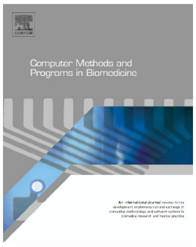

journal homepage: www.intl.elsevierhealth.com/journals/cmpb

Integrating biological knowledge based on functional annotations for biclustering of gene expression data

Juan A. Nepomuceno[a,*], *Alicia Troncoso*[b], *Isabel A. Nepomuceno-Chamorro*[a], *Jesús S. Aguilar-Ruiz*[b]

[a] *Departamento de Lenguajes y Sistemas Informáticos, Universidad de Sevilla, Avd. Reina Mercedes s/n, 41012 Seville, Spain*

[b] *Department of Computer Engineering, Pablo de Olavide University, Ctra. Utrera km. 1, 41013 Seville, Spain*

ARTICLE INFO

Article history:
Received 22 July 2014
Received in revised form
17 February 2015
Accepted 27 February 2015

Keywords:
Biclustering of gene expression data
Integration of biological knowledge
Scatter search

ABSTRACT

Gene expression data analysis is based on the assumption that co-expressed genes imply co-regulated genes. This assumption is being reformulated because the co-expression of a group of genes may be the result of an independent activation with respect to the same experimental condition and not due to the same regulatory regime. For this reason, traditional techniques are recently being improved with the use of prior biological knowledge from open-access repositories together with gene expression data.

Biclustering is an unsupervised machine learning technique that searches patterns in gene expression data matrices. A scatter search-based biclustering algorithm that integrates biological information is proposed in this paper. In addition to the gene expression data matrix, the input of the algorithm is only a direct annotation file that relates each gene to a set of terms from a biological repository where genes are annotated. Two different biological measures, FracGO and SimNTO, are proposed to integrate this information by means of its addition to-be-optimized fitness function in the scatter search scheme. The measure FracGO is based on the biological enrichment and SimNTO is based on the overlapping among GO annotations of pairs of genes. Experimental results evaluate the proposed algorithm for two datasets and show the algorithm performs better when biological knowledge is integrated. Moreover, the analysis and comparison between the two different biological measures is presented and it is concluded that the differences depend on both the data source and how the annotation file has been built in the case GO is used. It is also shown that the proposed algorithm obtains a greater number of enriched biclusters than other classical biclustering algorithms typically used as benchmark and an analysis of the overlapping among biclusters reveals that the biclusters obtained present a low overlapping. The proposed methodology is a general-purpose algorithm which allows the integration of biological information from several sources and can be extended to other biclustering algorithms based on the optimization of a merit function.

[*] Corresponding author. Tel.: +34 954559769.
E-mail address: janepo@us.es (J.A. Nepomuceno).

http://dx.doi.org/10.1016/j.cmpb.2015.02.010
0169-2607/© 2015 Elsevier Ireland Ltd. All rights reserved.

1. Introduction

Gene expression data matrices show the expression profile of thousands of genes along dozens of samples, which are studied in different microarray experiments. Each value in these matrices is a numerical value that represents the expression value of a gene in a specific sample. It is assumed that groups of genes that share a similar expression profile also share the same regulatory regime and hence the same functionalities. This assumption is called the guilt-by-association heuristic [1]. It is lately being reformulated because of the co-expression of a group of genes may be the result of an independent activation with respect to the same experimental condition and not due to the same regulatory regime is captured. Biclustering of gene expression data is an unsupervised machine learning technique that searches groups of genes with a similar expression profile under a subset of conditions. Recently, integration methods based on the combination of multiple sources from open-access data have been proposed in other fields as clustering or classification. These approaches can outperform the traditional algorithms and increase the possibilities of correcting the spurious information existing in high-throughput technology data as gene expression data or other “omic” data. The most important difference of biclustering with respect to traditional clustering is that biclustering aims to cluster simultaneously genes as well as conditions, rather than focusing solely on either one. Clustering techniques split the data matrix into groups of co-expressed genes along all samples in the matrix such that the union of all the clusters constitutes the complete matrix and all of them are disjoint. However, biclustering finds co-expressed genes only under a subset of samples. Therefore, the overlapping among results is considered and the motivation is to discover hidden patterns more than to describe the gene expression matrix. Note that some recently published traditional clustering algorithms allow the overlapping between clusters [2,3]. The goal of biclustering is to search groups of locally co-expressed genes more than to describe the gene expression matrix. The motivation is to find hidden patterns to discover potential biomarkers or formulate new hypothesis. Although biclustering was studied firstly in the 1960s [4] when it was proved to be a NP-hard problem, in the context of gene expression data it was firstly introduced by Cheng and Church [5].

In the context of biclustering, public databases and repositories such as *the Gene Ontology project* (GO) or *Kyoto Encyclopedia of Genes and Genomes* (KEGG) have been commonly used to validate the quality of biclusters from a biological point of view. Concretely, the enrichment analysis of a set of genes in the context of GO is usually used as a standard framework of comparison among biclustering algorithms [6]. The results are compared according to a ranking based on the characterization of each group of genes belonging to a bicluster with respect the information stored in GO. GO is an ontology with a hierarchical structure with three roots or domains: molecular function, biological process and cellular component. Each gene is related to a set of GO annotations with different levels of specificity. These annotations are terms in the ontology which are linked with groups of genes. These genes are annotated in the term. Low-level terms in the tree structure report more detailed information than high-level terms which are more general. Functional annotation files are built such that each gene is associated with the set of terms where it is annotated.

These open-access biological data are commonly used in biclustering literature to validate results and their use is a common factor in most of the papers. Many biclustering algorithms have been proposed using different heuristic strategies or search criteria to find biclusters [7–9]. Several algorithms such as Cheng and Church’s algorithm (CC) [5], Iterative Signature Algorithm (ISA) [10], Order-preserving Submatrix Algorithm (OPSM) [11] or xMotifs [12] are usually used as benchmark algorithms in order to establish a comparison among biclustering algorithms. CC was the foundational algorithm and it is based on a deterministic greedy iterative search method. This method finds biclusters with a residue less or equal than a threshold that is given as an input parameter. Although the residue is a measure usually used in biclustering [5], it cannot capture coherent evolution patterns [13]. The ISA algorithm uses a nondeterministic greedy algorithm that finds up- and down-regulated patterns. The input matrix is reordered to find blocks of coherent values with respect to rows and columns that are reported as biclusters. The OPSM algorithm searches for biclusters according to a model based on linear ordering among rows. This algorithm sequentially finds each bicluster and although coherent evolution patterns are captured, it cannot find inverse coherent evolution patterns. The xMotifs algorithm iteratively searches the largest bicluster according to some constraints by removing samples. Direct and inverse coherent evolution patterns are captured by this algorithm and a huge number of biclusters are usually reported. Moreover, it is important to highlight the family of biclustering algorithms based on evolutionary computation and metaheuristics that optimize a certain quality measure [14–17]. Likewise, algorithms of this family that use measures based on correlations among genes as a mechanism to find co-expressed genes have been recently published in literatures [18–26].

