Algorithmic Issues in Reverse Engineering of Protein and Gene Networks via the Modular Response Analysis Method

PIOTR BERMAN,[a] BHASKAR DASGUPTA,[b] AND EDUARDO SONTAG[c]

[a]*Department of Computer Science and Engineering, Pennsylvania State University, University Park, Pennsylvania 16802, USA*

[b]*Department of Computer Science, University of Illinois at Chicago, Chicago, Illinois 60607-7053, USA*

[c]*Department of Mathematics, Rutgers University, New Brunswick, New Jersey 08903, USA*

ABSTRACT: This paper studies a computational problem motivated by the modular response analysis method for reverse engineering of protein and gene networks. This set-cover problem is hard to solve exactly for large networks, but efficient approximation algorithms are given and their complexity is analyzed.

KEYWORDS: reverse engineering; biological networks; set multicover; randomized approximation algorithms

INTRODUCTION

The reverse engineering problem is, loosely speaking, that of unraveling the web of interactions among the components of protein and genetic regulatory networks. A major goal is to map out the direct functional interactions among components, a problem that is difficult to approach by means of standard statistical and machine-learning approaches, such as clustering into co-expression patterns. Information on direct functional interactions throws light upon the possible mechanisms and architecture underlying the observed behavior of complex molecular networks.

An intrinsic difficulty in capturing such interactions in intact cells by traditional genetic experiments, RNA interference, hormones, growth factors,

Address for correspondence: Bhaskar DasGupta, Department of Computer Science (MC 152), University of Illinois at Chicago, 851 South Morgan Street, Chicago, IL 60607-7053. Voice: 312-355-1319; fax: 312-413-0024.
dasgupta@cs.uic.edu
berman@cse.psu.edu
sontag@math.rutgers.edu

Ann. N.Y. Acad. Sci. 1115: 132–141 (2007).
doi: 10.1196/annals.1407.001

or pharmacologic interventions, is that any perturbation to a particular gene or signaling component may rapidly propagate throughout the network, thus causing *global* changes which cannot be easily distinguished from direct (*local*) effects. Thus, a central goal is to use the observed global responses (such as steady-state changes in concentrations of activated activities of proteins, mRNA levels, or transcription rates) in order to infer the local interactions between individual nodes.

One potentially very powerful approach to solve the global to local problem is the *modular response analysis* (MRA) method originally introduced by Kholodenko *et al*.[1,2] and further elaborated upon by Andrec *et al.*[3] and Sontag *et al.*[4] The MRA technique (see Crampin *et al.*[5] and Stark *et al.*[6] for reviews) was recently employed by Santos *et al.*[7] in order to discover positive and negative feedback effects in the Raf/Mek/Erk MAPK network in rat adrenal pheochromocytoma (PC-12) cells. The MRA approach uncovered connectivity differences, depending on whether the cells are stimulated with epidermal growth factor or instead with neuronal growth factor. The perturbations used by Santos *et al*.[7] consisted of downregulating protein levels by means of RNAi.

In this paper, we first describe the MRA method, and then we formulate an experimental design problem that arises when using the approach. For large networks, this problem will not scale well, and is computationally hard. This suggests an interesting computational complexity theoretical problem, closely related to set-cover questions. We are able to provide an efficient approximate-solution algorithm. The main ideas are explained intuitively, but details of the proofs, which may be found in Berman *et al.*,[8] are not given here.

The Idea of the Method

Mathematically, the basic description of the method is as follows. We assume that there are n quantities $x_i(t)$ that can be in principle measured, such as the levels of activity of selected proteins, or transcription rates of certain genes. These quantities are thought of as state variables in a dynamical system described by a set of ordinary differential equations and collected into a time-dependent vector $x(t) = (x_1(t), \ldots, x_n(t))$. The dynamical system is described by a system of differential equations:

$$\begin{aligned}
\dot{x}_1 &= f_1(x_1, \ldots, x_n, p_1, \ldots, p_m) \\
\dot{x}_2 &= f_2(x_1, \ldots, x_n, p_1, \ldots, p_m) \\
&\vdots \\
\dot{x}_n &= f_n(x_1, \ldots, x_n, p_1, \ldots, p_m)
\end{aligned}$$

(dot indicates time derivative) or, in more convenient vector form, $\dot{x} = f(x, p)$. The p_i's are parameters, collected into a vector $p = (p_1, \ldots, p_m)$. These parameters can be manipulated, but once changed they remain constant for the duration of the experiment. They represent quantities that can be perturbed,

perhaps indirectly, such as total levels of proteins whose half-lives are long compared to the rate at which the variables evolve. A basic assumption (but see Sontag *et al.*[4] 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, for all pairs of variables, in obtaining information regarding the signs and relative magnitudes of the partial derivative

$$\frac{\partial f_i}{\partial x_j}(\bar{x}, \bar{p}),$$

which quantifies the direct effect of a variable x_j upon another variable x_i. For example, if one determines that $\partial f_i/\partial x_j > 0$, this means that x_j has a positive (catalytic) effect upon the rate of formation of x_i, while a negative sign indicates inhibition.

