Computational Complexities of Combinatorial Problems With Applications to Reverse Engineering of Biological Networks

Piotr Berman*
Department of Computer Science and Engineering
Pennsylvania State University
University Park, PA 16802
Email: berman@cse.psu.edu

Bhaskar DasGupta†
Department of Computer Science
University of Illinois at Chicago
Chicago, IL 60607
Email: dasgupta@cs.uic.edu

Eduardo Sontag‡
Department of Mathematics
Rutgers University
New Brunswick, NJ 08903
sontag@math.rutgers.edu

September 17, 2005

1 Introduction

The problems discussed here are motivated by a central concern of contemporary cell biology, that of unraveling (or "reverse engineering") the web of interactions among the components of complex protein and genetic regulatory networks. Notwithstanding the remarkable progress in genetics and molecular biology in the sequencing of the genomes of a number of species, the inference and quantification of interconnections in signaling and genetic networks that are critical to cell function is still a challenging practical and theoretical problem. High-throughput technologies allow the monitoring the expression levels of sets of genes, and the activity states of signaling proteins, providing snapshots of the transcriptional and signaling behavior of living cells. Statistical and machine learning techniques, such as clustering, are often used in order to group genes

*Supported by NSF grant CCR-0208821.
†Supported in part by NSF grants CCR-0206795, CCR-0208749 and IIS-0346973.
‡Partly supported by NSF grants EIA 0205116 and DMS-0504557.

into co-expression patterns, but they are less able to explain functional interactions. An intrinsic difficulty in capturing such interactions in intact cells by traditional genetic experiments or pharmacological interventions is that any perturbation to a particular gene or signaling component may rapidly propagate throughout the network, causing global changes. The question thus arises of how to use the observed global changes to derive interactions between individual nodes. In this chapter we discuss some computational problems that arises in the context of experimental design for reverse engineering of protein and gene networks. Biological networks may have a very large number of species and parameters. For example, the *E. coli* transcription network identified in [15] has 577 interactions involving 116 transcription factors and 419 operons. For such large-scale networks, exhaustive calculations are not practically possible due to combinatorial explosion and this necessitates the design of *provably efficient* approximation algorithms.

2 Motivations

We will first pose our problems in linear algebra terms, and then recast it as a combinatorial question. After that, we will discuss its motivations from systems biology.

2.1 Linear Algebraic Formulations and the Combinatorial Questions

Our problem is described in terms of two matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ such that:

- A is *unknown*;
- B is *initially unknown*, but each of its columns $B_1, B_2, \ldots, B_m$ can be retrieved with a *unit-cost query*;
- the columns of B are in *general position*, *i.e.*, each subset of $\ell \leq n$ columns of B is *linearly independent*;
- the *zero structure* of the matrix $C = AB = (c_{ij})$ is known, *i.e.*, a binary matrix $C^0 = \left(c^0_{ij}\right) \in \{0,1\}^{n\times m}$ is given, and it is known that $c_{ij} = 0$ for each i, j for which $c^0_{ij} = 0$.

The objective is to obtain as much information as possible about A (which, in the motivating application, describes regulatory interactions among genes and/or proteins), while performing "few" queries (each of which may represent the measuring of a complete pattern of gene expression, done under a different set of experimental conditions). For each query that we perform, we obtain a column B_i, and then the matrix C^0 tells us that certain rows of A have zero inner product with B_i.

As a concrete example, let us take $n = 3$, $m = 5$, and suppose that the known information is given by the matrix:

$$C_0 = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

and the two unknown matrices are:

$$A = \begin{bmatrix} -1 & 1 & 3 \\ 2 & -1 & 4 \\ 0 & 0 & -1 \end{bmatrix}, \qquad B = \begin{bmatrix} 4 & 3 & 37 & 1 & 10 \\ 4 & 5 & 52 & 2 & 16 \\ 0 & 0 & -5 & 0 & -1 \end{bmatrix}$$

(the matrix C_0 has zero entries wherever AB has a zero entry). Considering the structure of C_0, we choose to perform four queries, corresponding to the four columns 1,3,4,5 of B, thus obtaining the following data:

$$\begin{bmatrix} 4 & 37 & 1 & 10 \\ 4 & 52 & 2 & 16 \\ 0 & -5 & 0 & -1 \end{bmatrix}. \tag{1.1}$$