All the above-mentioned algorithms search biclusters composed of co-expressed genes and the biological knowledge is only used as a posteriori criterion to determine the relevance of the biclusters found. However, the author in [27] considered that the studies based on gene expression data had some limitations. In particular, the co-expression of a group of genes may be the result of a parallel and independent activation with respect to the same experimental condition and not due to capture the same biological functionality. Therefore, the assumption that co-expression means co-regulation should be reinterpreted. For this reason, the biological knowledge has been incorporated during the search process to avoid groups of co-expressed genes that do not show biologically representative connections. For example in the field of clustering, the algorithm presented in [28] uses the K-means algorithm and integrates gene annotation files extracted from GO with a concept of distance based on the co-expression and functional similarity. A GO-based measure has been also applied as part of the workflow of a predictive algorithm to classify genes in [29]. Namely, the average of the Pearson correlation, which evaluates similarities among gene expression profiles, and a GO-based measure are used to define a distance. This distance is used to cluster genes and to elaborate a ranking of candidate genes. A gene feature selection method is presented in [30] which integrates information from KEGG instead of from GO. This algorithm is a KEGG-improved evolutionary strategy that shows a better performance for selecting genes than classical algorithms.

Functional similarity measures based on GO involve a concept of distance between two genes. There are several semantic similarity measures to compare GO terms to which genes are annotated [31]. They can be basically classified in two groups: edge-based measures and information content (IC)-based measures. The first group of measures assumes that the specificity of a term can be directly inferred from its depth in the GO graph and the second group is based on the frequency of a term in the GO graph. However, a gene is usually annotated in several terms and not in only one, and hence, it is better to use similarity measures that compare sets of terms rather than single terms. A comparison among this kind of measures is presented in [31] where the *simGIC* measure shows the best performance. This measure computes the similarity between two genes using the IC associated to each term in the genes. Note that the GO graph structure is needed to compute the *IC* in addition to the gene annotation file. Also there are other measures based on a preprocessed binary matrix instead of the tree structure of GO. This matrix is composed of genes as rows and GO terms as columns and the elements are 1 or 0 depending on if the gene is or is not annotated in the GO term, respectively. These measures are based on the Vector Space Model (VSM) originally developed in the context of information retrieval. The measure presented in [32] can be also classified as a semantic similarity measure but neither a preprocessed matrix nor the GO graph structure are necessary. This measure is based on the overlapping among gene annotations from flat GO annotation files where the tree structure of GO is captured. In particular, the annotations for each gene are propagated to upper levels in the ontology and all the associated parent terms are considered.

It can be stayed that the integration of biological information from different sources is actually one of the challenges and research directions in Bioinformatics [33]. There are many works that use other data sources, not only information from GO or KEGG. For example, several data sources can be merged to integrate biological knowledge as protein–protein interaction networks, genome-wide binding data and information from the literature and not only information from GO [34]. The COALESCE algorithm [35] uses the gene expression data together with DNA sequence data as input data and other supporting data as additional information during the process in order to discover regulatory modules. This algorithm finds biclusters that are used as a guide to discover these modules. The proposed algorithm in [36] uses protein–protein interaction networks and gene expression data to discover cancer biomarkers, which are found through the discovering of groups of genes in biclusters that are also highly connected in the network. Recently, a biclustering algorithm [37], which works with microRNA and target genes data, uses GO information to do a ranking of the results.

From the best of our knowledge, the biological information integration in biclustering of gene expression data has not been still investigated. The aim of this paper is to introduce this idea in the biclustering field by means of functional annotation files. These files are flat files where each gene is linked with its corresponding terms in the biological repository. The proposed algorithm is a scatter search-based metaheuristic that optimizes a fitness function which defines a criterion to evaluate the quality of the biclusters. This algorithm is based on the algorithm presented in [25] and although several procedures differ, the most relevant contribution is the fitness function definition to integrate the biological information. This function consists of three parts: the first one is a term to control the size of the biclusters; the second one is the correlation among genes to capture co-expressed genes; finally, the third term is an additional term to integrate the biological information. The goal is to establish an equilibrium in the fitness function to find biclusters composed of co-expressed genes that capture similar biological functionalities. Additionally, two different biological integration possibilities are experimentally studied: firstly, the biological information is included with a measure proposed here based on an enrichment study of a set of genes, and secondly, with a GO-based measure that computes the overlapping among the annotated terms of a group of genes. Both measures only use as input data to integrate the functional information an annotation file that relates genes and biological terms. Therefore, the input data of the algorithm are only the gene expression matrix and a functional annotation file. The proposed methodology is a general-purpose algorithm which allows the integration of biological information from several sources such as GO, pathways KEGG or any biological information provided by flat annotation files and can be extended to other biclustering algorithms based on the optimization of a merit function.

The remainder of this paper is organized as follows. Section 2 presents the algorithm where firstly the fitness function is defined and secondly the search procedure is described. In the fitness function section, two different biological integration measures are also defined. Experiments are described and discussed in Section 3. Namely, the experimental results are analyzed from three points of view: the integration of biological information improves the algorithm performance; the performance of our approach is better than that obtained by several benchmark algorithms; and the results from the two different biological integration measures are discussed. Finally, Section 4 is devoted to conclusions and future work.

2. Methodology

The proposed methodology is divided into two phases clearly differentiated. In a first step, a fitness function integrating biological knowledge from functional annotation files is designed to measure the quality of biclusters in Section 2.2. A search process based on a scatter search algorithm is applied by minimizing the measure provided by this fitness function in Section 2.3. In a sense, the methodology separates the searching and the characterization of the biclusters to be found. That is, the search process is independent of the established criterion, which is defined by the fitness function, to find biclusters.

2.1. Input data

The input data of the algorithm are basically the gene expression matrix and a direct annotation file. The gene expression matrix is composed of gene expression profiles and samples that are represented in rows and columns, respectively. Each element in the matrix represents the level of expression of a gene under a particular sample or experimental condition. Direct annotation files are flat files where each line is constituted by a gene with a set of terms from a biological repository. These terms are labels for a determined biological functionality where the gene is involved. For example, this sentence *TVP15 GO:0006810,GO:0016192* is a line in a direct annotation file extracted from GO, where *TVP15* is the gene name and *GO:0006810* and *GO:0016192* are two GO terms where this gene is annotated. Note that direct annotation files may be downloaded from repositories in several ways. Nevertheless, the proposed methodology is a general-purpose algorithm which allows the integration of biological information from several sources of information. Additionally, the number of biclusters to obtain is also provided as an input parameter.

