

Finding clusters of positive and negative coregulated genes in gene expression data

Non Peer-reviewed author version

KOCH, Kerstin; Schonauer, Stefan; JANSEN, Ivy; VAN DEN BUSSCHE, Jan & BURZYKOWSKI, Tomasz (2007) Finding clusters of positive and negative coregulated genes in gene expression data. In: Yang, MQ & Zhu, MM & Zhang, Y & Arabnia, HR & Deng, Y & Bourbakis, N (Ed.) PROCEEDINGS OF THE 7TH IEEE INTERNATIONAL SYMPOSIUM ON BIOINFORMATICS AND BIOENGINEERING, VOLS I AND II. p. 93-99..

DOI: 10.1109/BIBE.2007.4375550

Handle: <http://hdl.handle.net/1942/8020>

# Finding Clusters of Positive and Negative Coregulated Genes in Gene Expression Data

Kerstin Koch

Hasselt University & Transnational University of Limburg  
Theoretical Computer Science  
Agoralaan, 3590 Diepenbeek, Belgium  
kerstin.koch@uhasselt.be

Stefan Schönauer

University of Helsinki  
Helsinki Institute for Information Technology  
P.O. Box 68  
00014 University of Helsinki  
stefan.schoenauer@cs.helsinki.fi

Jan van den Bussche

Hasselt University  
& Transnational University of Limburg  
Theoretical Computer Science  
Agoralaan, 3590 Diepenbeek, Belgium  
jan.vandenbussche@uhasselt.be

Ivy Jansen

Hasselt University  
Center for Statistics  
Agoralaan, 3590 Diepenbeek, Belgium  
ivy.jansen@uhasselt.be

Tomasz Burzykowski

Hasselt University  
Center for Statistics  
Agoralaan, 3590 Diepenbeek, Belgium  
tomasz.burzykowski@uhasselt.be

## Abstract

In this paper, we propose a system for finding partial positive and negative coregulated gene clusters in microarray data. Genes are clustered together if they show the same pattern of changing tendencies in a user defined number of condition pairs. It is assumed that genes which show similar expression patterns under a number of conditions are under the control of the same transcription factor and are related to a similar function in the cell. Taking positive and negative coregulation of genes into account, we find two types of information: (1) clusters of genes showing the same changing tendency and (2) relationships between two such clusters whose respective members show opposite changing tendency.

Because genes may be coregulated by different transcription factors under different environmental conditions, our algorithm allows the same gene to fall into different clusters. Overlapping gene clusters are allowed because coregulation normally takes place in only a fraction of the investigated condition pairs, and because the gene expression

data is noisy so that the approach should be tolerant to errors. In a first step, the gene expression matrix is transformed to a binned matrix of changing tendencies between all condition pairs. For the binning of the gene expression levels, a statistical technique is used, for which no arbitrary threshold needs to be chosen, which automatically corrects for multiple testing, and which is able to handle replicates for the different conditions, immediately accounting for the random variability of gene expression data. To present the results of a clustering a new structure called coregulation graph is proposed.

## 1. Introduction

The metabolism of all organisms is tightly controlled by internal and external conditions so that not all proteins are produced under all circumstances. Products which function together in the cell are often under common regulatory control and expressed coordinately [2]. Because many gene products have multiple roles in the metabolism, genes may be coexpressed with different other genes under different environmental conditions. A clustering to find coregulated

genes should therefore allow for a gene to be a member of different clusters under different conditions.

There are different patterns of coregulated genes suggested by [4, 10]. Lee et al. [4] mention different loops of regulatory networks found in Yeast cells, e.g. the regulatory chains with at least three regulators where the product of one regulator binds to the promoter sequence for the next regulator. Examples of other motifs mentioned are the single input motif, where a single regulator binds to the promoter of a set of genes, and a multiple input motif where a set of regulators bind to a set of genes. Yu and coworkers [10] relate these motifs to different time patterns of gene expression. They point out that there are 4 different relationships of gene expression which are coregulated, time shifted coregulated, negative and negative time shifted.