What can we say about the unknown matrix A? Let us first attempt to identify its first row, which we call A_1. The first row of the matrix C_0 tells us that the vector A_1 is orthogonal to the first and second columns of (1.1) (which are the same as the first and third columns of B). This is the *only* information about A that we have available, and it is not enough information to uniquely determine A_1, because there is an entire line that is orthogonal to the plane spanned by these two columns, However, we can still find *some* nonzero vector in this line, and conclude that A_1 is an unknown multiple of this vector. This nonzero vector may be obtained by simple linear algebra manipulations. For example, we might add a linearly independent column to the two that we had, obtaining a matrix

$$B_1 = \begin{bmatrix} 4 & 37 & 0 \\ 4 & 52 & 0 \\ 0 & -5 & 1 \end{bmatrix},$$

then pick an arbitrary vector v whose first two entries are zero (to reflect the known orthogonality), let us say $v = [0, 0, 1]$, and finally solve $A_1 B = v$, thus estimating A_1 as vB^{-1}:

$$\hat{A}_1 \;=\; [0,0,1]\, B^{-1} \;=\; [0,0,1] \begin{bmatrix} 13/15 & -37/60 & 0 \\ -1/15 & 1/15 & 0 \\ -1/3 & 1/3 & 1 \end{bmatrix} = [-1/3, 1/3, 1]\,.$$

Notice that this differs from the unknown A_1 only by a scaling. Similarly, we may employ the last two columns of (1.1) to estimate the second row A_2 of A, again only up to a multiplication by a constant, and we may use the first and third columns of (1.1) (which are the same as the first and fourth columns of B) to estimate the last row, A_3.

Notice that there are always intrinsic limits to what can be accomplished: if we multiply each row of A by some nonzero number, then the zero structure of C is unchanged. Thus, as in the example, the best that we can hope for is to identify the rows of A up to scalings (in abstract mathematical terms, as elements of the projective space $\mathbb{P}^{n-1}$). To better understand these geometric constraints, let us reformulate the problem as follows. Let A_i denote the i^{th} row of A. Then the specification of C^0 amounts to the specification of *orthogonality relations* $A_i \cdot B_j = 0$ for each pair i, j for which $c^0_{ij} = 0$. Suppose that we decide to query the columns of B indexed by $J = \{j_1, \ldots, j_\ell\}$. Then, the information obtained about A may be summarized as $A_i \in \mathcal{H}^{\perp}_{J,i}$, where "$\perp$" indicates *orthogonal complement*, $\mathcal{H}_{J,i} = \text{span}\,\{B_j, j \in J_i\}$,, and $J_i = \{j \mid j \in J \text{ and } c^0_{ij} = 0\}$ .. Suppose now that the set of indices of selected queries J has the property:

$$\text{each set } J_i,\ i = 1, \ldots, n, \ \text{ has cardinality } \ \geq n - k, \tag{1.2}$$

for some given integer k. Then, because of the general position assumption, the space $\mathcal{H}_{J,i}$ has dimension $\geq n - k$, and hence the space $\mathcal{H}^{\perp}_{J,i}$ has dimension at most k.

The case $k = 1$

The most desirable special case is that in which $k = 1$. Then $\dim \mathcal{H}^{\perp}_{J,i} \leq 1$, hence each A_i is uniquely determined up to a scalar multiple, which is the best that could be theoretically achieved. Often, in fact, finding the sign pattern (such as "$(+, +, -, 0, 0, -, \ldots)$") for each row of A is the main experimental goal (this would correspond, in our motivating application, to determining if the regulatory interactions affecting each given gene or protein are *inhibitory* or *catalytic*). Assuming that the degenerate case $\mathcal{H}^{\perp}_{J,i} = \{0\}$ does not hold (which would determine $A_i = 0$), once that an arbitrary nonzero element v in the line $\mathcal{H}^{\perp}_{J,i}$ has been picked, there are only two sign patterns possible for A_i (the pattern of v and that of $-v$). If, in addition, one knows at least one nonzero sign in A_i, then the sign structure of the whole row has been *uniquely* determined (in the motivating biological question, typically one such sign is indeed known; for example, the diagonal elements a_{ii}, i.e. the ith element of each A_i, is known to be negative, as it represents a degradation rate). Thus, we will be interested in this question:

$$\text{find } J \text{ of minimal cardinality such that } |J_i| \geq n - 1,\ i = 1, \ldots, n. \tag{Q1}$$

If queries have variable unit costs (different experiments have a different associated cost), this problem must be modified to that of minimizing a suitable linear combination of costs, instead of the number of queries.