The critical assumption, indeed the main point in References 1, 2, and 4, is that, while one may not know the algebraic form of the vector field f, often it is known which parameters p_j directly affect which variables x_i. For example, x_i may be the level of activity of a particular protein and p_j might be the total amount (active plus inactive) of that protein in a cell.

In order to use this prior information, we summarize it by a binary matrix

$$C^0 = (c^0_{ij}) \in \{0, 1\}^{n \times m},$$

where "$c^0_{ij} = 0$" means that p_j does not appear in the equation for $\dot{x}_i$, that is, $\partial f_i/\partial p_j \equiv 0$. We assume that an experimental protocol has been designed which allows one to perturb any one of the parameters, let us say the kth one, while leaving the remaining ones constant. (Generalizations to allow for the simultaneous perturbation of more than one parameter will be studied in a future paper.) For the perturbed vector $p \approx \bar{p}$, it is assumed that measurements are available of the perturbed steady-state vector $x = \xi(p)$, which is assumed to be unique as a function of p. (To be mathematically precise: 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.) This readout might be done through Western blots, microarray methods, etc. When the parameter p_j is perturbed, the n "sensitivities"

$$b_{ij} = \frac{\partial \xi_i}{\partial p_j}(\bar{p}) \approx \frac{1}{\bar{p}_j - p_j}(\xi_i(\bar{p} + p_j e_j) - \xi_i(\bar{p})), \quad i = 1, 2, \ldots, n,$$

where $e_j \in \mathbf{R}^m$ is the jth canonical basis vector, can be computed. (As discussed in Kholodenko *et al.*,[1,2] division by $\bar{p}_j - p_j$, which is numerically undesirable, is in fact not necessary.) We arrange these numbers into a matrix $B = (b_{ij})$. Finally, we let $A = \partial f/\partial x$ be the Jacobian matrix with respect to state variables, and let C be the negative of $\partial f/\partial p$, the Jacobian matrix, with respect to the

parameters, Since $f(\xi(p), p)$ is identically zero, we may take derivatives with respect to p, and use the chain rule to obtain that $C = AB$.

The experimental design question that we wish to address is as follows. We would like to obtain as much information as possible about the matrix A. However, each experiment (parameter perturbation) involves an added cost, which we would like to minimize. We think of these experiments as "queries" that return a column B_i of B (if the ith parameter is perturbed). Observe that the matrix C^0 tells us which rows of A have zero inner product with which B_i. We make a *general position assumption* that all subsets of n columns of B are linearly independent; this entails no theoretical loss of generality, since the entries of B correspond to experimental data, although in actual implementations this may lead to numerical instabilities. (See Ref. 3 for an analysis of numerical aspects as well as the effect of errors and noise; the algorithm implemented in Santos *et al.*[7] uses ideas from Andrec *et al.*[3])

We thus are led to the following linear algebra question, which is later recast as a combinatorial question and shown to be dual to the set multicover problem. We describe the problem in terms of two matrices: $A \in \mathbf{R}^{n \times n}$ and $B \in \mathbf{R}^{n \times m}$, such that:

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

There is a limit 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 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 $\mathcal{P}^{n-1}$).

A geometric reformulation is as follows. Let A_i denote the ith 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

$$J = \{j_1, \ldots, j_l\}$$

Then, the information obtained about A may be summarized by the property:

$$A_i \in H^{\perp}_{J,i}$$

where "$\perp$" indicates *orthogonal complement*, and

$$H_{J,i} = \text{span}\{B_j,\ j \in J_i\}$$

$$J_i = \{j | j \in J \quad \text{and} \quad c_{ij}^0 = 0\}.$$

Suppose now that the set of indices of selected queries J has the property:

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

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

In particular, when $k = 1$ one has that

$$\dim\ H_{J,i}^{\perp} \leq 1,$$

and it follows that 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, −, ...)") for each row of A is the main experimental goal, corresponding to knowing if the regulatory interactions affecting each given gene or protein are *inhibitory* or *catalytic*.