To find interesting coregulated genes, the gene expression matrix is transformed into a binned matrix which captures changing tendencies between condition pairs (increase, decrease or no change) [3]. In our approach, a threshold arbitrarily to decide if a gene is differently expressed and needs to be chosen by the user ahead of time ([3, 11]), is avoided. Instead we use a statistical technique called SAM by Tusher et al. [7], that only needs a prespecified significance level (usually 5%). SAM also automatically corrects for multiple testing (since many genes and conditions are involved in this process), and it can handle replicates for the different conditions, meanwhile accounting for the random variability present in gene expression data. Clustering methods which use the normalised gene expression levels directly often have the problem that they mistake small fluctuations in gene expression levels as significant. This can result in clusterings with little or no biological meaning.

Our clustering is based on finding similar patterns of changing tendencies, not on finding similar absolute values of gene expression. Genes which have different gene expression values in some condition pairs may still show the same changing tendency between these condition pairs. These shift based clusters will be missed by methods based on distance measures [9]. Because for microarray data with many condition pairs it is unlikely that all condition pairs of the two genes are coregulated and because of the intrinsic noise of the data, our algorithm allows to find clusters with matches in only a part of consecutive condition pairs. Because we are also looking for genes with the opposite changing tendency, our approach is able to find the negative gene expression motifs as well.

The outcome of the clustering is visualised in a coregulation graph. In this graph, positively coregulated genes are found in the same vertex of the graph. Negative coregulation between gene clusters can be recognised by edges which connect two (positive) clusters that show opposite changing tendencies and therefore are negatively coregu-

lated. Clusters which show no negative coregulation with any other cluster of the graph are not connected.

## 2. Related Work

There are different clustering methods with different distance measures used for finding groups in gene expression data. Fuzzy k-means clustering is used by Eisen and coworkers [2]. In contrast to k-means clustering where genes are partitioned into a defined set of discrete clusters attempting to maximize the expression similarity in each cluster, each gene belongs to every cluster with a variable degree of membership using the fuzzy k-means method. To overcome the seeding problem where the random initialisation of the centroids of the clusters can have an impact on the results, they seed prototype centroids with eigen vectors identified by PCA. To identify patterns missed in the first round, they continue the clustering on a subset of the data in a second and third round. While this approach allows genes to be assigned to more than one cluster, it does not address the issue of negative coregulation.

Clustering approaches finding objects based on similar patterns which might not be close concerning distances like Euclidian distance are proposed by Wang et al. [9, 8]. Their algorithm, which finds *pclusters* [9], groups objects that exhibit a coherent pattern on a subset of dimensions. This is interesting for gene expression data, because the magnitude of the expression levels might not be close although the two genes show a similar pattern of expression. They introduce the *pscore* and a user defined threshold  $\delta$  and cluster together two objects if their *pscore* is less than  $\delta$ .

As this approach does not scale well to large data sets, they proposed an extension suitable for larger data sets [8]. They introduce a distance measure to decide whether two objects are similar in a subspace. Again, both methods do not address the issue of negative coregulated genes.

An algorithm to extract clusters of coregulated genes was proposed by Ji and Tan [3]. They introduce the concept of positive and negative coregulated gene clusters to cover positive and negative gene regulation. Positive coregulated gene clusters are defined as clusters which show a similar behaviour under a number of condition pairs, whereas negative coregulated gene clusters are defined to show the opposite behaviour in a number of condition pairs. However, their definition of positive and negative coregulated clusters is not symmetrical, which means that under the same conditions two genes may be identified as coregulated in one case and may not be identified as such if the genes are processed in a different order. This is due to their precision threshold which is the dependent probability of  $gene_i$  being upregulated given  $gene_j$  is upregulated, which may lead to different clusters if either  $gene_i$  or  $gene_j$  is taken as a reference gene to form the cluster.

They transform the gene expression data in a binned matrix with pairwise changing tendencies for all conditions. For the decision if a gene is up- or downregulated or not differentially expressed, the user has to choose an arbitrary normalisation threshold. The outcome of the clustering is depending on this threshold. With a low normalization threshold, lots of genes will be classified as significantly expressed which leads to more clusters. With a higher normalization threshold, on the other hand, many genes are classified as not significantly expressed which leads to less clusters. Their algorithm is proposed as an improvement over the support confidence framework used in A-priori-based data mining methods which reduces the large number of rules that may be generated by uncorrelated genes. Their negative clusters are based on one reference gene which shows the opposite behaviour compared to a positive cluster. The output of their algorithm is gene-centered, where positive and negative clusters are reported for each gene. This leads to multiple appearances of positive clusters in their output if the cluster contains more than one gene, which makes it hard to read the clustering result.

