Gene expression

Composite Module Analyst: a fitness-based tool for identification of transcription factor binding site combinations

A. Kel1,*, T. Konovalova2, T. Waleev3, E. Cheremushkin3, O. Kel-Margoulis1 and E. Wingender1,4

1BIOBASE GmbH, Halchtersche Str. 33, D-38304 Wolfenbüttel, Germany, 2Institute of Cytology and Genetics, Novosibirsk, Russia, 3A, P. Ershov’s Institute of Informatics Systems, 6, Lavrentiev avenue, 630090 Novosibirsk, Russia and 4Department of Bioinformatics, UKG/University Göttingen, Goldschmidsdtstr. 1, 37077 Göttingen, Germany

Received on November 30, 2005; revised on January 9, 2006; accepted on February 3, 2006
Advance Access publication February 10, 2006
Associate Editor: Thomas Longauer

ABSTRACT
Motivation: Functionally related genes involved in the same molecular-genetic, biochemical or physiological process are often regulated coordinately. Such regulation is provided by precisely organized binding of a multiplicity of special proteins [transcription factors (TFs)] to their target sites (cis-elements) in regulatory regions of genes. Cis-element combinations provide a structural basis for the generation of unique patterns of gene expression.

Results: Here we present a new approach for defining promoter models based on the composition of TF binding sites and their pairs. We utilize a multicomponent fitness function for selection of the promoter model that fits best to the observed gene expression profile. We demonstrate examples of successful application of the fitness function with the help of a genetic algorithm for the analysis of functionally related or co-expressed genes as well as testing on simulated and permuted data.

Availability: The CMA program is freely available for non-commercial users. URL http://www.gene-regulation.com/pub/programs.html#CMAAnalyst. It is also a part of the commercial system ExPlain® (www.biobase.de) designed for causal analysis of gene expression data.

Contact: alexander.kel@biobase-international.com

1 INTRODUCTION

Massive application of microarray measurements of gene expression is a common route in the current studies of disease mechanisms. Hundreds of genes are revealed whose change of expression is associated with the disease. However, changed expression of these genes often represents just an ‘echo’ of the real molecular processes in the cells. Still, the available means of analysis and interpretation of these mechanisms are very limited.

Regulation of gene expression is accomplished through binding of transcription factors (TFs) to distinct regions of DNA. It is clear by now that combinations of TFs rather than single a TF drive gene transcription and define its specificity. At the level of DNA, the blueprints for assembling such variable TF complexes on promoter regions may be seen as specific combinations of TF binding sites located in close proximity to each other. We call such structures ‘composite regulatory modules’.

Known functional combinations of TF binding sites were used before in a number of approaches of promoter analysis: for the identification of muscle-specific promoters (Frech et al., 1998; Wasserman and Fickett, 1998), promoters of liver-enriched genes (Tronche et al., 1997), of yeast genes (Brazma et al., 1997), of immune-specific genes (Boehlk et al., 2000; Fessele et al., 2001; Kel et al., 1999), promoters of genes regulated during cell cycle (Kel et al., 2001) or antibacterial defense response (Shelest and Wingender, 2005). In the cases when the functional site composition is not known, a number of new predictive approaches identifying composite motifs can be used: BioProspector (Liu et al., 2001), Co-Bind (GhiaThakurta and Stormo, 2001), MITRA (Eskin and Pevzner, 2002), dyad search (van Helden et al., 2000). These programs help to discover new regulatory sites for yet unknown TFs, but an ‘ab initio’ motif finding method is limited in their ability to distinguish signal from noise in the long regulatory regions (10 Kb and more) of genes of human and other higher eukaryotic organisms. Novel methods have been developed that utilize information on known and potential binding sites in promoters (ClusterScan: Kel-Margoulis et al., 2002b; TOUCAN system: Aerts et al., 2003; probabilistic methods: Sinha et al., 2003). But still analysis of long regulatory regions remains a challenge for these approaches.

In this paper we are describing a novel tool, Composite Module Analyst (CMA), which has been created to make the next step in applying combinatorial approach to the analysis of regulatory regions of functionally related or co-expressed genes. This tool applies a novel approach for defining the promoter model based on composition of single TF binding sites as well as their pairs located inside local regulatory domains (corresponding to enhancer/silencer subregions). We utilize a multicomponent fitness function for selection of the promoter model, which fits best to
the observed gene expression profile. We demonstrate some successful applications of the fitness function with the help of a genetic algorithm for the analysis of functionally related or co-expressed genes as well as testing of the tool on simulated data.