Suppose that we do not have the degenerate case $H_{J,i}^{\perp} = \{0\}$ (which would force $A_i = 0$). Then, once that any arbitrary nonzero element v in the line $H_{J,i}^{\perp}$ is 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 will have been uniquely determined. Typically one such sign is indeed known: for example, the diagonal elements a_{ii}, i.e., the ith element of each A_i are negative if they represent a dilution or degradation kinetic rate. The problem is, then:

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

When queries have variable unit costs, meaning that different experiments have different associated costs, this problem would have to 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), with $k > 1$ but still small. It is no longer true that there are only two possible sign patterns for any given A_i. However, 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 $H_{J,i}^{\perp}$ (assume dim $H_{J,i}^{\perp} = k$; 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 $v_r = \text{col}(v_{1r}, \ldots, v_{nr})$, for each $r = 1, \ldots, k$. Since no coordinate of A_i is zero, we know that A_i belongs to the set

$$C = \mathbf{R}^{k-1} \backslash (L_1 \cup \cdots \cup L_n)$$

where, for each $1 \leq s \leq n$, L_s is the hyperplane in $\mathbf{R}^{k-1}$ consisting of all those vectors

$$(\lambda_2, \ldots, \lambda_k) \text{ such that } \sum_{r=2}^{k} \lambda_r v_{sr} = -v_{s1}.$$

On each connected component of 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 Cover,[9] Schläfli,[10] and Sontag[11] 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,

$$\Phi(n, d) = \sum_{i=0}^{d} \binom{n}{i} \leq 2\frac{n^d}{d!} \leq \left(\frac{en}{d}\right)^d.$$

Doubling the estimate to include $\lambda_1 < 0$, we have the upper bound $2\Phi(n, k-1)$. For example, one has $\Phi(n, 0) = 1$, $\Phi(n, 1) = n + 1$, and $\Phi(n, 2) = 1/2(n^2 + n + 2)$. Thus, we have an estimate of two 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 $|J_i| = n - k$ for all $i = 1, \ldots, n$.

Recalling the definition of J_i, we see that $J_i = J \cap T_i$, where $T_i = \{j | c_{ij}^0 = 0\}$. 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:

$$\begin{array}{l} \textit{find } J \subseteq \{1, \ldots, m\} \textit{of minimal cardinality} \\ \textit{such that} |J \cap T_i| = n - k, \quad 1 = i = n. \end{array} \tag{Q2}$$

For example, suppose that $k = 1$, and pick the matrix $C^0 \epsilon \{0,1\}^{n \times n}$ 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 two-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 $n(n - 1)/2$.

THE SET MULTICOVER PROBLEM

The set multicover problem with a "coverage factor" of $k > 0$, which we denote by $\mathbf{SC}_k$, is well-known in the combinatorial algorithms community (e.g., see Vazirani[12]) and is defined as follows. We are given a set of n elements $U = \{1, 2, \ldots, n\}$, usually termed as the universe, and m sets $S_1, S_2, \ldots, S_m \subseteq U$. Our goal is to select a sub-collection of these sets of *minimum cardinality* such that every element of U occurs in at least k of the selected sets. The case $k = 1$, namely $\mathbf{SC}_1$, is simply called the set-cover problem. Usually, the problem is parameterized by a, the maximum number of elements in any set.

A brief summary of some of the known relevant results for these problems is as follows. In general, for arbitrary a, Feige showed that assuming $NP \not\subseteq DTIME(n^{\log \log n})$, $\mathbf{SC}_1$ cannot be approximated to within a factor of $(1 - \varepsilon)\ln n$ for any constant $0 < \varepsilon < 1$ in polynomial time.[13] A slightly weaker lower bound under the more standard complexity-theoretic assumption of P $\neq$ NP was obtained by Raz and Safra,[14] who showed that there is a constant $0 < c < 1$ such that it is NP-hard to approximate **SC** to within a factor of $c \ln n$. The result of Feige was generalized by Trevisan[15] by showing that for all sufficiently large a $\mathbf{SC}_1$ cannot be approximated to within a factor of $(1 - \varepsilon)\ln a$ for any constant $0 < \varepsilon < 1$ in polynomial time unless P = NP. On the positive side, the $\mathbf{SC}_k$ problem can be $(1 + \ln a)$-approximated in $O(nmk)$ time by a simple greedy heuristic that, at every step, selects a new set that covers the maximum number of those elements that has not been covered at least k

times yet. It is also possible to design randomized approximation algorithms with similar expected approximation ratios.[12]