An approach also able to detect negative coregulation between genes was proposed by Zhao et al. [11]. They use a model based on so called g-cluster to find positive and negative coregulation between genes. They allow partial coregulation of genes by taking into account submatrices of conditions. For finding significant up- or downregulated neighbour conditions, they check if the difference of the absolute gene expression values of conditions  $c_1$  and  $c_2$  is larger than  $\delta \times$  expression  $c_1$ , where  $\delta$  is chosen by the user and is restricted to values between 0 and 1. By restricting the threshold to 1, only cases with a 2-fold up- or downregulation can be taken into account, whereas in gene expression data, often up- or downregulations higher than 3-fold are taken as significant. Again this static choice of a threshold can lead to certain regulation patterns being missed.

### 3. Methods

The goal of the coregulated gene mining process is to construct a coregulation graph for a given microarray experiment. In this graphical representation, genes with a similar gene expression pattern are clustered together at a vertex of the graph making it easy to see which genes share a similar expression pattern and therefore might be regulated by the same transcription factors. In addition, edges in the graph indicate which gene clusters show opposite expression patterns and are negatively coregulated.

### 3.1 Binning of the Gene expression matrix

In a first phase of the algorithm, the gene expression matrix is transformed into a binned matrix showing the pairwise changing tendency between conditions. We assume that there are  $n$  genes in total on a microarray, that  $m$  conditions need to be considered, and  $l_j > 1$  replicates (arrays) are available for each condition  $j$ . The gene expression matrix  $Y$  then has  $n$  rows and  $\sum_{j=1}^m l_j$  columns with elements  $Y_{ijk}$ , where  $Y_{ijk}$  is the measured gene expression of gene  $i$  ( $i = 1, \dots, n$ ) in condition  $j$  ( $j = 1, \dots, m$ ) for replicate  $k$  ( $k = 1, \dots, l_j$ ). From the matrix  $Y$  we want to produce a binned matrix, which will have  $n$  rows and  $m(m-1)/2$  columns, corresponding to the pairs of conditions. The creation of the binned matrix will be based on the SAM method by Tusher *et al.* [7], which is used to analyse microarray experiments and detect significant genes.

For each gene  $i$  and pair of conditions  $j_1, j_2$  with  $j_1 < j_2$ , the score

$$d_{ij_1j_2} = \frac{r_{ij_1j_2}}{s_{ij_1j_2} + s_0}$$

is calculated. It is based on the difference in average gene expression  $r_{ij_1j_2} = \bar{Y}_{ij_1} - \bar{Y}_{ij_2}$ , between conditions  $j_1$  and  $j_2$ , relative to its standard deviation  $s_{ij_1j_2}$ , augmented by a small positive constant  $s_0$ , called a fudge factor. This fudge factor ensures that the variance of the difference is independent of the mean gene expression level. Its value is chosen to minimize the coefficient of variation of the test statistic  $d_{ij_1j_2}$ .

Determining whether the value of  $d_{ij_1j_2}$  is significantly different from zero is not straightforward because one should control for multiple testing, and, due to the small numbers  $l_j$  of replicates, the test statistic  $d_{ij_1j_2}$  cannot be assumed to be normally distributed. In SAM, both problems are solved. Since SAM needs to use all  $d_{ij_1j_2}$  for all genes  $i$  and all pairs of conditions  $j_1, j_2$  simultaneously, we will simplify notation to  $d_p$ , with  $p = 1, \dots, N$ , where  $N = n \times m(m-1)/2$ .

The idea is to use a number  $B$  of arbitrary permutations of the columns of the matrix  $Y$  (recall that these columns represent all replicates of all conditions). For each permutation  $b$ , we recalculate  $d_p$  but on the permuted matrix  $Y^b$ , denoted by  $d_p^b$ . For each  $b$ , we sort the values  $d_1^b, \dots, d_N^b$ , resulting in the order statistics  $d_{(1)}^b \leq \dots \leq d_{(N)}^b$ . We now determine, for each  $p$ , the average of  $d_{(p)}^b$  over all  $b$ 's, denoted by  $\bar{d}_{(p)}$ . We also sort the original values  $d_1, \dots, d_N$ , resulting in the order statistics  $d_{(1)} \leq \dots \leq d_{(N)}$ .

