JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 9, Number 2, 2002

Pp. 317–330

Analysis Techniques for Microarray Time-Series Data

VLADIMIR FILKOV,[1] STEVEN SKIENA,[1] and JIZU ZHI[2]

ABSTRACT

We address possible limitations of publicly available data sets of yeast gene expression. We study the predictability of known regulators via time-series analysis, and show that less than 20% of known regulatory pairs exhibit strong correlations in the Cho/Spellman data sets. By analyzing known regulatory relationships, we designed an edge detection function which identified candidate regulations with greater fidelity than standard correlation methods. We develop general methods for integrated analysis of coarse time-series data sets. These include 1) methods for automated period detection in a predominately cycling data set and 2) phase detection between phase-shifted cyclic data sets. We show how to properly correct for the problem of comparing correlation coefficients between pairs of sequences of different lengths and small alphabets. Finally, we note that the correlation coefficient of sequences over alphabets of size two can exhibit very counterintuitive behavior when compared with the Hamming distance.

Key words: yeast microarray data, gene regulation, time-series data, correlation coefficient.

1. INTRODUCTION

New experimental technologies in molecular biology (particularly oligonucleotide and cDNA arrays) now make it possible to quickly obtain vast amounts of time-series data on gene expression in a particular organism under various conditions. Extensive time-series data on gene expression in yeast (*Saccharomyces cerevisiae*) have been obtained by Cho (Cho *et al.*, 1998) and Spellman (Spellman *et al.*, 1998) using microarrays, greatly expanding our knowledge of which genes are involved in cell cycle regulation.

The importance of the Cho and Spellman data sets is perhaps best revealed by the variety of methodologies being applied to analyze it. Clustering studies and promoter analysis (Eisen *et al.*, 1998; Spellman *et al.*, 1998) have been used to classify genes according to where they are active in the cell cycle. We (Chen *et al.*, 1999a) have developed a system for proposing putative gene regulatory networks by identifying activators and inhibitors using signal processing and combinatorial optimization. Friedman *et al.* (2000) have built a system for analyzing the same data, similar in spirit, but based instead on Bayesian networks. Techniques for analyzing these data sets using differential equation modeling (Chen *et al.*, 1999b), wavelets

[1]Department of Computer Science, State University of New York, Stony Brook, NY 11794.
[2]Center for Biotechnology, State University of New York, Stony Brook, NY 11794.

(Klevecz and Dowse, 2000), and singular value decomposition (SVD) (Alter *et al.*, 2000; Holter *et al.*, 2000) have also been explored.

In general, this analysis has succeeded in revealing certain gross periodicities in the data corresponding to the cell and other cycles and has generated untested predictions of putative regulators. Microarray technology is limited in sensitivity when comparing the relative concentrations of different genes in a single experiment. Further, the costs associated with current microarray technologies, as well as the nature of these experiments, make it difficult for the experiments to be repeated. This means that the available microarray data, as of this writing, cannot be analyzed for accuracy or precision, since there can be no statistical study of a single measurement. Therefore, the available gene expression data is intrinsically noisy and difficult to analyze.

In this paper, we address several fundamental questions concerning possible limitations to which informed predictions can be generated using these data sets:

- **Predictability of known regulations via time-series analysis.** Previous systems for inferring regulatory networks from these microarray data sets (Chen *et al.*, 1999a; Friedman *et al.*, 2000) have bravely assumed that the Cho/Spellman data sets contain sufficient information to identify a substantial fraction of all regulators. To test this hypothesis, we analyzed the literature and compiled a database of known regulatory relations in yeast. Less than 20% of these regulatory pairs exhibited strong correlations in the Cho/Spellman data set, clearly insufficient to infer large regulatory pathways by pairwise similarity analysis.
- **Improved edge detection for regulatory relations.** By analyzing the known regulatory relations which were actively expressed in the Cho/Spellman data, we were able to design and analyze an edge detection function which identifies these relations with greater fidelity than standard correlation methods. We believe that our edge detector is now significantly better at identifying interesting regulatory candidates than was previous work (Chen *et al.*, 1999a).
- **Periodicity and phase shift analysis of time-series data sets.** The Cho/Spellman data sets comprise four distinct time-series data sets, each measuring gene expression for all 6,178 ORFs in yeast. Each data set uses cell cultures synchronized by a particular method (cdc15, cdc28, alpha, and elutriation), which starts each in distinct phases of the cell cycle, with varying cell cycle lengths and sampling frequency. Integrating these distinct time-series into a common reference frame requires methods for inferring periodicity and phase shifts on noisy data. Our techniques yield period length and shift values consistent with those observed by the experimenters and are of independent interest.
- **Comparing correlations of distinct length sequences.** Our techniques for periodicity and phase shift analysis require comparing the significance of correlations of short sequences of different lengths. Properly interpreting these correlations is a subtle problem. For example, two random sequences of length five will have a correlation of 0.8 or higher roughly 60% of the time, a likelihood which reduces sharply with increased sequence length. We show how to correct for such anomalies and apply it successfully to identify periodicity/phase shift in the Cho/Spellman data.
- **Correlation significance of small alphabet sequences.** The limited resolution inherent in measuring noisy experimental data can be modeled by grouping data values into a small number of bins for analysis. This yields data streams over a small alphabet and alphabets of size two in the limiting case. We explore the significance of the standard correlation coefficient measure on sequences over alphabets of length two and show that the results can be inherently misleading. For example, we show that two sequences of any length can have a correlation coefficient near 0 (i.e., essentially uncorrelated) even though they are identical in all but two positions!

This paper is organized as follows. In Section 3, we demonstrate that only a small fraction of all known regulatory relations are exhibited in these data sets and use these results to guide the design of an effective edge detector for identifying interesting pairs of expressed genes. In Section 2, we describe the particulars of the Cho/Spellman data sets, which we analyze throughout this paper. In Section 4, we explore techniques to extract cycle period length and phase shifts by appropriately weighting correlation coefficients by the length of the associated sequences. Finally, in Section 5, we compare the correlation coefficient with the Hamming distance metric as a similarity measure for sequences of size-two alphabets, with surprising results.

Table 1. Cho/Spellman Data Sets and Their Properties[a]

Data set	*Period obs.*	*Period det.*	δt	*# samples*	*# full orfs*
alpha	66 ± 11 min.	70 ± 7 min.	7	18	3361
cdc28	90 ± 10 min.	100 ± 10 min.	10	17	1188
cdc15	70 ± 10 min.	90 ± 10 min.	10/20	24	3453
elu	—	—	30	14	4753

[a]The data sets are named by the method used to synchronize the yeast cells. *Period obs.* indicates the cell cycle period observed by the original researchers, *Period det.* is the period we detect in this study, δt is the time interval between measurements, *# samples* is the number of measurements, and *# full orfs* is the number of time-series that have no missing values in them.