The general case $k > 1$

More generally, suppose that the queries that we performed satisfy (1.2), with $k > 1$ but small k. It is not true anymore that there are only two possible sign patterns for any given A_i, but the number of possibilities is still very small. For simplicity, let us assume that we know that no entry of A_i is zero (if this is not the case, the number of possibilities may increase, but the argument is very similar). We wish to prove that the possible number of signs is much smaller than 2^n. Indeed, suppose that the queries have been performed, and that we then calculate, based on the obtained B_j's, a basis $\{v_1, \ldots, v_k\}$ of $\mathcal{H}_{J,i}^{\perp}$ (assume $\dim \mathcal{H}_{J,i}^{\perp} = k$; otherwise pick a smaller k). Thus, the vector A_i is known to have the form $\sum_{r=1}^{k} \lambda_r v_r$ for some (unknown) real numbers $\lambda_1, \ldots, \lambda_k$. We may assume that $\lambda_1 \neq 0$ (since, if $A_i = \sum_{r=2}^{k} \lambda_r v_r$, the vector $\varepsilon v_1 + \sum_{r=2}^{k} \lambda_r v_r$, with small enough ε, has the same sign pattern as A_i, and we are counting the possible sign patterns). If $\lambda_1 > 0$, we may divide by λ_1 and simply count how many sign patterns there are when $\lambda_1 = 1$; we then double this estimate to include the case $\lambda_1 < 0$. Let $v_r = \mathrm{col}\,(v_{1r}, \ldots, v_{nr})$, for each $r = 1, \ldots, k$. Since no coordinate of A_i is zero, we know that A_i belongs to the set $\mathcal{C} = \mathbb{R}^{k-1} \setminus (L_1 \bigcup \ldots \bigcup L_n)$ where, for each $1 \leq s \leq n$, L_s is the hyperplane in $\mathbb{R}^{k-1}$ consisting of all those vectors $(\lambda_2, \ldots, \lambda_k)$ such that $\sum_{r=2}^{k} \lambda_r v_{sr} = -v_{s1}$. On each connected component of $\mathcal{C}$, signs patterns are constant. Thus the possible number of sign patterns is upper bounded by the maximum possible number of connected regions determined by n hyperplanes in dimension $k-1$. A result of L. Schläfli (see [6, 14], and also [17] for a discussion, proof, and relations to Vapnik-Chervonenkis dimension) states that this number is bounded above by $\Phi(n, k-1)$, provided that $k - 1 \leq n$, where $\Phi(n, d)$ is the number of possible subsets of an n-element set with at most d elements, that is, $\Phi(n, d) = \sum_{i=0}^{d} \binom{n}{i} \leq 2\frac{n^d}{d!} \leq \left(\frac{\mathbf{e}n}{d}\right)^d$. Doubling the estimate to include $\lambda_1 < 0$, we have the upper bound $2\Phi(n, k-1)$. For example, $\Phi(n, 0) = 1$, $\Phi(n, 1) = n + 1$, and $\Phi(n, 2) = \frac{1}{2}(n^2 + n + 2)$. Thus we have an estimate of 2 sign patterns when $k = 1$ (as obtained earlier), $2n + 2$ when $k = 2$, $n^2 + n + 2$ when $k = 3$, and so forth. In general, the number grows only polynomially in n (for fixed k).