2 COMPOSITE MODULES

2.1 Databases and tools for identification of TF binding sites

In this study we used three databases: (1) TRANSFAC® is a database on gene regulation (Matys et al., 2006). It provides data on TFs and their binding sites in promoters and enhancers of eukaryotic genes as well as a library of position weight matrices (PWMs). This work has been done with release 8.4. (2) TRANSCompel® (rel.6.4) which contains known composite regulatory elements in mammalian genes (Kel-Margoulis et al., 2002a). (3) TRANSPRO® is a database on promoters (Chen et al., 2006). Positions of transcription start sites (TSS) in genomes (human, mouse, rat) are collected from EPD, DBTSS and Ensembl, providing the possibility to retrieve about 13 000 human and about 10 000 murine promoters. Potential binding sites for TFs were identified by the TRANSFAC® tool Match® (Kel et al., 2003) that uses a library of about 500 PWMs for vertebrate TFs (TRANSFAC®, release 8.4).

2.2 Definition of composite promoter model

We model the promoter structure by a Boolean function \( \theta \) from which the promoter score \( ps \) can be computed:

\[
ps = \theta(b_1, b_2, \ldots, b_m),
\]

where \( b_i \) are the (0,1) outputs of \( m \) independent composite modules (CMs) \( cm \), \( i = 1, 2, \ldots, m \).

Each composite module is a triplet \((\Phi, M, \Psi)\), where \( \Phi \) is the set of TFs regulating the promoters; \( M \) is a set of PWMs that are used to identify potential binding sites for the TFs from the set \( \Phi \); and \( \Psi \) is a set of rules and parameters that are applied to the set \( M \) to search for the potential DNA binding sites in promoters. In the current realization of the method we consider the following rules and parameters \( \Psi \):

1. \( w_i \), the length of the window in promoter sequences that covers all sites included in the CM;
2. \( K_i \), the number of individual PWMs in the module,
3. \( R_i \), the number of pairs of PWMs,
4. cut-off value \( q_{k \text{cut-off}} \), relative impact values \( \varphi_{k} \), maximal number of best matches \( k_{\text{max}} \) are assigned to every individual weight matrix \( k \), \( k = 1, K_i \),
5. cut-off value \( q_{r \text{cut-off}} \), relative impact values \( \varphi_{r} \), orientation values \( \delta_{r} \) and maximal \( d_{\text{max}} \) and minimal \( d_{\text{min}} \) distances and maximal number of best pairs of matches \( k_{r} \) are assigned to every matrix pair \( r \), \( r = 1, R_i \) in the CM.

First of all, the program searches for potential TF sites in the promoters using Match™ algorithm by applying the matrices M and corresponding cut-off values. Next, in each sliding window of the length \( w \), the program selects the predefined maximal number of the best matrix matches and checks if the found sites obey the predefined distance and orientation rules. A maximal normalized composite score value \( cm_i \) is calculated among all window positions using the following equation:

\[
\text{cm}_i = \frac{\left[ \sum_{k=1}^{K_i} \varphi_{k} \times \sum_{j=1}^{k_{\text{max}}} q_{k}(x) + \sum_{r=1}^{R_i} \varphi_{r} \times \left(q_{r}^{(1)}(x) + q_{r}^{(2)}(x)\right) \right]}{\max(\text{cm})},
\]

where \( q_{k}^{(1)}(x) < q_{k \text{cut-off}} \) and \( q_{r,2}^{(1)}(x) < q_{r \text{cut-off}} \); and distance between matches of two matrices of a pair \( r \) : \( d_{\text{max}} < d_{r} < d_{\text{min}} \); \( \text{Max (cm)} \) is the maximal possible value of cm.

A threshold parameter \( P_i \) is defined for each composite module. It is used to compute the corresponding Boolean value:

\[
b_i = \begin{cases} 
0, & \text{cm}_i < P_i \\
1, & \text{cm}_i \geq P_i 
\end{cases}.
\]

Finally, the promoter score \( ps \) is calculated using the Boolean function \( \theta \). If, for a given promoter, \( ps = 1 \) we consider that this promoter matches the defined promoter model.