2.2. Fitness function

The main goal is to define a fitness function that integrates biological knowledge to find biologically relevant biclusters, in addition to other measures such as the correlation to find biclusters with enclosed interesting patterns and the volume to find non-trivial biclusters.

The gene expression matrix can be seen as a numerical matrix D where an element (i, j) is the expression level of gene i under the sample or condition j. A bicluster B is a submatrix of D with N genes and M conditions, that is, $B = \{(g_i, c_j)\}_{i,j}$ where $i \in \{1, \ldots, N\}$ and $j \in \{1, \ldots, M\}$. The proposed fitness function to evaluate B is defined as follows:

$$f(B) = M_1 \cdot f_1(B) + M_2 \cdot f_2(B) + M_3 \cdot f_3(B) \quad (1)$$

where f_1 measures the volume of the bicluster, f_2 the patterns found in the bicluster and f_3 the quality of the bicluster from a biological view point, and M_1, M_2 and M_3 are parameters to weight the relevance of the measures f_1, f_2 and f_3, respectively.

The measure f_1 is used to control the volume of bicluster. It is defined as

$$f_1(B) = \frac{1}{N \cdot Q} \quad (2)$$

where N is the number of genes and Q the number of conditions in the bicluster B. This term is important to deal with the size of biclusters during the search process and to avoid irrelevant information [38]. It is used to avoid finding trivial biclusters with only a very low number of genes or conditions, as for example a bicluster with only two conditions.

The measure f_2 is based on the average correlation among the genes of the bicluster. Note that only the conditions in the bicluster are considered and not all conditions in the gene expression matrix. This term is considered in order to capture most of the relevant patterns in biclusters such as shifting and scaling patterns [13] or activation-inhibition patterns [22]. The correlation has been previously used as merit function to determine co-expressed genes and it evaluates the grade of dependence among genes [25]. It is defined as follows:

$$corr(B) = \frac{1}{\binom{N}{2}} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} |\rho_{ij}| \quad (3)$$

where ρ_{ij} is the Pearson correlation coefficient between the genes g_i and g_j. Note that only $\binom{N}{2}$ elements have been contemplated due to the symmetry of the correlation coefficient. The absolute value is considered to avoid that groups of genes with high positive correlation values could eliminate the effect of groups of genes with high negative correlation values. It is noteworthy to mention, the average correlation evaluates a bicluster considering both genes and conditions, that is, two biclusters with the same set of genes but a different set of conditions provide different values for the correlation. As the scatter search is applied to minimize the fitness function f, the function f_2 can be defined as

$$f_2(B) = 1 - corr(B) \quad (4)$$

The measure f_3 is based on a direct annotation file from a biological knowledge repository, and hence, it determines how the biological information is integrated in the process. Note that the term $f_3(B)$ evaluates the set of genes of the bicluster B but not the conditions, that is, f_3 has the same value for two biclusters with the same set of genes and different sets of conditions. Therefore, f_3 evaluates the biological relevance of a bicluster but it cannot find interesting patterns in a bicluster and it cannot differentiate biclusters with the same set of genes. Two different functions, which use only as biological data source a direct annotation file with the information for the genes in the gene expression matrix, are proposed for the measure f_3 in the following Sections 2.2.1 and 2.2.2.

2.2.1. Fractional Gene Ontology measure

A measure based on the analysis of the enrichment of a bicluster [39] is proposed in this paper to measure the biological relevance of a bicluster. The measure is the proportion or fraction of genes in a bicluster associated to enriched GO terms, hereinafter *FracGO*. The terms with an adjusted p-value under a given threshold are said to be enriched or overrepresented. This threshold is known as significance level and is usually set to 0.05. The adjusted p-value for each annotated term in the annotation file is computed with respect to the group of genes that belong to the bicluster. The universe of genes is the complete set of genes in the gene expression matrix. Fisher's exact test has been used to determine statistically overrepresented terms and Bonferroni test has been used to correct the adjusted p-values for multiple comparisons as the number of hypotheses tested is the number of terms in the annotation file. Thus, *FracGO* is defined as follows:

$$FracGO(B) = \begin{cases} 0 & \text{if} J = 0 \\ \dfrac{1}{J \cdot N} \displaystyle\sum_{i=1}^{J} x_i & \text{if} J \geq 1 \end{cases} \tag{5}$$

where *J* is the number of enriched GO terms, namely the number of GO terms with an adjusted *p*-value less than 0.05, *N* is the number of genes in the bicluster and x_i is the number of genes of the bicluster that presents the GO term *i* in the annotation file. Note that *FracGO* is equal to 1 if all the genes of the bicluster are associated with all enriched GO terms ($x_i = N$, $\forall i = \{1, \ldots, J\}$) and 0 when there is not an enriched GO term. It can be concluded that *FracGO* has a value of 1 for biologically relevant biclusters and 0 otherwise.

In this case, the proposed function f_3 can be directly defined as

$$f_3(B) = 1 - FracGO(B) \tag{6}$$

It can be noted that the bicluster *B* is considered a high-quality bicluster if f_3 is equal to 0 and a bad bicluster if its value is set to 1.

2.2.2. *Normalize term overlap measure*

Several gene pairwise GO-based measures have been proposed in the literature. These measures compute the similarity between two genes based on their GO annotations. The simNTO measure, which is based on the *term overlap* defined in [32], only uses annotation files as input. This measure is faster and simpler than other measures based on information content (IC). These IC-based measures use the GO tree structure as additional information along with the annotation file. SimNTO captures the GO hierarchical structure if the annotation files are built by containing all parent terms for each term.

A measure based on the functional similarity of a bicluster using simNTO measure is proposed as a measure to evaluate the biological relevance of a bicluster in this work. This measure is defined by means of the average overlapping between the pairs of genes of a bicluster according to the information from the GO annotation file. That is,

$$SimNTO(B) = \frac{1}{\binom{N}{2}} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} simNTO(g_i, g_j) \tag{7}$$

where *N* is the number of genes in the bicluster *B* and *simNTO* is the normalized term overlap measure proposed in [32]. In particular, *simNTO* measures the overlapping between two genes g_1 and g_2 with respect to GO terms and is defined as

$$simNTO(g_1, g_2) = \frac{|annot_{g_1} \cap annot_{g_2}|}{\min(|annot_{g_1}|, |annot_{g_2}|)} \tag{8}$$

where $annot_{g_i}$ is the set of GO terms associated to the gene g_i and $|\cdot|$ is the number of elements of a set. It is important to mention that the direct annotation file must be built by propagating the terms toward the upper levels in the ontology to compute this measure. Therefore, $annot_{g_i}$ is defined by considering the set of all direct annotations of the gene g_i and the associated parent terms in the ontology, excluding the root of the hierarchy. Accordingly, the annotation file must capture the GO tree structure. Note that the *simNTO* measure ranges from 0 to 1. Two genes share the same annotations and are very similar in the ontology if *simNTO* has a value of 1, and 0 otherwise. Moreover, if one gene *g* is not contemplated in the annotation file because there is not any information in GO, $annot_g$ is the empty set, and in this case, the *simNTO* measure is directly set to 0.