2. MICROARRAY DATA SETS

Our experiments were conducted on the data of Cho *et al.* (1998) and Spellman *et al.* (1998), which are available at *http://genome-www.stanford.edu/cellcycle*. They are comprised of four time-series data sets, each data containing temporal concentration measurements for all 6,178 ORFs in yeast. Each of the experiments starts with a cell population which has been synchronized with distinct methods (cdc15, cdc28, alpha factor, and elutriation elu), which arrest the cells in the same state by introducing external substances, changing environmental conditions, or selecting cells that are of the same size and hence are likely in the same state. The time-series courses have been repeated through one+ periods for Elu, two+ periods for alpha and cdc28, and three+ periods for cdc15.

Table 1 gives the cycle-times for each data set, as reported by the original researchers (Spellman *et al.*, 1998), the cycle-times for each data set that we detected (as reported in this writing), the time interval between sampling points, the number of sampling points in each experiment, and the number of curves (out of 6,178) that had no missing points in them.

Out of the four data sets, we found the cdc28 (Cho) and alpha data sets to be most yielding to analysis because the cdc15 data set had a lot of missing points and the sampling intervals were not constant (10 or 20 minutes), while the elu data set had been sampled for one period, and only coarsely at that.

3. ASSESSING AND IMPROVING REGULATORY PAIR DETECTION

At least three software systems (Chen *et al.*, 1999a, 1999b; Friedman *et al.*, 2000) have been developed to predict gene regulations using the Cho/Spellman time-series data. However, neither the data nor these systems were rigorously evaluated as to the quality of the resulting predictions.

In this section, we demonstrate that the Cho/Spellman data is inherently inadequate to identify the vast majority of known regulatory relations, when subjected to simple signal analysis techniques. We use insights from this and a previous microarray data analysis (Chen *et al.*, 1999a) to construct an edge detector which appears better at selecting biologically interesting pairs of genes.

3.1. Evaluating known regulatory pairs

To analyze the potential for determining regulatory pairs from the Cho/Spellman data sets, we began by constructing a database of known regulatory relations in yeast. Specifically, a keyword search (regulat*) on the Yeast Protein Database (YPD) (*www.proteome.com*) performed in February 2000 yielded 1,007 regulated genes in yeast. By reviewing the published literature on these 1,007 genes, we collected 888 transcriptional regulations of which 647 were activations and 241 were inhibitions. Altogether, 486 genes were involved in these transcriptional regulations.

We then mapped these 486 genes to the two highest-quality time-series data sets, cdc28 and alpha. Genes mapped successfully if they had the same name in the Cho/Spellman data sets and our database and in addition had no missing points in the time-series. The results are described in Table 2.

Only 366 genes were successfully mapped to the cdc28 data set, with the balance of 120 genes failing primarily because of differing naming conventions between the literature and the data set. Of the original

TABLE 2. MAPPING KNOWN REGULATION PAIRS (FROM YPD) ONTO CHO/SPELLMAN DATA SETS[a]

YPD mapped onto data set	*# genes mapped (out of 486)*	*# activations (out of 647)*	*# inhibitions (out of 241)*
alpha	335	343	96
cdc28	366	469	155

[a]Four hundred eighty-six genes involved in 888 transcriptional regulations (647 activations and 241 inhibitions) were mapped onto alpha and cdc28 data sets. The heading *# genes mapped* indicates the number of successfully mapped genes (out of 486), *# activations* indicates the number of successfully mapped activation regulations, and *# inhibition* indicates the number of successfully mapped inhibition regulations (out of 647 and 241 respectively).

888 regulations, 469 activations and 155 inhibitions mapped correctly to the cdc28 data set. Onto the alpha data set, we mapped successfully only the 335 genes for which all 18 time-series points were available. These genes were involved in 343 known activations and 96 known inhibitions.

We assessed the predictability of regulation pairs of the Cho/Spellman data sets, by signal similarity analysis, with the following three studies.

1. We performed correlation coefficient analysis on each pair of activations in the 469 and 343 known activating relations that were mapped on the data sets. The result, shown in Fig. 1, demonstrates a poor correlation between the activators and the corresponding activated genes. Less than 20% of known regulations (93/469 for the cdc28 data set, 44/343 for the alpha data set) scored > 0.5 correlation between an activator gene and activated gene.

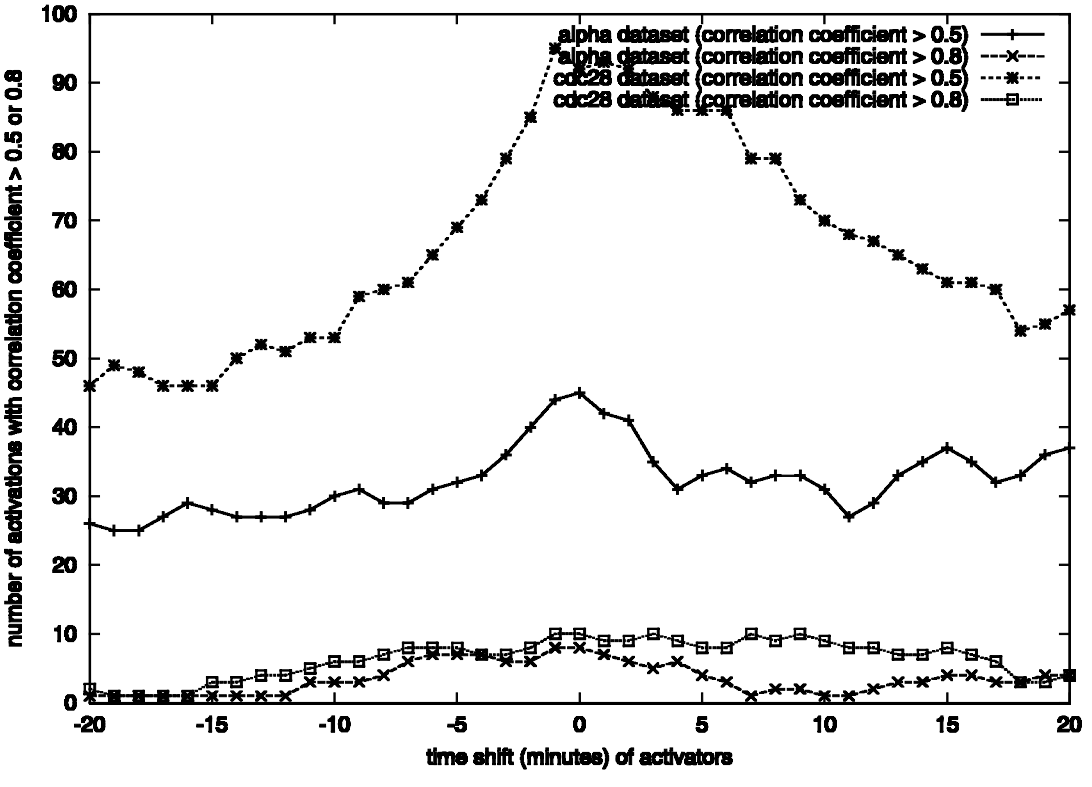

FIG. 1. Correlation coefficient studies of known activation relationships. Correlating between curves of known activations yields poor correlation for all 4 data sets—only 20% correlate over 0.5 (shift = 0). Shifting the time-series of known activation pairs along each other (in the same data set) did not improve the correlation between them in general.

