vertex set $V(M)$ consists of all the connected components of $A_M = (O, M)$, and the edge set $E(M)$ is created dynamically by connecting any two connected components of A_M. The weight of the edge e, which is created by connecting two connected components, say C_1 and C_2, is defined as follows: Let M_1 and M_2 be the

current maximum matching of C_1 and C_2, respectively, and be the maximum matching of the subgraph C_{12}, which is created by combining C_1 and C_2, then the weight of edge $e = (C_1, C_2)$ is defined as $w(e) = |M_{1,2}| - |M_1| - |M_2|$. In fact, the weight of an edge is the number of augmenting paths [12] related to M in the subgraph C_{12}. A schematic diagram of our algorithm is shown in Figure 3.1.

Initially, $M = \phi$ (the empty set) and $G(M) = [V[\phi], E(\phi)]$. The algorithm starts to find and merge two connected components where the edge between them has the maximum weight among all edges in $E(M)$ (Figure 3.1); then the algorithm updates the auxiliary graph and the connected components, and repeats the merge operation. The iterative process stops when the maximum weight of edges in $E(M)$ reaches zero. At this point, the final matching M and the final connected components are reached. Though the algorithm does not guarantee to find the globally optimal matching, we found that in practice, the maximal matching M identified by this algorithm is often the globally optimal solution (data not shown). Our algorithm outputs M, which gives orthologous gene pairs between G_1 and G_2, and the connected components determined by M correspond to (part of) uber-operons to be detected.

3.2.3 Uber-operon prediction using multiple reference genomes

For a target genome, our higher-layer algorithm makes the final uber-operon prediction, which is 'maximally' consistent with all the initial predictions by the lower-layer algorithm based on all reference genomes. Generally, the uber-operons predicted based on different reference genomes may be different, because each reference genome might provide different 'reference' information. By effectively combining all these predictions, we could possibly (i) eliminate accidental false predictions due to various reasons, such as false operon prediction in a particular reference genome and (ii) reduce false negative predictions due to the incomplete (reference) information given by any specific reference genome. While more sophisticated 'integration' strategies could be employed, our strategy is to capture the consensus of the initial predictions. This is achieved through a clustering algorithm, described as follows.

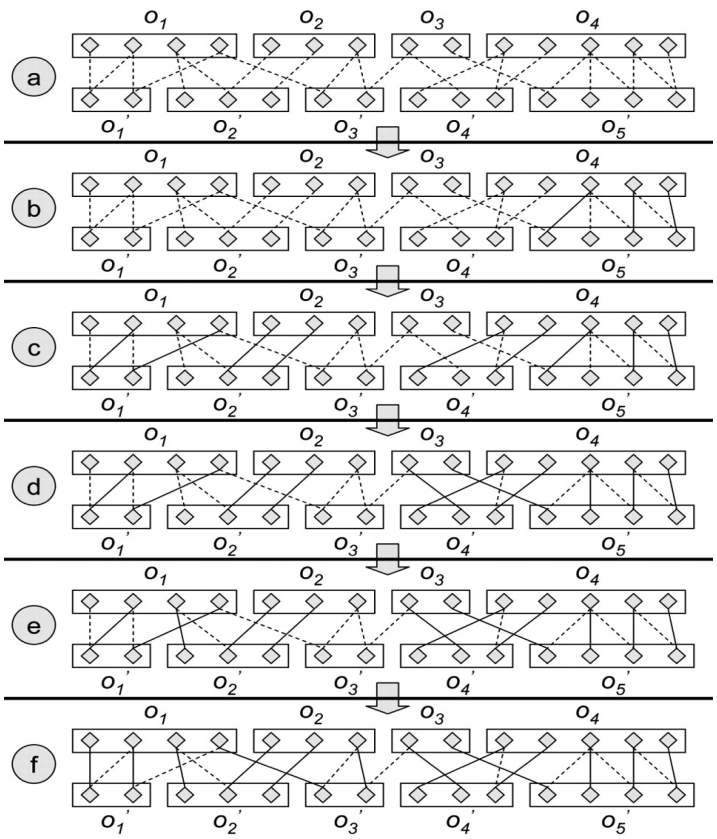


Figure 3.1: A schematic diagram showing how our algorithm works

In each (a, b, c, d, e and f), each row represents genes (diamonds) and operons (rectangles) in each of two genomes. (a) The initial homologous relationship (dashed lines) between the two genomes; each operon is considered as a vertex; (b) the weight of $O_4 - O'_5$ is 3 (because the maximum mapping between them is 3), and it is the maximal among all the weights, so they are merged to one operon group, where the solid lines represent orthologous relationship, and this operon group becomes a new vertex; (c) the weights of $O_1 - O'_1$, $O_2 - O'_2$ and $O'_4 - O_4O'_5$ are 2; they are merged to operon groups and become the new vertices; (d) the weight of $O_3 - O'_4O_4O'_5$ are 2; they are merged into one operon group; it should be noted that when the maximum mapping is re-calculated, one pair of orthologues between O_4 and O'_5 has been re-predicted; the new prediction is more accurate when all the four operons are considered, which represents a correcting mechanism in this algorithm; (e) $O_1O'_1$ and $O_2O'_2$ are merged into one operon group; (f) O'_3 and $O_1O'_1O_2O'_2$are merged into one operon group; it should be noted that when the maximum mapping is re-calculated, some of the predicted orthologous relationships could be different from that by the previous iteration. At the end two uber-operons in each genome are generated.

For N ($N = 90$ in our study) sets of uber-operon predictions based on N reference genomes, we define a weighted graph G as follows: (i) each predicted operon in the target genome is represented as a vertex; (ii) two vertices have an edge between them if and only if the two corresponding operons are predicted to be in the same uber-operon by at least one of the N predictions; and (iii) the weight of an edge is defined to be the number of times that the two corresponding operons are in the same uber-operon among all the N predictions. In general, G consists of a number of connected sub-graphs. A naïve prediction might predict each such connected sub-graph as a uber-operon. However, we have observed that many of these connected sub-graphs are only intra-linked through 'thin' edges (*e.g.* edges with weight 1), which we suspect to be accidental predictions due to various reasons (*e.g.* false operon predictions). To uncover the 'true' uber-operons (with dense linkages), we have used the Markov cluster algorithm (MCL) [`http://micans.org/mcl/`] [45] to partition G into a set of non-overlapping subgraphs (or clusters) whose vertices are densely intra-linked. MCL is used because of its previous successes in graph partitioning with similar characteristics to ours [`http://micans.org/mcl/lit/\#3party`] [48, 99, 124].

The MCL algorithm simulates random walks on a graph using Markov matrices to determine the transition probabilities among the vertices of the graph [45]. By alternating expansion and inflation steps in random walks iteratively, MCL eventually separates a graph into unconnected or loosely connected subgraphs, each of which is densely intra-connected among its vertices. Using a parameter that controls the inflation rate, the MCL algorithm partitions a graph into 'densely' intra-connected subgraphs at different levels of granularity. The inflation rate in MCL varies from 2.0 to 5.0. We have applied the algorithm using four different inflation rates (2.0, 3.0, 4.0 and 5.0) and obtained graph partitions with different levels of granularity. For any fixed inflation rate, we predict the vertices of each partitioned subgraph as a uber-operon. The overall procedure is given in Figure 3.2.

In our prediction for *E. coli* K12, we have compared our predicted uber-operons using different inflation rates, 2.0, 3.0, 4.0 and 5.0 with KEGG pathways, EcoCyc regulons and

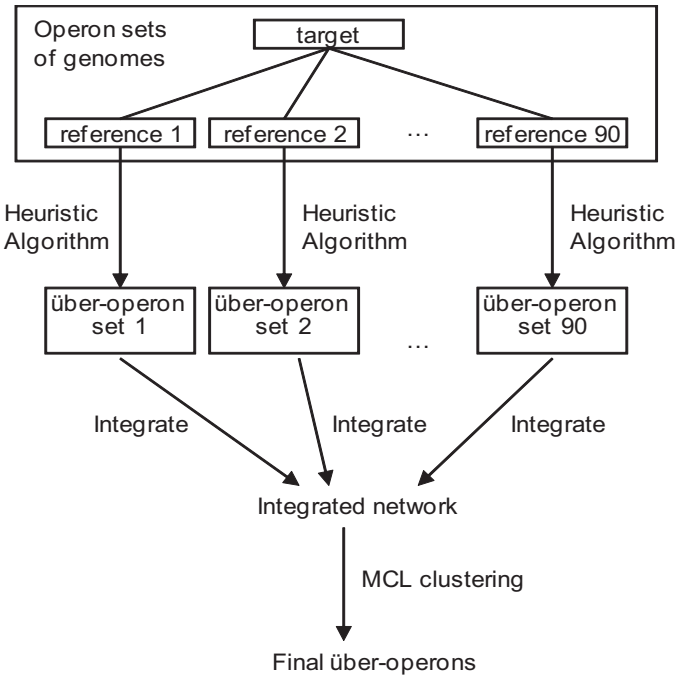


Figure 3.2: An overview of the uber-operon prediction procedure

GO annotations (see Results and Discussion), and found that the difference between uber-operon predictions by using different inflation rates is small. There are two possible reasons: (i) MCL has indeed captured some intrinsic 'cluster' information in the graph, so it is not very sensitive to the inflation rates. A similar observation is also made for a recent study on accurate orthologs predictions, using MCL [166]. (ii) Our comparison is against biological processes at different levels, including pathways, super- and sub-pathways [84]. Hence a slight over- or under-prediction of uber-operons may not quite be reflected by such comparisons. We have chosen uber-operon predictions at the inflation rate = 5.0 as our default prediction.

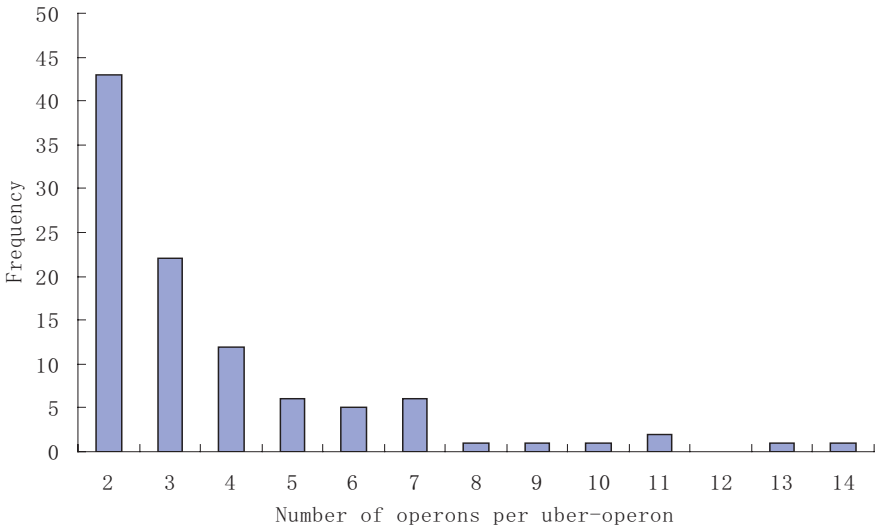


Figure 3.3: Frequency distribution of the number of operons in a uber-operon in *E. coli*

3.3 Results and Discussions

We have predicted uber-operons for 91 genomes using the method described in Materials and Methods. For prediction for each genome, we use the other 90 genomes as the references. To evaluate these predictions, we have performed a detailed analysis on the uber-operon predictions in *E. coli*, and assessed the prediction reliability based on known information about *E. coli*. For this genome, we have predicted 158 uber-operons, covering 578 operons and 1830 genes. The size distribution of all the predicted uber-operons in *E. coli* in terms of the number of included operons is given in Figure 3. 3. As we can see, most of the predicted uber-operons contain two or three operons, though a few uber-operons have more than ten operons. It can be checked that this distribution follows a power law distribution.

3.3.1 Analysis of predicted *E. coli* uber-operons

Because there is no dataset of experimentally verified uber-operons, we have used three types of information to assess the soundness of our predicted uber-operons, in terms of both biology and statistics. They are (i) experimentally verified regulons of *E. coli* (15), (ii) experimentally verified pathways in *E. coli* [79], and (iii) GO assignments for *E. coli* genes [4].

Comparison between predicted uber-operons and regulons

We have collected 153 *E. coli* regulons from the EcoCyc database [84]. Our hypothesis is that many of the predicted uber-operons each belong to a regulon. So we use the following approach to compare the consistency between the predicted uber-operons and the known regulons in *E. coli*. We understand that both the uber-operon predictions and known regulons represent only a fraction of all the uber-operons and regulons in the genome, due to the possible incompleteness of our prediction and experiments. So we have taken this into consideration in our analysis. The basic idea of our comparison is given as follows [some of the ideas have been used for a different application [165].

Let $A = \{a_i\}$ be a gene list and $P = \{p_i, 1 \leq i \leq m\}$ and $Q = \{q_j, 1 \leq j \leq n\}$ be its two partitions. The matching degree ($MD_{i,j}$) between p_i and q_j is defined as:

$$MD_{i,j} = \frac{|p_i \cap q_j|}{|p_i \cup q_j|} \tag{3.1}$$

and the highest matching degree (HMD) achieved by Q for p_i is defined as:

$$HMD_{p_i} = max_{j=1}^{n}(\frac{|p_i \cap q_j|}{|p_i \cup q_j|}) \tag{3.2}$$

The average highest matching degree ($AHMD$) achieved by Q for P is defined as:

$$AHMD_P = \frac{\sum_{i=1}^{m} HMD_{p_i}}{m} \tag{3.3}$$

The matching degree ($MD_{i,j}$) gives the similarity between two subsets: p_i and q_j. The HMD for p_i (HMD_{p_i}) gives the subset in Q that achieves the highest similarity with p_i. The $AHMD$

measures the similarity between P and Q. In our analysis, P represents the available regulons or pathways while Q is the predicted uber-operons. Though some of regulons/pathways may have overlaps, it should not have serious effects on our overall evaluation because of the overlaps in general are small compared to the size of the gene list. We have found that when both P and Q are fully available, we can use a more accurate formula as given in definition (3.4) to more accurately measure the similarity between P and Q.

$$AHMD = \frac{\sum_{i=1}^{m} HMD_{p_i} + \sum_{j=1}^{n} HMD_{q_j}}{m+n} \tag{3.4}$$

Note that in this definition (3.4), $AHMD$ is symmetrical with respect to P and Q.

For each set of the predicted uber-operons U, we calculated the $AHMD_R(U)$ between U and the known regulons R using definition (3.3). The $AHMD_R(U)$ value is 0.159 (Table 3.1). To assess the statistical significance of this obtained $AHMD_R(U)$ value, we have calculated its Z-score as follows. We first constructed a set of pseudo uber-operons U', by randomly combining the predicted operons such that the i^{th} pseudo uber-operon has the same number of operons as the i^{th} uber-operon in U. We constructed 100 such sets of pseudo uber-operons, and calculated their $AHMD_R(U')$ values. The Z-score of $AHMD_R(U)$ is computed as

$$Z_R(U) = \frac{AHMD_R(U) - \overline{AHMD_R(U')}}{\sigma_{AHMD_R(U')}} \tag{3.5}$$

with $\overline{AHMD_R(U')}$ being the average $AHMD_R(U')$ value and $\sigma_{AHMD_R(U')}$ the standard deviation. We obtain a Z-score 4.091 for $AHMD_R(U) = 0.159$, indicating that the matching between the predicted uber-operons and the known regulons is highly significant.

Comparison between predicted uber-operons and pathways

We have carried out a similar comparison between the predicted uber-operons (denoted as U) and all the known pathways (denoted as P) in *E. coli* as given in KEGG [79], and calculated the $AHMD_P(U)$ value and its Z-score, using the same procedures outlined above. The value of $AHMD_P(U)$ is 0.115, and its Z-score is 4.091 (Table 3.2). This result again suggests that the matching between the predicted uber-operons and known pathways is highly significant.

Table 3.1: AHMDs between predicted uber-operons and regulons

IR	PathAHMD	Random PathAHMD(sd)	Z-score
2.0	0.098	0.066(0.0058)	5.637
3.0	0.107	0.082(0.0063)	3.991
4.0	0.112	0.085(0.0069)	3.93
5.0	0.115	0.085(0.0071)	4.091

Table 3.2: AHMDs of between predicted uber-operons and pathways

Inflation rate	RegAHMD	Random RegAHMD(sd)	Z-score
2.0	0.125	0.078(0.0093)	4.979
3.0	0.166	0.104(0.011)	5.76
4.0	0.166	0.107(0.011)	5.173
5.0	0.159	0.110(0.012)	4.145

We have also assessed the matching between known regulons and pathways by calculating $AHMD_P(R)$. The $AHMD_P(R)$ is 0.090, slightly smaller than $AHMD_P(U)$. This seems to make good sense because the uber-operons cover not only the operons that are co-regulated but also the operons that work together, say, in the same pathway, while being regulated possibly by different mechanisms. These two similar $AHMD_P$'s indicate the relationship between genes in the same uber-operon is at least as tight as genes in the same regulon.

Relationship between GO assignments and predicted uber-operons

We have assessed the statistical significance of the predicted uber-operons in terms of their GO assignments. Among the three levels of GO functionalities, namely, molecular function, biological process and cellular component, we have used GO's biological processes to compare genes assigned to the same uber-operon. The GO term assignments for *E. coli* were

retrieved from Integr8 (`http://www.ebi.ac.uk/integr8/EBI-Integr8-HomePage.do`). We have previously developed a method for comparing two GO biological processes [165]. For two genes g_1 and g_2, we define as the similarity score between their GO biological processes, as defined in [165]. We then measured the overall consistency of GO assignments for the genes in a predicted uber-operon using the following formula.

$$S_{go} = \frac{1}{L} \sum_{i=1}^{r} \sum_{j=i+1}^{r} \sum_{k=1}^{s_i} \sum_{l=1}^{s_j} d_{k,l} \tag{3.6}$$

where L is the total number of gene pairs across operons in the uber-operon, r is the number of operons in the uber-operon, s_i and s_j are the numbers of genes in the i^{th} operon and j^{th} operon of the uber-operon, respectively, and $d_{k,l}$ is the similarity score for the k^{th} genes in i^{th} operon and l^{th} gene in j^{th} operon. We have calculated the average S_{go} for all the predicted uber-operons in *E. coli*, denoted as AS_{go}, as,

$$AS_{go} = \frac{1}{n} \sum_{i=1}^{n} S_{go}(U_i) \tag{3.7}$$

where n is the number of uber-operons in the genome, and $S_{go}(U_i)$ is S_{go} for i^{th} uber-operon. We have obtained $AS_{go} = 3.561$ (see Table 3.3). For Z-score estimation, we have calculated the AS_{go} values for 100 pseudo uber-operons, and obtained a Z-score 9.579 for $AS_{go} = 3.561$, indicating that the similarity among the functionalities of genes from the same uber-operons across all predicted uber-operons are highly significant. As a reference, we have also calculated AS_{go} for all known *E. coli* regulons, and obtained $AS_{go} = 4.32$. The similar values between the two AS_{go} indicate that the functional similarity among genes from the same uber-operon is quite comparable to that among genes from the same regulon.