In this case, the proposed function f_3 is defined as follows:

$$f_3(B) = 1 - SimNTO(B) \tag{9}$$

It can be appreciated that $f_3(B) = 0$ when *B* is composed of a group of genes that share similar biological functionalities in GO, and therefore, *B* is a relevant bicluster according to GO.

2.3. *Description of the algorithm*

An algorithm based on a metaheuristic scheme has been considered due to the computational nature of biclustering. The search scheme derives from the algorithm presented in [25] where a scatter search was also applied in biclustering of gene expression data. The proposed algorithm is a sequential covering algorithm, namely, each bicluster is obtained by applying an independent scatter search procedure. This iterative process, jointly with the bias introduced in the search through the fitness function definition, controls the non-deterministic nature of the process. The scatter search is a population-based evolutionary metaheuristic where a population of solutions evolves until an optimal solution is reached. The basic idea in the scatter search is to perform the optimization with a small set of solutions, called the *reference set*, instead of a complete population of solutions as in other population-based metaheuristics [40]. The *reference set* is composed of the best solutions according to intensification and diversity strategies. Fig. 1 shows the basic idea behind the proposed algorithm. All the steps composing this algorithm are going to be briefly described in Sections 2.3.1 and 2.3.2.

2.3.1. *Scatter search*

A solution represents a bicluster, which is codified by two binary strings where the bits indicate if the gene or condition is present or not in the bicluster. Firstly, an *initial population* is generated by the *diversification generation method* and is composed of solutions as scatter as possible, which are then improved by a local search procedure that will be described in Section 2.3.2. Contrary to the genetic algorithms the initial population is built following a mechanism to achieve diversification and not a general randomization process. The *diversification generation method* generates a collection of solutions from a seed solution. If *x* is a binary string used as seed, a new string *x*′ is generated for each value of an integer *h* = 1, 2, 3, . . ., h_{max} as follows:

$$x'_{1+kh} = 1 - x_{1+kh} \text{ for } k = 0, 1, 2, 3, \ldots, \lfloor n/h \rfloor \tag{10}$$

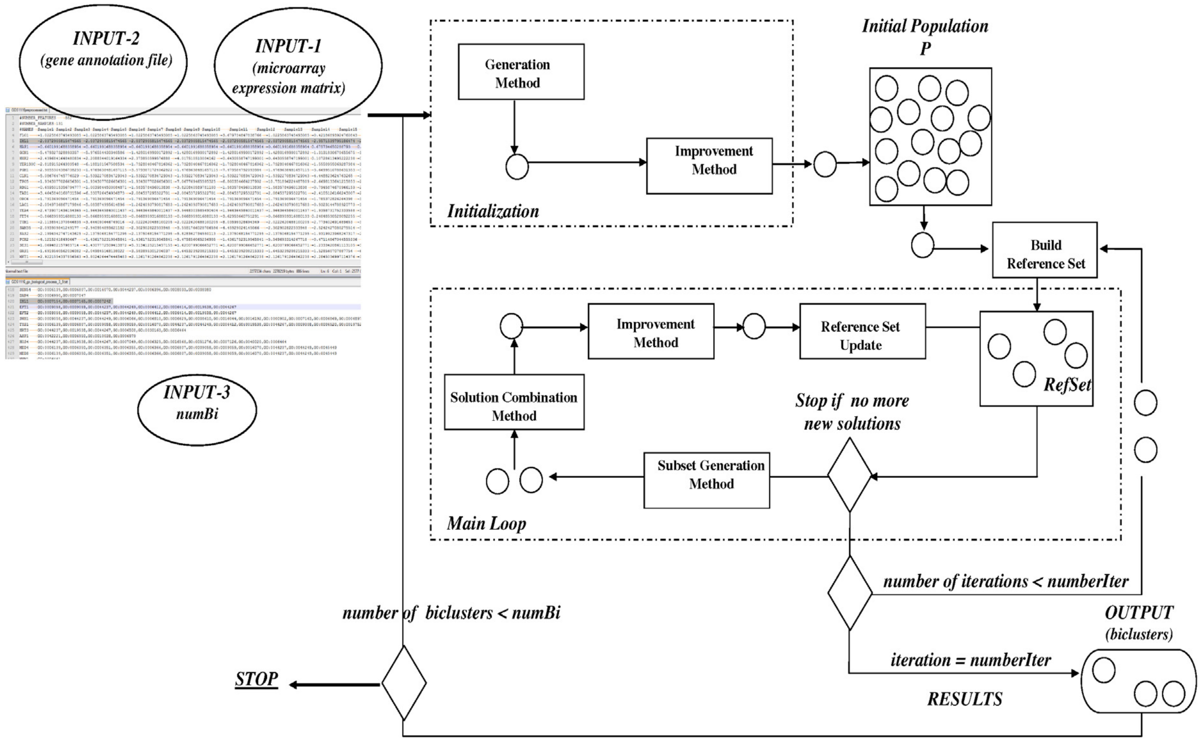

Fig. 1 – The proposed scatter search for biclustering.

where $x=(x_1, \ldots, x_n)$, n is the number of bits, k takes values from 0 to the largest integer satisfying $k \leq n/h$. Due to the binary string size and to the scatter search literature recommendations [40], the maximum value for h is $h_{max} = n/5$. All the remaining bits of x' are equal to those of x. After generating all the possible solutions with that seed, if more solutions were needed in the initial population, the rule will be applied again using the last solution as a new seed. This method is the standard procedure usually used in scatter search algorithms with binary codification [40]. The idea of scatter between two solutions is introduced by the *Hamming distance*. The Hamming distance between two binary strings is defined as the number of positions at which the corresponding 0's and 1's are different.