2. To compensate for a possible time lag of the biological process of activation, we shifted the activation pair curves with respect to each other, but that did not improve the number of correlations for any shift between ±20 minutes, as shown in Fig. 1. These curve shifts were performed on resampled interpolated data to facilitate short shifts.
3. Finally, eyeball examination by a PhD biologist (Zhi) of all plotted pairs of known regulations failed to extract a common pattern which could indicate a possible transcription regulation in more than 20% of these cases. The biological considerations were mainly signal similarity, knowledge of timing of biological processes in yeast, and specific properties of the experimental procedures.

What these studies demonstrate is that simple signal similarity analyses cannot with certainty extract more than 20% of regulation relationships from the Cho/Spellman data sets. This negative result is probably due to a combination of the following: 1) many of the genes are regulated post-transcriptionally, which is true for eukaryotes in general (this limits the use of microarrays in network prediction to transcriptional networks), 2) the data sets describe the genes' behavior under a very limited set of environmental conditions, and so many regulators have not been expressed at all, 3) genes in eukaryotes have more than one activator, likely more than two (Yuh *et al.*, 1998), 4) the Cho/Spellman data sets are not precise enough to capture biological phenomena essential for regulation (highly unlikely in general).

We can say the following though, that together these results confirm the complexity of biological systems and demonstrate that gene regulation is much more complicated than this model and data set can deal with.

3.2. An improved edge detection function

Although two genes within a regulated pair do not necessarily have similar expression patterns, expressionally correlated genes usually suggest a functional correlation. Thus, correlation coefficient-based clustering has been widely used in analyzing time course expression data (Eisen *et al.*, 1998). The correlation coefficient method fails to pick up strong local signals, as opposed to global similarity, particularly in noisy data. Here we propose an edge function, which was aimed to more accurately predict functional correlated genes.

We set up two goals for our edge detector: a) to favor similarity of local signals on the curves, i.e., identify meaningful peaks in expression, and b) to act as a conservative and biologically significant filter on the signals, so as to exclude meaningless events. As a result of these requirements, we expected very similar curves to score high scores, but also visually dissimilar curves to perhaps score high enough to be considered as in a regulation.

To facilitate further discussion we define an *edge* as a piece-wise linear, monotonic function. Any piecewise linear function (e.g., a time-series curve) is an ordered list of edges of alternating direction (i.e., nonincreasing or nondecreasing). For our purposes, an edge can be considered to be a vector of the y coordinates of monotonic linear functions, since the x coordinates are the sampling times—which are same for all curves. Of importance to us will be, in particular, the edge's minimum, maximum, and middle $((minumum + maximum)/2)$ points.

We want our edge detector to filter very conservatively, so we deem a point of the curve insignificant if the difference in expression to both of its adjacent points is less than 10%. In addition, we find an edge to be of biological significance if its relative expression is more than 30%. Finally, we would like to score all such biologically significant edges (local signals) across time-series curves (genes).

All these considerations went into the following, locally based, edge detector algorithm:

1. For each gene, label the points as local minima, maxima, and in-between. While comparing the expression value of neighboring points, allow a given expression error level, typically 10%, between a point and both of its adjacent points and erase the points that do not pass this threshold.
2. For each gene, the labeled minima and maxima are connected using a four-step edge construction process:
   - **Primary edges.** Connect neighboring local maxima and minima. In this way, each curve is decomposed into monotone segments (edges).
   - **Secondary edges.** Erase all primary edges whose normalized expression, $(max - min)/average$, is smaller than a threshold, typically 30%. Here, max, min, and $average$ pertain to the expression level. This accounts for the minimal biologically significant expression level change. Any changes below this level are probably due to experimental error.

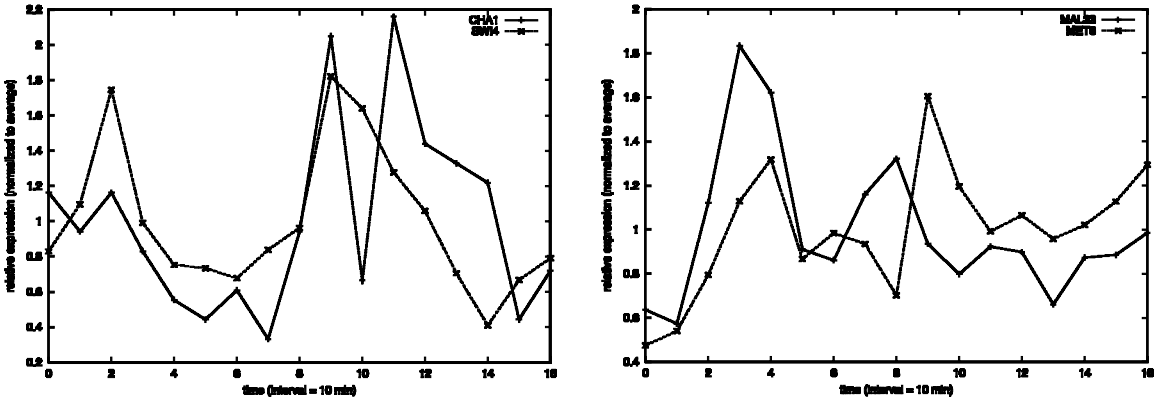

FIG. 2. Interesting regulatory pairs detected by our edge function but not by correlation.

- **Tertiary edges.** Merge (if possible) consecutive secondary edges of the same direction. Possibly, at the beginning of this step, we could have a number of consecutive edges that really are just continuations of the same edge. So we connect them in this step. Note that although two consecutive edges can have the same direction, it may not be possible to merge them because of monotonicity. Those edges will stay separate.
- **Quadrary edges.** Eliminate narrow peaks or troughs, i.e., stray points, likely resulting from error in measurement, that vary significantly in relative expression from their adjacent points, typically a factor of 2. For example, in Fig. 2(a), the narrow trough of CHA1 from time point 9 to 11 is probably due to an error at time point 10. Thus, the edges from time points 9 to 10 and from 10 to 11 are eliminated.

Finally, each gene is represented by an array of quadrary edges which we consider as biologically significant and reliable expression level changes.

3. Pairs of genes are *scored* solely based on their quadrary edges. To score genes G_a and G_b, we score each quadrary edge (e_a) of G_a against each quadrary edge (e_b) of G_b, provided the time difference between the two edges is $\leq \delta_{max}$. The similarity score S_g between G_a and G_b is given as

$$S_g = \sum_{all\ e} d\left(1 - \frac{\delta}{\delta_{max}}\right) \Big/ \sqrt{n_a n_b}$$

where d denotes the agreement of the slopes of e_a and e_b. Specifically, $d = 1$ if the signs of the slopes agree; otherwise, $d = -1$. The parameter δ_{max} defines the maximum allowable time difference (in minutes) between the middle of e_a and e_b. A typical setting is $\delta_{max} = 15$ minutes for yeast expression data sets. The parameter δ denotes the observed time difference (minutes) between the middle of e_a and e_b, and $0 \leq \delta \leq \delta_{max}$. Interactions between edge pairs with time difference $> \delta_{max}$ are considered biologically meaningless and are simply ignored. Larger values of δ imply less similarity between the edges. The counts n_a and n_b denote the total number of quadrary edges in G_a and G_b, respectively.