In summary, these analyses have shown that our predicted uber-operons are biologically and statistically meaningful. A detailed list of 158 predicted uber-operons in *E. coli*, in terms of its component operons, genes and their functions, is provided in the Supplementary Table S2 of [24]. Numerous predicted uber-operons contain ABC transporter systems, which is consistent with the KEGG where many pathways contain ABC transporter systems. Some of the predicted uber-operons are not associated with any known KEGG pathways, which

Table 3.3: GO scores of predicted uber-operons

Inflation rate	ASgo	Random ASgo(sd)	Z-score
2.0	3.419	2.861(0.079)	7.102
3.0	3.511	2.864(0.069)	9.378
4.0	3.509	2.850(0.068)	9.691
5.0	3.561	2.855(0.074)	9.579

might indicate that they belong to pathways that are yet to be elucidated. For instance, two operons containing *csgA*, *csgB*, *csgD*, *csgE*, *csgF* and *csgG* have been predicted to form a uber-operon. These genes have not been previously reported to be involved in any known KEGG pathway, but their genes are known to belong to the same regulon based on known EcoCyc regulons. We expect such predicted uber-operons, particularly the ones not known to belong to the same regulons, will provide a highly useful information source for discovery of novel pathways and regulons.

3.3.2 Case studies: detailed analyses of three examples of uber-operons

We now further demonstrate the quality of the predictions by providing detailed analysis of three predicted uber-operons, which are involved in the flagellar system, tricarboxylic acid (TCA) cycle and sulfur metabolism, respectively. These examples highlight the possibility of using uber-operon prediction for elucidation of regulons and the component genes of pathways.

Flagellar assembly

The bacteria flagellum is the motor organelle for propulsion, driven by the transmembrane proton motive force. The full function of flagella requires the expression of more than 50 genes, including structural genes, chemotaxis-related genes, and possibly other related genes

[33]. We have predicted one uber-operon consisting of 54 genes from 10 operons. Among the 54 genes, 30 genes (*flgB*, *flgC*, *flgD*, *flgE*, *flgF*, *flgG*, *flgH*, *flgI*, *flgJ*, *flgK*, *flgL*, *flhE*, *flhA*, *flhB*, *fliA*, *fliD*, *fliS*, *fliF*, *fliG*, *fliH*, *fliI*, *fliJ*, *fliK*, *fliL*, *fliM*, *fliN*, *fliO*, *fliP*, *fliQ*, *fliR*) are known to be in the pathway of the flagellar assembly according to the KEGG database, 12 genes (*cheZ*, *cheY*, *cheB*, *cheR*, *tap*, *tar*, *cheW*, *cheA*, *motB*, *motA*, *flhC*, *flhD*) are known to be in the chemotaxis pathway, and the remaining 12 genes are involved in cell division and other biological processes, based on their GO annotation. In [94], Lathe *et al.* used four reference genomes to predict uber-operons and identified flagellar uber-operon genes. While we found some level of agreement between our uber-operon prediction and the corresponding uber-operon in [94], we noticed that a number of genes in our uber-operon, annotated as possibly flagellar-related by GO, are not reported by Lathe *et al.*, such as *fliZ* and *fliT*. For instance, *fliZ* is annotated as the putative regulatory gene on *fliA*. Interestingly, a cell division related gene, *minD*, seemingly not related to the flagellar system, is found both in our predicted uber-operon and in the Nebulon system [74]. The association of the cell division and the flagellar system clearly warrants further experimental investigation.

TCA cycle

TCA cycle is a common pathway in mitochondria. It starts with oxidizing acetyl CoA, which is the product from the oxidative decarboxylation of pyruvate, and goes through a ten-step reaction process that yields energy and CO_2. We have predicted one uber-operon consisting of three operons covering nine genes: *sdhC*, *sdhD*, *sdhA*, *sdhB*, *b0275*, *sucA*, *sucB*, *sucC*, *sucD*, eight of which, except for *b0275*, are known to be involved in the TCA cycle pathway, as reported in KEGG. Further analysis indicates that these eight genes are predicted to be in one operon in six other genomes, *i.e. Candidatus Blochmannia* floridanus, *Coxiella burnetii* RSA 493, *Legionella pneumophila* str. Paris, *Neisseria meningitidis* MC58, *Photobacterium profundum* SS9 and *Vibrio vulnificus* YJ016. The functionality of gene *b0275* is unknown at this point. Our BLAST search did not reveal any homologous genes in other genomes,

suggesting that it might represent a unique gene involved in the *E. coli* TCA process. This uber-operon does not include other genes known to be in the pathway, such as *frdD*, which encodes fumarate redutase. This indicates that the gene rearrangement might have occurred locally, *i.e.* within succinate related genes.

Sulfur metabolism

Sulfur metabolism is one of the most important components in energy metabolism in *E. coli*, which consists of synthesis and catabolism of the sulfur-containing amino acids, such as cysteine and methionine [140]. Our predicted uber-operon contains two operons covering seven genes, six of which are involved in the sulfur metabolism pathway, *i.e. cysC*, *cysN*, *cysD*, *cysH*, *cysI* and *cysJ*. *ygbE* is annotated as a putative cytochrome oxidase subunit. We have not been able to find its homologous gene in the corresponding uber-operons in other genomes, and so far no literatures have suggested that cytochrome oxidase is involved in the process of this metabolism. This seemingly displaced gene could possibly be explained by the 'selfish operon' [95] hypothesis. In the 'selfish operon' model, an operon deletes its unused genes through horizontal transfers, and only useful genes are retained. The gene *ygbE* may represent a trace of incomplete evolution, and the *cysCNDHIJ* genes may represent the 'useful' genes suggested by the 'selfish operon' hypothesis. This in turn indicates that our method could tolerate some level of noise, *i.e.* irrelevant genes in some operons.

3.3.3 Novel uber-operons

Our prediction includes a set of putative uber-operons, which haven not been confirmed by any known pathways or regulons, though they are highly statistically significant. GO assignments cannot reveal much clue about the biological processes in which the involved genes participate, either. Supplementary Table S3 in [24] summarizes this set of predicted uber-operons and the possible biological processes in which they might be involved, according to individual gene annotations. To show what possible biological processes these putative

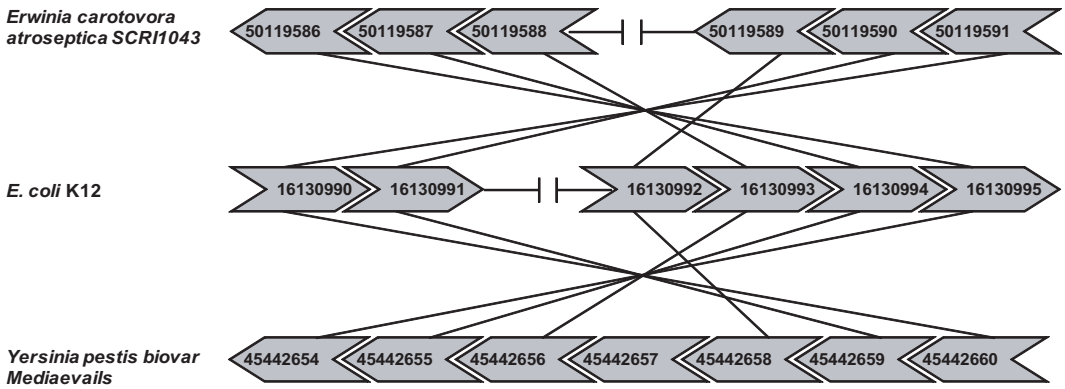


Figure 3.4: Membrane protein-related uber-operon.

uber-operons might suggest biologically, we provide two examples to show how the uber-operon prediction could possibly be further explored.

Membrane proteins

One of the predicted uber-operons contains six genes (*yqjA*, *yqjB*, *yqjC*, *yqjD*, *yqjE* and *yqjK*) from two operons in *E. coli*, and has its corresponding uber-operons predicted in a few reference genomes, including *Erwinia carotovora atroseptica SCRI1043* and *Yersinia pestis biovar Medievalis*. All these six genes in *E. coli* have their orthologous genes belong to one operon in *Y. pestis biovar Medievalis*, and have orthologous genes that belong to two operons in *Erwinia carotovora atroseptica SCRI1043* (see Figure 3.4). The conservation of these genes indicates the significance of this novel uber-operon. The genes of this uber-operon encode integral membrane proteins, although their detailed collective functionality is unknown to date.

Rhs-family related proteins

The *Rhs* family consists of at least five *Rhs* elements in *E. coli*, with the most prominent *Rhs* component containing extended repeated regions and often participating in ligand-binding processes in the cell surface [68]. Our uber-operon prediction indicates that gene *b0499* and gene *b1456*, belonging to two different operons in *E. coli*, have their predicted orthologous genes *SF0267* and *SF0268* belong to the same operon in *Shigella flexneri 2a* str. 301. We have also observed that gene *b0499* and gene *b3428*, belonging to two different operons in *E. coli*, have their predicted orthologous genes *CV1238* and *CV1239* belong to the same operon in *Chromobacterium violaceum* ATCC 12 472. All these genes are annotated as *Rhs*-family proteins or putative *Rhs*-family proteins in the NCBI microbial genome database. The initial prediction of two uber-operons, one based on *S.flexneri 2a* str. 301 and the other based on *C.violaceum* ATCC 12 472, respectively, ultimately leads to the final prediction of a combined uber-operon, which contains three operons including three genes *b0499*, *b1456* and *b3428*. Unlike the previous example, this putative uber-operon does not seem to have a corresponding prediction in other genomes. While we do not rule out the possibility of a false prediction, we do suspect that these genes work together as a unit as their proteins are mostly annotated as the *Rhs* family related. We believe that this prediction warrants further experimental investigation.

3.4 Concluding Remarks

We have developed a new framework for identification of uber-operons, which represent a class of genomic structure yet to be fully investigated, and record the footprints of operon evolution. Uber-operons may prove to be highly useful for elucidation of biological pathways. Our analyses on the predicted uber-operons, in terms of the statistical significance, evolutionary conservation and functional relatedness among their component genes, have indicated that this concept is well founded, though further investigation and refinement

might be needed. We can see a number of important applications of our uber-operon prediction capability. (i) The component genes of a predicted uber-operon could suggest possible candidate genes in a particular biological process, such as a pathway, which has higher gene coverage than operons. (ii) Many of the predicted uber-operons seem to be parts or even whole regulons, based on our analyses. Hence, this could possibly lead to an effective way for regulon prediction. As of today there is no publicly available computer program for regulon prediction. Uber-operon-based approach could become the first general approach to regulon prediction. (iii) If we consider genes in an operon as tightly coupled working unit in a biological process, uber-operons might provide lists of genes that are less tightly coupled, possibly including genes responsible for different biological functions in a complex biological network. Specifically, a uber-operon might include genes involved in both metabolic and regulatory functions, providing richer information for elucidation of complex biological networks.

Chapter 4

PFP: A Computational Framework for Phylogenetic Footprinting in Prokaryotic Genomes[1]

[1]D. Che, G. Li, S.T. Jensen, J.S. Liu, and Y. Xu. *Lecture Notes in Bioinformatics*, 2008, 110-121. With kind permission of Springer Science+Business Media

4.1 Introduction

Phylogenetic footprinting is a method for identification of *cis* regulatory elements in promoter regions of orthologous genes across species [147]. This strategy attempts to find conserved sequence motifs in the provided promoter regions based on the assumption that functional elements, such as transcription factor binding sites, evolve more slowly than non-functional elements over time. A prerequisite for using a phylogenetic footprinting approach is the mapping of orthologous genes across multiple genomes (often called *reference* genomes). A number of orthology mapping approaches, mainly sequence similarity-based such as COG [151] and OrthoMCL [99], have been widely used. By applying such orthology mapping methods to eukaryotic genomes, a number of research groups have carried out studies on identification of *cis* regulatory motifs at a genome scale. For example, Wang *et al.* [162] developed PhyloNet to search for regulatory motifs in *Saccharomyces cerevisiae* by using three other yeast genomes as reference genomes and identified more than 90% of the known TFBSs in *Saccharomyces cerevisiae*. Using several mammalian genomes as references, Xie *et al.* [167] successfully identified a number of transcription regulatory motifs in the human genome.

A similar phylogenetic footprinting strategy may not be directly applicable to prokaryotic genomes due to their different genomic structures from the eukaryotic ones. Typically about half of the genes in a prokaryotic genome are *polycistronic*, *i.e.*, they are organized into multi-gene transcriptional units (or multi-gene operons), genes of each of which share a common promoter and terminator. Multi-gene operons add a new challenge to the identification problem of orthologous promoter regions: promoters are associated with operons rather than individual genes and may not necessarily be conserved across multiple genomes. Thus, relationships between operons across genomes are more complex in general than those between orthologous genes. In addition, the sequence similarity-based approach cannot correctly characterize orthologous relationships in some cases. For prokaryotes, the true orthology can be elucidated by deriving conserved operons across multiple genomes. This is because that

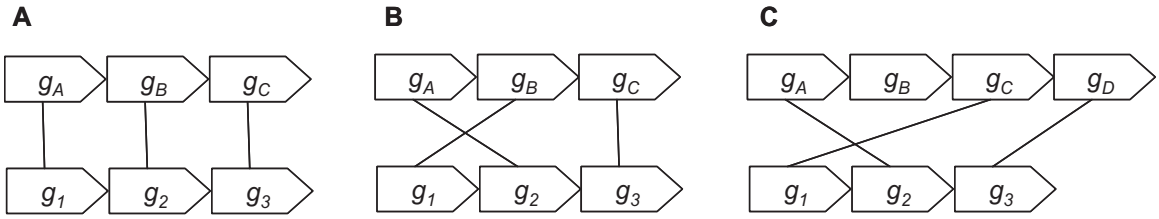


Figure 4.1: Three categories of operon conservation.

Boxes represent genes and consist of an operon. Lines indicate sequence similarity between two genes. (A) Conserved with both gene list and order; (B) Conserved with gene list only; and (C) Partially conserved.

homologous genes are more likely to be orthologous if their neighboring genes within an operon are also homologous [166].

Numerous computational methods have been developed to predict operons in prokaryotic genomes, including OFS [163], OPERON [50], OperonDT [26], VIMSS [126], and UNIPOP [98]. The prediction accuracy of the best programs has reached $\sim$ 90% on several model genomes such as *E. coli* and *Bacillus subtillis* [40]. It has been previously observed that "conserved" operons may only have their gene list conserved but not necessarily the gene order within the list. In this study, we consider both cases: *category-1* for conserved operons with both conserved gene list and order and *category-2* for conserved operons with only conserved gene list (Figure 4.1A and 4.1B). In addition, we also have considered *category-3* for partially conserved operons, which is defined as follows: two operons from different genomes are partially conserved if they have at least one pair of orthologous genes (Figure 4.1C). Clearly, the multiple scenarios of operon conservation complicate the derivation of orthologous upstream sequences for the purpose of phylogentic footprinting analysis in prokaryotic species.

Previous work on extracting promoter sequences of orthologous genes for phylogenetic footprinting analysis has been done in a simplistic manner. Basically, orthologous genes are

collected using sequence similarity-based approaches, then the intergenic sequences of individual genes with the upstream region of its predicted operon are concatenated [107, 108, 109]. This strategy has also been used in a recent computational tool 'microFootPrinter' [116]. To address the issue of including upstream sequences for internal genes in an operon, Jensen *et al* [76] considered only the "promoter" regions of genes with upstream intergenic regions longer than 50 bp (called *beginning* genes of an operon). This approach is also problematic since it considers only operons that have both conserved gene list and gene order. There remains a need for more careful and more accurate treatment of the "corresponding" promoters of orthologous genes in prokaryotes.

In this paper, we derive conserved operons among multiple genomes for phylogenetic footprinting analysis and provide a superior treatment of promoter regions of orthologous genes. To fully consider all operons with different levels of evolutionary conservation, we designed an algorithm, *OPERMAP*, to find operons across reference genomes. By applying this algorithm, we have identified 2,706 *E. coli* operons that are conserved across multiple (reference) genomes. In addition, we have developed a pipeline consisting of multiple motif discovery programs for the prediction of conserved sequence motifs. Performance comparison on known binding sites of *E. coli* suggests that our approach tend to generate more reliable orthologous promoter regions (*i.e.*, regions containing the binding sites for orthologous TFs) than previous approaches for motif finding at the genome scale in prokaryotes.

4.2 Methods

We divide our procedure of phylogenetic footprinting in prokaryotes into five steps:

1. Selecting reference genomes for a target genome;

2. Predicting operons of all selected genomes at genome-scale;

3. Predicting conserved operons across selected genomes;

4. Obtaining promoter sequences of conserved operons;

5. Predicting binding sites using our motif-finding pipeline.

Below, we present the details of each step.

Reference Genome Selection

Selecting suitable reference genomes for comparison to the target genome of interest is a key step in the phylogenetic footprinting process. A candidate reference genome should be phylogenetically close to the target genome. A large list of candidate genomes is not essential since using a large number of genomes for motif discovery does not seem to improve performance [72]. This has also been observed in our experiments (data not shown). Accordingly, our selection strategy is to choose 10-15 reference genomes belonging to the same class with similar genome sizes to that of the target genome.

In this study, *E. coli* K12 is our target genome and 11 other γ-proteobacteria were chosen as reference genomes. The names and genome sizes of 12 genomes are listed as follows: *Aeromonas hydrophila* ATCC_7966 (4.6 Mb), *Erwinia carotovora atroseptica* SCRI1043 (4.9 Mb), *E. coli* K12 (4.5 Mb), *Photobacterium profundum* SS9 (6.3 Mb), *Photorhabdus luminescens* (5.6 Mb), *Pseudomonas fluorescens* Pf-5 (6.9 Mb), *Salmonella enterica* Choleraesuis (4.9 Mb), *Shewanella ANA* 3 (5.2 Mb), *Shigella sonnei* Ss046 (4.9 Mb), *Sodalis glossinidius morsitans* (4.2 Mb), *Vibrio parahaemolyticus* (5.1 Mb) and *Yersinia pestis* Antiqua (4.8 Mb).