Once the initial population is generated, the reference set is built by the *build reference set method*. This method consists in selecting the five best solutions and the five most scattered solutions with respect to the remaining of existing solutions in the set from the initial population. Therefore, the reference set contains the most representative solutions from the initial population according to quality and diversity criteria. The scatter solutions provide diversity in the search process to avoid local optima. This diversity strategy plays a similar role to the mutation operators in genetic algorithms, for example. On the other hand, the quality solutions are considered as mechanism to define an intensification strategy in the search. It is important to update the initial population by removing the solutions selected by this method.

The reference set evolves by using the *subset generation method*, the *solution combination method* and the *reference set update method* until the reference set is stable. Once the reference set does not change, the reference set is rebuilt with the five best solutions from the previous reference set and the five most scattered solutions with respect to the remaining solutions in the set from the initial population. The *subset generation method* generates subsets of pairs of solutions from the reference set to be combined by the *solution combination method* with the purpose of creating new solutions. Once the new solutions are obtained, the local search procedure is again applied to improve them. The *solution combination method* is based on the uniform crossover operator commonly used in evolutionary computation. The *reference set update method* consists in choosing the 10 best solutions, according to the fitness function, from the joining of the solutions of the reference set and the new solutions obtained by the *solution combination method*.

The output is the best bicluster in the last reference set. The whole process is repeated as many times as number of biclusters to be obtained, which is an input parameter of the algorithm. No mechanism of redundancy control among results has been considered and it could be thought that the same bicluster is always found. Several works based on metaheuristics include mechanisms of redundancy control but these algorithms usually use only an initial population in the complete process. Concretely, the algorithm published in [41] presents an additional term in the fitness function to control the redundancy in order to avoid similarities among biclusters and to find repeatedly the same bicluster. However, if a different initial population is built for each bicluster obtained by the proposed algorithm, this additional term can be removed and it is enough filtering those highly redundant biclusters. Moreover, it is interesting to obtain biclusters sharing groups of genes from a biological point of view [2,3]. Consequently, the proposed algorithm does not contemplate an additional term in the fitness function and builds an initial population for each found bicluster. An overlapping threshold of 30% is chosen and the biclusters with an overlapping greater than this threshold are removed.

The parameters of the algorithm have been chosen according to recommendations of the literature of scatter search [40] and previous works [25]. In particular, 200 for the size of the initial population, 10 for the size of the reference set and 20 for the number of iterations of the evolutionary process (the inner loop in Fig. 1).

2.3.2. *Improvement method*

The improvement method is a local search procedure designed to improve the quality of solutions according to the fitness function. Namely, given a bicluster this method generates a new bicluster with a better value for its fitness function. In general, the improvement method in a scatter search is specifically designed for each problem as it depends on the nature of the fitness function [25]. Although an improvement method guided by a heuristic is better than a blind improvement method, the heuristic to improve the quality of biclusters is related to the fitness function of the problem. In this work, several different fitness functions are analyzed, namely one for each measure proposed to integrate biological knowledge, and therefore, the same heuristic is not good for all the fitness functions. For this reason, a blind improvement method is proposed with the purpose of being used with all them. This independence of the improvement method regarding the fitness function motivates a blind search among solutions close to the original solution. Sometimes this search does not find a better solution, and therefore, in these cases the original solution is not improved. This method plays an important role to speed up the convergence of the search.

The improvement method aims at selecting the best bicluster from a certain number of new biclusters generated by the combination of different binary strings. These new solutions are generated by means of permutations (see Fig. 2) in order to be close (in the sense of the hamming distance) to the original bicluster/solution. Note that the new solutions must improve the original solution but in the same “neighborhood”. If new biclusters do not improve the search, the output is the original bicluster. Fig. 2 shows how binary strings are combined. These binary strings have been obtained from the binary strings of the original bicluster to be improved. In particular, each bit of the binary string of the original bicluster is analyzed and if the bit is set to 0 then the bit of the new binary string is also set to 0 and if the bit is 1, then the following four cases are considered:

- Case 1: The next bit is changed to 1 and the current bit does not change its value.
- Case 2: The next bit is changed to 1 and the current bit is changed to 0.
- Case 3: The previous bit is changed to 1 and the current bit does not change its value.
- Case 4: The previous bit is changed to 1 and the current bit is changed to 0.

Twelve new biclusters have been generated by applying this method. If the new biclusters do not improve the value of the fitness function of the original bicluster, the output of the improvement method is the original bicluster.

3. Experiments

The goal of the experiments is to analyze how the integration of biological information has a relevant influence on the performance of the proposed algorithm. In particular, the results obtained from FracGO and SimNTO measures proposed here in order to integrate the biological knowledge are compared. Finally, the results obtained by the proposed algorithm are compared with those of the classical biclustering algorithms such as ChCh [5], ISA [10], OPSM [11] and xMotifs [12] typically used as benchmark in the literature. The standard comparison methodology among biclustering algorithms is usually based on the gene enrichment in the obtained biclusters [6]. In this work, a bicluster is said to be enriched if at least one enriched GO term is associated to it.

This section presents the experiments carried out to assess the performance of the proposed algorithm on the datasets described in Section 3.1. The results are gathered in Section 3.2 and a discussion can be found in Section 3.3.

3.1. *Data sets*

Two yeast datasets with accession numbers *GDS*1116 and *GDS*2914 from the GEO repository [42] have been used in the experimentation. The first one is composed of 7085 expression profiles and 131 samples and recollects the genetic variation in the gene expression between parents and progenies from a cross of two different kinds of yeast strains. The second one has 15,488 expression profiles and 36 samples and is a time course experiment in which yeast cells dealt with a low dose of caffeine are analyzed. The raw data have been processed with the Babelomics web tool [43]. For both datasets, the expression profiles with more than a 30% of missing values have been filtered and the remaining missing values have been replaced with the mean of the values in the profile. The profiles, which appear several times but representing the same gene, have been summed up by means of the median of the values. After processing the raw data, *GDS*1116 expression matrix is a matrix composed of 882 genes and 131 samples or experimental conditions and *GDS*2914 a matrix of 975 genes and 36 conditions.