It can be proven that, like the correlation coefficient, the range of S_g is between -1 and 1, by noticing that the slopes of the quadrary edges in any G_i alternate in sign.

3.3. Results

Our results showed that this algorithm identifies certain interesting putative pairs missed by correlation coefficient methods, because we focus on local rather than global features.

We used the following methodology to evaluate the performance of our edge detector. A PhD level biologist (Zhi) manually compared all known regulatory relations in the alpha and cdc28 data sets, classifying

them into sets: a) those which clearly revealed a possible regulatory relation and b) the rest (which clearly *did not* reveal a regulatory relations or were inherently ambiguous). Only 27 of the 343 activations in the alpha and 63 of the 469 activations in the cdc28 data sets were classified in the positive group *a*, with the remainder classified in the negative group *b*. We looked only at activations since we had more of them to work with; there were less than 10, and 20, respectively, inhibitions classified in the positive group.

The result of scoring all gene pairs from the 335 alpha genes and 366 cdc28 genes are in Table 3. A perfect classifier/threshold pair would score only positive pairs above the threshold; i.e., it would simulate our biologist, thus underwriting our belief that biological knowledge can be captured by looking at local signals.

Table 3 demonstrates that our proposed edge scoring function misclassified no pair above the 0.5 level.

The established efficacy of our edge detection function makes it interesting to study other high-scoring pairs of genes, namely the 192 alpha pairs which correlated > 0.8 and the 223 alpha pairs which scored > 0.5 using our edge function. For the cdc28 data set, 628 pairs correlated > 0.8 and 398 pairs scored > 0.5 using our edge function. There is some agreement between the two scoring measures for 88 alpha pairs and 127 cdc28 pairs occurring in both sets. Eyeball examination showed that all these common pairs carried a characteristic strong and smooth oscillation pattern. However, we observed that certain interesting pairs of genes using the edge function correlated poorly. For example, the pairs in Fig. 2: MAL33/MET6 correlated 0.32 and CHA1/SWI4 correlated 0.47. For MAL33/MET6, shifting the curve of MAL33 5 min, 10 min and 15 min to the right still brings the correlation coefficient score to only 0.41, 0.42, 0.27, respectively. In the case of CHA1/SWI4, the low expression of CHA1 at time point 10 is the main reason why the correlation coefficient scored low. Our edge function, which decided this time point was probably experimental error and ignored it, scored it well.

In conclusion, we believe our edge function provides an interesting method to analyze time-course gene-expression data. More results of our experiments are available at *www.cs.sunysb.edu/~skiena/gene*. Figure 3 shows that the collection of highest scoring regulatory pairs in alpha are well distributed among many genes. While we do not propose that such a diagram yields any semblance of the complete regulatory network, we do believe these pairs warrant further study. As more accurate experimental data emerges, perhaps with error bars, such edge detection functions should become even more reliable and predictive.

Table 3. Comparing Classification Performance of the Correlation Coefficient and Our Edge Function[a]

Alpha data							
Correlation coefficient				*Edge function*			
Thresh	*Total*	*Good*	*Bad*	*Thresh*	*Total*	*Good*	*Bad*
> 0.85	107	5	0	> 0.6	96	5	0
> 0.8	192	5	2	> 0.5	223	5	0
> 0.7	703	5	7	> 0.4	557	7	6
> 0.6	1852	9	13	> 0.3	1581	11	15
Cdc28 data							
Correlation coefficient				*Edge function*			
Thresh	*Total*	*Good*	*Bad*	*Thresh*	*Total*	*Good*	*Bad*
> 0.85	289	2	2	> 0.6	146	1	0
> 0.8	628	5	4	> 0.5	398	3	0
> 0.7	1826	22	15	> 0.4	1236	11	3
> 0.6	3903	31	19	> 0.3	3401	19	20

[a]For each dataset, the two measures classify the positive pairs present among all pairs of genes passing a given threshold. A positive pair is classified either correctly (good), or incorrectly (bad). The heading *total* indicates the total number of gene pairs (out of 335*334/2 for alpha and 366*365/2 for cdc28) passing a given threshold.

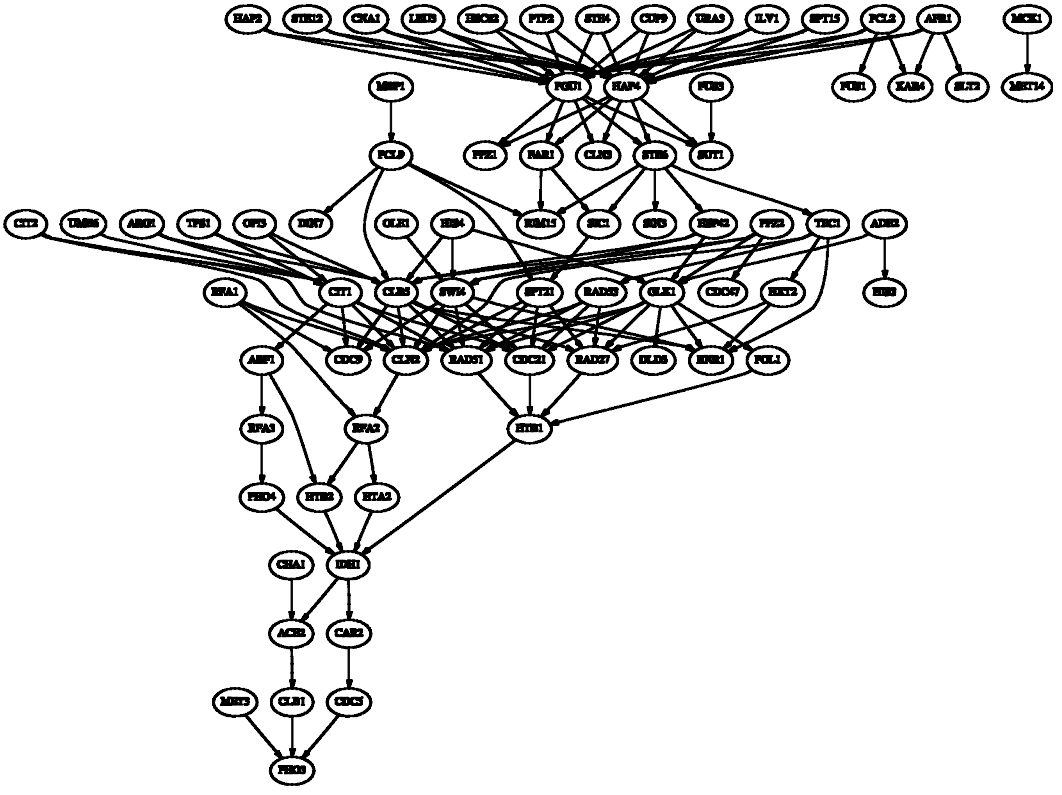

FIG. 3. Network of highest scoring activations in alpha data set.