Operon Prediction

For each of the selected genomes, operon prediction at the genome scale is performed using our own program UNIPOP [98]. We choose UNIPOP because it outperforms other operon prgrams in terms of prediction accuracy. In addition, unlike most of operon programs, UNIPOP does not need extra feature information (*i.e.*, gene function annotation), which is not available for newly sequenced genomes. The key idea of UNIPOP is to predict operons through identification of conserved gene clusters across multiple genomes. Briefly,

given a target genome and N reference genomes, we predict N versions of operon maps for the target genome by comparing and deriving conserved gene clusters between the target genome and each of the reference genomes. We consider two sets of contiguous genes from two genomes to be conserved gene clusters (or operons) if the following conditions are satisfied: a). Each member of a gene cluster is transcribed in the same direction; b). The total intergenic distance within each group is less than the maximum allowed distance (MAD); c). The number of mappings of homologous gene pairs between two groups is at least two. We then obtain a consensus version of operon map using a voting scheme on N versions of operon maps. In this study, operon structures for each of the 12 genomes were predicted by using 348 reference genomes from the NCBI GenBank database (`http://www.ncbi.nlm.nih.gov/genomes/lproks.cgi`).

Identification of Conserved Operons

Having predicted operon structures for the 12 species, we need to identify "orthologous" operons among these prokaryotes. We have developed an algorithm, called *OPERMAP*, to identify the corresponding conserved operon in a particular reference genome for a given query operon in the target genome. We now describe the *OPERMAP* approach in detail as follows.

The input to the algorithm consists of

1. a query operon U in the target genome,
2. a collection of all predicted operons $[V_1,\ V_2,\ \ldots,\ V_k]$ in the reference genome, and
3. a threshold for the degree of conservation (TDC) between two operons.

The output of the program is the operon pair $(U,\ V^*)$ between the query operon U and the best conserved operon V^* from the reference genome. The algorithm proceeds as follows:

1. Calculate the degree of conservation between query operon U and each candidate operon $[V_1,\ V_2,\ \ldots,\ V_k]$ in the reference genome.

(a) For each operon $V_i \in [V_1, V_2, \ldots, V_k]$, construct a bipartite graph $G_i = (U, V_i, E_i)$, where all the genes in U and all the genes in the i-th operon V_i are represented as vertices. A pair of genes is considered to be *homologous* if their reciprocal BLAST e-values are both $< 10^{-6}$, and a homologous relationship between a gene in U and a gene in V_i is represented by an edge in E_i. The weight of each edge in E_i is set to be the average of -log(e-value) of the BLAST between the pair of genes.

(b) Calculate the maximum weighted maximum cardinality bipartite matching (*mwmcm*) M_i on each graph G_i, in a similar fashion to that of [110]. Each matched edge in *mwmcm* reflects the orthology relationship between the pair of genes.

(c) Calculate the degree of conservation $DC_i = |M_i| \ / \ max\ (|U|, |V_i|)$, where $|X|$ is the cardinality of the set X.

2. The best conserved operon pair (U, V^*) is the operon pair with the highest degree of conservation DC_i. This best operon pair is reported only if the degree of conservation is higher than the predefined threshold TDC; otherwise, no conserved operon pair is returned.

The core of this algorithm is to calculate *mwmcm*. A *matching* in a graph $G = (V, E)$ is a subset M of the edges E such that no edges in M share a common vertex, and a *maximum cardinality matching* (*mcm*) is a matching with the highest possible cardinality. An *mwmcm* is a *mcm* with the maximum total weight (see Figure 4.2 for an example). In this study, the edge relationship in an *mwmcm* represents the orthology relationship between the two corresponding operons. Using the scheme of *mwmcm* to identify the best conserved operon in *OPERMAP* has several advantages. First, it is guaranteed to find the maximum number of homologous gene relationships between two operons. Second, it can find the true orthogolous

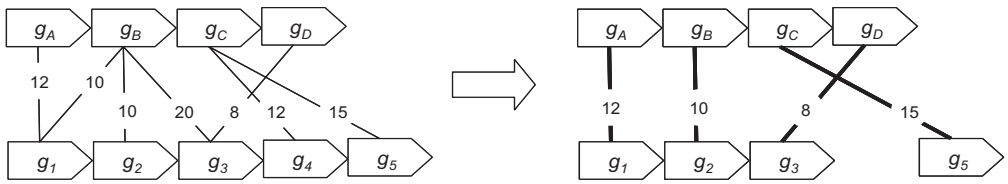


Figure 4.2: An illustration of a maximum weight maximum cardinality matching (*mwmcm*)

The resulting matching is shown on the right, with the matching size of 4. While the weight of the edge g_B - g_3 is 20, the *mwmcm* does not choose it. Otherwise, the matching size will be 3.

gene pair based on sequence similarities in the case where there are several *mcms*, provided that an appropriate weighting scheme is given.

By applying *OPERMAP* on all reference genomes, we can obtain a set of conserved operons for a given query operon in the target genome. For each query operon out of 2,706 predicted operons in *E. coli*, we have applied *OPERMAP* on the 11 reference genomes. In this study, we want to cover not only fully conserved operons (*category*-1 and *category-2*), but also partially conserved operons (*category-3*). Including partial conserved operons has its biological reasoning. Some large operons can break into multiple smaller operons with some part of these smaller operons still maintaining the same regulation mechanism. For instance, a Crp-regulated *xylFGHR* operon in *E. coli* breaks into *xylFGH* and *xylR* in *H. influenzae*, with *xylFGH* maintaining Crp regulation, but *xylR* not [148]. Setting a low value of *TDC* (*i.e.*, < 0.5) may introduce partial conserved operons with different regulation mechanisms. On the other hand, setting a high value of *TDC*(*i.e.*, > 0.8) will exclude most of partial conserved operons with *category-3* since the sizes of most operons are less than five. We have chosen 0.6 for *TDC* in this study.

Collection of regulatory sequences

The gene annotations and the genomic sequences of the 12 genomes in this study were downloaded from the NCBI GenBank database (`http://www.ncbi.nlm.nih.gov/genomes/lproks.cgi`). For each operon obtained in the previous step, we extract the upstream sequence up to 400 base-pairs (bp) from the translation start site, without overlap of the next upstream gene.

Motif Discovery

The upstream promoter sequences for each conserved operon are the input for our motif discovery pipeline to identify (possibly multiple) TFBSs. The pipeline is similar to our previously developed tool BEST [24], which contains four motif-finding programs: AlignACE [133], BioProspector [100], CONSENSUS [67] and MEME [7], as well as BioOptimizer [77] for optimizing the predictive power of each program. However, BEST is a graphic tool which makes it less suitable for the genome scale motif discovery. Our pipeline overcomes this drawback to produce top-ranked motifs for each sequence dataset in a fully automatic fashion. We outline our motif discovery pipeline in three stages (also see Figure 4.3).

1. Run the four motif-finding programs mentioned above. Since the motif length in all the four programs must be specified by the user, each program is run multiple times with different motif lengths ranging from 10 to 20 bp. The range of motif lengths chosen is based on the fact that most experimentally verified motifs fall in this range. For each width and each program, the top-ranked motif is collected, giving a set of $4 \times 11 = 44$ top-ranked motifs.

2. The BioOptimizer program is run on each of the 44 motifs. BioOptimizer takes each predicted motif as the starting point and optimizes it using a local hill-climbing technique [77].

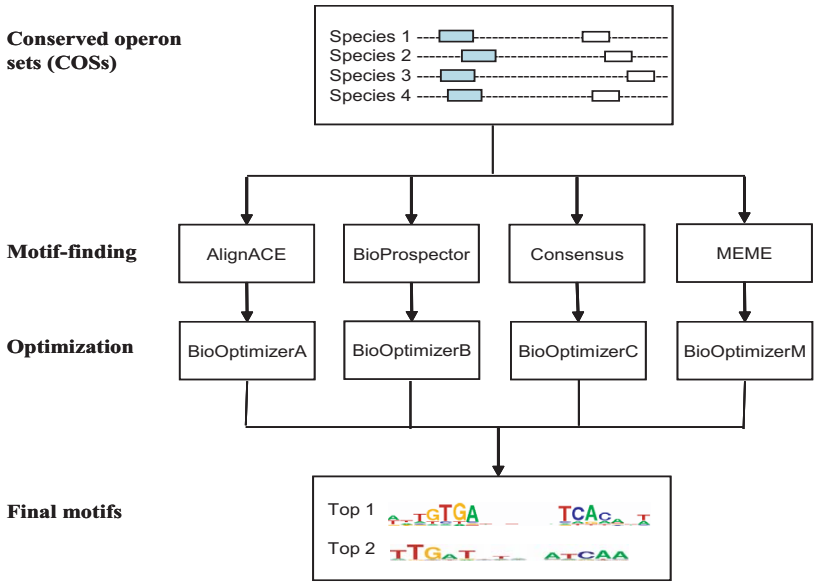


Figure 4.3: The workflow of motif discovery.

3. Rank all 44 optimized motifs based on their score values calculated by BioOptimizer, and output the top five.

Performance Evaluation

We validate our motif predictions with a similar approach to past motif discovery investigations. We define as *true positives* (TP) the predicted binding sites which overlap with the true binding sites by at least 50%; *false positives* (FP) are the predicted binding sites which have no such overlap; *false negatives* (FN) are the true binding sites that have no overlap with any of the predicted binding sites. We focus on four validation measures, sensitivity (Sn), specificity(Sp), performance coefficient (PC), and F-measuer (F), which are defined as follows:

$$Sn = TP/(TP + FN) \tag{4.1}$$

$$Sp = TP/(TP + FP) \tag{4.2}$$

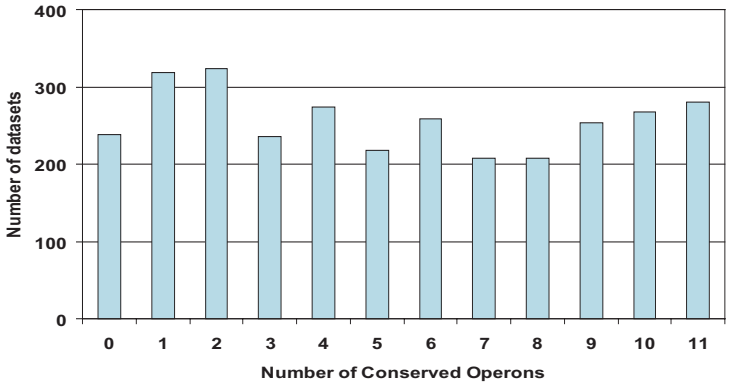


Figure 4.4: The operon conservation histogram for 2706 predicted operons of *E. coli*.

X-axis indicates the number of conserved operons in 11 other species, and y-axis indicates the number of conserved operons with the conservation number ranging from 0 to 11.

$$PC = TP/(TP + FN + FP) \tag{4.3}$$

$$F = 2 * Sn * Sp/(Sn + Sp) \tag{4.4}$$

4.3 Results

Collection of Conserved Operons

The genome sizes of our 12 genomes range from 4.2 Mb to 6.9 Mb, and the numbers of predicted operons ranged from 1596 to 4468. For each of the 2,706 predicted operons in *E. coli*, we ran *OPERMAP* to identify conserved operons in the 11 reference genomes. The distribution of the number of conserved operons across the twelve genomes is shown in Figure 4.4. Two hundred and thirty-eight operons (8.8%) from *E. coli* do not have a corresponding operon match in any of the 11 reference genomes, which may indicate that those operons are either unique to *E. coli* or have been predicted incorrectly by UNIPOP. At the opposite extreme, 280 operons (10.3%) are conserved across all 11 reference genomes.

Table 4.1: Prediction accuracy of motif-findings on 10 TFBSs of *E. coli* using the PFP approach.

TFs	ArgR	Crp	Fis	Fnr	Fur	IHF	LexA	Lrp	MetJ	SoxS
Sn	0.682	0.64	0.5	0.655	0.761	0.5	0.926	0.467	0.818	0.722
Sp	0.205	0.094	0.113	0.113	0.181	0.066	0.116	0.109	0.138	0.088
PC	0.188	0.089	0.102	0.107	0.172	0.061	0.115	0.097	0.134	0.086
F	0.316	0.163	0.185	0.193	0.293	0.116	0.206	0.177	0.237	0.158

Performance of TFBS Predictions

Our evaluation was restricted to predicted motifs in conserved operon sets in *E. coli* since experimentally-verified binding-sites are not available in the 11 reference genomes. We retrieved verified binding sites of *E. coli*, grouped by transcription factors, from the PRODORIC database [113]. We focus on the binding sites regulated by the following ten transcription factors: ArgR, Crp, Fis, Fnr, Fur, IHF, LexA, Lrp, MetJ and SoxS, totally covering 424 verified binding sites. Table 4.1 shows individual performance statistics for each transcription factor. Prediction accuracies vary among 10 TFs. For example, the prediction sensitivity was 92.6% for LexA, but only 46.7% for Lrp with the known motif. Further studies have shown that Lrp-associated motif was quite degenerate, with the pattern of "NNNNNNTTTATTCT", thus making motif-finding quite difficult. In contrast, LexA-associated motif was a 16-bp palindrome, with a conserved pattern of "CTGTATATATAT-ACAG". In general, our motif discovery pipeline has a high sensitivity but low specificity, similar to other motif prediction results [72]. However, some of this low specificity could be due to unverified but true sites. As more binding sites are verified and deposited in the PRODORIC database, some predicted false positives could become true positives.

Comparison to other approaches

We also compared the performance of our conserved operon-based approach with two orthologous gene-based (specifically sequence similarity-based) approaches, which were used in MicroFootprinter [116] and PHYLOCLUS [76] respectively. In both methods, orthologous genes in other species were identified using a reciprocal BLAST best-hit procedure, with a threshold of 10^{-6}. For each method, we generated sequence data sets, ran our motif pipeline for TFBSs prediction, and then evaluated predictions based on 424 binding sites from the PRODORIC database. As shown in Table 4.2, our approach was more sensitive than the two other ones (63.6% versus 60.5% and 60.3%). The higher sensitivity of our approach over the other two can be attributed to the reliability of our generated orthologous promoter regions. For example, our approach could detect the true binding-sites of the glutamine permease operon *glnHPQ* in *E. coli*, while the orthologous gene-based couldn't. An investigation of the datasets showed that our approach identified 7 conserved operons for *glnHPQ*, while 'OrthM' identified 10, and 'OrthB' identified 6 "orthologous" genes for *glnH* (shown in Table 4.3). Further analysis has shown that three 'orthologous' genes (*e.g.*, 117619357, *artI*, 2800492) found by 'OrthM' were actually arginine ABC transporters. In addition, both 'OrthB' and 'OrthM' considered '70728423' from *P. fluorescens* to be an 'orthologous' gene for *glnH*, while our approach did detect a conserved operon *glnHP*-70733921. All these indicate that these four identified genes by OrthB and OrthM are not true orthologs, and introduction of the sequences of these genes in OrthB and OrthM lead to the reduction of information content for motif finding.

4.4 Conclusion

We have presented a computational framework of phylogenetic footprinting in prokaryotes. The major contributions of our work include: a) the introduction of the conserved operon approach, rather than the orthologous gene approach, to collect promoter sequence datasets, and b) the development of motif-discovery pipeline for identifying TFBSs from the sequences

Table 4.2: Performance comparison between the conserved operon-based (PFP) and the orthologous gene based approaches (OrthM and OrthB).

Methods	*Sn*	*Sp*	*PC*	*F*
OrthM	0.605	0.109	0.102	0.184
OrthB	0.603	0.105	0.098	0.179
PFP	0.636	0.106	0.100	0.182

Table 4.3: A list of *glnHPQ* associated orthologous genes and conserved operons predicted by OrthM, OrthB and PFP.

Species	**OrthM**	**OrthB**	**PFP**	**Degree of Conservation**
E. coli	glnH	glnH	glnQ glnP glnH	
A. hydrophila	117619357		117619722 117620284 117620427	0.67
E. carotovora	glnH	glnH	glnH glnP glnQ	1
P. profundum	54302807			
P. luminescens	artI			
P. fluorescens	70728423	70728423	glnH glnP 70733921	1
S. enterica	glnH	glnH	glnQ glnP glnH	1
S. ANA	117919334	117919334	117919332 117919333 117919334	1
S. sonnei	glnH	glnH	glnQ glnP glnH	1
V. parahaemolyticus	28900492			
Y. pestis	108807999	108807999	108807999 108808000 108808001	1

we have identified. Performance omparison of TFBSs prediction between our approach and others has shown that our approach could identify more experimentally verified binding-sites.

The better performance of our approach over previous ones is mainly due to the followings: the correct characterization of operon structures in the recent research efforts, and the correct determination of orthology relationships by relying on multiple homologous gene relationships within an operon. In addition, our algorithm *OPERMAP* can nicely incorporate three different categories of conserved operons that maintain the same regulation mechanism.

In our future work, we will predict TFBSs of prokaryotes at the genome scale using our computational framework. By clustering these predicted TFBSs, we can utimately decipher regulons, which is the set of operons whose promoter regions share the similar binding motif patterns regulated by the same transcription factor.

Chapter 5

Computational prediction and analyses of regulons at a genome scale

5.1 Introduction

A regulon is a collection of (individually transcribed) genes and operons that are regulated by the same transcription factor(s) (TFs). Under certain conditions such as heat shock or starvation, specific transcription factors are activated, which will in turn activate or repress the expression of their regulated genes. To date, several hundreds of TFs and their regulated genes have been identified in prokaryotes using experimental techniques. They are limited, however, to a few model organisms such as *E. coli* [57] and *B. subtilis* [141]. Furthermore, they represent probably only a small fraction of all the regulatory relationships among the TFs and their regulated genes in these organisms. One of the challenging issues in experimental elucidation of such regulatory relationships at a global level is that our knowledge about what conditions may activate which TFs is limited, which has limited our ability to design the right experimental conditions to activate a specific set of TFs and their related genes.

Computational techniques have proven to be useful in providing complementary information to experimental techniques for studying various complex biological systems such as for elucidation of regulons. For example, Tan *et al.* [148] predicted new members of the Crp and Fnr regulons based on known transcription factor binding sites (TFBSs). The basic idea is to first construct a position-specific weight matrix (PWM) of the known TFBSs of a given TF, and then search all the promoter sequences for additional binding sites in the target genome using the PWM. Genes with promoter regions matching the PWM will be predicted as possible members of the TF's regulon. A general issue with such a scanning strategy is that it generally suffers from high false positive prediction rates [148, 111].