These considerations lead us to formulating the generalized problem, for each fixed k: *find J of minimal cardinality such that $|J_i| \geq n - k$* for all $i = 1, \ldots, n$. Recalling the definition of J_i, we see that $J_i = J \bigcap T_i$, where $T_i = \{j \mid c_{ij}^0 = 0\}$. Thus, we can reformulate our question purely combinatorially, as a more general version of Question **(Q1)** as follows. Given sets $T_i \subseteq \{1, \ldots, m\}$, $\quad i = 1, \ldots, n$, and an integer $k < n$, the problem is:

$$\textit{find } J \subseteq \{1, \ldots, m\} \textit{ of minimal cardinality such that } \; |J \bigcap T_i| \geq n - k,\ 1 \leq i \leq n. \qquad \textbf{(Q2)}$$

For example, suppose that $k = 1$, and pick the matrix $C^0 \in \{0, 1\}^{n \times n}$ in such a way that the columns of C^0 are the binary vectors representing all the $(n-1)$-element subsets of $\{1, \ldots, n\}$ (so $m = n$); in this case, the set J must equal $\{1, \ldots, m\}$ and hence has cardinality n. On the other hand, also with $k = 1$, if we pick the matrix C^0 in such a way that the columns of C^0 are the binary vectors representing all the 2-element subsets of $\{1, \ldots, n\}$ (so $m = n(n-1)/2$), then J must again be the set of all columns (because, since there are only two zeros in each column, there can only be a total of 2ℓ zeros, $\ell = |J|$, in the submatrix indexed

by J, but we also have that $2\ell \geq n(n-1)$, since each of the n rows must have $\geq n-1$ zeros); thus in this case the minimal cardinality is $n(n-1)/2$.

2.2 Set Multicover Formulations

The algorithmic questions posed in the previous section can be cast as variations of a generic combinatorial set multicover problem, defined as follows. Suppose that we are given an universe U, a set of subsets Γ of U and a positive integer k with $\mid \{u \in \gamma \mid \gamma \in \Gamma\} \mid \geq k$ for every $u \in U$. Then, our problem is the following integer programming problem:

$$\text{minimize} \sum_{\gamma \in \Gamma} x_\gamma \text{ subject to } \begin{array}{ll} \sum_{u \in \gamma \in \Gamma} x_\gamma \geq k & \text{for each } u \in U \\ x_\gamma \in \{0,1\} & \text{for each } \gamma \in \Gamma \end{array}$$

Basic versions of question **(Q1)** and **(Q2)** in the next section can be cast in a similar formulation; see references [4, 5]. An appropriate *on-line* variation of the set-multicover problem, as outlined in the reference [3], is also appropriate for the reverse engineering problems as will be mentioned in the next section.

2.3 Motivations from Systems Biology

The biological motivation stems from an effort by many research groups whose goal is to infer mechanistic relationships underlying the observed behavior of complex molecular networks. We focus our attention here solely on one such approach, originally described in [9, 10], further elaborated upon in [2, 16], and reviewed in [7, 18]. In this approach, the architecture of the network is inferred on the basis of observed global responses (namely, the steady-state concentrations in changes in the phosphorylation states or activities of proteins, mRNA levels, or transcription rates) in response to experimental perturbations (representing the effect of hormones, growth factors, neurotransmitters, or of pharmacological interventions).

In the setup in [9, 10, 16], the time evolution of a vector of state variables $x(t) = (x_1(t), \ldots, x_n(t))$ is described by a system of differential equations:

$$\dot{x} = f(x,p) \equiv \begin{array}{l} \dot{x}_1 = f_1(x_1, \ldots, x_n, p_1, \ldots, p_m) \\ \vdots \\ \dot{x}_n = f_n(x_1, \ldots, x_n, p_1, \ldots, p_m) \end{array}$$

where the dot indicates time derivative and $p = (p_1, \ldots, p_m)$ is a vector of parameters, which can be manipulated but remain constant during any given experiment. The components $x_i(t)$ of the state vector represent quantities that can be in principle measured, such as levels of activity of selected proteins or transcription rates of certain genes. The parameters p_i represent quantities that can be manipulated, perhaps indirectly, such as levels of hormones or of enzymes whose half-lives are long compared to the rate at which