4. INTEGRATED ANALYSIS OF DATA SETS

An integrated analysis of the Cho/Spellman data sets could yield more information than would analyzing each data set separately, especially since the experiments entailed different environmental conditions. Two evident applications could benefit from such an analysis: 1) interleaving the signals from all data sets for each gene, to obtain an integrated data set of higher precision and accuracy than any of the original ones, through smaller gaps between sampling times; and 2) comparing the integrated time-series to reveal genes whose expression grossly differs in one data set from the rest. This would be indicative of either experimental error or genes whose expression is effected by one of the synchronization factors.

Thus, we sought to integrate the data, so as to render the curves from different data sets comparable to each other. For a similarly motivated study of the same data sets, independent of ours and done in roughly the same period of time, see Aach and Church (2001).

Our approach is to simply interleave the 4 time-series curves for each gene over the same coordinates. Such interleaving of the data sets requires identifying the cell cycle periods of each data set as well as detecting the cell cycle phase each experiment started in after synchronization. Since the experimenters (Spellman *et al.*, 1998) grossly observed the cell cycle length of each time course and the cycle phases in which the cells were arrested, we have a way to assess the performance of our algorithms.

4.1. Cycle detection in microarray data

Fourier analysis is the standard method of identifying cycles in periodic data sets when the number of sampling points is large, but this is not the case with the gene expression data sets. It has been shown (Klevecz and Dowse, 2000) that Fourier analysis does not pick up some important characteristics of microarray data sets, possibly because of their coarseness. Instead, we explored how well discrete similarity methods work, which we used for analyzing short sequences (Chen *et al.*, 1999a). We have previously explored geometric approaches to cycle detection (Filkov, 2000).

To detect the cell cycle of a data set, we first analyze the similarity of a time-series curve with itself. Since each yeast gene's expression follows the general cell cycle if it exhibits any cycling behavior, we expect that this method will detect the periods of the cycling genes while extracting random periods for the noncycling genes. Thus, we expect the period of the data set to become apparent when we average over all genes.

4.1.1. Methods and results. We chose the Pearson correlation coefficient ("correlation coefficient," or simply "correlation" in what follows) to be our similarity measure of two time-series curves with equal sampling times, x_i, y_i of length n,

$$r(X, Y) = \frac{\sum x_i y_i - \sum x_i \sum y_i / n}{\sqrt{(\sum x_i{}^2 - (\sum x_i)^2/n)(\sum y_i{}^2 - (\sum y_i)^2/n)}}. \tag{1}$$

This correlation coefficient is known to give good results when employed in clustering and analyzing time-series (Eisen *et al.*, 1998; Spellman *et al.*, 1998). Curves that are visually very similar typically score as being highly correlated.

Consider a time-series curve (vector) $C = (c_1, c_2, \ldots, c_k)$. If C is periodic with period T, the similarity of a prefix "window" $P_t(C) = (c_1, c_2, \ldots, c_t)$ with the same-length suffix "window" immediately following it $S_t(C) = (c_t, c_{t+1}, \ldots, c_{2t-1})$ should be greatest when their length equals (up to an integer) the period of the whole curve, i.e., $t = T$. Thus, by varying the length t of the window, we can find the optimal period for each gene. For each data set, we correlated $P_t(C)$ with $S_t(C)$ over all curves, for all t, where $5 \leq t \leq 12$. The constant 5 denotes the length of the shortest sequence for which we deemed the correlation coefficient as having predictive power. Length 12 was one half of the longest sequence we had. When

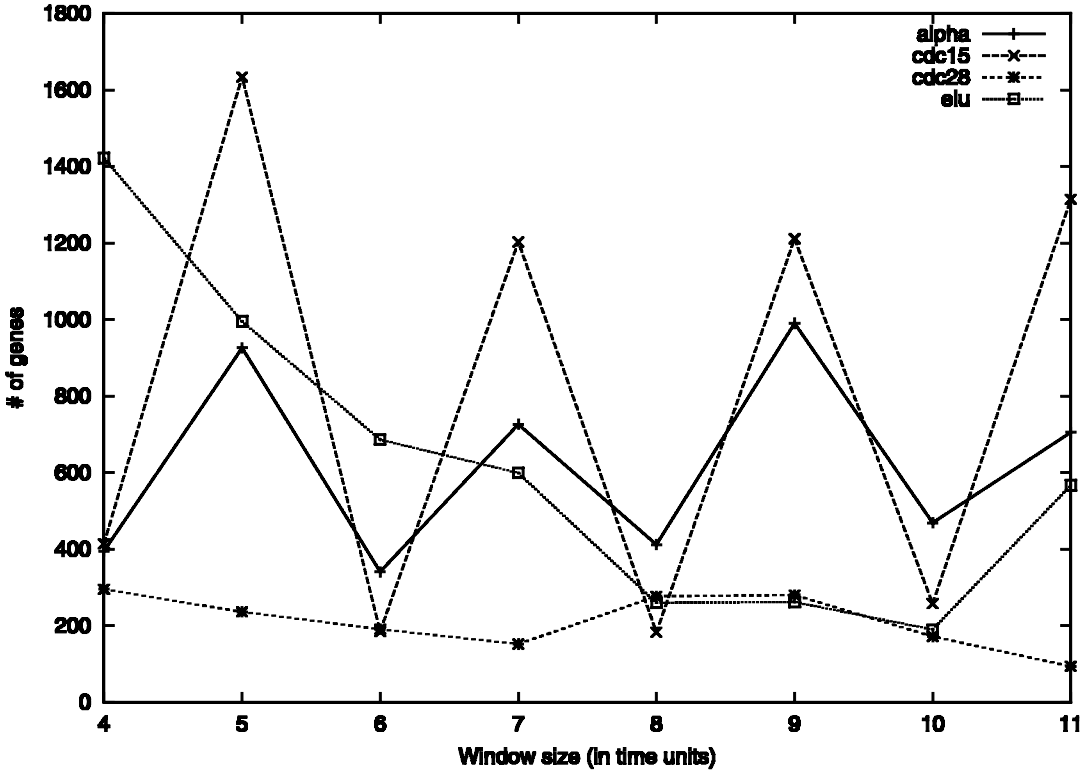

FIG. 4. Inferred periods for all data sets using standard correlation coefficients. The window size (x axis) indicates the number of sampling intervals for each data set, and the # of genes (y axis) indicates the number of genes for which, when correlating a prefix and a suffix of equal lengths (of the gene's time-series), the length "window size" correlated the highest.

$2t - 1 > k$, we truncated the suffix "window" on the left, i.e., $S_t(C) = (c_{k+1-t}, c_{k-t}, \ldots, c_k)$. For each gene C_i, the inferred period was the t that maximized $r(P_t(C_i), S_t(C_i))$. Then, for each window length $t = 5, \ldots, 12$, we counted the number of genes, with inferred period equal to t. The results of repeating the above procedure over all data sets are shown in Fig. 4, where window lengths are converted from number of sampling points to time intervals for easier conversion to absolute time.