The problem becomes even more challenging when there is no prior knowledge about TFBSs when attempting to predict such regulatory relationships. Nevertheless, there have been a few published studies on regulon prediction at a genome scale, such as Regulog [1] and PHYLOCLUS [77]. The general approach employed in these programs includes the use of the *phylogenetic footprinting* [147] for identifying orthologous genes across related genomes and then identifying *cis* regulatory motifs in their promoter regions using motif-finding tools

[100, 67] and the use of motif clustering for prediction of transcriptionally co-regulated genes [77, 127]. Application of these methods on organisms of *E. coli* and *B. subtilis*, however, revealed that these methods suffer from the general problem of low prediction specificity as well as low sensitivity [1, 77].

Among the various challenging issues in predicting regulons, one is to predict accurately the *cis* regulatory motifs in the promoter regions. The phylogenetic footprinting technique proves to be effective for eukaryotic genomes, but it could not be applied to prokaryotic genomes directly since more than half of the genes in a prokaryotic genome are organized into multi-gene operons, and their promoter sequences are located in the inter-operonic rather than inter-genic regions. Thus, in order to apply this technique, additional care needs to be taken.

We have previously developed an algorithm [27] for applying the phylogenetic footprinting analysis to prokaryotes, by taking into consideration of operon structures. The algorithm first identifies conserved operons (instead of orthologous genes) across multiple prokaryotic genomes, and then applies the phylogenetic footprinting technique to their promoter sequences for motif-finding. We have used this algorithm as part of our regulon prediction framework for finding possibly co-regulated operons and associated *cis* motifs.

A second key component of our regulon prediction scheme is to use predicted uber-operons as candidate components of the to-be-identified regulons since our previous study [24], as well as other studies [2], has shown that regulons and uber-operons are closely related. Operationally, while regulons have traditionally been predicted through identification of operons sharing similar *cis* regulatory motifs, uber-operons are predicted through identification of evolutionarily related operons. We have integrated the two types of complementary information in our regulon prediction method.

We have applied our regulon-prediction framework to the genome of *E. coli* K12. Our test results indicate that our method gives a much improved prediction performance over the existing ones. Application of our prediction method has led to the discovery of new members

of previously known regulons as well as of new regulons, some of which are partially verified based on the analyses of functional enrichment and microarray gene expression data.

We have applied our regulon-prediction method to the genome of *E. coli* K12, which has the largest number of experimentally verified regulons among all prokaryotic genomes, and predicted 554 regulons with at least four operons. Our prediction covers 41 out of 88 previously known regulons in the regulonDB database, where a previously known regulon is considered being covered by our prediction if at least 50% of its genes are covered by a single predicted regulon. We have predicted new members of previously known regulons as well as of new regulons, some of which are partially (computationally) validated based on our analyses of functional relatedness and microarray gene expression data.

5.2 Materials and Methods

5.2.1 Data and preliminary data processing

Reference genomes for phylogenetic footprinting. The selection of reference genomes used for phylogenetic footprinting was based on three factors, including namely a) the phylogenetic relationship between the reference genome and the target genome; b) the number of orthologous genes in the reference genome corresponding to TF genes in the target genome; and c) the genome sizes of the target genome and the reference genome. To do so, we have collected 16s ribosomal-RNA sequences of all non-redundant γ-proteobacterial species from the NCBI GenBank database, aligned them using CLUSTALX1.8 [154], and then constructed the phylogenetic trees by using the maximum likelihood-based program PROML in the PHYLIP package (J. Felsenstein, Department of Genome Sciences, University of Washington, Seattle; `http://evolution.genetics.washington.edu/phylip.html`). In addition, we collected all the 155 TFs genes of *E. coli* from the regulonDB [57], and used them for identifying TF orthologs in each of 39 species, by using a reciprocal BLAST best-hit procedure, with a significance threshold of 10^{-6}. Based on the results of phylogenetic trees, TF orthologs, and genome sizes (Figure 5.1), we selected 11 reference genomes, including *Aeromonas hydrophila*

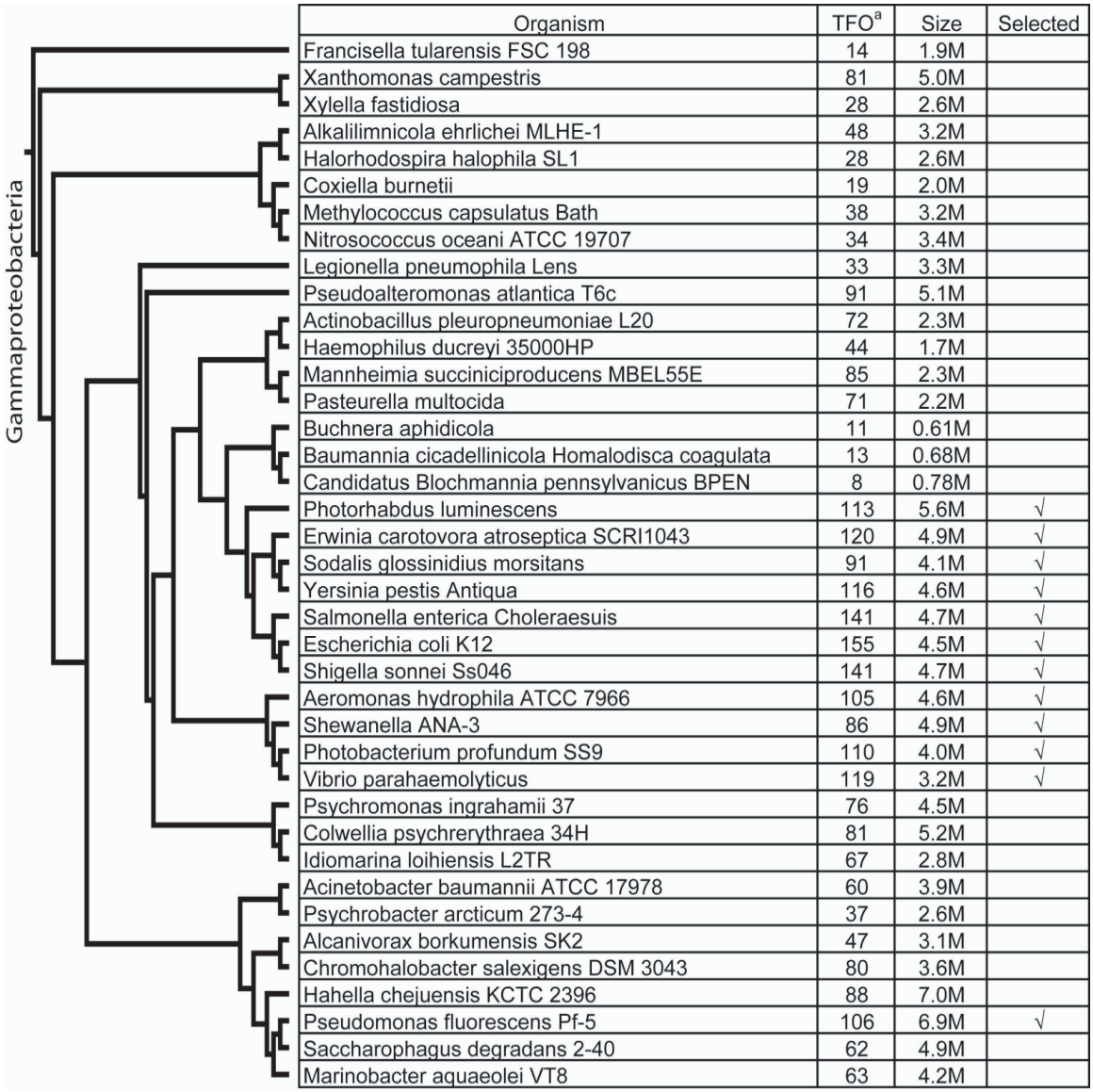
ATCC_7966, Erwinia carotovora atroseptica SCRI1043, Photobacterium profundum SS9, Photorhabdus luminescens, Pseudomonas fluorescens Pf-5, Salmonella enterica Choleraesuis, Shewanella ANA3, Shigella sonnei Ss046, Sodalis glossinidius morsitans, Vibrio parahaemolyticus, and *Yersinia pestis Antiqua*.

Promoter sequence sets for motif-finding. We first predicted operons for each of twelve selected genomes, using our operon program, UNIPOP [98]. We then identified a 'conserved' operon set in reference genomes for each of 2,706 operons in *E. coli*, using our algorithm OPERMAP [27]. Finally, for each operon in each conserved operon set, we collected its promoter sequence by extracting its upstream sequence up to 400 base-pairs (bp) from the translation start site, without overlap of the next upstream gene. All sets of collected promoter sequences were used for identifying motifs in our computational framework.

The predicted uber-operon data. In our previous work, we predicted 158 sets of uber-operon predictions for *E. coli* based on 90 reference genomes [24]. Thus, we collected these predicted uber-operon data used in regulon prediction.

5.2.2 The computational framework

A unique feature of our regulon prediction program is that it combines the information of operons that are evolutionarily related and operons sharing similar motifs for regulon prediction. Previously, we developed an algorithm for predicting uber-operons, which are groups of evolutionarily-related as well as functionally related operons [24]. Our comparison of 158 predicted uber-operons and known regulons in *E. coli* revealed that they overlapped significantly, with instances of uber-operons contained in regulons, and vice-versa. The high degree of overlap between the predicted uber-operons and known regulons indicate that many operons within an uber-operon are transcriptionally co-regulated. Thus, the co-regulated relationships between operons could be derived directly from uber-operons as well as from their binding motifs, the latter of which has been widely used for regulon predictions. The reason for combining these two sources of information in our regulon prediction is that



Organism	TFO[a]	Size	Selected
Francisella tularensis FSC 198	14	1.9M	
Xanthomonas campestris	81	5.0M	
Xylella fastidiosa	28	2.6M	
Alkalilimnicola ehrlichei MLHE-1	48	3.2M	
Halorhodospira halophila SL1	28	2.6M	
Coxiella burnetii	19	2.0M	
Methylococcus capsulatus Bath	38	3.2M	
Nitrosococcus oceani ATCC 19707	34	3.4M	
Legionella pneumophila Lens	33	3.3M	
Pseudoalteromonas atlantica T6c	91	5.1M	
Actinobacillus pleuropneumoniae L20	72	2.3M	
Haemophilus ducreyi 35000HP	44	1.7M	
Mannheimia succiniciproducens MBEL55E	85	2.3M	
Pasteurella multocida	71	2.2M	
Buchnera aphidicola	11	0.61M	
Baumannia cicadellinicola Homalodisca coagulata	13	0.68M	
Candidatus Blochmannia pennsylvanicus BPEN	8	0.78M	
Photorhabdus luminescens	113	5.6M	√
Erwinia carotovora atroseptica SCRI1043	120	4.9M	√
Sodalis glossinidius morsitans	91	4.1M	√
Yersinia pestis Antiqua	116	4.6M	√
Salmonella enterica Choleraesuis	141	4.7M	√
Escherichia coli K12	155	4.5M	√
Shigella sonnei Ss046	141	4.7M	√
Aeromonas hydrophila ATCC 7966	105	4.6M	√
Shewanella ANA-3	86	4.9M	√
Photobacterium profundum SS9	110	4.0M	√
Vibrio parahaemolyticus	119	3.2M	√
Psychromonas ingrahamii 37	76	4.5M	
Colwellia psychrerythraea 34H	81	5.2M	
Idiomarina loihiensis L2TR	67	2.8M	
Acinetobacter baumannii ATCC 17978	60	3.9M	
Psychrobacter arcticum 273-4	37	2.6M	
Alcanivorax borkumensis SK2	47	3.1M	
Chromohalobacter salexigens DSM 3043	80	3.6M	
Hahella chejuensis KCTC 2396	88	7.0M	
Pseudomonas fluorescens Pf-5	106	6.9M	√
Saccharophagus degradans 2-40	62	4.9M	
Marinobacter aquaeolei VT8	63	4.2M	

Figure 5.1: Reference genome selection for *E. coli* K12

Thirty-nine represented species from γ-proteobacteria were chosen for phylogeny analysis using 16s rRNA sequences. TFO was the number of orthologs in other species corresponding to 155 transcription factors in *E. coli*. Twelve species were selected for phylogenetic footprinting.

they provide complementary information about regulons. We found that this strategy is particularly effective in identifying regulons where the *cis* regulatory motifs of operons are degenerative and hence may not be easily identifiable using statistical approaches.

The basic idea of our prediction program is to build a unified graph, where vertices are operons and edges are operon pairs which either have high motif similarities, or belong into the same uber-operon, or have both. The weight of each edge reflects the strength of these two sources. Thus, those densely intra-connected subgraphs in the graph should either contain highly conserved motifs, or belong to the same uber-operon, or have both. We consider such an intra-connected subgraph to be a regulon as it contains operons that share similar *cis* regulatory patterns or belong to the same uber-operons. Thus, the problem of predicting regulons can be converted into the problem of the partition of the unified graph into densely intra-connected subgraphs.

Our prediction program has four major steps: 1) identification of motifs and construction of a motif graph, which is used to record motif similarities among operons; 2) construction of an operon graph, which is used to record the strength of operon pairs belonging into the same uber-operons; 3) construction of a unified graph based on the motif and operon graphs; and 4) partition of the unified graph into subgraphs (*i.e.*, regulons) using clustering algorithms. A schematic diagram of the framework is given in Figure 5.2. The details of our prediction algorithm follow.

Step 1. The input to this step is a collection of promoter sequence datasets, and the output is a motif graph. We first predict all the *cis* regulatory motifs on all sequence sets, using the method described in the Section "Motif prediction and ranking"; and we then calculate pair-wise motif similarities among all identified motifs using an average log-likelihood ratio (ALLR) [162]. Using this computed information, we then construct a *motif* graph $G_1 = (V_1, E_1, W_1)$, where each predicted motif of an operon is represented as a vertex of V_1, and each pair of motifs with an ALLR score above a pre-defined cutoff (7.0 in this study) is represented as edge $e \in E_1$, with its weight $w(e) \in W_1$ being its ALLR score. We label each

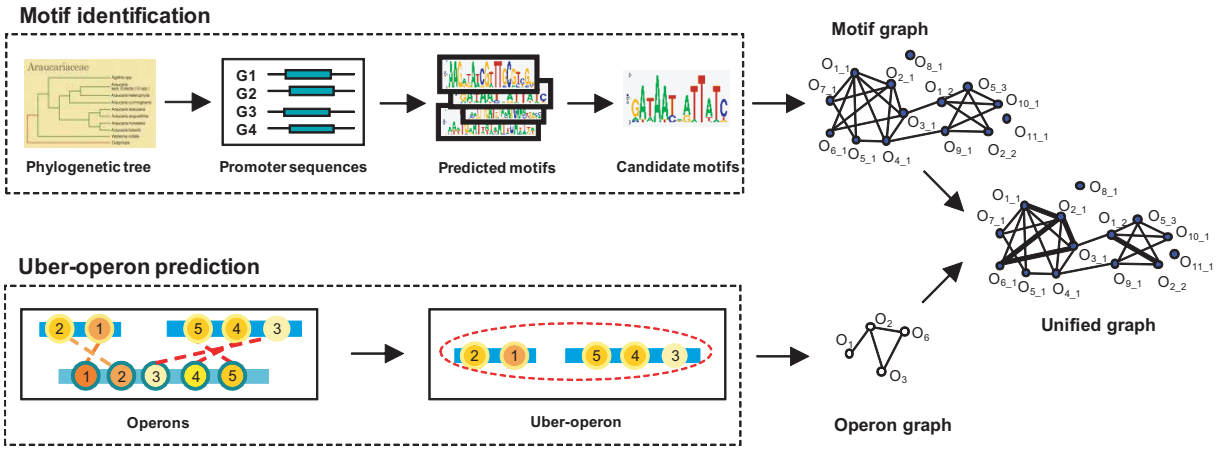


Figure 5.2: Computation framework for regulon prediction

Motif identification includes the following steps: selection of close-related genomes, operon prediction, conserved operon analysis and corresponding promoter sequence collection, motif prediction and motif filtering. A motif graph is built using motifs as vertices and motif similarities as edges. Uber-operon prediction is employed by using comparative genomic data and our graph-theoretic algorithms. An operon graph is built using operons as vertices and evolutionary relationships of operons within uber-operons as edges. A unified graph based on the motif graph and the operon graph is finally built for clustering it into "densely" intra-connected subgraphs, *i.e.*, regulons.

vertex in the graph by 'O_{i_j}', where i is the ID of the corresponding operon and j is the jth predicted motif of the operon.

Step 2: The input to this step is the predicted uber-operon data, and the output is an operon graph. We measure the strength of operon connection for each pair, which is simply the number of times that this pair is in the same uber-operon among all N (90 in this study) predictions by using N reference genomes. We then construct an operon graph $G_2 = (V_2, E_2, W_2)$, where each operon is represented as a vertex of V_2, and each pair of operons that appears in the same uber-operon at least once in N predictions is represented as edge each edge $e \in E_2$, with its weight $w(e) \in W_2$ being the strength of operon connection.

Step 3: The input to this step are the motif graph and the operon graph, and the output is the unified graph. We define the unified graph $G = (V, E, W)$, where $V = V_1$, $E = E_1 \bigcup E_2$, and the weight of each edge (w(e)) is computed as follows,

$$w(e) = \begin{cases} cw(e_1)^\alpha + w(e_2) \\ cw(e_1)^\alpha \\ w(e_2) \end{cases} \tag{5.1}$$

i.e., for any edge $e \in E$, $w(e)$ is the combined weight of $w(e_1)$ and $w(e_2)$ if the vertex pairs of two edges share the same operon IDs, *e.g.*, edge $(O_{1\_1}, O_{2\_1}) \in E_1$ and edge $(O_1, O_2) \in E_2$. Otherwise, the weight is contributed only by $w(e_1)$ or $w(e_2)$. Multiple parameter settings of c (1, 2, 4, 8)and α (1, 1.5, 2) are used in this study.

Once the unified graph is constructed, it remains to partition the unified graph into "densely" intra-connected subgraphs using a clustering algorithm. We apply Markov cluster (MCL) algorithm [48] to partition it into subgraphs at the different granularity levels (2.0, 3.0, 4.0, 5.0) in MCL. The operons corresponding to the vertices in the subgraphs are predicted to be within a regulon.