the variables evolve. A basic assumption (but see [16] for a time-dependent analysis) is that states converge to steady state values, and these are the values used for network identification. There is a reference value $\bar{p}$ of p, which represents "wild type" (that is, normal) conditions, and a corresponding steady state $\bar{x}$. Mathematically, $f(\bar{x},\bar{p})=0$. We are interested in obtaining information about the Jacobian of the vector field f evaluated at $(\bar{x},\bar{p})$, or at least about the signs of the derivatives $\partial f_i/\partial x_j(\bar{x},\bar{p})$. For example, if $\partial f_i/\partial x_j > 0$, this means that x_j has a positive (catalytic) effect upon the rate of formation of x_i. The critical assumption, indeed the main point of [9, 10, 16], is that, while we may not know the form of f, we often do know that *certain parameters p_j do not directly affect certain variables x_i.* This amounts to *a priori* biological knowledge of specificity of enzymes and similar data. In the current context, this knowledge is summarized by the binary matrix $C^0=\left(c^0_{ij}\right)\in\{0,1\}^{n\times m}$, where "$c^0_{ij}=0$" means that p_j does not appear in the equation for $\dot{x}_i$, that is, $\partial f_i/\partial p_j\equiv 0$.

The experimental protocol allows one to perturb any one of the parameters, let us say the kth one, while leaving the remaining ones constant. (A generalization, to allow for the simultaneous perturbation of more than one parameter, is of course possible.) For the perturbed vector $p\approx\bar{p}$, one then measures the resulting steady state vector $x=\xi(p)$. Experimentally, this may for instance mean that the concentration of a certain chemical represented by p_k is kept are a slightly altered level, compared to the default value $\bar{p}_k$; then, the system is allowed to relax to steady state, after which the complete state x is measured, for example by means of a suitable biological reporting mechanism, such as a microarray used to measure the expression profile of the variables x_i. Mathematically, we suppose that for each vector of parameters p in a neighborhood of $\bar{p}$ there is a unique steady state $\xi(p)$ of the system, where ξ is a differentiable function. For each of the possible m experiments, in which a given p_j is perturbed, we may estimate the n "sensitivities" $b_{ij}=\frac{\partial\xi_i}{\partial p_j}(\bar{p})\approx\frac{1}{\bar{p}_j-p_j}\left(\xi_i(\bar{p}+p_je_j)-\xi_i(\bar{p})\right)\ ,\quad i=1,\ldots,n$, where $e_j\in\mathbb{R}^m$ is the j^{th} canonical basis vector. We let B denote the matrix consisting of the b_{ij}'s. (See [9, 10] for a discussion of the fact that division by $\bar{p}_j-p_j$, which is undesirable numerically, is not in fact necessary.) Finally, we let A be the Jacobian matrix $\partial f/\partial x$ and let C be the negative of the Jacobian matrix $\partial f/\partial p$. From $f(\xi(p),p)\equiv 0$, taking derivatives with respect to p, and using the chain rule, we get that $C=AB$. This brings us to the problem stated in the previous section; the general position assumption is reasonable, since we are dealing with experimental data.

2.4 Online Versions of Questions of the Type (Q1) or (Q2)

The online versions of the questions of the type **(Q1)** or **(Q2)** are more suited to the case when one performs an experimental protocol which is slightly different from the one described in Section 2.1 and described below:

- Let $J_i\subseteq\{j\mid c_{ij}=1\}$ be the indices of the sets chosen in our set-multicover. Then, each $j\in J_i$ is associated with an experiment of the following type:

- Change (perturb) only the parameter p_j.
- For the perturbed vector $p \approx \bar{p}$, we measure the resulting steady state value $x_i = \xi_i(p)$. Experimentally, this may for instance mean that the concentration of a certain chemical represented by p_j is kept are a slightly altered level, compared to the default value $\bar{p}_j$; then, the system is allowed to relax to steady state, after which the steady state x_i is measured, for example by means of a suitable biological reporting mechanism, such as a fluorescent proteins[1]. Mathematically, we suppose that for each vector of parameters p in a neighborhood of $\bar{p}$ there is a unique steady state $\xi_i(p)$ of x_i, where ξ_i is a differentiable function.
- Estimate the corresponding "sensitivity"

$$b_{ij} = \frac{\partial \xi_i}{\partial p_j}(\bar{p}) \approx \frac{1}{\bar{p}_j - p_j}\left(\xi_i(\bar{p} + p_j e_j) - \xi_i(\bar{p})\right)$$

(where $e_j \in \mathbb{R}^m$ is the j^{th} canonical basis vector).

The cost of doing these experiments is amortized against the weights of the sets, the unweighted case being the simplest case when we just wish to minimize the number of experiments.