In order to score the values obtained with the Boolean function \( \theta \) we applied the fuzzy logic approach. For each promoter we define a fuzzy score \( \lambda \) (or fuzzy promoter score fps) which is based on the scores \( \lambda_i = \text{cm}_i \), of each component of the predicate of the function \( \theta \). The following calculation rules are applied:

\[
\lambda = \lambda_1 \lambda_2 \text{ for logical AND;}
\]

\[
\lambda = 1 - (1 - \lambda_1)(1 - \lambda_2) \text{ for logical OR;}
\]

\[
\lambda = (1 - \lambda) \text{ for logical NOT.}
\]

2.3 Fitness function

Based on the assumption that promoters of co-regulated genes should share a common composition of TF binding sites we propose an approach which allows defining a composite promoter model for sets of co-regulated genes. We have constructed a fitness function that provides a means to direct the search in a wide space of various parameters and to find the optimal promoter model settings. In this approach we take as input a set of promoters of potentially co-regulated genes (set A) and a background set of promoters (set B) and design a fitness function, which is based on several properties of distributions of the \( \theta \) function and \( \text{cm}_i \) values in these two promoter sets. As an alternative input we consider a single set of promoters with assigned expression values.

The components of our fitness function are the following:

- \( R \), regression value;
- \( T \), Student t-test value;
- \( E \), specificity and sensitivity value;
- \( N \), normality index;
- \( P \), penalty on the complexity of the model.

The fitness function is defined as linear combination of these components with the specified relative weights:

\[
Z = \frac{(aR + bT + cE + dN + eP)}{(a + b + c + d + e)}.
\]

The weights \( a, b, c, d, e \) can be modified by users. We will describe the components in detail.
2.4 Regression value

This component shows how well the fuzzy promoter scores $fps(n)$ of the promoters ($n = 1$, NP-total number of promoters) fit the expression values $\xi(n)$ assigned to the corresponding genes. The $\xi(n)$ values can be assigned either as 1/0 values to the promoters of two corresponding sets of genes A and B, or as any continuous function of the observed differential expression value (for instance as the log ratio of gene expression provided by Affymetrix software). The linear regression is done by fitting the $fps(n)$ to the curve $\alpha \xi(n) + \beta$. We compute then the regression value $R^2$ for the promoter model and normalize it to the maximal possible value. It is then taken as the component $R$ in Equation (4) which shows the fit of the considered composite modules in the promoters to the differential expression of the corresponding genes.

2.5 Student’s $t$-test value calculation using fuzzy logic

This component is equal to the value of the two-sided Student’s $t$-test with different variances. Here we consider two alternative promoter sets, A and B, described above or, in the case of the alternative input of a single set of promoters with assigned expression values, the promoter set is divided into two groups A and B: with high and low differential expression values (in this case the value $P_0$ dividing these two groups is specified by the user).

After that, the $t$-test value $T$ is calculated:

$$T = \frac{\bar{n} - \bar{y}}{\sqrt{\sigma_x^2/n_1 + \sigma_y^2/n_2}} T_{\alpha/2}$$

Here, $nA_1, \sigma_r$, are the number of promoters in each group; an average value of the fuzzy score $\lambda$ ($fps$) in each group of promoters and the standard deviation, respectively. $T_{\alpha/2}$ is the 99.9% confidence interval values of the Student’s $t$-distribution for the corresponding number of degrees of freedom. If $T$ becomes higher than $T_{\alpha/2}$ then $T = 1.0$. The $T$ component of the fitness function shows the statistical significance of the difference between two distributions of composite module scores.

2.6 Specificity and sensitivity value

This component of the fitness function regulates the mean error rate. It equals to

$$E = \gamma(1 - FN) + (1 - \gamma)(1 - FP).$$

where FN is the false-negative rate, i.e. proportion of promoters in group A (with $\xi = 1$, or $\xi \geq P_0$ in the alternative input), getting $ps = 0$ (Equation 1), and FP is the false positive rate, i.e. proportion of promoters in group B (with $\xi = 0$, or $\xi < P_0$ in the alternative input), getting $ps = 1$. The constant $0 \leq \gamma \leq 1$ is defined by the user and gives relative impact of true positives versus true negatives. It is set to 0.5 by default.