5.2.3 Motif prediction and ranking

For each sequence set, we use two motif-finding programs, BioProspector [100] and CONSENSUS [67], to predict TFBSs with the motif lengths ranging from 6 to 30 (*i.e.*, 6, 8, 10, ..., 30, with the total of 13 different motif lengths selected). For each motif length of each program, 4 predicted motifs are collected. Thus, 104 ($2 \times 13 \times 4$) motifs are predicted for each sequence set.

We select top-five non-redundant motifs out of 104 predicted motifs, based on two criteria: 1) how conserved the predicted *cis* motif is across all the genomes in the promoter sequence set; and 2) how frequent this *cis* motif is across all other promoter sequence sets in the genomes. Intuitively, the higher fraction of orthologs that share the same *cis* motif (*i.e.*, relative conservation), the more reliable the motif. Such a measure has been applied in several studies [1, 59, 170]. On the other hand, a reliable motif should be consistent in the whole genome, *i.e.*, if a motif has high relative conservation score in a sequence set, and it has high relative conservation score in other sequence sets, then this motif is reliable. This is because a transcription factor usually regulates multiple genes (or operons), which has the same motif pattern. Based on these criteria, we design the following score function to evaluate the reliability of a predicted motif.

$$score = \frac{1}{K}\sum_{i=1}^{K}(\frac{s_i}{m_i})^{\alpha} + \frac{c_0}{K}\sum_{i=1}^{K}(\frac{s'_i}{m_i})^{\alpha} \tag{5.2}$$

where K is the number of sequence sets that share the same motif pattern under investigation (*i.e.*, one motif in 104 motifs), m_i is the number of sequences in the ith sequence set, s_i is the number of sequences that contain at least one binding site of the *cis* motif. α ($\alpha = 2$ in this study) is used so that motifs highly conserved across genomes have high scores. A similar treatment for highly conserved motifs has been done in previous studies [170]. Previous studies [148] also showed that predicted multiple binding sites in one sequence were likely to be the true binding sites. Thus we added this information as an additional source into the second part in the equation, where s'_i is the number of sequences that contains at least two

binding sites, and c_0 is small constant ($c_0 = 0.1$). For two motifs with same score, we pick the longer motif for consideration.

Using this score function to evaluate each of 104 predicted motif, we rank all 104 motifs for each sequence dataset, and we choose up to top-five non-redundant motifs for regulon prediction.

5.2.4 Validation of predicted regulons

We obtained 185 sets of microarray expression data from Stanford Microarray Database (SMD) (urlhttp://smd.stanford.edu/) [43]. These expression data included the following conditions: absolute transcript levels, amino acid metabolism, DNA damage, DNA metabolism, media comparisons, mutants, and RNA decay.

Since the expression data were the normalized ratios (*i.e.*, Log(base2)) of differential expressions on cDNA, we transformed them into three discrete categories, up-regulated, down-regulated and unchanged. Specifically, we simply treated it up-regulated if the expression value was higher than 0.1; down-regulated if it was lower than -0.1; and unchanged if it ranged between -0.1 and 0.1.

To check whether genes within our predicted regulons contained more same expression patterns (either up-regulated or down-regulated) than those of randomly generated regulons, we did the following: for each regulon size, we generated 10,000 artificial regulons, recorded the number of conditions out of 185 conditions, where the majority of genes (70% in the gene cluster, which was similar to the measure used in [97]) shared the same expression patterns (either up-regulated or down-regulated), and plotted the frequency distribution of the number of conditions that the majority of genes shared the same expression pattern. For each set of artificial regulons with the size of k, we obtained the number of conditions in the fifth percentile of the right tail, denoted as defined C_k (*i.e.*, cutoff value at $p < 0.05$). For each predicted regulon with the size of k, we obtained the number of conditions that

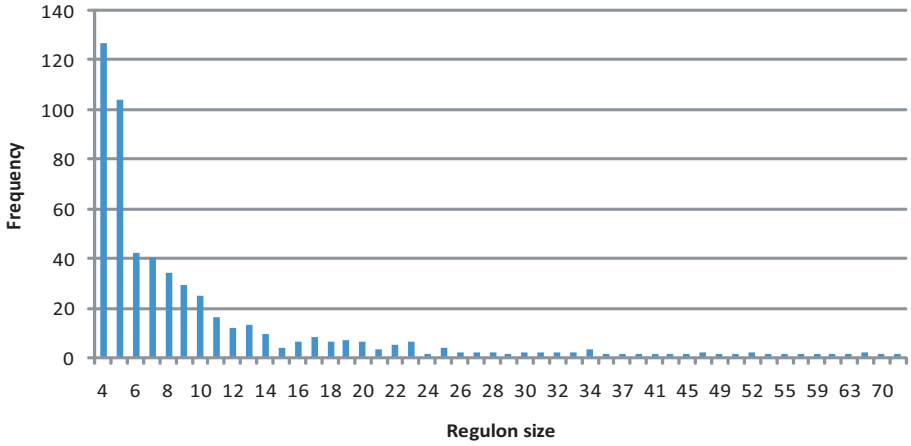


Figure 5.3: Size distribution of our predicted reguons of *E. coli*

the predicted regulon shared the same patterns, denoted as C'_k. We considered the predicted regulon with the size of k as *expression coherent* if C'_k was greater than C_k.

5.3 Results

5.3.1 Regulon prediction at genome scale on *E. coli* K12: a summary

By applying our phylogenetic footprinting approach, we obtained 11,088 *cis*-regulatory motifs in the whole genome. Sequence similarity based analyses of all these motifs gave 55,675 pairs of motifs with high sequence similarities. In addition, 33,596 pairs of operons, out of a total of 578 predicted operons, were predicted to be evolutionarily related based on our analyses of all the predicted uber-operons in *E. coli*. By combining these results, we predicted 554 regulons, each containing at least 4 operons, in *E. coli*, with their size distribution summarized in Figure 5.3. The size distribution of our predicted regulons is similar to that of the known regulons [80], which follows a power-law distribution.

We have examined the localization of operons within each regulon in the genomic sequence (treated as a circular genome), and have observed the following (Figure 5.4a). For small-sized regulons, in some cases, all operons within a regulon were localized, while in other cases, component operons within it covered about a half of the whole genome. For large sized regulons, their component operons typically are scattered around the whole genome. A similar coverage pattern was observed of the known regulons (Figure 5.4b).

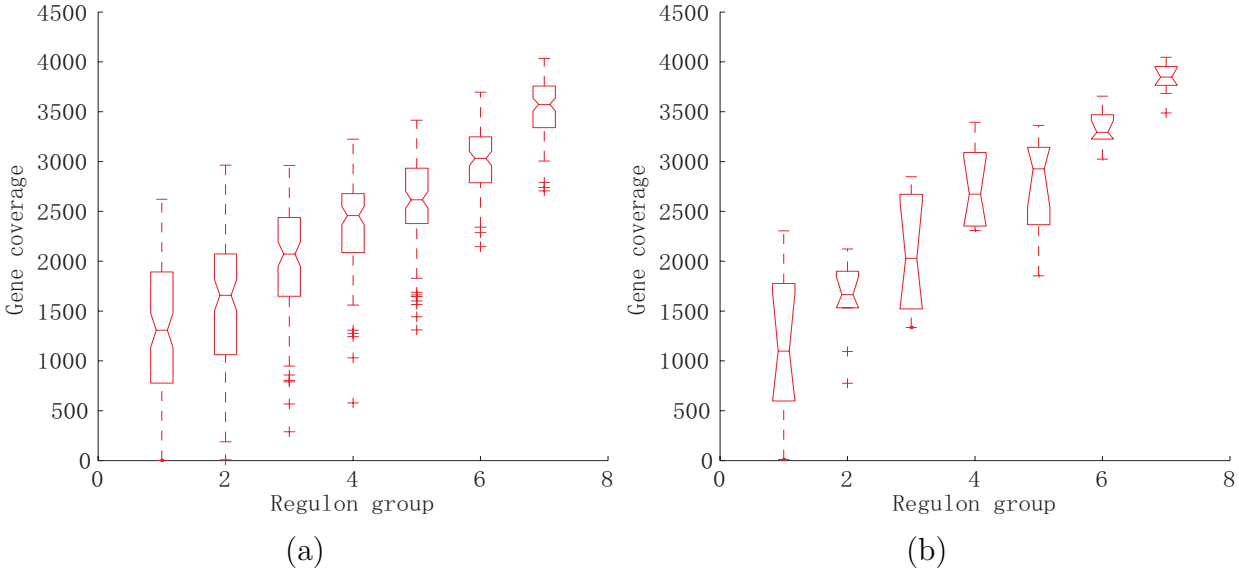


Figure 5.4: Distribution of physical distance coverage of (a) predicted regulons, and (b) known regulons.

We also examined the biological processes of the predicted regulons based on Gene Ontology (GO) [4], and found that our predicted regulons covered all major biological processes such as biosynthesis, cell growth and maintaince, organic acid metabolism. The detailed frequency distribution of GO biological processes that predicted regulons covered is shown in Figure 5.5. Furthermore, we examined individual regulons to check how many biological processes were involved in for each regulon. Our analyses showed that, most of our predicted regulons were associated with one or two biological processes only, while a few 'global' regulons could cover as many as 9 biological processes (Figure 5.6).

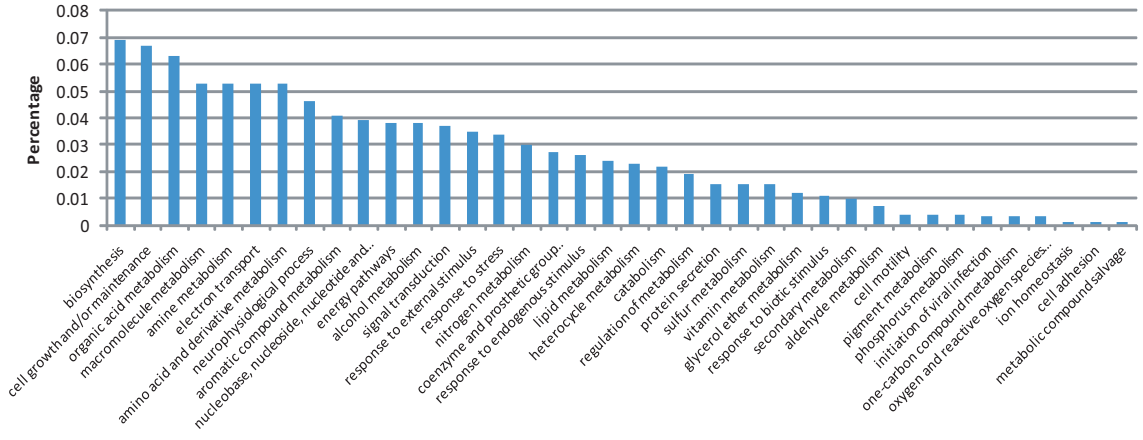


Figure 5.5: The distribution of biological processes of predicted regulons

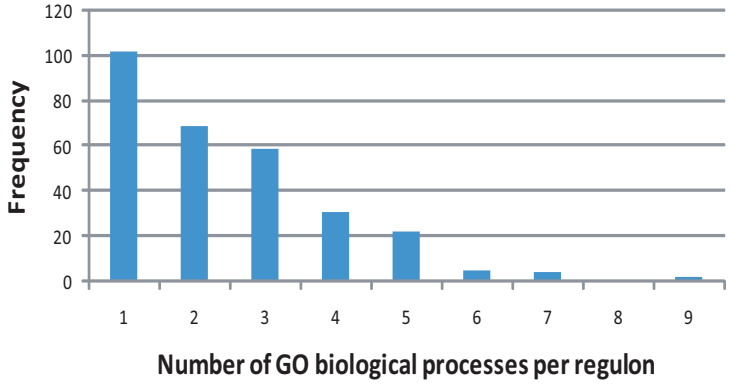


Figure 5.6: Frequency distribution of the number of biological processes per regulon

5.3.2 Assessment of predicted regulons

We have assessed the quality of our predicted regulons using three measures: (a) consistency with experimentally confirmed regulons, (b) functional relatedness among genes within each predicted regulon, and (c) consistency with microarray gene expression data.

Consistency with known regulons. To evaluate how accurate of our predicted regulons of *E. coli*, we downloaded experimentally confirmed regulons from the regulonDB database [57], covering 155 transcription factors and 2801 co-regulated genes. We have measured the consistency between the known regulons and the predicted regulons using the concept of matching degree. The basic idea is given as follows, where the details can be found in [165]).

Let $P = \{p_i, 1 \leq i \leq m\}$ and $Q = \{q_j, 1 \leq j \leq n\}$ be the set of known regulons and predicted regulons, respectively. The matching degree ($MD_{i,j}$) between p_i and q_j is defined as:

$$MD_{i,j} = \frac{|p_i \cap q_j|}{|p_i \cup q_j|} \tag{5.3}$$

and the highest matching degree (HMD) achieved by Q for p_i is defined as:

$$HMD_{p_i} = max_{j=1}^{n}(\frac{|p_i \cap q_j|}{|p_i \cup q_j|}) \tag{5.4}$$

The average highest matching degree ($AHMD$) achieved by Q for P is defined as:

$$AHMD_P = \frac{\sum_{i=1}^{m} HMD_{p_i}}{m} \tag{5.5}$$

The matching degree ($MD_{i,j}$) gives the similarity between a known regulon p_i and a predicted regulon q_j. The HMD for p_i (HMD_{p_i}) gives the subset in Q that achieves the highest similarity with p_i. The $AHMD$ measures the similarity between P and Q.

We calculated the $AHMD_P(Q)$ between Q and the known regulons P using definition (5.5). The $AHMD_P(Q)$ value is 0.206. To assess the statistical significance of this obtained $AHMD_P(Q)$ value, we constructed a set of random regulons Q', by randomly combining the predicted operons such that the i^{th} random regulon has the same number of operons as

the i^{th} regulon in Q. We constructed 10,000 such sets of random regulons, and calculated their $AHMD_P(Q')$ values. Out of 10,000 calculations, all 10,000 $AHMD_P(Q')$ were less than $AHMD_P(Q)$, indicating that the matching between the predicted uber-operons and the known regulons is highly significant (p-value < 0.0001).

Analyses of functional relatedness. It is generally believed that genes in the same regulon are functionally related, while randomly generated regulons should not. To check the level of functional relatedness among genes in the same regulons, we downloaded the GO term assignments of biological processes for all *E. coli* genes from Integr8 (`http://www.ebi.ac.uk/integr8/EBI-Integr8-HomePage.do`).

A possible measure of the functional relatedness between two genes is through calculating the common path between their assigned functional terms from the root node in the GO directed acyclic graph [165]. Generally, the longer the common path a gene pair has, the more functionally related they are. Therefore, the average length of common paths of all gene pairs within the randomly generated regulon should be shorter than that of the gene pairs within the predicted regulon.

We have generated 10,000 sets of artificial regulons, by randomly grouping predicted operons together so that their size distribution is consistent with the size distribution of the predicted regulons, and calculated the average common path of all gene pairs for each artificial regulon, denoted as *ASgo*. We plotted the distribution of *ASgo* for all random regulons, and obtained the value of *ASgo* in the fifth percentile of the right tail, denoted as *AS'go* (*i.e.*, a cutoff value for $p < 0.05$). For each predicted regulon, we calculated *ASgo* and considered it to be *functionally related* if its *ASgo* was greater than *AS'go*. Based on this calculation, we found that 39.9% of predicted regulons were functionally related. As a comparison, we also measured the functional relatedness of known regulons and found that 52.3% of them were functionally related. The functional relatedness analysis of predicted regulons suggested that quite a few regulons were functionally related.

We further divided our predicted regulons into two groups, global regulons (containing at least 10 operons) and local regulons. We found that 26.1% of the global regulons were functionally related, while 46.4% of local regulons were functionally related. This is because global regulons usually contain many operons, which might be involved in different biological processes. There were a few 'global' regulons, however, that are highly functionally related. For instance, the functional enrichment score of a 32-operon regulon was 5.70, significantly higher than that of random one, 2.89 ($p < 2.8 \times 10^{-25}$).

Consistency with gene expression data. We have also compared the predicted regulons with the publicly available microarray gene expression data done on *E. coli*. We believed that in general, component genes within the same regulon tend to have more similar expression patterns in microarray than genes that are randomly grouped together. Thus, we have designed an evaluation approach to check whether each of predicted regulons has a statistical significance in terms of the number of similar expression patterns (see Material and Methods section), and considered a predicted regulon to be *expression coherent* if its number of similar expression patterns is statistically significant.

We found that 36.7% predicted regulons were expression coherent based on our measurement, compared to 64.7% for those of known regulons. The low percentage of our predicted regulons indicates that our prediction might include extra unrelated genes in our predicted regulons. On the other hand, current microarray expression data could bias towards known regulons since some of known regulons directly derived from microarray data based on similar expression patterns in microarray conditions. As more microarray expression data are available, they might cover our predicted regulons, and thus improving the percentage of expression coherence.

Based on the analyses of functional relatedness and microarray data, we found that 335 predicted regulons (61.2% of all predicted regulons) have support by either the functional relatedness analyses or by gene expression coherence analyses. Among them, 107 regulons

were supported by both measures. The detailed analysis of biologically significant regulons is given in the following Section.

5.3.3 New biological findings

By comparing our predicted regulons with currently documented known regulons, we found that our predictions contain novel regulons, as well as new members of known regulons. The followings summarize some of interesting findings in our predictions.

Novel regulons

We consider those predicted regulons to be *novel* if they do not overlap any known regulons in regulonDB, and their genes are highly functionally related and have coherent expression patterns. We found that 49 novel regulons satisfy these criteria. Most of the novel regulons are local regulons, though there are a number of global ones can cover up to 22 operons. These novel regulons are mostly involved in cell growth and maintenance, nucleotide metabolism, amino acid metabolism, transport systems, and energy pathways. We now provide more detailed information of five such predicted new regulons.

Transporter related regulon. This predicted regulon contains five operons, *i.e.*, *argT*, *glnHPQ*, *gltIJKL*, *hisJMPQ* and *yieP*,covering 13 genes. Among them, *argT*, *glnHPQ*, *gltIJKL* and *hisJMPQ* are amino acid associated transporters, while *yieP* is a putative transcriptional regulator. Previous study has shown that *argT* and *hisJMPQ* were responsive to nitrogen stress and were induced by an AAA+ family member, NtrC [138]. *gltIJKL* was reported to be regulated by the master regulator FlhDC [143]. We suspect that *yieP*, acting as a local regulon, specifically regulate these amino-acid associated transporter genes.