Biological information is provided by a direct annotation file from biological process domain (BP) of Gene Ontology (GO). This file shows the GO terms associated with each gene of the data set. Note that this information could be downloaded from GO in different ways. In this work, the tree structure of GO is considered and the annotation file has been obtained by propagating the annotations to the upper levels in the ontology. That is, each gene is also related to GO terms corresponding to all parent nodes until the root node. The annotation files for both datasets have been generated by the Babelomics tool with default options. For *GDS*1116 data set, 632 genes are annotated in the file from the 882 genes of the expression matrix. This file contains 245 different GO terms and the average number of GO terms per gene is equal to 10.6. For *GDS*2914 data set, 658 genes are annotated from a total number of 975 genes, the file

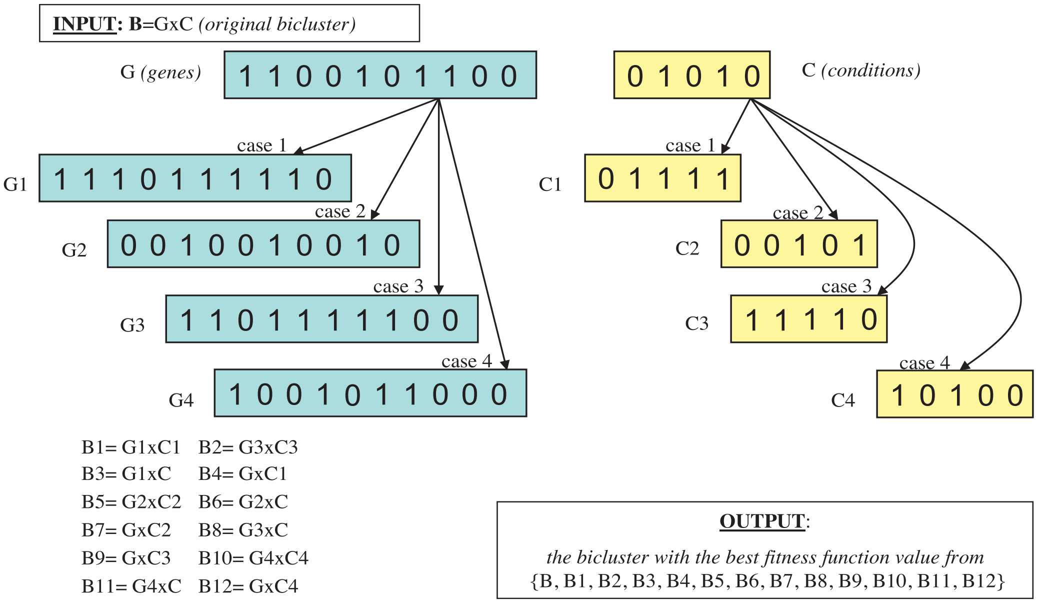

Fig. 2 – The proposed improvement method: a test example.

contains 256 GO terms and the average number of GO terms per gene is 10.1.

3.2. *Results*

In this section the results are gathered with the purpose of evaluating the importance of integrating biological knowledge to find high-quality biclusters. The results obtained from the proposed algorithm for different configurations for the fitness function are presented and compared with respect to several typical variables such as the size of biclusters, overlapping among biclusters and enrichment of biclusters. Also, the results obtained from different repositories such as GO, KEGG pathways and InterPro are presented. Each run of the algorithm obtains 100 biclusters. This number has been chosen because to obtain a high number of biclusters is desirable due to the comparison among biclustering algorithms is usually established in terms of percentages of enriched biclusters. Note that the algorithm obtains each bicluster through an independent non-deterministic procedure. Table 1 presents the percentage of enriched biclusters obtained by the proposed algorithm for different numbers of biclusters (namely, 10, 50 and 100). It can be observed that the percentage does not depend on the number of biclusters considered as input parameter. The fitness function parameter setting is experimental studied to analyze the performance of the biological integration measures.

Table 2 summarizes the quality of biclusters obtained by the proposed algorithm for *GDS*1116 and *GDS*2914 datasets when different configurations of the fitness function have been considered. Specifically, the column *Measure*, indicates if the fitness function takes into account biological information by means of the measures SimNTO (Eq. (7)) or FracGO (Eq. (6)) or not ($f_3 = 0$ in Eq. (1)). The column *Parameters* represents the weight corresponding to each term of the fitness function. It is important to highlight that all terms f_i vary between 0 and 1.

Given the importance of avoiding trivial biclusters with a low number of genes or conditions, for example only two genes or conditions, the parameter M_1 associated to the volume of the bicluster always has been set to 2 [25]. Namely, the following three configurations have been analyzed when the biological knowledge is integrated in the fitness function: to find biclusters with underlying correlated patterns and to find biological high-quality biclusters are equally important ($M_2 = 1$, $M_3 = 1$), to find biclusters with correlated patterns is more important than to find biological high-quality biclusters ($M_2 = 2$, $M_3 = 1$) or vice-versa ($M_2 = 1$, $M_3 = 2$); when the algorithm searches for biclusters without a priori biological knowledge ($M_3 = 0$), two configurations have been analyzed depending on the relevance of finding biclusters with enclosed patterns ($M_2 = 1$ or $M_2 = 2$). Note that M_1 cannot be equal to zero to avoid trivial biclusters [25]. On the other hand, if $M_2 = 0$ the number of conditions is not correctly controlled by the fitness function, and therefore, the algorithm is not a biclustering algorithm because can cluster genes but not conditions. Finally, $M_3 = 0$ is considered when biological information is not considered.

The next columns in Table 2, *size*, *enriched biclusters (%)*, *GO terms per bicluster* and *time*, show the average number of genes and conditions for 100 biclusters obtained by the proposed algorithm, the percentage of enriched biclusters, the average number of enriched GO terms per bicluster and the computation time of the algorithm to obtain a bicluster, respectively. Although the enrichment or biological significance of GO terms is usually studied regarding the biological process (BP) domain of GO, in this work the biological significance has been also studied for molecular function (MF) and cellular component (CC) domains. Note that the percentage of enriched biclusters is the standard biological evaluation criterion commonly used in biclustering [6,44]. It should also

Table 1 – Percentage of enriched biclusters obtained by the proposed algorithm for different numbers of biclusters.

Fitness function configuration	Number of biclusters	(M_1, M_2, M_3)	Enriched biclusters (%)		
			BP	MF	CC
1-SimNTO	10	(2,1,1)	100	100	100
1-SimNTO	50	(2,1,1)	100	98	100
1-SimNTO	100	(2,1,1)	99	97	97
1-FracGO	10	(2,1,1)	100	70	50
1-FracGO	50	(2,1,1)	100	72	64
1-FracGO	100	(2,1,1)	100	73	67
0	10	(2,1,0)	85	82	88
0	50	(2,1,0)	82	72	84
0	100	(2,1,0)	85	82	88

Table 2 – Results obtained by different fitness function configurations for both datasets. Each row shows a run that obtains 100 biclusters.