2.7 Normality index of the distribution

This component shows how close the distribution of the fuzzy promoter score is to the normal distribution. Let $\lambda, \sigma$ be the average fuzzy score and their standard deviation for promoters of the group $A$; $p$ is the fraction of promoters whose fuzzy score belongs to $(-\infty; \lambda - \sigma) \cup (\lambda + \sigma; \infty)$. Then the $N$ component of the fitness function is defined as

$$N = 1 - |0.32 - p|.$$
population of ‘organisms’ subjected to mutations, recombination, selection and multiplication. Mutations are done by random exchange of matrices in the model into any other matrices from the library and by random small increase or decrease of the associated parameters such as cut-off values, impact values and others. Recombination is done by random exchange of the parts of chromosomes between different models. During a number of iterations the program tries to find the ‘best’ model that is characterized by the highest value of fitness function. It identifies the set of matrices, optimizes their cut-offs, the relative impact values, maximum number of best matches in promoters and some other optional parameters such as site orientation and distance range and in pairs. The output of the CMA tool is the best discriminative promoter model with the optimized parameters.

More details of the encoding schema of the models in the genetic algorithm and definition of the mutation and recombination operators as well as the selection and multiplication rules and other options of CMA program are described in the documentation of CMA (see e.g. http://www.gene-regulation.com/pub/programs.html#CMAnalyst).

3 RESULTS

3.1 Testing on the simulated data

First, we did initial testing of the ability of the program to reveal combinations of single matrices. To test this we performed a series of simulation experiments. We generated sets of 1000 random nucleotide sequences of a length of 1000 bp each, ‘implanted’ a certain number of site combinations in random positions of these sequences based on a predefined composite module and let the program ‘blindly’ reveal the composite module back or at least some parts of it.

First of all, we observed that with increase of the size of the library the accuracy of recognition (the percentage of correctly identified matrices as components of implanted composite modules) falls linearly (data not shown), but this can be generally recovered by increasing the number of iterations of the genetic algorithm. We also observed that the algorithm is able to recapture correctly not only CMs containing unique matrices but also modules comprising matrices that are similar to other matrices for other TFs characterizing by similar DNA binding patterns (e.g. VSGATA3_03 and VSMYOD_01). We found that the optimal parameters of the algorithm are the ratio of the population size to the number of iterations should be between 0.1 and 2; the optimal selection pressure is ~40% (so the best 40% of the population goes into the next generation) and the optimal mutation level is ~30% of population and recombination level ~60%. With these optimal parameters we performed further testing. The results of the testing are shown in Tables 1–3.

The result of this simulation shows that the CMA program is able to determine implanted matrices correctly in most cases, even if just a few sequences of set A contain all of the sites of the module (e.g. for P = 0.3, only about 8 sequences out of 300 contain all 3 sites).

Microarray data usually contain a high level of noise and reveal no clear differentiation between ‘changed’ and ‘unchanged’ genes. The expression values vary considerably. In order to test the ability of our method to deal with such kind of noisy data we generated test data with various ratios of differential expression values (e) and the random variance of the expression value A e (as a measure of noise).

<table>
<thead>
<tr>
<th>Table 1. Revealing combination of three single matrices (VSAP4_01, V$E2F_02, V$GATA1_02) in the window of 200 bp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site scores</td>
</tr>
<tr>
<td>-------------</td>
</tr>
<tr>
<td>High</td>
</tr>
<tr>
<td>Low</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

We implanted sites into 30% of the sequences (set A); other sequences in the set are taken as background (set B). Scores of the implanted sites are high (optimum to reduce false positives) or low (optimum to reduce false negatives). Sites were implanted probabilistically, so the probability that a site is implanted in a given sequence (P) varies from 0.3 to 0.9. +++ indicates that all implanted matrices were correctly identified back, ++ that only two were identified correctly and one was wrongly identified, in 100 iterations; population size was 200.

<table>
<thead>
<tr>
<th>Table 2. Testing the CMA program to analyze noisy expression data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression parameters</td>
</tr>
<tr>
<td>------------------------</td>
</tr>
<tr>
<td>Δe = 0.3</td>
</tr>
<tr>
<td>Δe = 0.4</td>
</tr>
<tr>
<td>Δe = 0.6</td>
</tr>
</tbody>
</table>

‘Changed genes’ comprised 30% of the random sequences. Sites were implanted in these sequences only (P = 0.9). Matrices were the same as in the Table 1. High cut-offs were used for implantation. We assigned to each ‘changed gene’ an expression value randomly generated from the interval [10, 10 + x]. Other ‘non-changed’ genes got the lower expression value from the interval [0,1]. By varying x = 5, 10, 15 we simulated 3 variants of the data with gradually increased noise. ‘Random fitness’—values of fitness function obtained in the shuffling experiments.