Now, for a fixed threshold  $\Delta > 0$ , all gene – condition pair combinations for which  $d_{(p)} - \bar{d}_{(p)} > \Delta$  are called “significant positive”. Similarly, all gene – condition pair combinations for which  $d_{(p)} - \bar{d}_{(p)} < -\Delta$  are called “significant negative”. This is repeated for a grid of  $\Delta$  values, and a

	C2	C3	C4
C1	1	1	1
C2		1	1
C3			-1

**Figure 1. Row of the binned matrix for one gene in triangular matrix form.**

list of significant gene – condition pair combinations is obtained for each value. Moreover, the False Discovery Rate (FDR) is estimated for each  $\Delta$ . FDR is the expected proportion of false positive gene – condition pair combinations among all gene – condition pair combinations called significant. Based on an a priori chosen level for FDR (mostly 5%), the corresponding  $\Delta$  value is chosen, and the significant gene – condition pair combinations are listed. More details can be found in Chu *et al.* [1].

Finally, when, based on the analysis of the test statistic  $d_{ij_1j_2}$ , the average change  $r_{ij_1j_2}$  in gene expression for gene  $i$  between conditions  $j_1$  and  $j_2$  is determined to be significantly different from zero, the value  $O_{ij_1j_2}$  in the binned matrix is taken to be equal to 1 when  $r_{ij_1j_2} > 0$ , and equal to  $-1$  when  $r_{ij_1j_2} < 0$ . For all other genes and pairwise comparisons,  $O_{ij_1j_2} = 0$ .

### 3.2 Clustering

After the gene expression matrix has been binned, we use a clustering technique to identify groups of similarly expressed and hence coregulated genes within the matrix. Our goals with our new approach where to identify coregulated as well as negative coregulated genes and present them in an easy, human-readable form. To achieve those goals, we first need to define what we mean by the terms "similarly expressed" or "coregulated" in a clustering sense.

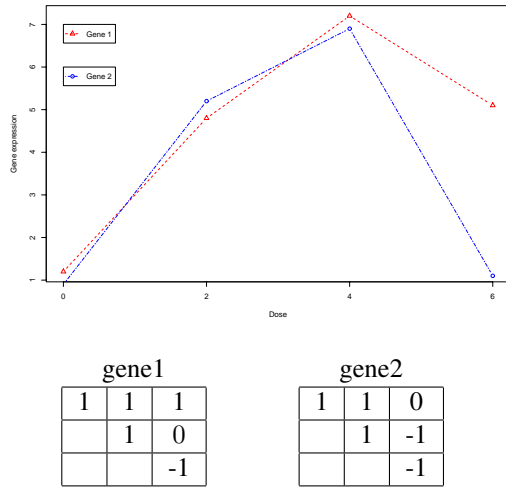
One aspect important for clustering expression data is the fact that a gene can be coregulated with different other genes under different conditions. Consequently, the expression behavior of two coregulated genes is usually not the same for all conditions within one microarray experiment. To take this into account, the clustering technique should not be partitioning, i.e. assign an object to exactly one cluster. On the contrary, a gene could be part of different clusters, depending on the conditions under which they were expressed.

As described above, each gene in the microarray experiment is represented in the binned matrix by a row of  $\frac{m(m-1)}{2}$  values indicating the pairwise changing tendency between the  $m$  conditions of the original microarray data. One such row in the binned matrix can itself be viewed as an upper triangular matrix. The first entry of the first row in the binned matrix for a gene denotes the change of ex-

pression behavior when transitioning from condition C1 to Condition C2 during the microarray experiment. The next entry in the first row denotes the overall change in expression behavior when transitioning from condition C1 to C3. The values change for a transition from C2 directly to C3 is then stored in the next row of the triangular view. Figure 1 illustrates the principle and Figure 2 shows an example for two genes. It should be noted that for a microarray experiment with  $m$  conditions, there are exactly  $(m - 1)$  rows to such a triangular matrix.