As is apparent from Fig. 4, the uncorrected correlations are not sufficient to detect the periods, for even though the actual cell cycles are in fact strongly expressed in all the graphs, they are not necessarily the most strongly expressed. Moreover, we notice that there is an overall tendency towards smaller periods, i.e., all graphs being somewhat skewed towards left.

In addition, there are very strong indications of underlying oscillatory phenomena with periods smaller than the observed cell cycles. One is on the order of 3 sampling points or approximately 20 minutes for alpha, and another of a period around 40 minutes. This last phenomenon was also reported by Klevecz and Dowse (2000). These frequencies are apparently undetectable by Fourier analysis, but can be found with methods suitable for analysis of short, coarsely sampled sequences like ours or by wavelet analysis.

We note that the elu graph did not show any oscillations, due probably to the coarseness of the sampling, the fact that the experiment was performed for less than one cell cycle, and possibly the inability of the above method to detect it.

4.1.2. Correcting for correlations of sequences of different length. Our cycle detection algorithm required comparing correlations of sequences as short as 5 (4 time units) with correlations of sequences as long as 12 (11 time units) in analyzing the Cho/Spellman data sets. However, high correlations are more likely to occur by chance on short sequences than long sequences. This phenomenon caused the curves of Fig. 4 to be skewed towards smaller shifts. Here we propose a statistical method to correct for it.

To determine the likelihood of each correlation value occurring by chance, we ran a Monte Carlo simulation study by generating twenty-million pairs of sequences of different lengths, each pair yielding

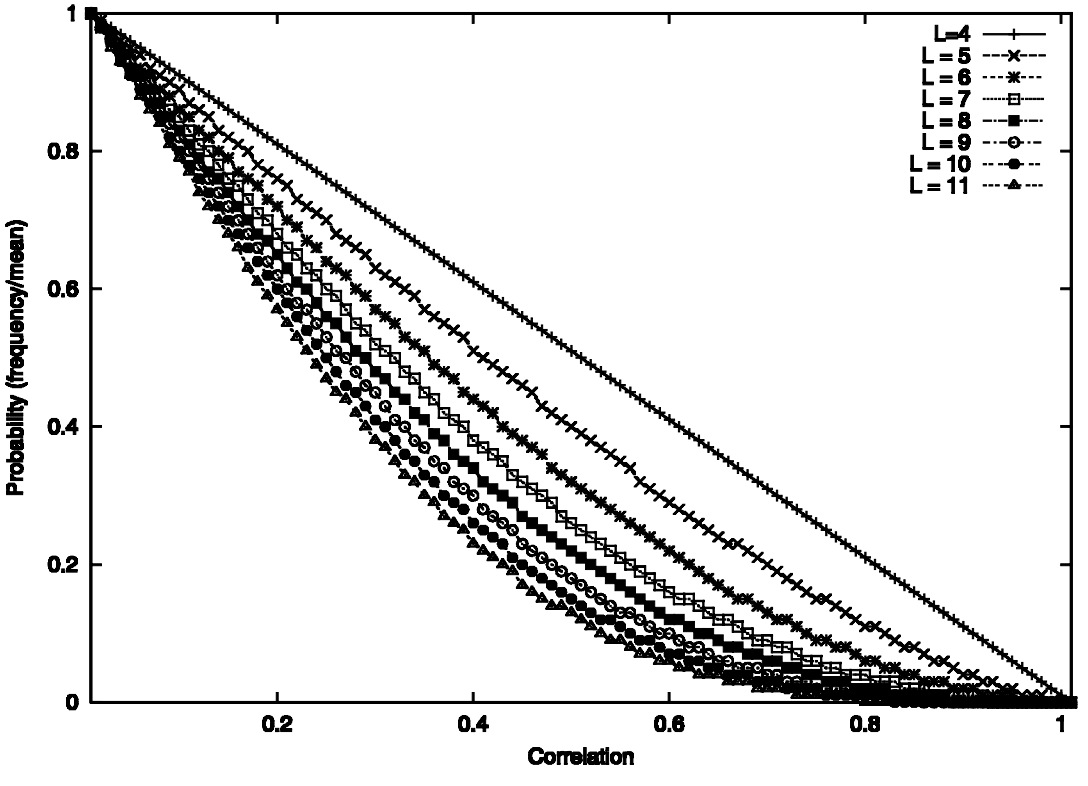

FIG. 5. Cumulative distribution function of positive correlation coefficients for sequences of length 4 to 11, from a Monte Carlo study of 20,000,000 sequence pairs, with values distributed as the measured values in the Cho/Spellman data sets.

a correlation value. Each sequence was generated from the distribution of the experimental expression values, where consecutive values in the sequence were assumed to be independent. The results, given in Fig. 5, give the statistical distributions of the correlation coefficient for various lengths. Finding analytical forms for these distributions is known to be a difficult problem (Snedecor and Cochran, 1980).

Given any two sequences of length bounded by our study, Fig. 5 gives the likelihood of their correlation occurring by chance. We used those likelihoods to correct for the observed "skewing" phenomenon. Namely, we used the same procedure as in the previous section, but this time the period for each curve was inferred to be the window length for which the correlation value had the smallest likelihood of occurring by chance.

Corrected plots are provided in Fig. 6. Clearly, the oscillations in the raw data have been dampened and even eliminated. Moreover, the corrections have revealed peaks around a single time point for all data sets except the elu data, which in fact did not cycle. For each data set, the period is given by the window length corresponding to the peak. Thus, we deduce that the periods are 70 minutes for alpha, 100 minutes for cdc28, and 90 minutes for cdc15. As shown in Table 1, these computational results match the observed cycle times very well.

4.2. Phase shift detection

In addition to changing period lengths, cell synchronization procedures leave cells in different stages of the cell cycle. These phase shifts must also be recovered to interleave the data.

To detect shift we used a method similar to the one for cell cycle detection or above, but this time we correlated across data sets. Using the periods of the time-series computed in Section 4.1, we normalized the periods of the cdc28 and cdc15 time-series curves with respect to the alpha data set so that all resulting time-series had the same period length. Then, by correcting for different length sequences, we correlated