These considerations motivate us to look at the online versions of questions **(Q1)** and **(Q2)** which can be abstracted as an online set multicover problem as follows. we have an universe V of elements, a family $\mathcal{S}$ of subsets of V with a positive real cost c_S for every $S \in \mathcal{S}$, and a "coverage factor" (positive integer) k. A subset $\{i_0, i_1, \ldots\} \subseteq V$ of elements are presented online in an arbitrary order. When each element i_p is presented, we are also told the collection of all (at least k) sets $\mathcal{S}_{i_p} \subseteq \mathcal{S}$ and their costs in which i_p belongs and we need to select additional sets from $\mathcal{S}_{i_p}$ if necessary such that our collection of selected sets contains *at least* k sets that contain the element i_p. The goal is to *minimize* the *total cost* of the selected sets.

3 Algorithms and Computational Complexities

References [9, 10, 18] survey biological motivations for Jacobian estimation under the assumptions given above, prove various results, and provide simulations of realistic biological systems in which the technique successfully recovers the Jacobian. In the next two subsections, we discuss most recent algorithmic developments for both the offline version and the online version of the problems.

3.1 Offline version

In [4, 5] we investigated the algorithmic complexity of Question **(Q2)** and provided *randomized* approximation algorithms with *expected* performance ratios of about 2 for $k = 1$. This was obtained in two steps. In the first

[1] Fluorescent proteins can be used to know the rate at which a certain gene trabscribes in a cell under a set of conditions.

step, we showed an equivalence of this problem with the set-multicover formulation outlined in Section 2.2. We then considered a randomized approximation algorithm for this problem in the following manner via the "linear programming with nontrivial rounding" approach:

- Let $c = \begin{cases} \ln a & \text{if } k = 1 \\ \ln(a/(k-1)) & \text{if } a/(k-1) \geq \mathbf{e}^2 \text{ and } k > 1 \\ 2 & \text{if } \frac{1}{4} < a/(k-1) < \mathbf{e}^2 \text{ and } k > 1 \\ 1 + \sqrt{\frac{a}{k}} & \text{otherwise} \end{cases}$.
- Find a solution vector $\mathbf{x}^* \in \mathbb{R}^{|U|}$ to the LP relaxation of the formulation in Section 2.2 via algorithms such as [8]. Let x_j^* denote the j^{th} component of this solution vector.
- Form a family of sets $\mathcal{C}_0 = \{\gamma \ : \ cx_\gamma^* \geq 1\}$.
- Form a family of sets $\mathcal{C}_1 \subseteq \mathcal{S} - \mathcal{C}_0$ by selecting a set $\gamma \in \Gamma \backslash \mathcal{C}_0$ with probability cx_γ^*.
- Form a family of sets $\mathcal{C}_2$ by greedy choices: if an $u \in U$ belongs to fewer than k sets in $\mathcal{C}_0 \cup \mathcal{C}_1$, choose any of the remaining sets that contains u.
- Return $\mathcal{C} = \mathcal{C}_0 \cup \mathcal{C}_1 \cup \mathcal{C}_2$ as the solution.

Then, we were able to prove the following result on the performance of this algorithm.

Theorem 1.1 *The expected performance ratio of our algorithm is given by*

$$\begin{cases} 1 + \ln a, & \text{if } k = 1 \\ \left(1 + \mathbf{e}^{-(k-1)/5}\right) \ln(a/(k-1)), & \text{if } a/(k-1) \geq \mathbf{e}^2 \approx 7.39 \text{ and } k > 1 \\ \min\{\, 2 + 2 \cdot \mathbf{e}^{-(k-1)/5},\ 2 + \left(\mathbf{e}^{-2} + \mathbf{e}^{-9/8}\right) \cdot \frac{a}{k} \,\} \\ \quad \approx \min\{\, 2 + 2 \cdot \mathbf{e}^{-(k-1)/5},\ 2 + 0.46 \cdot \frac{a}{k} \,\} & \text{if } \frac{1}{4} < a/(k-1) < \mathbf{e}^2 \text{ and } k > 1 \\ 1 + 2\sqrt{\frac{a}{k}} & \text{if } a/(k-1) \leq \frac{1}{4} \text{ and } k > 1 \end{cases}$$

3.2 Online version

In [3] we describe a new randomized algorithm for the online multicover problem based on a randomized version of the *winnowing approach* of [12]. The winnowing algorithm has two scaling factors: a multiplicative scaling factor $\frac{\mu}{c_S}$ that depends on the particular set S containing i and another additive scaling factor $|\mathcal{S}_i|^{-1}$ that depends on the number of sets that contain i. These scaling factors quantify the appropriate level of "promotion" in the winnowing approach.