For microarray data with a natural order within the conditions, not all parts of the binned matrix do have the same importance for answering the different questions posed to gene expression data. Examples for commonly used microarrays with such an order are time series experiments where conditions are time points and dose response microarrays where the effect of different doses of a substance are tested. For time series data, the change between non-consecutive time points which are not on the diagonal of the triangular matrix provides additional information about the course of the gene expression over longer time spaces, but the main trend can be seen by comparing the values on the diagonal. The same applies for dose response data, where the diagonal values describe the effect on gene expression from one amount of the substance given to the next higher amount of the substance (e.g. from amount 0 to 2 or from amount 2 to 4 in Figure 2). The off-diagonal values of the matrix describe the effect on the gene expression between different amounts of the substance which are not consecutive. Using this additional information in Figure 2, it can e.g. be seen that concerning dose 6, gene 1 and gene 2 show a different behaviour. For gene 2, dose 6 has the same effect as giving dose 0, whereas for gene 1 there is clearly an effect for this dose of the substance, although the trend in both genes (the diagonal values) are the same. How this information can be used for filtering clusters is explained below. While the algorithm described in this paper is focused on such data with a natural order in the conditions, it can easily be adapted to data sets where all positions of the binned matrix are of equal importance.

Zeros in the binned matrix indicate that the gene did not show any significant change in expression level between the condition pair for this cell. This is normally due to the fact that the gene is not part of the specific cell reaction under investigation. Therefore, genes which exhibit a large number of zeros in the binned matrix should be excluded from the investigation because they do not belong to the differentially expressed genes for the investigated condition. A certain number of zeros should be tolerated in the clusters in contrast to the Ji and Tan algorithm [3], since even a gene which shows a specific reaction under the investigated conditions does not necessarily show it for each condition pair. For example, this can happen due to the intrinsic noise of



**Figure 2. Dosage response of two genes**

the gene expression data, which makes it difficult to decide whether a gene is differentially expressed. The number of accepted zeros in a cluster is a parameter of the clustering algorithm which can be easily adopted by the user.

Considering all of the above, we define genes as being coregulated in the following way:

**Definition 1 (coregulated and negative coregulated genes)**

Let  $S = \{g_1, g_2, \dots, g_i\}$  be a set of genes, each represented by a sequence of  $(m - 1)$  expression level change values. The genes in  $S$  are  $(k, l)$ -coregulated if there exists a subsequence of at least  $k \leq (m - 1)$  consecutive components common to all genes in  $S$  and this subsequence contains at most  $l < k$  zeros.

Two genes  $g_1$  and  $g_2$  are negative coregulated if they are coregulated after all values for  $g_1$  in the binned matrix have been inverted. The value in the binned matrix are inverted by changing each value 1 to -1 and vice versa.

Two points have to be noted about this definition. One is that a coregulation relationship between two genes as defined above is symmetrical contrary to the definition in [3]. Another important point is that the definition covers the fact that a gene can be coregulated with different and even the same genes under different experimental conditions. Consequently, any clustering algorithm used to identify coregulated genes has to allow a gene to be assigned to several clusters based on the different condition subsets.

With the above definition, we can now describe our algorithm for finding all cluster of coregulated genes from a microarray experiment. The algorithm consists of four main steps: binning, preprocessing, clustering coregulated genes, detecting inverse coregulations.

After the binning described in section 3.1, the resulting matrix is subject to a pre-processing step. As discussed earlier, there should not be too many cases for a gene in which no change in expression behavior happens between condition transitions. Otherwise, the gene can not be assumed to be part of the cell specific answer to the different conditions tested in the experiment. For genes which are always up- or downregulated, the same assumption holds for some cases as well. Genes which react with an up- or downregulation to each change of the conditions during the experiment do not belong to the specific response tested in the microarray and are therefore normally not interesting for the researcher who is interested in this specific response to the conditions applied. Therefore, we remove all genes whose expression behavior always stays the same or is always up-regulated or always down-regulated from the binned matrix and cluster only the remaining genes. If the user is interested in these genes as well, the preprocessing step can be omitted.

After the binning and preprocessing of the gene expression data, all clusters of coregulated genes are detected. This is done in an iterative manner. As a first step for the clustering, a gene is chosen and its entry removed from the matrix. For this gene all possible subsequences which fulfill the clustering condition, i.e. have length  $k$  and contain at most  $l$  zeros, are generated. For each one of those subsequences a new cluster is created and all remaining genes from the matrix which show this subsequences are assigned to the respective cluster. After all subsequences have been processed, a new gene from the matrix is chosen and the process repeats.

As the next step of the clustering, the inverse coregulation relationships are detected. For all clusters found in the previous step, the inverted coregulated clusters are searched. One cluster is chosen and the associated subsequence is inverted. Coregulated clusters with a subsequence equal to the inverted sequence are noted to be inverse coregulated.