Translation associated regulon. This regulon contains six operons (*gsk, ihfA-pheST, thrS-infC-rplT-rpmI, tynA, yfcJ, ygeR*), covering 11 genes. Interestingly, this regulon also contains a putative TF gene, *ygeR*, which probably regulates the remaining genes as well as itself. Most of the component genes of this regulon are protein synthesis-associated, including:

infC (translation initiation factor IF-3), *pheS* and *pheT* (phynelalanyl-tRNA synthetase subunit alpha and beta), and *thrS* (threonyl-tRNA synthetase). Previous studies have shown that the cell growth rate influences the expression of aminoacyl-tRNA synthetase genes [66], we suspect that the cell growth condition might trigger the putative TF, *YgeR*, and in turn induce or repress the expression of these regulon members.

Membrane related regulon. This regulon contains five operons (*cyAB, ybgC-tolQRAB, pal-ybgF, exbBD, trkD-ybgCEFT-ydfG*), covering 15 genes. Among them, operon *ybgC-tolQRAB* and *pal-ybgF* are membrane proteins that play an important role in maintaining membrane integrity and cell morphology [102]. A recent study has shown that the expression of these genes was responsive to the extracytoplasmic stress [157]. Another operon *exbBD*, a membrane protein complex, is involved in the transmission of the energy source of the electrochemical potential to the outer membrane [16]. We suspect the expression of this membrane protein complex is also regulated under the extracytoplasmic stress.

Insertion sequence-related regulon. This regulon covers six single-gene operons, *i.e.*, *insA-1, insA-3, insA-4, insA-5, insA-6* and *insA-7*, all encoding for insertion element IS1 repressor proteins. Interestingly, these operons disperse around the *E. coli* chromosome. While we do not know the transcriptional regulation mechanism of these genes, we do believe that one or more TFs control the expression of these IS1-repressor genes, which are used to control transposition activities [139].

Energy associated regulon. This regulon contains 4 operons (*glgABCPX, malPQ, ybaL, ycjW*), covering 9 genes. Among them, a five-gene operon *glgABCPX*, is involved in glycogen synthesis, and another operon, *malPQ*, is also involved in the sugar pathway. Interestingly, this regulon contains the gene *ycjW*, encoding for a putative DNA-binding TF. We suspect that this gene regulates the expression of glycogene associated genes, as well as itself.

Table 5.1: Summary of known regulons and predicted regulons of *E. coli* K12

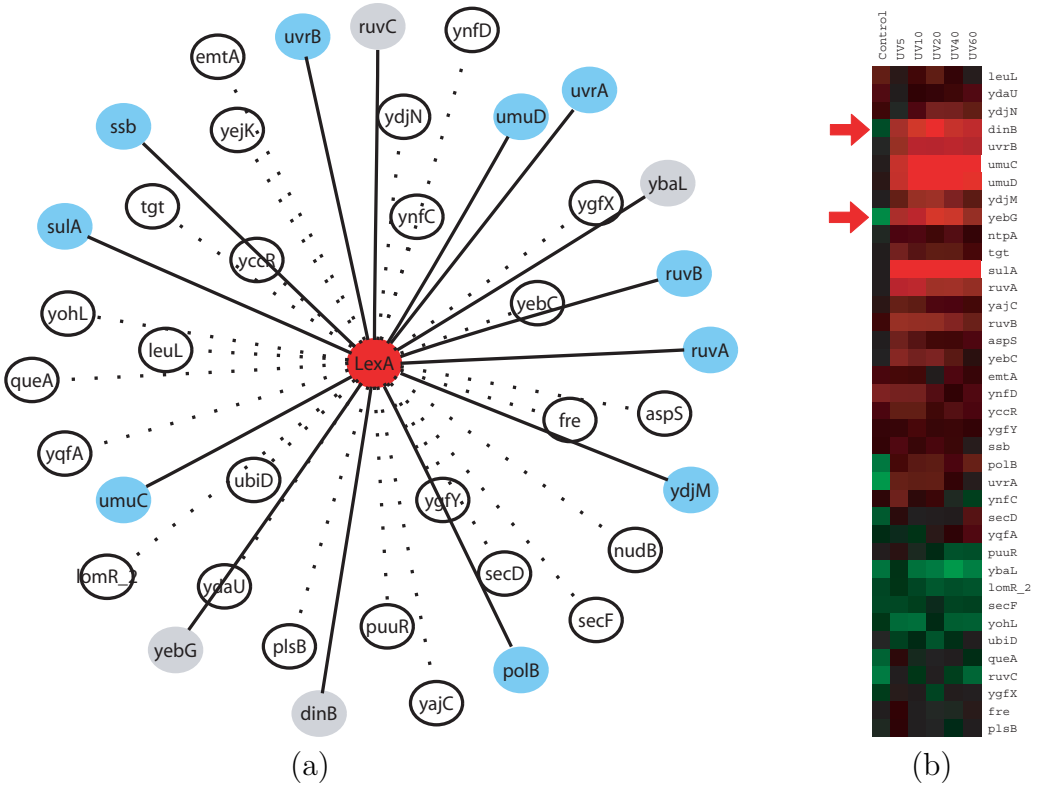
Transcription factor	Known members	Predicted members	Correct members	Missing members	New members
CRP	404	20	15	389	5
FNR	265	42	24	241	18
IHF	207	14	13	194	1
ArcA	151	21	14	137	7
NarL	98	26	25	73	1
Fis	91	111	8	83	103
H-NS	89	14	12	77	2
FlhDC	84	74	31	53	43
Fur	77	38	17	60	21
Lrp	55	21	6	49	15
CpxR	55	6	6	49	0
ModE	46	20	15	31	5
NtrC	44	19	12	32	7
NarP	40	53	22	18	31
PhoB	35	15	14	21	1
FruR	33	50	5	28	45
PhoP	31	13	6	25	7
PurR	31	81	14	17	67
FhlA	30	23	7	23	16
GadE	27	5	5	22	0
ArgR	27	22	11	16	11
IscR	26	27	10	16	17
LexA	25	39	10	15	29
CysB	24	14	9	15	5
SoxS	24	16	3	21	13
RcsAB	22	64	5	17	59
MarA	21	6	2	19	4
NagC	18	9	6	12	3
GadX	17	76	3	14	73
OxyR	17	34	6	11	28
Nac	15	4	2	13	2
MetJ	13	92	4	9	88
Rob	13	11	2	11	9
GntR	12	18	5	7	13
PaaX	12	26	12	0	14
TrpR	12	11	5	7	6
CytR	12	15	6	6	9

New regulon members

We have compared the known regulons in regulonDB with the corresponding regulons based on prediction, and summarize their relationships in Table 5.1. As we can see from the table, the predicted regulons do miss a few members in known regulons, especially for large known regulons, such as CRP regulon. This is probably because some regulon members contain binding sites, which are far from similar to the consensus motif patterns. Interestingly, our predicted regulons do contain a number of new members that were confirmed in recent experiments to be part of this regulon, but not included in the regulonDB yet. In the following, we report five predicted regulons with additional members compared to the known ones, namely LexA, FlhDC, Fnr, PaaX and ArgR regulons.

LexA regulon. The LexA protein is a transcriptional repressor, and inhibits the expression of its regulated genes by binding the 20-bp LexA boxes in their promoter regions under the normal condition [18]. When exposed to ultraviolet light or genotoxic agents, the complex of ssDNA-RecA becomes active and serves as a co-protease to cleave the LexA protein, thus inducing the over-expression of LexA-regulated genes and triggering the SOS response system [83]. The current regulonDB database documents 26 genes that are LexA-regulated.

Our approach predicted a 39-gene regulon, out of which 10 genes were also present in RegulonDB (Figure 5.7). To check whether the remaining predicted ones were LexA-regulated, we searched the literature related to the studies of the LexA regulon [5, 37, 41, 49, 51, 53]. We found that four of predicted ones were experimentally supported but not included in regulonDB. For example, the predicted *dinB* (DNA polymerase IV) and *yebG* (a conserved protein regulated by LexA) were confirmed in several studies [37, 49, 51, 53]. The UV-damaged expression data, collected from Stanford Microarray Database [43], showed that expression levels of *dinB* and *yebG* increased dramatically after five-minute exposure of ultraviolet light (see the arrows in Figure 5.7). Some other predicted LexA-regulated genes were confirmed in other species, though not yet been confirmed in *E. coli*. For example, *ruvC*, encoding for holliday junction resolvase, was experimentally confirmed in *Mycobacterium tuberculosis* [41]



(a) (b)

Figure 5.7: Predicted LexA-regulated regulon: (a) Regulon members, with blues confirmed in regulonDB and gray ones confirmed in recent experiments; (b) Expression profiles of predicted regulon members under the ultraviolet light after 5, 10, 20, 40, 60 minutes, and control. Two arrows show that *dinB* and *yebG* are UV-induced dramatically.

and *Sinorhizobium meliloti* [49]. Another predicted regulon member ybaL, which encodes for transporter with NAD(P)-binding Rossmann-fold domain, was experimentally verified as the LexA target in *B. subtilis* [5].

FlhDC regulon. The FlhDC complex is a master regulator that regulates many flagellar and non-flagellar genes in bacteria. Under the pH or carbon stress, FlhDC is over-expressed and triggers the flagellar and motility system by inducing the expression of the associated genes [106]. Our predicted regulon includes 74 FlhDC-regulated genes (Figure 5.8). By comparing our predicted FlhDC regulon with the one in regulonDB, which contains 84 genes, we

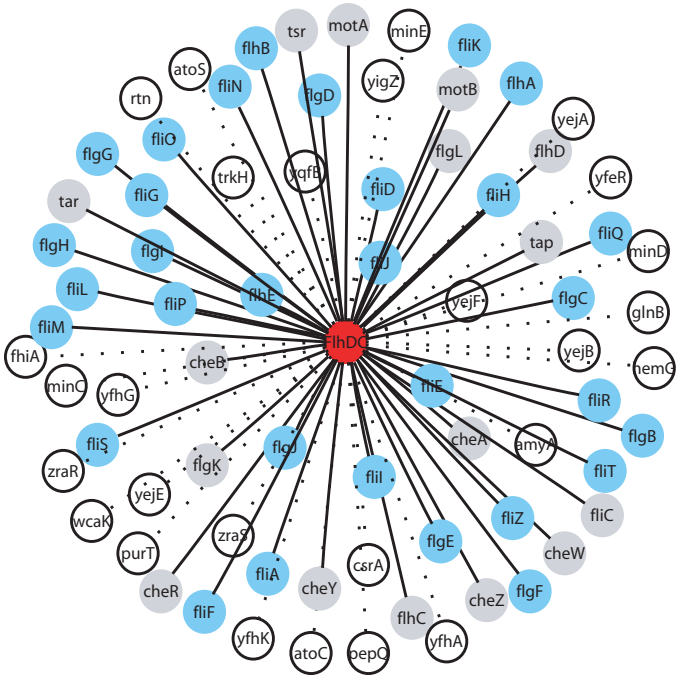


Figure 5.8: Predicted FlhDC regulon members

Blues were confirmed in regulonDB, gray ones were confirmed in recent experiments, and white ones were not confirmed yet.

found that 31 genes overlapped between the two. In a recent study [169] of the FlhDC regulon in *E. coli*, the deletion of FlhDC in vivo led to 2-fold decrease or more of 117 genes expression, out of which 65 are experimentally confirmed to be FlhDC-regulated. We compared our predicted 74-gene regulon with those 65 down-regulated genes, and discovered that 47 genes overlapped. Functional annotations of these 16 genes not included in regulonDB have revealed that they are involved in chemotaxis and motility, further supporting the possible correctness of our prediction.

Fnr regulon. Fumarate and nitrate reduction (Fnr) protein is a helix-turn-helix transcriptional regulator, and regulates 265 genes in *E. coli* according to regulonDB. We predicted a 42-gene regulon, 24 of which overlap with the regulon in regulonDB (Figure 5.9). We found

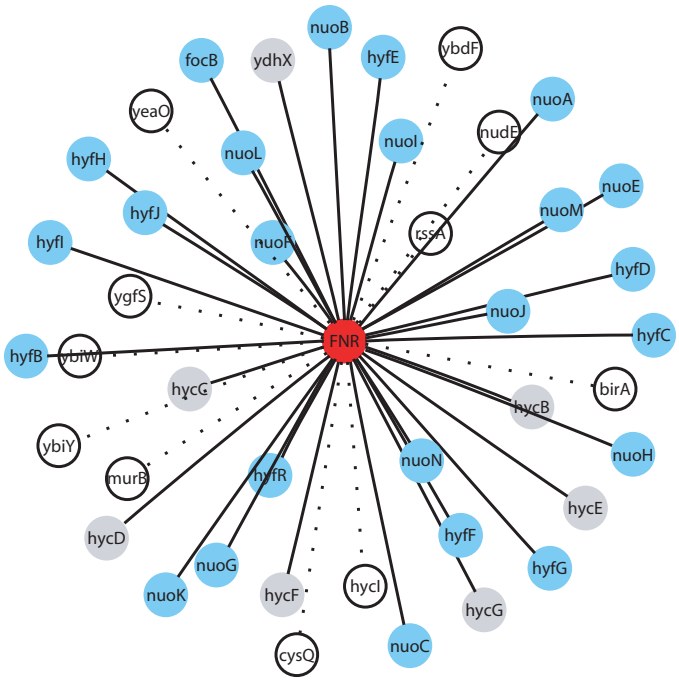


Figure 5.9: Predicted FNR regulon members

Blues were confirmed in regulonDB, gray ones were confirmed in recent experiments, and white ones were not confirmed yet.

from the published literature that some other members of our predicted Fnr regulon were experimentally supported. For example, a six-gene operon *hycBCDEFG* was confirmed in a reassessment study of the Fnr regulon [35]. More recently, a six-gene operon *ydhYVWXUT* in *E. coli* was found to be regulated by multiple transcription factors of Fnr, NarL and NarP [122], and our predicted regulon has *ydhX* to be Fnr-regulated.

PaaX regulon. Paax is a negative transcription factor that represses the genes involved in phenylacetic acid (PA) degradation [54]. In current regulonDB, this regulon contains two operons (*moaA* and *paaABCDEFGHIJK*), totally covering 12 genes. Our prediction of this regulon contains 26 genes, including all the 12 regulon members in regulonDB, and 14 others.

Among 14 other predicted members, genes *ydiR* and *ydiS*, encode for coenzymes of electron transfer, possibly participating oxidation/reduction steps in the PA degradation.

ArgR regulon. ArgR is a hexameric repressor protein that mainly inhibits the transcription of genes of arginine biosynthesis. The ArgR regulon in regulonDB has 27 members, while our predicted regulon contains 22 genes, with 11 overlapping with the known ones in regulonDB. Interestingly, a 4-gene operon *artPIJM*, which is included in our prediction but not in regulonDB, has been recently confirmed to be a member of the ArgR regulon [21]. Transcriptome analysis showed that the expression levels of *artP*, *artI, artJ* and *artM* were at least 2-3 folds decreased under the repression of ArgR [21]. The ArgR-regulated binding site of this operon was also confirmed by DNase I footprinting experiments [22]. In addition, we found that *ybcF*, a carbamate kinase, was involved in arginine and proline metabolism [79], suggesting that *ybcF* is the possible ArgR regulon member.

5.4 Discussion

We have developed a computational method for predicting regulons for prokaryotic genomes, by using two sources of information, namely predicted similar cis regulatory motifs and evolutionary relationships among operons. Different sources of validations have shown that our predicted regulons were consistent with the data of known regulons, functional relatedness and miroarray gene xpression data. Based on these analyses, we have identified a number of novel regulons with strong experimental data support.

Since the current regulon database represents only a small fraction of all regulons encodced in the *E. coli* genome, it is impossible to accurately to estimate the prediction accuracy. Our comparison of several predicted regulons with those experimentally confirmed in recent studies has shown that our predicted regulons did contain regulon members that were missing in known regulons. Therefore, we expect the prediction specificity of our approach to be much higher, with more regulon members discovered and deposited in the regulon databases in the future.

Our predicted regulon do miss a number of known regulons, thus affecting our prediction sensitivity. As we know, our approach relies on two sources, *cis* regulatory motifs and uber-operon data. Therefore, for those operons which do not show evolutionary relationships based on our uber-operon data, our prediction only relies on the motifs among operons, which is the determining factor for prediction accuracy. The missing of known members in our predictions might be due to the poor motif conservation for such regulons. In fact, we did observe that a number of binding site outliers for regulon members in regulonDB. In our future work, we may incorporate additional sources, such as microarray and CHIP-chip data [129], to improve our prediction accuracy.

Chapter 6

Conclusions

The tremendous efforts of high-throughput genome sequencing have made more than 700 genomes fully sequenced to date. We expect this number to increase dramatically in the near future, as thousands of genomes are currently in the sequencing pipeline. There is lots of meaningful information encoded in these genomic data which needs to be deciphered and annotated. As part of genomic data studies, our work is mainly on identifying operon, uber-operon and regulon structures in prokaryotic genomes.

In Chapter 2, we have presented a graph-based approach to predict operon structures, without using any training set or experimental data. The approach only needs sequence information, and guarantees high prediction accuracy. The assumption of our approach is the existence of similar gene blocks between two genomes, and our maximum bipartite matching-based algorithm can detect such similar gene blocks.

To identify those operons with evolutionary relationships, *i.e.*, uber-operons, we have used comparative genomic data and developed a maximum bipartite matching-based algorithm. The key idea of the approach is to identify a set of linker genes, each of which refers to a pair of genes in one genome where each gene is in a different operon and their orthologous genes are in the same operon in another genome.

The computational identification of operons and uber-operon makes it possible to predict regulon structure. In Chapater 4 we introduced a conserved operon approach to extract promoter sequences used for *cis* motif identification. By combining two sources of data, *cis* motif and uber-operon, and formulating it a graph as described in Chapter 5, we were able to identify regulon structures in genomes.

It is worth noting that many algorithms rely on experimental data to predict genomic structures; this unavoidably leads to the limitation that these approaches are applicable only in a few genomes. Our algorithms and frameworks, however, depend on minimal annotations, and thus are general enough for identifying genomic structures of any sequenced genome.

For future work, we plan to develop automatic computational tools to automatically extract genomic data in the Internet whenever the genome sequence data are available, predict their genomic structure, and display the prediction results for biological studies.

Bibliography

[1] Alkema WB, Lenhard B, Wasserman WW (2004) Regulog analysis: detection of conserved regulatory networks across bacteria: application to Staphylococcus aureus. *Genome Res* 14: 1362-1373.