// definition //
D1 **for** $(i \in V)$
D2 $\quad \mathcal{S}_i \leftarrow \{s \in \mathcal{S} : \ i \in S\}$

// initialization //
I1 $\mathcal{T} \leftarrow \emptyset$ // $\mathcal{T}$ is our collection of selected sets //
I2 **for** $(S \in \mathcal{S})$
I3 $\quad \alpha p[S] \leftarrow 0$ // accumulated probability of each set //

// after receiving an element i //
A1 $deficit \leftarrow k - |\mathcal{S}_i \cap \mathcal{T}|$ // k is the coverage factor //
A2 **if** $deficit = 0$ // we need $deficit$ more sets for i //
A3 $\quad$ finish the processing of i
A4 $\mathcal{A} \leftarrow \emptyset$
A5 **repeat** $deficit$ **times**
A6 $\quad S \leftarrow$least cost set from $\mathcal{S}_i - \mathcal{T} - \mathcal{A}$
A7 $\quad$ insert S to $\mathcal{A}$
A8 $\mu \leftarrow c_S$ // μ is the cost of the last set added to $\mathcal{A}$ //
A9 **for** $(S \in \mathcal{S}_i - \mathcal{T})$
A10 $\quad p[S] \leftarrow \min\left\{\frac{\mu}{c_S}\left(\alpha p[S] + |\mathcal{S}_i|^{-1}\right), 1\right\}$ // probability for this step //
A11 $\quad \alpha p[S] \leftarrow \alpha p[S] + p[S]$ // accumulated probability //
A12 $\quad$ **with** probability $p[S]$
A13 $\quad\quad$ insert S to $\mathcal{T}$ // randomized selection //
A14 $deficit \leftarrow k - |\mathcal{S}_i \cap \mathcal{T}|$
A15 **repeat** $deficit$ **times** // greedy selection //
A16 $\quad$ insert a least cost set from $\mathcal{S}_i - \mathcal{T}$ to $\mathcal{T}$

Figure 1.1: Algorithm **A-Universal**

This algorithm generalizes and improves some earlier results in [1]. We proved the following performance bounds for this algorithm.

Theorem 1.2 *The expected performance ratio of Algorithm* **A-Universal** *is at most* $\log_2 m \ln d$ *plus lower order terms, where* d *is the maximum number of elements in any set and* m *is the number of sets.*

We also discussed in [3] lower bounds on competitive ratios for *deterministic algorithms* for general k based on the approaches in [1]

4 Conclusions and Further Research Problems

Obviously, much research remains to be done regarding the algorithmic and computational complexity of Questions **(Q1)** and **(Q2)** and their generalizations, extensions, and specific applications to gene and protein networks. For example:

- Can we design randomized algorithms with expected performance ratios better than the ones in Theorem 1.1, especially for $k \in [\omega(n), o(n)]$? It seems that a different rounding strategy with a considerably more non-trivial probabilistic analysis may be necessary in order to achieve this goal.

- The set system Γ may have a structure depending on the biological nature of the dependence of the variables p_j on the variable x_i's. This requires a new integer programming formulation in which, for example, "forbidden" (either mutual or as a group) combination of sets may arise (analogously to what is done in [11], for a different problem in reverse engineering). For example, a basic version of the problem that one might consider involves a given set $S \subseteq 2^U$ of forbidden combinations and adding the constraint $\sum_{\gamma \in s} x_\gamma \leq 1$ for every $s \in S$. Interestingly, the computational complexity of the problem changes substantially with these additional constraints.

- How do we design *deterministic* algorithms to derandomize such algorithms efficiently to provide deterministic algorithms? The greedy strategy is shown not to work effectively in [4, 5], hence another strategy may be necessary. A direct derandomization of the randomized algorithm, via standard techniques such as the method of conditional probabilities or the two-point sampling techniques [13], does not seem to generate a computationally efficient deterministic procedure.