Since the number of clusters found with the above approach is potentially very large, we propose a new structure to visualize the clusters and the negative coregulation relationships between them. This structure is called the coregulation graph. This graph consists of nodes which each are one of the clusters of coregulated genes found in the first step of the clustering process. There exists an edge between two nodes in the coregulation graph, if the cluster in the adjacent nodes are in a negative coregulation relationship. Presenting the clustering results as a coregulation graph allows to quickly visually identify which groups of genes are negative coregulated to each other.

**Filtering Cluster.** Displaying the clustering result as a coregulation graph improves the readability significantly and thereby eases interpretation of results. But still, the number of clusters deduced from a large microarray exper-

	A	B	C	D
1	1	1	1	1
2		1	1	1
3			-1	0
4				1

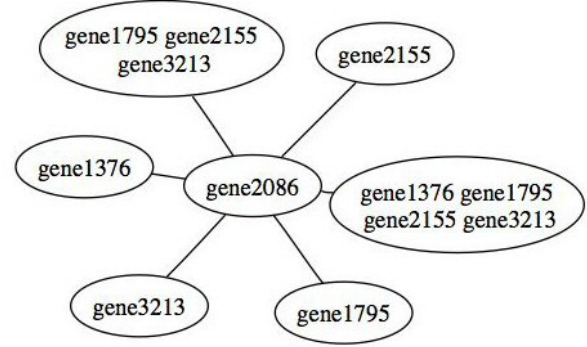
**Figure 3. Binned expression profile of a gene in triangular matrix form.**

iment can be very large. To identify promising clusters for further investigation, a ranking based on the quality of the clusters would be needed. So far we have only used the values for changes between immediately adjacent conditions in the microarray experiment for our clustering. While these are the most important ones to describe the change of expression behavior of a gene under the different conditions of the experiment, changes between not directly adjacent conditions can provide additional insight. These values refine the information about the details of the changes between the values along the main diagonal as illustrated by the example in Figure 3. In this example, the upregulation shown on position C2 indicates that the downregulation in C3 is smaller than the preceding upregulation in B2. Therefore, genes agreeing on the values in positions B2, C2 and C3 can be seen as stronger coregulated than genes only agreeing on positions B2 and C3. We propose two measures to be used for filtering the clustering result and reducing its size to the most interesting structures. Both measures are based on the number of values off the main diagonal which also support the coregulation relationship among the cluster members. Which off-diagonal elements have to be considered, depends on which elements on the diagonal are taken into account to define the respective cluster.

As an example consider again the situation depicted in Figure 3. If the diagonal elements A1, B2 and C3 establish a coregulation relationship between the genes of a cluster, the values in cells B1, C1 and C2 provide additional information about the change of expression levels between the conditions represented by the diagonal elements. Therefore, we count the number of those additional elements on which the genes in the cluster also agree. In the following, we denote the the number of cells on the diagonal, which establish the coregulation relationship, with  $\lambda$  and the number of corresponding cells off the diagonal on which the genes of a cluster agree  $\kappa$ . We use the following two numbers to filter and rank cluster:

- $m_1 = \frac{\kappa}{\kappa + \lambda}$
- $m_2 = \frac{\kappa}{\rho}$ , with  $\rho$  the number of cells in a row of the binned matrix.

Both measures are normalized allowing the user to choose values applicable to a wide range of data sets. For



**Figure 4. Part of the coregulation graph obtained from the GEO data set.**

both measures higher values indicate a stronger correlation between the values of the genes in the cluster. While  $m_1$  favours stronger correlations of shorter subsequences,  $m_2$  gets largest for longer subsequences. Depending on the microarray experiment and the user intentions, one or the other will be more favorable for limiting the size of the clustering result and support the evaluation purposes. An example of a coregulation graph is shown in Figure 4.

## 4 Experiments

The binning part of the algorithm was compared to the binning of the Ji-Tan algorithm. We use a data set of 500 genes for which gene expression levels are measured under 3 conditions. There are 20 replicates for condition 1 and 15 replicates for condition 2 and 3 in the data set. This data set is part of the SAM plugin for Excel which can be downloaded at <http://www-stat.stanford.edu/~tibs/SAM/>. To be able to compare the performance of the binning with the Ji-Tan algorithm [3], we need to summarise the replicates into a single observation per condition, such as the mean value and use the input for the Ji-Tan algorithm [3]. Using the mean values, we found 388 positive and negative coregulated gene clusters taking a normalisation threshold of 0.3, a frequency threshold of 0 and a precision threshold of 0.8. Using our algorithm and allowing a false discovery rate of 5%, we found 38 gene clusters which are not at all a subset of the 388 clusters of Ji and Tan. This is due to the fact that multiple gene expressions are summarised into a single value, not taking into account the variability of the data. This might result in assuming a different gene expression level while it is not differentially expressed or in the assumption of no difference when the difference is



**Figure 5. Part of the coregulation graph obtained from the GEO data set.**

significant.