Combinatorial Formulation of Questions (Q1) and (Q2)

A combinatorial formulation of Questions (**Q1**) and (**Q2**) can be obtained via a generalization of the the so-called *hitting set problem* (e.g., see Garey and Johnson[16] [p. 222]). We denote this problem by $\mathbf{CP}_k$. We are given a set of m elements $U = \{1, 2, \ldots, m\}$, usually termed as the universe, and n sets $T_1, T_2, \ldots, T_n \subseteq U$ and a "coverage factor" $k > 0$. Our goal is to select a subset of elements of U of *minimum cardinality* such that every set contains at least k of the selected elements. The hitting set problem is precisely the case $k = 1$.

Equivalence of CP_k and SC_k

One can easily establish a one-to-one correspondence between an instance of $\mathbf{CP}_k$ and an instance of $\mathbf{SC}_{n-k}$ by taking an instance of $\mathbf{CP}_k$ and creating an instance of $\mathbf{SC}_{n-k}$ in which we have an element for every set of $\mathbf{CP}_k$ and a set for every element x of $\mathbf{CP}_k$ that contains those sets of $\mathbf{CP}_k$ in which the element x was contained. It is easy to verify that U' is a solution to the instance of $\mathbf{CP}_k$ if and only if the collection of sets S_u for each $u \in U'$ is a solution to the instance of $\mathbf{SC}_k$.

SUMMARY OF OUR RESULTS

Our algorithmic contributions can be summarized as follows; see Berman *et al.*[8] for more details. A polynomial time algorithm for a minimization problem is said to have a performance or approximation ratio of $\varepsilon > 1$ if it provides a solution with an objective value no larger than ε times the value of the optimum.

We first observe that the standard greedy algorithm $\mathbf{SC}_k$, namely a procedure that selects a set which contains the maximum number of elements that has not been covered k times yet, produces an approximation ratio of $\Omega(\log n)$ even if k is "large," i.e., $k = n - c$ for some constant $c > 0$. This is obtained by giving an explicit example in which the greedy performs in such a manner. For $k = 1$, such a result was already known.[17] This indicates that such a greedy procedure cannot have an improved approximation ratio for larger values of k.

Recall that $a > 1$ denotes the maximum number of elements in any given set in our set multicover problem. We show that a non-trivial analysis of a simple randomized polynomial-time algorithm for this problem yields an expected approximation ratio $E[r(a, k)]$ that is an increasing function of a/k.

TABLE 1. Precise mathematical upper bounds on $E[r(a, k)]$

Upper bound on $E[r(a, k)]$	Parameter range
$1 + \ln a$	$k = 1$, a arbitrary
$(1 + e^{-(k-1)/5}) \ln\ (a/(k-1))$	$a/(k-1) \geq e^2 \approx 7.39$, $k > 1$
$\min\{2 + 2 \cdot e^{-(k-1)/5}, 2 + (e^{-2} + e^{-9/2}) \cdot (a/k)\}$ $\approx \min\{2 + 2 \cdot e^{-(k-1)/5}, 2 + 0.46 \cdot (a/k)\}$	$1/4 < a/(k-1) < e^2$, $k > 1$
$1 + 2\sqrt{(a/k)}$	$a/(k-1) \leq 1/4$, $k > 1$

The behavior of $E[r(a, k)]$ is "roughly" as follows: it is about $\ln(a/k)$ when a/k is at least about $e^2 \approx 7.39$, and for smaller values of a/k it decreases toward 1 as a linear function of $\sqrt{(a/k)}$ with $\lim_{a/k \to 0} E[r(a, k)] = 1$. More precise bounds for our results are shown in TABLE 1.

Can E[r(a, k)] Converge Toward 1 at a Faster Rate?

Is it possible to design randomized or deterministic approximation algorithms for which $E[r(a, k)]$ or $r(a, k)$ converges to 1 at a *significantly* faster rate as a function of a/k? Assuming P $\neq$ NP, this may be difficult to achieve and, in particular, $E[r(a, k)]$ or $r(a, k)$ cannot be $1 + o(1)$ for $a \geq k$ since the set multicover problem is MAX-SNP-hard for this case. To illustrate the last assertion, consider the special case of $k = a = n - 1$. Then, the set multicover problem is still MAX-SNP-hard as shown in the following. One could have $n - 1$ sets of the form $V \setminus \{i\}$ that cover every element, except one, exactly $n - 2$ times (the last element is covered $n - 1$ times). Moreover, we can have a family of sets of size exactly 3 that form an instance of the set-cover problem restricted to $a = 3$. This restricted problem is MAX-SNP-hard, and a solution of size m for that instance gives solution of size $n + m - 1$ for our instance. Because $m \geq n/3$, this is an approximation-preserving reduction.