Dataset	Fitness function		Size	Enriched biclusters (%)			GO terms per biclus. (BP)	Time (s)
	Measure f_3	Parameters (M_1, M_2, M_3)		BP	MF	CC		
GDS1116	1-SimNTO	(2, 1, 1)	(11.6 × 15.6)	99	97	97	5.74	28.65
		(2, 1, 2)	(8.7 × 15.2)	87	75	84	3.67	15.78
		(2, 2, 1)	(10.1 × 5.1)	95	83	89	4.5	24.03
	1-FracGO	(2, 1, 1)	(181.6 × 19.4)	100	73	67	1	5410.98
		(2, 1, 2)	(193.9 × 19.3)	100	79	79	1	5546.12
		(2, 2, 1)	(109.2 × 3)	100	47	52	1	5682.65
	0	(2, 1, 0)	(23.5 × 14.7)	85	82	88	2.16	38.29
		(2, 2, 0)	(46.8 × 3.2)	25	17	21	0.4	11.51
GDS2914	1-SimNTO	(2, 1, 1)	(10.9 × 10.9)	87	33	33	5.45	27.76
		(2, 1, 2)	(8.0 × 10.2)	65	24	33	2.94	15.67
		(2, 2, 1)	(9.5 × 3.3)	87	25	30	5.39	22.18
	1-FracGO	(2, 1, 1)	(180.6 × 15.1)	100	97	94	1.01	5420.84
		(2, 1, 2)	(193.2 × 15.8)	100	98	94	1	5920.09
		(2, 2, 1)	(102.8 × 3)	100	72	67	1	6311.43
	0	(2, 1, 0)	(24.1 × 9.0)	2	5	6	0.02	13.14
		(2, 2, 0)	(37.1 × 3.1)	13	15	17	0.19	7.16

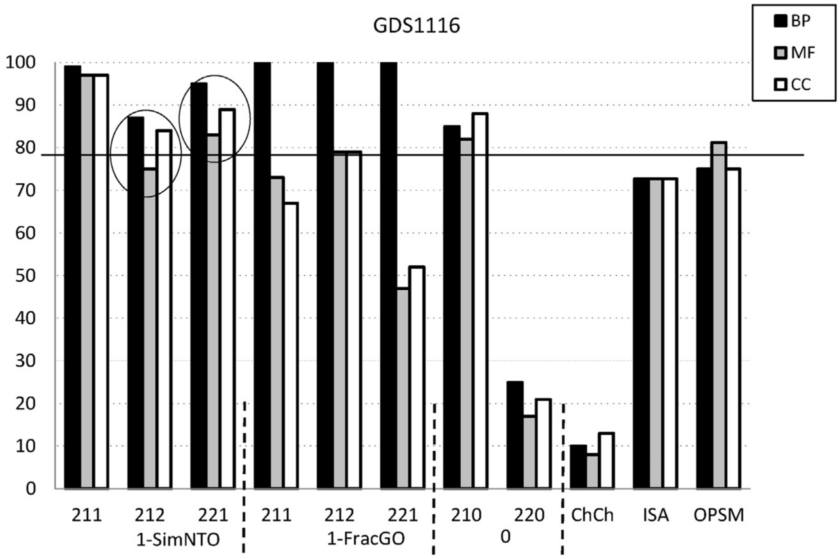

Fig. 3 – Percentage of enriched biclusters for GDS1116 from Tables 2 and 3.

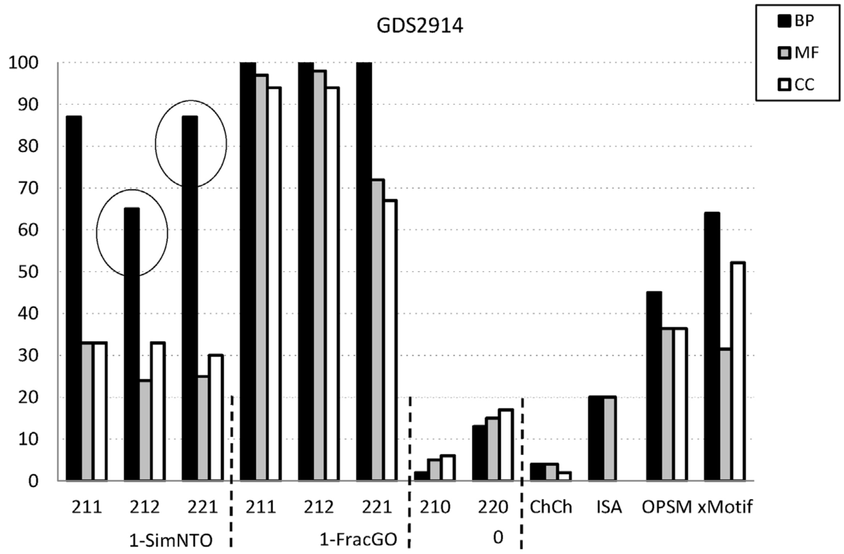

Fig. 4 – Percentage of enriched biclusters for GDS2914 from Tables 2 and 3.

Table 3 – Results obtained by classical biclustering algorithms for both datasets.

Dataset	Algorithm	Number of biclusters	Size	Enriched biclusters (%)			GO terms per bicluster (BP)
				BP	MF	CC	
	ChCh	100	(21.9 × 18.4)	10	8	13	0.30
GDS1116	ISA	11	(50.7 × 6.5)	72.7	72.7	72.7	6.45
	OPSM	16	(128.1 × 10.4)	75	81.2	75	20.06
	xMotifs	–	–	–	–	–	–
	ChCh	100	(17.2 × 8.8)	4	4	2	0.04
GDS2914	ISA	5	(28.6 × 2)	20	20	0	0.60
	OPSM	11	(164.4 × 7.2)	45.45	36.4	36.4	28.64
	xMotifs	999	(61.2 × 5)	64.76	31.5	52.15	1.13

be noted that the average number of enriched GO terms is computed over all the biclusters and not only over those that are enriched. Figs. 3 and 4 show the percentage of enriched biclusters reported in Tables 2 and 3 for *GDS*1116 and *GDS*2914 datasets, respectively. The black bar represents the BP domain and the gray and the white bars depict the MF and CC domains, respectively. Note that 211, 212 and 221 represent $(M_1 = 2, M_1 = 1, M_3 = 1)$, $(M_1 = 2, M_1 = 1, M_3 = 2)$ and $(M_1 = 2, M_1 = 2, M_3 = 1)$, respectively.

Figs. 5–7 present the size of biclusters obtained by the proposed algorithm for *GDS*1116 dataset for three configurations analyzed when the biological knowledge based on the SimNTO and FracGO measures has been integrated. A point represents a bicluster, where the number of genes is represented in the *x*-axis and the number of conditions is represented in the *y*-axis. Table 4 shows the average, variance, maximum and minimum number of genes and conditions for the aforementioned figures.