The clustering part of the algorithm is tested on 2 different data sets of Yeast cell cycle data which are both parts of the Spellman data set [6] as well as a data set from the GEO database <http://www.ncbi.nlm.nih.gov/geo/>. As first data set, the 17 time points for Yeast synchronised in the cell cycle by *alpha*-factor were taken. This test set contains 6178 Yeast genes. As second data set, a subset of this data of 2884 genes used by Ji and Tan for their clustering algorithm was taken [3]. This data can be downloaded at <http://www.comp.nus.edu.sg/~jiliping/pl/Yeast%20Matrix.txt>. Because this data does not contain replicates, it was binned using the first phase of the Ji and Tan algorithm with a normalisation threshold of 0.3. The third data set is freely available Gene expression data from the GEO database. The data set downloaded here was the GDS1804 data set containing 16 microarray experiments with expression levels from *E. coli* K12 cells at different time points after inducing an alternative sigma factor (Sigma32) which plays a role in transcriptional regulation during stress (heat shock). For this data set, replicates for different time points are available so that the new binning method could be used. Because there are many genes where at least one condition in one of the microarray experiments is a NULL value, these genes are excluded leading to a data set with 3766 genes instead of the original 6400 genes (including controls). The clusters are interpreted biologically using textual description from <http://db.yeastgenome.org/> for yeast ORFs and the *E. coli* K12 Genome Annotation from the EBI <http://www.ebi.ac.uk/GOA/proteomes.html>.

Our clustering algorithm finds 436 positive clusters for

the first Yeast test set containing 6178 genes matching a subsequence of length 15 and allowing one position to be zero within this subsequence. Because of the graph structure which facilitates the examination, interesting clusters can be easily identified. Some examples for interesting clusters found are given below. Closer examinations of the results are still work in progress.

We could identify an interesting cluster in the second data set, *i. e.* the smaller Yeast data set with 2884 genes, containing two genes involved in the linkage of transcriptional regulation to RNA Polymerase II (Gene 432 (YCR081W) and Gene 2870 (YPR168W)). Both genes are annotated with the same *GO*-term (*GO* : 0016455) for biological function and interact both with the same mediators (Med2 and SRB6). Using the larger first data set, meaningful clusters were found as well. An example is a cluster with 4 genes where 3 encode for structural protein of ribosomes are found (Gene 879 (YDL130W), Gene 2777 (YHR203C) and Gene 4781 (YNL067W)). For the fourth gene, no information in the used annotation was available.

Another very interesting relationship can be found in the clustering obtained from the third data set from *E. coli* (cf. Fig. 5). Gene 3025 has a very central role repressing many other clusters. This gene is the *arcA* Gene, which is one of the main regulators in the *E. coli* metabolism. The protein coded by this gene is a sensor for oxygen in the environment. It represses many genes involved in anaerobic metabolism of *E. coli*. In cells, different metabolic pathways are activated in the presence or absence of oxygen. This gene is negatively co-regulated with gene number 3080, which is a Lactat Dehydrogenase (LdhA). Lactat DH is used under anaerobic conditions to gain energy from Pyruvate.



This enzyme is known to be repressed under aerobic conditions. These examples show that our found clusters can be interpreted biologically. Using the binning of the Ji-Tan algorithm on this data set (normalisation threshold 0.3) leads to different clusters. The central genes number 3025 and 3080 do not appear in our clusters using this binned matrix.

As Sigma32 was induced during the the experiment for the third data set, we also looked for cluster containing at least two genes that are known to be regulated by Sigma32. The Sigma regulated genes were downloaded from the Regulon DB [5]. Using our binning on this dataset, we could find clusters for 20 of these genes looking for 3 consecutive matches on the diagonal and allowing at most 2 zeros per match. We were able to identify 20 genes in 4 clusters. The first cluster contained 9 Sigma32 regulated genes in 40 genes, a second smaller cluster contained 13 genes whereof 7 were Sigma32-regulated. In the last two clusters 3 out of 14 genes and 2 out of 4 genes share the property of being Sigma32 regulated. Due to the lack of reliable ground truth information we could not determine statistical significance measures for our clusterings. But the above ratios suggest that the clusters found are indeed of high significance.