[2] Alm EJ, Huang KH, Price MN, Koche RP, Keller K, et al. (2005) The MicrobesOnline Web site for comparative genomics. *Genome Res* 15: 1015-1022.

[3] Aravind, L., Walker, D.R. and Koonin, E.V. (1999) Conserved domains in DNA repair proteins and evolution of repair systems. *Nucleic Acids Res*, 27, 1223-1242.

[4] Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., et al. (2000) Gene Ontology: tool for the unification of biology. The Gene Ontology Consortium *Nature Genet*, . 25, 25-29.

[5] Au N, Kuester-Schoeck E, Mandava V, Bothwell LE, Canny SP, et al. (2005) Genetic composition of the Bacillus subtilis SOS system. *J Bacteriol* 187: 7655-7666.

[6] Bagchi, G., Chauhan, S., Sharma, D. and Tyagi, J.S. (2005) Transcription and autoregulation of the Rv3134c-devR-devS operon of Mycobacterium tuberculosis. *Microbiology*, 151, 4045-4053.

[7] Bailey, T.L., Elkan, C.: Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology.* AAAI Press, Menlo Park, California 28-36 (1994)

[8] Bardy, S.L., Ng, S.Y. and Jarrell, K.F. (2004) Recent advances in the structure and assembly of the archaeal flagellum. *J Mol Microbiol Biotechnol*, 7, 41-51.

[9] Berge C (1957) Two Theorems in Graph Theory. *Proc Natl Acad Sci U S A* 43: 842-844.

[10] Bischoff, M., Entenza, J.M. and Giachino, P. (2001) Influence of a functional sigB operon on the global regulators sar and agr in Staphylococcus aureus. *J Bacteriol*, 183, 5171-5179.

[11] Blumenthal, T., Evans, D., Link, C.D., Guffanti, A., Lawson, D., Thierry-Mieg, J., Thierry-Mieg, D., Chiu, W.L., Duke, K., Kiraly, M. et al. (2002) A global analysis of Caenorhabditis elegans operons. *Nature*, 417, 851-854.

[12] Bondy, J.A. and Murty, *Graph Theory with Applications*. U.S.R. (1976) Macmilliam Press Ltd., London, UK & Elsevier Science Publishing Co., Inc., NY, US .

[13] Bockhorst, J., Craven, M., Page, D., Shavlik, J. and Glasner, J. (2003) A Bayesian network approach to operon prediction. *Bioinformatics*, 19, 1227-1235.

[14] Braibant, M., Lefevre, P., de Wit, L., Peirs, P., Ooms, J., Huygen, K., Andersen, A.B. and Content, J. (1996) A Mycobacterium tuberculosis gene cluster encoding proteins of a phosphate transporter homologous to the Escherichia coli Pst system. *Gene*, 176, 171-176.

[15] Brandenberger, M., Tschierske, M., Giachino, P., Wada, A. and Berger-Bachi, B. (2000) Inactivation of a novel three-cistronic operon tcaR-tcaA-tcaB increases teicoplanin resistance in Staphylococcus aureus. *Biochim Biophys Acta*, 1523, 135-139.

[16] Braun V, Gaisser S, Herrmann C, Kampfenkel K, Killmann H, et al. (1996) Energy-coupled transport across the outer membrane of Escherichia coli: ExbB binds ExbD and TonB in vitro, and leucine 132 in the periplasmic region and aspartate 25 in the transmembrane region are important for ExbD activity. *J Bacteriol* 178: 2836-2845.

[17] Breidt, F., Jr., Hengstenberg, W., Finkeldei, U. and Stewart, G.C. (1987) Identification of the genes for the lactose-specific components of the phosphotransferase system in the lac operon of Staphylococcus aureus. *J Biol Chem*, 262, 16444-16449.

[18] Brent R, Ptashne M (1981) Mechanism of action of the lexA gene product. *Proc Natl Acad Sci U S A* 78: 4204-4208.

[19] Brouwer, R. W. W., Kuipers, O. P., and Hijum, S. A. F. T. v. (2008) The relative value of operon predictions. *Brief Bioinform*, April 17, bbn019v1.

[20] Cabrera, G., Xiong, A., Uebel, M., Singh, V.K. and Jayaswal, R.K. (2001) Molecular characterization of the iron-hydroxamate uptake system in Staphylococcus aureus. *Appl Environ Microbiol*, 67, 1001-1003.

[21] Caldara M, Charlier D, Cunin R (2006) The arginine regulon of Escherichia coli: whole-system transcriptome analysis discovers new genes and provides an integrated view of arginine regulation. *Microbiology* 152: 3343-3354.

[22] Caldara M, Minh PN, Bostoen S, Massant J, Charlier D (2007) ArgR-dependent repression of arginine and histidine transport genes in Escherichia coli K-12. *J Mol Biol* 373: 251-267.

[23] Casali, N., White, A.M. and Riley, L.W. (2006) Regulation of the Mycobacterium tuberculosis mce1 operon. *J Bacteriol*, 188, 441-449.

[24] Che, D., Jensen, S., Cai, L., Liu, J.S.: BEST: binding-site estimation suite of tools. *Bioinformatics* (Oxford, England) 21, 2909-2911 (2005)

[25] Che, D, Li. G, Mao. F, Wu H, Xu Y.: Detecting uber-operons in prokaryotic genomes.: *Nucleic Acids Res* 34, 2418-27 (2006)

[26] Che, D., Zhao, J., Cai, L., Xu, Y.: Operon Prediction in Microbial Genomes Using Decision Tree Approach.: *Proceedings of IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology* 135-142 (2007)

[27] Che D, Li G, Jensen S, Liu JS, Xu Y. PFP: a computational framework for phylogenetic footprinting in prokaryotic genomes 2008. pp. 110-121.

[28] Chen J, Yuan B (2006) Detecting functional modules in the yeast protein-protein interaction network. *Bioinformatics* 22: 2283-2290.

[29] Chen G, Jensen ST, Stoeckert CJ, Jr. (2007) Clustering of genes into regulons using integrated modeling-COGRIM. *Genome Biol* 8: R4.

[30] Chen, X., Su, Z., Dam, P., Palenik, B., Xu, Y., Jiang, T. (2004) Operon prediction by comparative genomics: an application to the Synechococcus sp. WH8102 genome *Nucleic Acids Res,*. 32, 2147-2157.

[31] Chen, X., Su, Z., Xu, Y., Jiang, T. (2004) Computational prediction of operons in Synechococcus sp. WH8102 *Genome Inform Ser Workshop Genome Inform,*. 15, 211-222.

[32] Cheng Y, Church GM (2000) Biclustering of expression data. Proc Int Conf Intell Syst Mol Biol 8: 93-103.

[33] Chilcott, G.S. and Hughes, K.T. (2000) Coupling of flagellar gene expression to flagellar assembly in Salmonella enterica serovar typhimurium and Escherichia coli Microbiol. *Mol. Biol. Rev*, . 64, 694-708.

[34] Constantinesco, F., Forterre, P., Koonin, E.V., Aravind, L. and Elie, C. (2004) A bipolar DNA helicase gene, herA, clusters with rad50, mre11 and nurA genes in thermophilic archaea. *Nucleic Acids Res*, 32, 1439-1447.

[35] Constantinidou C, Hobman JL, Griffiths L, Patel MD, Penn CW, et al. (2006) A reassessment of the FNR regulon and transcriptomic analysis of the effects of nitrate, nitrite, NarXL, and NarQP as Escherichia coli K12 adapts from aerobic to anaerobic growth. *J Biol Chem* 281: 4802-4815.

[36] Cormen, T.H., Leiserson, C.E., Rivest, R.L. and Stein, C. (2001) *Introduction to algorithms*. 2nd ed. MIT Press, Cambridge, Mass.

[37] Courcelle J, Khodursky A, Peter B, Brown PO, Hanawalt PC (2001) Comparative gene expression profiles following UV exposure in wild-type and SOS-deficient Escherichia coli. *Genetics* 158: 41-64.

[38] Craven, M., Page, D., Shavlik, J., Bockhorst, J. and Glasner, J. (2000) A probabilistic learning approach to whole-genome operon prediction. *Proc Int Conf Intell Syst Mol Biol*, 8, 116-127.

[39] Dam, P., Su, Z., Olman, V., Xu, Y. (2004) In silico construction of the carbon fixation pathway in Synechococcus sp. WH8102 *J. Biol. Syst*, . 12, 97-125.

[40] Dam, P., Olman, V., Harris, K., Su, Z., Xu, Y.: Operon prediction using both genome-specific and general genomic information. *Nucleic acids research* 35, 288-298 (2007)

[41] Davis EO, Dullaghan EM, Rand L (2002) Definition of the mycobacterial SOS box and use to identify LexA-regulated genes in Mycobacterium tuberculosis. *J Bacteriol* 184: 3287-3295.

[42] de Hoon, M.J., Makita, Y., Nakai, K. and Miyano, S. (2005) Prediction of transcriptional terminators in Bacillus subtilis and related species. *PLoS computational biology*, 1, e25.

[43] Demeter J, Beauheim C, Gollub J, Hernandez-Boussard T, Jin H, et al. (2007) The Stanford Microarray Database: implementation of new analysis tools and open source release of software. *Nucleic Acids Res* 35: D766-770.

[44] Dijkstra E (1959) A note on two problems in connecxion with graphs. *Numerische Mathematik* 1: 269-271.

[45] Dongen, S.v. (2000) A cluster algorithm for graphs. Technical Report INS-R0010, National Research Institute for Mathematics and Computer Science in the Netherlands Netherlands Amsterdam.

[46] Edwards, M.T., Rison, S.C., Stoker, N.G. and Wernisch, L. (2005) A universally applicable method of operon map prediction on minimally annotated genomes using conserved genomic context. *Nucleic Acids Res*, 33, 3253-3262.

[47] Eisen MB, Spellman PT, Brown PO, Botstein D (1998) Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci U S A* 95: 14863-14868.

[48] Enright, A.J., Kunin, V., Ouzounis, C.A. (2003) Protein families and TRIBES in genome sequence space *Nucleic Acids Res*, . 31, 4632-4638.

[49] Erill I, Jara M, Salvador N, Escribano M, Campoy S, et al. (2004) Differences in LexA regulon structure among Proteobacteria through in vivo assisted comparative genomics. *Nucleic Acids Res* 32: 6617-6626.

[50] Ermolaeva, M.D., White, O., Salzberg, S.L.: Prediction of operons in microbial genomes. *Nucleic acids research* 29, 1216-1221 (2001)

[51] Faith JJ, Hayete B, Thaden JT, Mogno I, Wierzbowski J, et al. (2007) Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biol* 5: e8.

[52] Feder T, Motwani R (1995) Clique partitions, graph compression and speeding-up algorithms. *J Comput System Sci* 51: 261-272.

[53] Fernandez De Henestrosa AR, Ogi T, Aoyagi S, Chafin D, Hayes JJ, et al. (2000) Identification of additional genes belonging to the LexA regulon in Escherichia coli. *Mol Microbiol* 35: 1560-1572.

[54] Ferrandez A, Minambres B, Garcia B, Olivera ER, Luengo JM, et al. (1998) Catabolism of phenylacetic acid in Escherichia coli. Characterization of a new aerobic hybrid pathway. *J Biol Chem* 273: 25974-25986.

[55] Fredman ML, Tarjan RE (1987) Fibonacci Heaps and Their Uses in Improved Network Optimization Algorithms. *Journal of the Acm* 34: 596-615.

[56] Gabow HN, Tarjan RE (1989) Faster Scaling Algorithms for Network Problems. *Siam Journal on Computing* 18: 1013-1036.

[57] Gama-Castro S, Jimenez-Jacinto V, Peralta-Gil M, Santos-Zavaleta A, Penaloza-Spinola MI, et al. (2008) RegulonDB (version 6.0): gene regulation model of Escherichia coli K-12 beyond transcription, active (experimental) annotated promoters and Textpresso navigation. *Nucleic Acids Res* 36: D120-124.

[58] Gertz, S., Engelmann, S., Schmid, R., Ziebandt, A.K., Tischer, K., Scharf, C., Hacker, J. and Hecker, M. (2000) Characterization of the sigma(B) regulon in Staphylococcus aureus. *J Bacteriol*, 182, 6983-6991.

[59] Gertz J, Riles L, Turnbaugh P, Ho SW, Cohen BA (2005) Discovery, validation, and genetic dissection of transcription factor binding sites by comparative and functional genomics. *Genome Res* 15: 1145-1152.

[60] Glanzmann, P., Gustafson, J., Komatsuzawa, H., Ohta, K. and Berger-Bachi, B. (1999) glmM operon and methicillin-resistant glmM suppressor mutants in Staphylococcus aureus. *Antimicrob Agents Chemother*, 43, 240-245.

[61] Goldberg AV, Kennedy R (1997) Global price updates help. Siam *Journal on Discrete Mathematics* 10: 551-572.

[62] Gollub, J., Ball, C.A., Binkley, G., Demeter, J., Finkelstein, D.B., Hebert, J.M., Hernandez-Boussard, T., Jin, H., Kaloper, M., Matese, J.C. et al. (2003) The Stanford Microarray Database: data access and quality assessment tools. *Nucleic Acids Res*, 31, 94-96.

[63] Grainger DC, Hurd D, Harrison M, Holdstock J, Busby SJ (2005) Studies of the distribution of Escherichia coli cAMP-receptor protein and RNA polymerase along the *E. coli* chromosome. *Proc Natl Acad Sci U S A* 102: 17693-17698.

[64] Grainger DC, Aiba H, Hurd D, Browning DF, Busby SJ (2007) Transcription factor distribution in Escherichia coli: studies with FNR protein. *Nucleic Acids Res* 35: 269-278.

[65] Groicher, K.H., Firek, B.A., Fujimoto, D.F. and Bayles, K.W. (2000) The Staphylococcus aureus lrgAB operon modulates murein hydrolase activity and penicillin tolerance. *J Bacteriol*, 182, 1794-1801.

[66] Grunberg-Manago M (1989) Escherichia coli and Salmonella typhimurium.; Neidhart F, ed, editor. Washington, D. C.: *American Society for Microbiology*. pp. 1386-1409

[67] Hertz, G.Z., Stormo, G.D.: Identifying DNA and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics* (Oxford, England) 15, 563-577 (1999)

[68] Hill, C.W., Sandt, C.H., Vlazny, D.A. (1994) Rhs elements of Escherichia coli: a family of genetic composites each encoding a large mosaic protein *Mol. Microbiol*, . 12, 865-871.

[69] Hiramatsu, T., Kodama, K., Kuroda, T., Mizushima, T. and Tsuchiya, T. (1998) A putative multisubunit Na+/H+ antiporter from Staphylococcus aureus. *J Bacteriol*, 180, 6642-6648.

[70] Hopocroft J, Karp R (1973) An n$\hat{2}$.5 algorithm for maximum matching in bipartite graphs. Siam Journal on Computing 2: 225-231.

[71] Horsburgh, M.J., Ingham, E. and Foster, S.J. (2001) In Staphylococcus aureus, fur is an interactive regulator with PerR, contributes to virulence, and Is necessary for oxidative stress resistance through positive regulation of catalase and iron homeostasis. *J Bacteriol*, 183, 468-475.

[72] Hu, J., Li, B., Kihara, D.: Limitations and potentials of current motif discovery algorithms. *Nucleic acids research* 33, 4899-4913 (2005)

[73] Jacob, E., Sasikumar, R. and Nair, K.N. (2005) A fuzzy guided genetic algorithm for operon prediction. *Bioinformatics*, 21, 1403-1407.

[74] Janga, S.C., Collado-Vides, J., Moreno-Hagelsieb, G. (2005) Nebulon: a system for the inference of functional relationships of gene products from the rearrangement of predicted operons *Nucleic Acids Res*, . 33, 2521-2530.

[75] Janga, S.C., Lamboy, W.F., Huerta, A.M. and Moreno-Hagelsieb, G. (2006) The distinctive signatures of promoter regions and operon junctions across prokaryotes. *Nucleic Acids Res.*

[76] Jensen, S.T., Shen, L., Liu, J.S.: Combining phylogenetic motif discovery and motif clustering to predict co-regulated genes. *Bioinformatics* (Oxford, England) 21, 3832-3839 (2005)

[77] Jensen, S.T., Liu, J.S.: BioOptimizer: a Bayesian scoring function approach to motif discovery. *Bioinformatics* (Oxford, England) 20, 1557-1564 (2004)

[78] Jensen,R.A. (2001) Orthologs and paralogswe need to get it right. *Genome Biol.*, 2(4): INTERACTIONS1002.

[79] Kanehisa, M., Goto, S., Kawashima, S., Okuno, Y., Hattori, M. (2004) The KEGG resource for deciphering the genome *Nucleic Acids Res*, . 32, D277-D280.

[80] Karp PD, Keseler IM, Shearer A, Latendresse M, Krummenacker M, *et al.* (2007) Multidimensional annotation of the Escherichia coli K-12 genome. *Nucleic Acids Res* 35: 7577-7590.

[81] Kao M, Lam T, Sung W, Ting H (2001) A decompostion theorem for maximum weight bipartite matchings. Siam Journal on Computing 31: 18-26.

[82] Karlin, S., Brocchieri, L., Campbell, A., Cyert, M. and Mrazek, J. (2005) Genomic and proteomic comparisons between bacterial and archaeal genomes and related comparisons with the yeast and fly genomes. *Proceedings of the National Academy of Sciences of the United States of America*, 102, 7309-7314.

[83] Kelley WL (2006) Lex marks the spot: the virulent side of SOS and a closer look at the LexA regulon. *Mol Microbiol* 62: 1228-1238.

[84] Keseler, I.M., Collado-Vides, J., Gama-Castro, S., Ingraham, J., Paley, S., Paulsen, I.T., Peralta-Gil, M., Karp, P.D. (2005) EcoCyc: a comprehensive database resource for Escherichia coli *Nucleic Acids Res*, . 33, D334-D337.

[85] Kies, S., Otto, M., Vuong, C. and Gotz, F. (2001) Identification of the sigB operon in Staphylococcus epidermidis: construction and characterization of a sigB deletion mutant. *Infect Immun*, 69, 7933-7936.

[86] Koonin,E.V. (2001) An apology for orthologsor brave new memes. *Genome Biol.*, 2(4): COMMENT1005.

[87] Kopp, U., Roos, M., Wecke, J. and Labischinski, H. (1996) Staphylococcal peptidoglycan interpeptide bridge biosynthesis: a novel antistaphylococcal target *Microb Drug Resist*, 2, 29-41.