References

[1] N. Alon, B. Awerbuch, Y. Azar, N. Buchbinder, and J. Naor. *The online set cover problem*, 35th annual ACM Symposium on the Theory of Computing, pp. 100-105, 2003.

[2] M. Andrec, B.N. Kholodenko, R.M. Levy, and E.D. Sontag, *Inference of Signaling and Gene Regulatory Networks by Steady-State Perturbation Experiments: Structure and Accuracy*, *J. Theoretical Biology*, 232, pp. 427–441, 2005.

[3] P. Berman and B. DasGupta. *Approximating the Online Set Multicover Problems Via Randomized Winnowing*, to appear in 9th Workshop on Algorithms and Data Structures (WADS), Waterloo, Canada, August 15-August 17, 2005.

[4] P. Berman, B. DasGupta and E. Sontag. *Randomized Approximation Algorithms for Set Multicover Problems with Applications to Reverse Engineering of Protein and Gene Networks*, to appear in Discrete Applied Mathematics (special issue on computational biology).

[5] P. Berman, B. DasGupta and E. Sontag. *Randomized Approximation Algorithms for Set Multicover Problems with Applications to Reverse Engineering of Protein and Gene Networks*, 7th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX), LNCS 3122, K. Jansen, S. Khanna, J. D. P. Rolim and D. Ron (editors), Springer Verlag, pp. 39-50, August 2004.

[6] T. Cover. *Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition*, *IEEE Trans. Electronic Computers* EC-14, pp. 326–334, 1965.

[7] E.J. Crampin, S. Schnell, and P.E. McSharry. *Mathematical and computational techniques to deduce complex biochemical reaction mechanisms*, Progress in Biophysics & Molecular Biology, 86, pp. 77-112, 2004.

[8] N. Karmarkar. *A New Polynomial-time Algorithm for Linear Programming*, Combinatorica, 4:373–395, 1984.

[9] B. N. Kholodenko, A. Kiyatkin, F. Bruggeman, E.D. Sontag, H. Westerhoff, and J. Hoek. *Untangling the Wires: A Novel Strategy to Trace Functional Interactions in Signaling and Gene Networks*, Proceedings of the National Academy of Sciences USA 99, pp. 12841-12846, 2002

[10] B. N. Kholodenko and E.D. Sontag. *Determination of Functional Network Structure from Local Parameter Dependence Data*, arXiv physics/0205003, May 2002.

[11] X. Lin, C.A. Floudas, Y. Wang, and J.R. Broach. *Theoretical and Computational Studies of the Glucose Signaling Pathways in Yeast Using Global Gene Expression Data*, *Biotechnol Bioeng.*, 84, pp. 864–886, 2003.

[12] N. Littlestone. *Learning Quickly When Irrelevant Attributes Abound: A New Linear-Threshold Algorithm*, Machine Learning, 2, pp. 285-318, 1988.

[13] R. Motwani and P. Raghavan. *Randomized Algorithms*, Cambridge University Press, New York, NY, 1995.

[14] L. Schläfli. *Theorie der Vielfachen Kontinuitat (1852)*, in *Gesammelte Mathematische Abhandlungen*, volume 1, pp. 177–392, Birkhäuser, Basel, 1950.

[15] S. Shen-Orr,, R. Milo, S. Mangan, S. and U. Alon. *Network Motifs in the Transcriptional Regulation Network of Escherichia Coli*, Nature Genetics, 31, pp 64-68, 2002.

[16] E. D. Sontag, A. Kiyatkin and B. N. Kholodenko. *Inferring Dynamic Architecture of Cellular Networks Using Time Series of Gene Expression, Protein and Metabolite Data*, Bioinformatics 20, pp. 1877-1886, 2004.

[17] E. D. Sontag. *VC dimension of Neural Networks*, in *Neural Networks and Machine Learning* (C.M. Bishop, ed.), Springer-Verlag, Berlin, pp. 69-95, 1998.

[18] J. Stark, R. Callard and M. Hubank. *From the Top Down: Towards a Predictive Biology of Signalling Networks*, Trends Biotechnol. 21, pp. 290-293, 2003.