We also clustered the same data after binning it with the method of Ji and Tan with thresholds 0.2 and 0.3. In neither case could any of the 29 above mentioned Sigma32-regulated genes be clustered together. This underlines that coregulated genes can be found using the combination of our new binning and clustering technique were competing methods fail.

## 5. Conclusion

Detecting coregulated genes is an important task in microarray data analysis. In this paper, we presented a new approach to detecting coregulated genes in time-series microarray data, using clustering techniques. We proposed a new approach to discretize expression data in order to detect the changing tendency between conditions. We formalized the notion of positive and negative coregulated genes and presented an algorithm to find all such relationships among the genes present on a microarray. Finally, we introduced the concept of a coregulation graph to present the clustering results in a visual and human-readable form. In several experiments, we showed that our approach produces biologically meaningful results.

A more thorough investigation of the obtained clusters of coregulated genes and their part in the regulatory network of the respective organism remains for the future. Another open question is how to integrate also time-shifted coregulation patterns into our approach.

## References

- [1] G. Chu, B. Narasimhan, R. Tibshirani, and V. Tusher. Sam “Significance Analysis of Microarrays” Users guide and technical document. Technical report, Stanford University, 2001.
- [2] A. P. Gasch and M. B. Eisen. Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. *Genome Biology*, 2002.
- [3] L. Ji and K.-L. Tan. Mining gene expression data for positive and negative co-regulated gene clusters. *Bioinformatics*, 2004.
- [4] T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, Z. Bar-Joseph, G. K. Gerber, N. M. Hannet, C. T. Harbison, G. M. Thompson, I. Simon, J. Zeitlinger, E. G. Jennings, H. L. Murray, D. B. Gordon, B. Ren, J. J. Wyrick, J.-B. Tagne, T. L. Volkert, E. Fraenkel, D. K. Giffort, and R. A. Young. Transcriptional regulatory networks in *saccharomyces cerevisiae*. *Science*, 2002.
- [5] H. Salgado, S. Gama-Castro, M. Peralta-Gil, E. Diaz-Peredo, F. Sanchez-Solano, A. Santos-Zavaleta, I. Martinez-Flores, V. Jimenez-Jacinto, C. Bonavides-Martinez, J. Segura-Salazar, A. Martinez-Antonio, and J. Collado-Vides. Regulondb (version 5.0): Escherichia coli k-12 transcriptional regulatory network, operon organization, and growth conditions. *Nucleic Acids Res*, 2006.
- [6] P. T. Spellman, G. Sherlock, M. Q. Zhang, W. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher. Comprehensive Identification of Cell Cycle-regulated Genes of the Yeast *saccharomyces cerevisiae* by Microarray Hybridization. *Molecular Biology of the Cell*, 1998.
- [7] V. G. Tusher, R. Tibshirani, and G. Chu. Significance analysis of microarray applied to the ionizing radiation response. *PNAS*, 2001.
- [8] H. Wang, F. Chu, W. Fan, P. S. Yu, and J. Pei. A fast algorithm for subspace clustering by pattern similarity. In *Proceedings of the 16th International Conference on Scientific and Statistical Database Management (SSDBM 2004)*, 21-23 June 2004, Santorini Island, Greece, pages 51–62, Los Alamitos, CA, USA, 2004. IEEE Computer Society.
- [9] H. Wang, W. Wang, J. Yang, and P. S. Yu. Clustering by pattern similarity in large data sets. In *Proceedings of the ACM SIGMOD 2002*, pages 394–405, June 4th to 6th 2002.
- [10] H. Yu, N. M. Luscombe, J. Qian, and M. Gerstein. Genomic analysis of gene expression relationships in transcriptional regulatory networks. *TRENDS in Genetics*, 2003.
- [11] Y. Zhao, G. Wang, Y. Yi, and G. Yu. Mining Positive and Negative Co-regulation Patterns from Microarray data. In *Proceedings of the Sixth IEEE Symposium on Bioinformatics and BioEngineering, BIBE’06*, pages 86–93, October 2006.