[88] Kremling, A., Jahreis, K., Lengeler, J.W., Gilles, E.D. (2000) The organization of metabolic reaction networks: a signal-oriented approach to cellular models *Metab. Eng*, . 2, 190-200.

[89] Kuhn H (1955) The hungarian method for the assignment-problem. Naval Research Logistics Quarterly 2: 83-97.

[90] Kullik, I.I. and Giachino, P. (1997) The alternative sigma factor sigmaB in Staphylococcus aureus: regulation of the sigB operon in response to growth phase and heat shock. *Arch Microbiol*, 167, 151-159.

[91] Kuroda, M., Hayashi, H. and Ohta, T. (1999) Chromosome-determined zinc-responsible operon czr in Staphylococcus aureus strain 912. *Microbiol Immunol*, 43, 115-125.

[92] Kuroda, M., Kobayashi, D., Honda, K., Hayashi, H. and Ohta, T. (1999) The hsp operons are repressed by the hrc37 of the hsp70 operon in Staphylococcus aureus. *Microbiol Immunol*, 43, 19-27.

[93] Laing, E., Mersinias, V., Smith, C.P. and Hubbard, S.J. (2006) Analysis of gene expression in operons of Streptomyces coelicolor. *Genome Biol*, 7, R46.

[94] Lathe, W.C., IIIrd, Snel, B., Bork, P. (2000) Gene context conservation of a higher order than operons *Trends. Biochem. Sci*, . 25, 474-479.

[95] Lawrence,J. (1999) Selfish operons: the evolutionary impact of gene clustering in prokaryotes and eukaryotes *Curr. Opin. Genet. Dev*, . 9, 642-648.

[96] Lecompte, O., Ripp, R., Thierry, J.C., Moras, D. and Poch, O. (2002) Comparative analysis of ribosomal proteins in complete genomes: an example of reductive evolution at the domain scale. *Nucleic Acids Res*, 30, 5382-5390.

[97] Lemmens K, Dhollander T, De Bie T, Monsieurs P, Engelen K, et al. (2006) Inferring transcriptional modules from ChIP-chip, motif and microarray data. *Genome Biol* 7: R37.

[98] Li G, Che D, Xu Y (2008) A universal operon predictor for prokaryotic genomes *Journal of Bioinformatics and Computational Biology.*

[99] Li, L., Stoeckert, C.J., Jr., Roos, D.S.: OrthoMCL: identification of ortholog groups for eukaryotic genomes. *Genome research* 13, 2178-2189 (2003)

[100] Liu, X., Brutlag, D., Liu, J.: BioProspector: discovering conserved DNA motifs in upstream regulatory regions of coexpressed genes. *Pac. Symp. Biocomput.* 127-138 (2001)

[101] Liu X, Noll DM, Lieb JD, Clarke ND (2005) DIP-chip: rapid and accurate determination of DNA-binding specificity. *Genome Res* 15: 421-427.

[102] Llamas MA, Ramos JL, Rodriguez-Herva JJ (2003) Transcriptional organization of the Pseudomonas putida tol-oprL genes. *J Bacteriol* 185: 184-195.

[103] Mao, F., Su, Z., Olman, V., Dam, P., Liu, Z., Xu, Y. (2006) Mapping of orthologous genes in the context of biological pathways: an application of integer programming *Proc. Natl Acad. Sci.* USA, 103, 129-134.

[104] Martin, F., and Jean-Stephane, V. 2004. Detecting uber-operons in Prokaryotics Genomes. Laboratoire d'Informatique Fondamentale de Lille, France.

[105] Masalha, M., Borovok, I., Schreiber, R., Aharonowitz, Y. and Cohen, G. (2001) Analysis of transcription of the Staphylococcus aureus aerobic class Ib and anaerobic class III ribonucleotide reductase genes in response to oxygen. *J Bacteriol*, 183, 7260-7272.

[106] Maurer LM, Yohannes E, Bondurant SS, Radmacher M, Slonczewski JL (2005) pH regulates genes for flagellar motility, catabolism, and oxidative stress in Escherichia coli K-12. *J Bacteriol* 187: 304-319.

[107] McCue, L., Thompson, W., Carmack, C., Ryan, M.P., Liu, J.S., Derbyshire, V., Lawrence, C.E.: Phylogenetic footprinting of transcription factor binding sites in proteobacterial genomes. *Nucleic acids research* 29, 774-782 (2001)

[108] McCue, L.A., Thompson, W., Carmack, C.S., Lawrence, C.E.: Factors influencing the identification of transcription factor binding sites by cross-species comparison. *Genome research* 12, 1523-1532 (2002)

[109] McGuire, A.M., Hughes, J.D., Church, G.M.: Conservation of DNA regulatory motifs and discovery of new motifs in microbial genomes. *Genome research* 10, 744-757 (2000)

[110] Mehlhorn, K., Nher, S.: Leda : a platform for combinatorial and geometric computing. *Cambridge University Press*, Cambridge, U.K.; New York (1999)

[111] Mironov AA, Koonin EV, Roytberg MA, Gelfand MS (1999) Computer analysis of transcription regulatory patterns in completely sequenced bacterial genomes. *Nucleic Acids Res* 27: 2981-2989.

[112] Morrissey, J.A., Cockayne, A., Hill, P.J. and Williams, P. (2000) Molecular cloning and analysis of a putative siderophore ABC transporter from Staphylococcus aureus. *Infect Immun*, 68, 6281-6288.

[113] Munch, R., Hiller, K., Grote, A., Scheer, M., Klein, J., Schobert, M., Jahn, D.: Virtual Footprint and PRODORIC: an integrative framework for regulon prediction in prokaryotes. *Bioinformatics* (Oxford, England) 21 (2005) 4187-4189

[114] Mushegian, A.R. and Koonin, E.V. (1996) A minimal gene set for cellular life derived by comparison of complete bacterial genomes *Proc. Natl Acad. Sci.* USA, 93, 10268-10273

[115] Nakao, A., Imai, S. and Takano, T. (2000) Transposon-mediated insertional mutagenesis of the D-alanyl-lipoteichoic acid (dlt) operon raises methicillin resistance in Staphylococcus aureus. *Res Microbiol*, 151, 823-829.

[116] Neph, S., Tompa, M.: MicroFootPrinter: a tool for phylogenetic footprinting in prokaryotic genomes. *Nucleic acids research* 34 (2006) W366-368

[117] Ng, S.Y., Chaban, B. and Jarrell, K.F. (2006) Archaeal flagella, bacterial flagella and type IV pili: a comparison of genes and posttranslational modifications. *J Mol Microbiol Biotechnol*, 11, 167-191.

[118] Norman, R. Z., and Rabin, M. O. 1959. An algorithm for a minimum cover of a graph. *Proc. Am. Math. Soc.* I0, 315-319.

[119] Okuda, S., Katayama, T., Kawashima, S., Goto, S. and Kanehisa, M. (2006) ODB: a database of operons accumulating known operons across multiple genomes. *Nucleic Acids Res*, 34, D358-362.

[120] Olman, V., Peng, H., Su, Z., Xu, Y. (2004) Mapping of microbial pathways through constrained mapping of orthologous genes *Proc IEEE Comput Syst Bioinform Conf*, . 363-370 .

[121] Ouyang, S., Sau, S. and Lee, C.Y. (1999) Promoter analysis of the cap8 operon, involved in type 8 capsular polysaccharide production in Staphylococcus aureus. *J Bacteriol*, 181, 2492-2500.

[122] Partridge JD, Browning DF, Xu M, Newnham LJ, Scott C, et al. (2008) Characterization of the Escherichia coli K-12 ydhYVWXUT operon: regulation by FNR, NarL and NarP. *Microbiology* 154: 608-618.

[123] Pavesi G, Mereghetti P, Mauri G, Pesole G (2004) Weeder Web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes. *Nucleic Acids Res* 32: W199-203.

[124] Pereira-Leal, J.B., Enright, A.J., Ouzounis, C.A. (2004) Detection of functional modules from protein interaction networks *Proteins*, 54, 49-57.

[125] Petsko,G.A. (2001) Homologuephobia. *Genome Biol.*, 2(2): COMMENT1002.

[126] Price, M.N., Huang, K.H., Alm, E.J., Arkin, A.P.: A novel method for accurate operon predictions in all sequenced prokaryotes. *Nucleic acids research* 33 (2005) 880-892

[127] Qin ZS, McCue LA, Thompson W, Mayerhofer L, Lawrence CE, et al. (2003) Identification of co-regulated genes through Bayesian clustering of predicted regulatory binding sites. *Nat Biotechnol* 21: 435-439.

[128] Reed, S.B., Wesson, C.A., Liou, L.E., Trumble, W.R., Schlievert, P.M., Bohach, G.A. and Bayles, K.W. (2001) Molecular characterization of a novel Staphylococcus aureus serine protease operon. *Infect Immun*, 69, 1521-1527.

[129] Ren B, Robert F, Wyrick JJ, Aparicio O, Jennings EG, et al. (2000) Genome-wide location and function of DNA binding proteins. *Science* 290: 2306-2309.

[130] Rogozin, I.B., Makarova, K.S., Murvai, J., Czabarka, E., Wolf, Y.I., Tatusov, R.L., Szekely, L.A., and Koonin, E.V. 2002. Connected gene neighborhoods in prokaryotic genomes. Nucleic Acids Res 30, 2212-2223.

[131] Romero, P.R. and Karp, P.D. (2004) Using functional and organizational information to improve genome-wide computational prediction of transcription units on pathway-genome databases *Bioinformatics*, 20, 709-717.

[132] Rosas-Magallanes, V., Deschavanne, P., Quintana-Murci, L., Brosch, R., Gicquel, B. and Neyrolles, O. (2006) Horizontal transfer of a virulence operon to the ancestor of Mycobacterium tuberculosis. *Mol Biol Evol*, 23, 1129-1135.

[133] Roth, F.P., Hughes, J.D., Estep, P.W., Church, G.M.: Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. *Nature biotechnology* 16 (1998) 939-945

[134] Sabatti, C., Rohlin, L., Oh, M.K. and Liao, J.C. (2002) Co-expression pattern from DNA microarray experiments as a tool for operon prediction. *Nucleic Acids Res*, 30, 2886-2893.

[135] Salgado, H., Gama-Castro, S., Martinez-Antonio, A., Diaz-Peredo, E., Sanchez-Solano, F., Peralta-Gil, M., Garcia-Alonso, D., Jimenez-Jacinto, V., Santos-Zavaleta, A., Bonavides-Martinez, C. et al. (2004) RegulonDB (version 4.0): transcriptional regulation, operon organization and growth conditions in Escherichia coli K-12. *Nucleic Acids Res*, 32, D303-306.

[136] Salgado, H., Moreno-Hagelsieb, G., Smith, T.F. and Collado-Vides, J. (2000) Operons in Escherichia coli: genomic analyses and predictions. *Proceedings of the National Academy of Sciences of the United States of America*, 97, 6652-6657.

[137] Savopoulos, J.W., Hibbs, M., Jones, E.J., Mensah, L., Richardson, C., Fosberry, A., Downes, R., Fox, S.G., Brown, J.R. and Jenkins, O. (2001) Identification, cloning, and expression of a functional phenylalanyl-tRNA synthetase (pheRS) from Staphylococcus aureus. *Protein Expr Purif*, 21, 470-484.

[138] Schmitz G, Nikaido K, Ames GF (1988) Regulation of a transport operon promoter in Salmonella typhimurium: identification of sites essential for nitrogen regulation. *Mol Gen Genet* 215: 107-117.

[139] Sekino N, Sekine Y, Ohtsubo E (1995) IS1-encoded proteins, InsA and the InsA-B'-InsB transframe protein (transposase): functions deduced from their DNA-binding ability. *Adv Biophys* 31: 209-222.

[140] Sekowska, A., Kung, H.F., Danchin, A. (2000) Sulfur metabolism in Escherichia coli and related bacteria: facts and fiction *J. Mol. Microbiol. Biotechnol*,. 2, 145-177.

[141] Sierro N, Makita Y, de Hoon M, Nakai K (2008) DBTBS: a database of transcriptional regulation in Bacillus subtilis containing upstream intergenic conservation information. *Nucleic Acids Res* 36: D93-96.

[142] Sinha S, Tompa M (2003) YMF: A program for discovery of novel transcription factor binding sites by statistical overrepresentation. Nucleic Acids Res 31: 3586-3588.

[143] Stafford GP, Ogi T, Hughes C (2005) Binding and transcriptional activation of non-flagellar genes by the Escherichia coli flagellar master regulator FlhD2C2. *Microbiology* 151: 1779-1788.

[144] Stranden, A.M., Roos, M. and Berger-Bachi, B. (1996) Glutamine synthetase and heteroresistance in methicillin-resistant Staphylococcus aureus. *Microb Drug Resist*, 2, 201-207.

[145] Su, Z., Olman, V., Mao, F., Xu, Y. (2005) Comparative genomics analysis of NtcA regulons in cyanobacteria: regulation of nitrogen assimilation and its coupling to photosynthesis *Nucleic Acids Res*,. 33, 5156-5171.

[146] Su, Z., Dam, A., Chen, X., Olman, V., Jiang, T., Palenik, B., Xu, Y. (2003) Computational inference of regulatory pathways in microbes: an application to the construction of phosphorus assimilation pathways in Synechococcus WH8102 *Genome Inform Ser Workshop Genome Inform*, . 14, 3-13.

[147] Tagle, D.A., Koop, B.F., Goodman, M., Slightom, J.L., Hess, D.L., Jones, R.T.: Embryonic epsilon and gamma globin genes of a prosimian primate (Galago crassicaudatus). Nucleotide and amino acid sequences, developmental regulation and phylogenetic footprints. *Journal of molecular biology* 203 (1988) 439-455

[148] Tan, K., Moreno-Hagelsieb, G., Collado-Vides, J., Stormo, G.D.: A comparative genomics approach to prediction of new members of regulons. *Genome research* 11 (2001) 566-584

[149] Tan K, McCue LA, Stormo GD (2005) Making connections between novel transcription factors and their DNA motifs. *Genome Res* 15: 312-320.

[150] Tanay A, Sharan R, Kupiec M, Shamir R (2004) Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data. *Proc Natl Acad Sci U S A* 101: 2981-2986.

[151] Tatusov, R.L., Koonin, E.V., Lipman, D.J.: A genomic perspective on protein families. *Science* 278 (1997) 631-637

[152] Tavazoie S, Hughes JD, Campbell MJ, Cho RJ, Church GM (1999) Systematic determination of genetic network architecture. *Nat Genet* 22: 281-285.

[153] Thijs G, Lescot M, Marchal K, Rombauts S, De Moor B, et al. (2001) A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling. *Bioinformatics* 17: 1113-1122.

[154] Thompson JD, Gibson TJ, Plewniak F, Jeanmougin F, Higgins DG (1997) The CLUSTAL_X windows interface: flexible strategies for multiple sequence alignment aided by quality analysis tools. *Nucleic Acids Res* 25: 4876-4882.

[155] Tran, T.T., Dam, P., Su, Z., Poole, F.L., 2nd, Adams, M.W., Zhou, G.T. and Xu, Y. (2007) Operon prediction in Pyrococcus furiosus. *Nucleic Acids Res*, 35, 11-20.

[156] van Helden J, Rios AF, Collado-Vides J (2000) Discovering regulatory elements in non-coding sequences by analysis of spaced dyads. *Nucleic Acids Res* 28: 1808-1818.

[157] Vines ED, Marolda CL, Balachandran A, Valvano MA (2005) Defective O-antigen polymerization in tolA and pal mutants of Escherichia coli in response to extracytoplasmic stress. *J Bacteriol* 187: 3359-3368.

[158] Wade JT, Reppas NB, Church GM, Struhl K (2005) Genomic analysis of LexA binding reveals the permissive nature of the Escherichia coli genome and identifies unconventional target sites. *Genes Dev* 19: 2619-2630.

[159] Wagner, R. Transcription Regulation in Prokaryotes, (2000) Oxford, UK Oxford University Press.

[160] Wall, D.P., Fraser, H.B., Hirsh, A.E. (2003) Detecting putative orthologs *Bioinformatics*, 19, 1710-1711.

[161] Wang T, Stormo GD (2003) Combining phylogenetic data with co-regulated genes to identify regulatory motifs. *Bioinformatics* 19: 2369-2380.

[162] Wang, T., Stormo, G.D.: Identifying the conserved network of cis-regulatory sites of a eukaryotic genome. *Proceedings of the National Academy of Sciences of the United States of America* 102 (2005) 17400-17405

[163] Westover, B.P., Buhler, J.D., Sonnenburg, J.L., Gordon, J.I.: Operon prediction without a training set. *Bioinformatics* (Oxford, England) 21 (2005) 880-888

[164] White, M.F. (2003) Archaeal DNA repair: paradigms and puzzles. *Biochem Soc Trans*, 31, 690-693.

[165] Wu, H., Su, Z., Mao, F., Olman, V., Xu, Y. (2005) Prediction of functional modules based on comparative genome analysis and Gene Ontology application *Nucleic Acids Res*,. 33, 2822-2837.

[166] Wu, H., Mao, F., Olman, V., Xu, Y.: Accurate prediction of orthologous gene groups in microbes. *Proceedings IEEE Computational Systems Bioinformatics Conference*, CSB (2005) 73-79

[167] Xie, X., Lu, J., Kulbokas, E.J., Golub, T.R., Mootha, V., Lindblad-Toh, K., Lander, E.S., Kellis, M.: Systematic discovery of regulatory motifs in human promoters and 3' UTRs by comparison of several mammals. *Nature* 434 (2005) 338-345

[168] Yada, T., Nakao, M., Totoki, Y. and Nakai, K. (1999) Modeling and predicting transcriptional units of Escherichia coli genes using hidden Markov models. *Bioinformatics*, 15, 987-993.

[169] Zhao K, Liu M, Burgess RR (2007) Adaptation in bacterial flagellar and motility systems: from regulon members to 'foraging'-like behavior in *E. coli*. *Nucleic Acids Res* 35: 4441-4452.

[170] Zeng E, Mathee K, Narasimhan G. (2007) IEM: An Algorithm for Iterative Enhancement of Motifs Using Comparative Genomics Data, *Proceedings of LSS Computational Systems Bioinformatics Conference* 6:227-235.

[171] Zheng, Y., Szustakowski, J.D., Fortnow, L., Roberts, R.J. and Kasif, S. (2002) Computational identification of operons in microbial genomes. *Genome Res*, 12, 1221-1230.