<table>
<thead>
<tr>
<th>Table 3. Testing the CMA program to reveal implanted pairs of matrices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters of the search of the CM structure</td>
</tr>
<tr>
<td>------------------------</td>
</tr>
<tr>
<td>1 pair</td>
</tr>
<tr>
<td>1–3 pair</td>
</tr>
<tr>
<td>0–1 pair, 0–2 single matrices</td>
</tr>
</tbody>
</table>

(++) indicates that the correct pair was found, ++ that two single matrices were revealed and they are correct components of the pair.

The result of this simulation (Table 2) shows that the CMA was able to reveal correct CM in cases of Δe < e and Δe ~ e, although the fitness is decreasing in the second case. In case of Δe > e the method was able to reveal correctly only two matrices used for the sites implantation.

In order to estimate the significance of the fitness values obtained in the analysis we have performed multiple shuffling experiments. In each such experiment we took all the expression values in the set and reassigned them to randomly chosen sequences. After that, we applied CMA to these shuffled samples and computed the average and standard deviation of the observed fitness values (Table 2, ‘random fitness’).

Next, we tested the functionality of the program on revealing pairs of matrices that reflect composite elements composed...
3.2 Analysis of promoters of co-regulated genes with the help of CMA

3.2.1 Test 1: T-cell specific genes. In order to check the ability of CMA to reveal known composite promoter modules we analyzed a set of promoters of T-cell specific genes that have been shown to be regulated by a very specific type of composite elements: NF-AT/AP-1. The set includes genes for several interleukins (IL-1,4,5,8); their receptors, signaling molecules such as TNF-alpha, IFN and others. It is based on the collection of experimentally proven NF-AT/AP-1 composite elements in TRANSCompel database (Kel-Margoulis et al., 2002a). In our earlier work (Kel et al., 1999) we showed that the promoters of these genes contain many copies of the NF-AT/AP-1 composite elements. Here, we would like to check if CMA would be able to reveal these composite elements without having explicit knowledge about their composition. Results of this test are given in Figure 1.

The (VSAP4_01@0.872000−VSNFAT_06@0.791000,3,1..14) line in the output describes a pair of matrices found by the system with all the parameters defined by the genetic algorithm. The values after the sign “@” give the cut-offs for the corresponding matrices; ‘<’ means that the orientation of the matches of the AP-1 matrix should have a definite orientation relative to the matches of the second matrix, whereas the orientation of the NF-AT is not fixed; values ‘3,1..14’ mean that the distance between matches of two matrices should be in the range from 1 to 14 bp and the maximal number of considered pairs in one window is limited to three pairs.

All these values found by the genetic algorithm correspond well to the known nature of NF-AT/AP-1 composite elements in T-cell specific promoters (Kel et al., 1999). The orientation of the matches was found correctly as well, since the crystal structure of the ternary complex of NF-ATp/AP-1-DNA reveals only one valid orientation of NF-ATp factor in this complex (Chen et al., 1998).

Another pair of matrices (V$PU1_Q6 − V$AP4_01) that was found by the program, in fact, corresponds to known composite elements P.U.I/AP-1 described in promoters of several genes regulating their expression in immune cells—macrophages (see TRANSCompel acc: C00251 in the promoter of mouse macrosialin gene). Interesting to note that consensus of NF-AT binding sites (AGGAAA) is similar to that of P.U.I binding sites (AGGAAC).

In Figure 2 we show the histogram of the fps score of the revealed composite module in promoters of T-cell specific genes in comparison with the randomly generated sequences. We see clear differentiation of these two sets of sequences. Although not in all promoters we can find sites for both matrix pairs, we still are able to identify the main components of the specific gene regulatory modules in the considered immune cells.

3.2.2 Test 2: 11 Kb upstream regions: T-cell specific genes versus housekeeping genes. In order to demonstrate the ability of the CMA algorithm to analyze long regulatory regions we compiled 5′ regulatory regions of the T-cell specific genes used in the test above. We extracted 11 Kb regions around TSS (10 Kb upstream and 1 Kb downstream). To test the algorithm on the real genomic data we have prepared a background set of 11 Kb upstream regions of 100 housekeeping genes from the list derived from an analysis of public gene expression data (Eisenberg and Levanon, 2003).

By running CMA we again revealed correctly the NFAT/AP-1 composite elements [with parameters of two modules slightly different from the test above: (VSAP1_Q400.871@<VSNFAT_0680.769,1,1..16) | (VSAP1_Q400.928@<VSNFAT_0680.834,2,1..16)]. In Figure 3 we show two histograms of the fuzzy promoter scores of the upstream regions of T-cell specific and housekeeping genes, respectively. The two distributions differ significantly, giving rise to the high fitness value = 0.738.