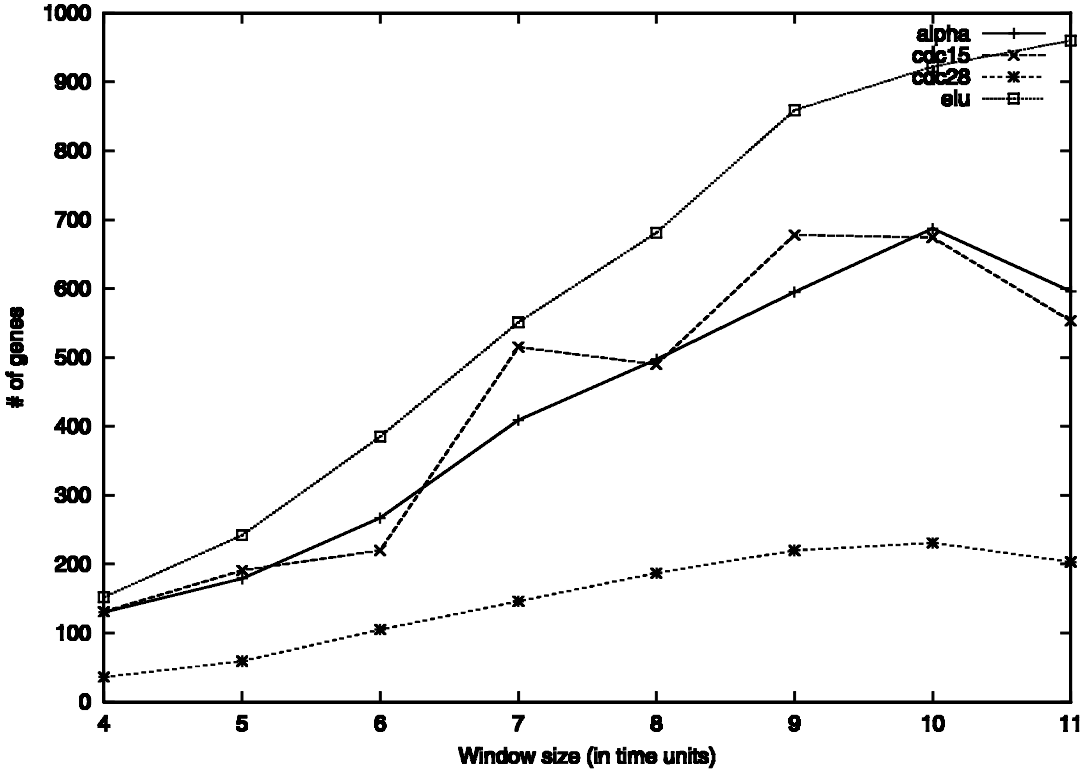

FIG. 6. Inferred periods for all data sets using length-corrected correlation coefficients. The window size (x axis) indicates the number of sampling intervals for each data set, and the # of genes (y axis) indicates the number of genes for which, when correlating a prefix and a suffix of equal lengths (of the gene's time-series), the length "window size" correlated the highest. The inferred cell cycle period corresponds to the "window size" at the peak (alpha—10, cdc15—9, cdc28—10, elu—/).

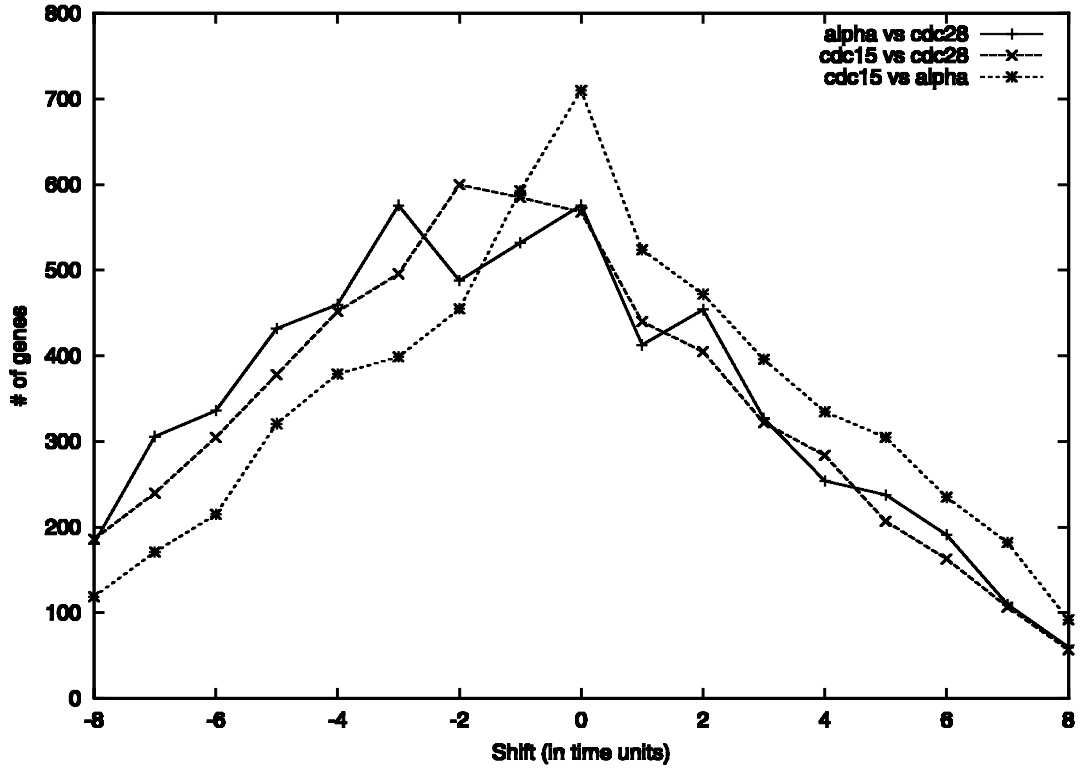

FIG. 7. Inferring phase shift for all pairs of alpha, cdc15, and cd28 data sets. Inferred phase shifts: approximately at the peaks of the curves.

the overlap of corresponding time-series curves from different data sets, by first aligning them at the experiment start point and then shifting one to the right, down the ticks of the other, across all possible shift lengths, up to the cell cycle of alpha, obtaining for each ORF pair a shift vector of correlations. By summing up these shift vectors for all curves, we obtain frequency distributions and can establish which shifts maximized the correlation for most curve pairs.

The results, provided in Fig. 7, demonstrate that alpha and cdc15 are unshifted relative to each other, while cdc28 shifts 1 to 2 time periods from cdc15. The results are less clear comparing alpha to cdc28, likely representing a shift of from 0–2 units. In fact, the alpha and cdc28 time-series both started in the $G1$ phase of the cell cycle, and cdc15 in the M phase (Spellman *et al.*, 1998). Our observed shifts are basically on target. For the transformed periods of roughly 70 minutes each, the computed shift offset of 14 minutes lies within the $M - G1$ time difference, since the M phase occupies roughly 50% of the cell cycle time and the $G1$ roughly 15%.

5. CORRELATION AND SMALL ALPHABETS

The correlation coefficient is highly regarded as a measure of similarity between pairs of sequences and has been widely used to analyze gene expression data (Eisen *et al.*, 1998). However, over the course of our work with this (and other) experimental data, indications arose that the correlation coefficient may perform very badly in comparing sequences drawn over small, finite alphabets. Such data arises naturally when attempting to smooth noisy data signals by quantizing them into a small number of bins, or as an output from a microarray scanning algorithm that classifies mRNA concentrations into three groups: up, down, and neutral (e.g., output from an Affymetrix chip scanner). Thus, we became interested in the behavior of the correlation coefficient on sequences over a small set of values, i.e., sequences over small alphabets.