ACKNOWLEDGMENTS

We would like to thank Uriel Feige for his constructive criticisms of an earlier draft of the manuscript that led to the improved bounds.

This work was supported in part by NSF Grant CCR-O208821 (to P.B.), by NSF Grants DBI-0543365, IIS-0612044 and a CAREER Award IIS-0346973 (to B.D.), and by NSF Grants DMS-0504557 and DMS-0614371 (to E.S.)

REFERENCES

1. KHOLODENKO, B.N., A. KIYATKIN, F. BRUGGEMAN, *et al.* 2002. Untangling the wires: a novel strategy to trace functional interactions in signaling and gene networks. Proc. Natl. Acad. Sci. USA **99:** 12841–12846.

2. KHOLODENKO, B.N. & E.D. SONTAG. 2002. Determination of functional network structure from local parameter dependence data. arXiv physics/0205003, May 2002.
3. ANDREC, M., B.N. KHOLODENKO, R.M. LEVY & E.D. SONTAG. 2005. Inference of signaling and gene regulatory networks by steady-state perturbation experiments: structure and accuracy. J. Theoret. Biol. **232:** 427–441.
4. SONTAG, E.D., A. KIYATKIN & B.N. KHOLODENKO. 2004. Inferring dynamic architecture of cellular networks using time series of gene expression, protein and metabolite data. Bioinformatics **20:** 1877–1886.
5. CRAMPIN, E.J., S. SCHNELL & P.E. MCSHARRY. 2004. Mathematical and computational techniques to deduce complex biochemical reaction mechanisms. Prog.Biophys. Mol. Biol. **86:** 77–112.
6. STARK, J., R. CALLARD & M. HUBANK. 2003. From the top down: towards a predictive biology of signaling networks. Trends Biotechnol. **21:** 290–293.
7. SANTOS, S.D.M., P.J. VERVEER & P.I.H. BASTIAENS. 2007. Growth factor-induced MAPK network topology shapes Erk response determining PC-12 cell fate. Nat. Cell Biol. **9:** 324–330.
8. BERMAN, P., B. DASGUPTA & E.D. SONTAG. 2007. Randomized approximation algorithms for set multicover problems with applications to reverse engineering of protein and gene networks. Discrete Appl. Math [special series on computational molecular biology] **155:** 733–749.
9. COVER, T. 1965. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. IEEE Trans. Electron. Comput. **EC-14:** 326–334. (Reprinted [1992] *in* Artificial Neural Networks: Concepts and Theory. P. Mehra and B. Wah, Eds. IEEE Computer Society Press. Los Alamitos, CA.)
10. SCHLÄFLI, L. 1950. Theorie der vielfachen Kontinuitat (1852). *In* Gesammelte Mathematische Abhandlungen, Vol. 1: 177–392. Birkhäuser. Basel.
11. SONTAG, E.D. 1998. VC dimension of neural networks. *In* Neural Networks and Machine Learning. C.M. Bishop, Ed.: 69–95. Springer-Verlag. Berlin.
12. VAZIRANI, V. 2001. Approximation Algorithms. Springer-Verlag. Berlin.
13. FEIGE, U. 1998. A threshold for approximating set cover. JACM **45:** 634–652.
14. RAZ, R. & S. SAFRA.1997. A sub-constant error-probability low-degree test and sub-constant error-probability PCP characterization of NP. Proceedings of the 29th Annual ACM Symposium on Theory of Computing, El Paso, Texas, ACM Press, NY, pp. 475–484.
15. TREVISAN, L. 2001. Non-approximability results for optimization problems on bounded degree instances. Proceedings of of the 33rd ACM Symposium on Theory of Computing, Hersonissos, Greece, ACM Press, NY, pp. 453–461.
16. GAREY, M.R. & D.S. JOHNSON. 1979. Computers and Intractability: a Guide to the Theory of NP-Completeness. W. H. Freeman & Co. New York.
17. JOHNSON, D.S. 1974. Approximation algorithms for combinatorial problems. J. Comput. Systems Sci. **9:** 256–278.