3.2.3 Permutation test. Since the method introduced here relies on the optimization of parameters of the model with the genetic algorithm there is a risk of overfitting. To evaluate the potential...
overfitting effect we performed a permutation test. We did 100 random shuffling of the grouping labels of the data used in the Test 2 (11 Kb upstream regions: 20 T-cell specific genes versus 100 housekeeping genes). Running the CMA algorithm 100 times with the randomized data resulted in the distribution of the obtained values of the fitness function that is shown in Figure 4.

As one can see from the distribution, the value of fitness function obtained in the analysis of T-cell specific promoters (0.738; see above) is much higher than expected by random chance. This provides a good evidence against the risk of overfitting and in favor of the realness of the promoter models revealed by the analysis.

3.2.4 Test 3: yeast cell cycle. In another example we performed an analysis of gene expression data on yeast cell cycle taken from Spellman et al. (1998). We selected five sets of genes according to
their expression in different cell cycle phases: genes specific for G₁, S phases and S/G₂, G₂/M transitional states. We retrieved promoters of the length 1100 bp (−1000, +100) for these genes and applied CMA program to find cell cycle phase-specific composite promoter models.

To estimate whether the results are consistent with the known facts we compared them with the experimental data on ChIP (chromatin immunoprecipitation) analysis and microarray gene expression data summarized in a recent publication (Kato et al., 2004). The parameters of the search were set to reveal a model, which consists of a single CM containing sites for 3–4 single matrices co-localized in a window of 200 bp. The total number of yeast matrices considered was 31 including all yeast matrices from TRANSFAC and several matrices derived from the DNA binding consensii for the corresponding factors published in Kato et al. (2004). In Table 4 we present the list of TFs whose weight matrices have been revealed by CMA as components of the best promoter models and compared these factors with the factors associated with the corresponding cell cycle phase in the mentioned paper (Kato et al., 2004).

Factors found by CMA for specific cell cycle phases are marked with ‘+’. Gray color of the table cells indicates this factor was shown experimentally to be involved in regulating genes in the corresponding cell cycle phases. Factors FKH1 and FKH2 are very similar to each other in their DNA pattern. That is why the results for these two factors are summarized together. One can see a very good agreement of the predictions made by CMA with the experimental knowledge.

4 CONCLUSION

In this paper we describe a novel method for the analysis and interpretation of gene regulatory regions. The method identifies composite modules—stable combinations of TF binding sites that are shared by the most of the co-regulated promoters. It is generally accepted that such modules are responsible for a function-specific regulation of transcription.

In comparison to most of the previously published approaches that consider combinatorics of TF binding sites, the method described here has several advantages, such as (1) capability to work with data of microarray experiments; (2) optimization not only of the matrix sets, but also of cut-off values for each matrix; (3) analysis of large regulatory regions; (4) search for pairs of matrices, selecting best distance and orientation.

Testing our method on simulated and real data has shown: (1) it is able to correctly reveal CMs that are overrepresented in the set of sequences; (2) it can be used to analyze data and propose factor combinations that are playing key roles in transcriptional regulation in the given biological context. In our previous work we have demonstrated that the combinatorial approach described here allows increasing significantly the precision of the computational prediction of target genes for TFs (Kel et al., 2004a; Kel et al., 2004b).
Application of this approach to the analysis of microarray gene expression data is very promising. The Composite Module Analyst is implemented now as a part of the commercial software system ExPlain™ that provides a wide range of tools and databases for causal interpretation of gene expression data.

ACKNOWLEDGEMENTS

The authors would like to thank all co-workers of the BIOBASE GmbH who have been contributing to the integration of CMA into the ExPlain™ system. Parts of the work were funded by grant of the German Ministry of Education and Research (BMBF) together with BioRegioN GmbH ‘BioProfil’, Grant No. 0313092; ‘Intergenomics’ Project, Grant No. 031U110C/031U210C; by European Commission under FP6-‘Life sciences, genomics and biotechnology for health’ contract LSHG-CT-2004-503568 ‘COMBIO’; by European Commission under ‘Marie Curie research training networks’ contract MRTN-CT-2004-512285 ‘TRANSISTOR’ and by INTAS Grant No. 03-51-5218. Funding to pay the Open Access publication charges for this article was provided by BIOBASE GmbH.

Conflict of Interest: none declared.

REFERENCES