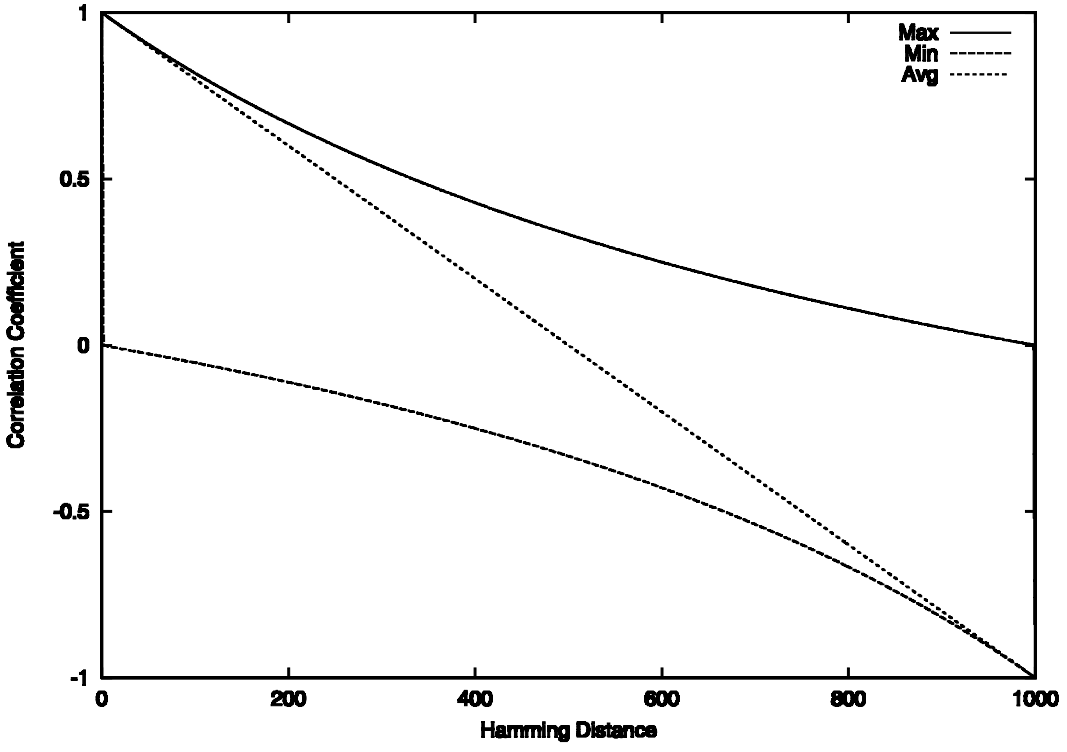

FIG. 8. Maximal, minimal, and average correlation coefficient values for any given Hamming distance between two two-letter-alphabet sequences of length 1,000.

We restricted our attention to the limiting case of sequences over two letter alphabets. For such sequences, we studied the behavior of the correlation coefficient as compared to that of the Hamming distance (number of positions in which two sequences disagree). Our study is described in detail in Filkov *et al.* (2001). Here we only summarize the results.

We computed all possible correlation values of any pair of two-letter-alphabet sequences using Formula 1. The obtained correlation values were grouped in classes—each class containing the correlations of sequence pairs with the same Hamming distance. In each class (i.e., for each Hamming distance), the maximum, minimum, and average correlation values were calculated and plotted as shown in Fig. 8. It is evident from the figure that even though the average correlation graph indicates expected behavior of the correlation coefficient, the upper and lower bounds demonstrate that even for very small Hamming distances, the possible range of correlations is distressingly large.

In fact, as shown by Filkov *et al.* (2001), this is true for two-letter-alphabet sequences of general length. For example, when correlating two sequences of length n which differ in only one position, it is possible to get a correlation of $1/\sqrt{2} \approx 0.7067$ for large n. Furthermore, two sequences which differ in two of the n positions can possibly correlate to $-1/(n-1) \approx 0$ for large n.

We conclude that the correlation coefficient is inadequate as a similarity measure of two-letter-alphabet sequences and must be viewed with caution in measuring the similarity of quantized or small alphabet sequences.

ACKNOWLEDGMENTS

The authors thank Prof. James Konopka for helpful discussions. We also thank the anonymous reviewers for their insightful comments. This work was partially supported by NSF Grant CCR-9988112 and ONR Award N00149710589.

REFERENCES

Aach, J., and Church, G.M. 2001. Aligning gene expression time series with time warping algorithms. *Bioinformatics* 17, 495–508.

Alter, O., Brown, P.O., and Botstein, D. 2000. Singular value decomposition for genome-wide expression data processing and modeling. *PNAS* 97, 10101–10106.

Chen, T., Filkov, V., and Skiena, S. 1999. Identifying gene regulatory networks from experimental data. *Proc. 3rd Annu. Int. Conf. Computational Molecular Biology (RECOMB'99),* 94–103.

Chen, T., He, H.L., and Church, G.M. 1999. Modeling gene expression with differential equations. *Proc. Pacific Symposium Biocomputing (PSB'99)*, 29–40.

Cho, R., Campbell, M., Winzeler, E., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T., Gabrielian, A., Landsman, D., Lockhart, D., and Davis, R. 1998. A genome-wide transcriptional analysis of the mitotic cell cycle. *Mol. Cell* 2, 65–73.

Eisen, M.B., Spellman, P.T., Brown, P.O., and Botstein, D. 1998. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA* 85, 14863–14868.

Filkov, V. 2000. Covering points on a circle with circular arcs. In D. Bremmer, ed., *Proc. Can. Conf. on Comp. Geom.*, Fredericton, Canada.

Filkov, V., Skiena, S., and Zhi, J. 2001. Analysis techniques for microarray time-series data. *Proc. 5th Annu. Int. Conf. Computational Molecular Biology (RECOMB'01),* 124–131.

Friedman, A., Linial, M., Nachman, I., and Péer, D. 2000. Using Bayesian networks to analyze expression data. *Proc. 4th Annu. Int. Conf. Computational Molecular Biology (RECOMB'00),* 127–135.

Holter, N., Mitra, M., Maritan, A., Cieplak, M., Banavar, J., and Fedoroff, N. 2000. Fundamental patterns underlying gene expression profiles: Simplicity from complexity. *Proc. Natl. Acad. Sci. USA* 97, 8409–8414.

Klevecz, R.R., and Dowse, H.B. 2000. Tuning in the transcriptome: Basins of attraction in the yeast cell cycle. *Cell Prolif.* 33, 209–218.

Snedecor, G., and Cochran, G. 1980. *Statistical Methods,* 7th ed. Iowa State University Press, Ames, IA.

Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D., and Futcher, B. 1998. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell* 9, 3273–3297.

Yuh, C.-H., Bolouri, H., and Davidson, E. 1998. Genomic cis-regulatory logic: Experimental and computational analysis of a sea urchin gene. *Science* 279, 1896–1902.

Address correspondence to:
Vladimir Filkov
Department of Computer Science
State University of New York
Stony Brook, NY 11794

E-mail: vlfilkov@cs.sunysb.edu