Figs. 8 and 9 show the overlapping among 100 biclusters obtained by the fitness function based on the SimNTO measure for *GDS*1116 dataset when biologically relevant biclusters are preferred versus biclusters with enclosed shifting and scaling patterns. In particular, each element in the matrix is the percentage of overlapping between two biclusters defined by the proportion of genes and conditions belonging to both biclusters. Although the proposed algorithm does not incorporate a control of the overlapping, it can be noticed that all biclusters have an overlapping less than 30% because of the reference set is always rebuilt by adding the most scattered solutions. A range of similar values for the overlapping has been obtained by the fitness functions based on the FracGO measure and the remaining configurations of the parameters.

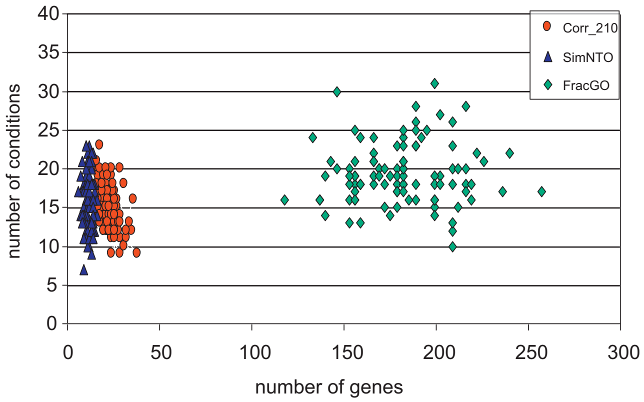

Fig. 5 – Size of biclusters for GDS1116 dataset when correlation and biological relevance are equally important.

Table 4 – Average, variance, maximum and minimum of the number of genes and conditions for biclusters from Figs. 5–7 and 10.

Run configuration		Genes				Conditions			
		Average	Variance	Max.	Min.	Average	Variance	Max.	Min.
Fig. 5	SimNTO	11.6	2.1	16	6	15.6	3.4	23	7
	FracGO	181.2	25.9	257	118	19.4	3.9	31	10
	Corr-210	23.5	4.6	38	16	14.7	2.8	23	9
Fig. 6	SimNTO	8.7	2.1	17	4	15.2	3.5	24	6
	FracGO	193.9	24.8	247	130	19.3	4.0	35	8
Fig. 7	SimNTO	10.1	2.6	16	5	5.1	2.1	11	3
	FracGO	109.1	17.6	162	65	3	0	3	3
	Corr-220	46.8	13.8	93	9	3.1	0.6	7	3
Fig. 10	OPSM	128.2	226.2	774	2	10.4	6.3	23	2
	CC	21.7	14.9	90	9	18.4	9.0	68	7
	ISA	50.7	38.2	97	4	6.4	2.3	9	2

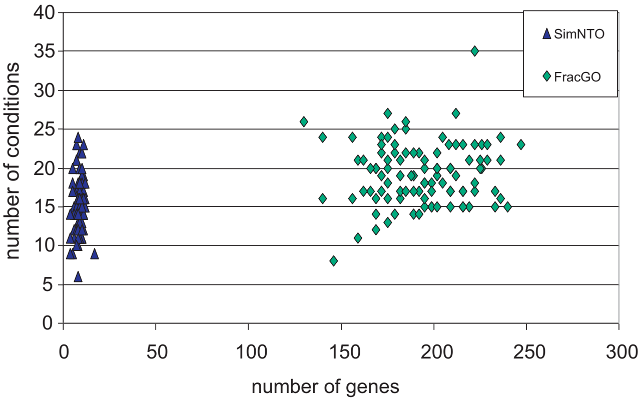

Fig. 6 – Size of biclusters for GDS1116 dataset when biological relevance is more important than correlation.

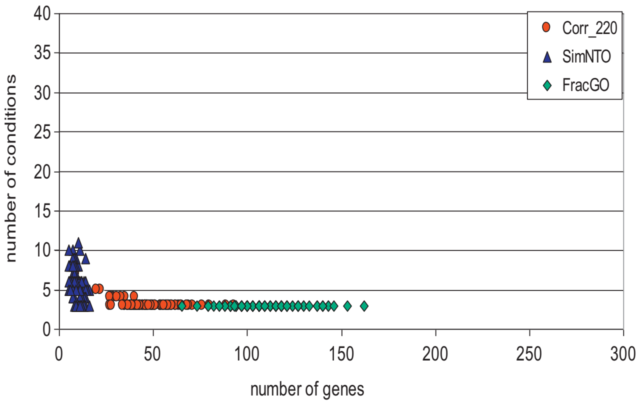

Fig. 7 – Size of biclusters for GDS1116 dataset when correlation is more important than biological relevance.

To make a comparison with the proposed algorithm, the biclustering algorithms ChCh [5], ISA [10], OPSM [11] and xMotifs [12], available in the BiCAT tool [45], have been chosen. All these algorithms constitute a classical framework of reference for biclustering and are commonly used in the literature to establish comparisons among biclusters [6]. Table 3 presents the number of biclusters, the average size of the biclusters, the average percentage of enriched bicluster for BP, MF and

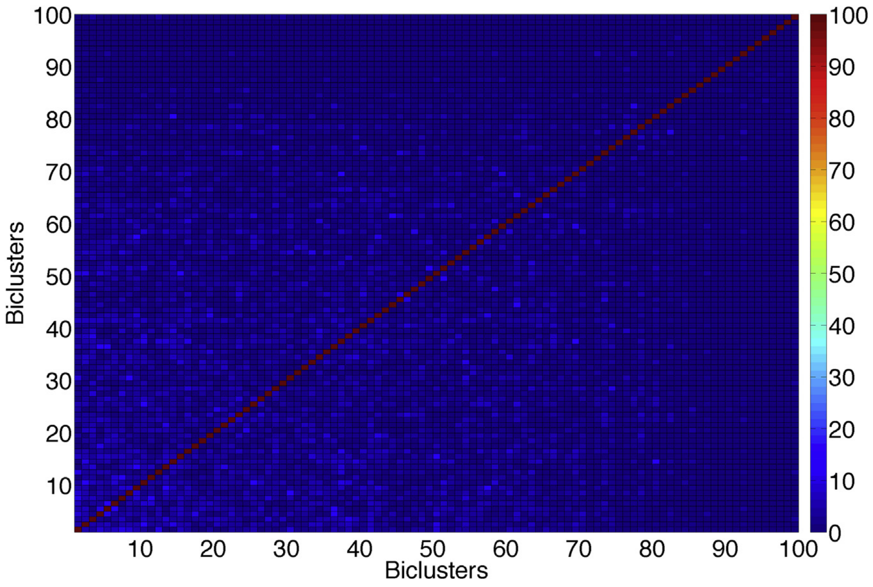

Fig. 8 – Percentage of overlapping among biclusters for GDS1116 dataset.

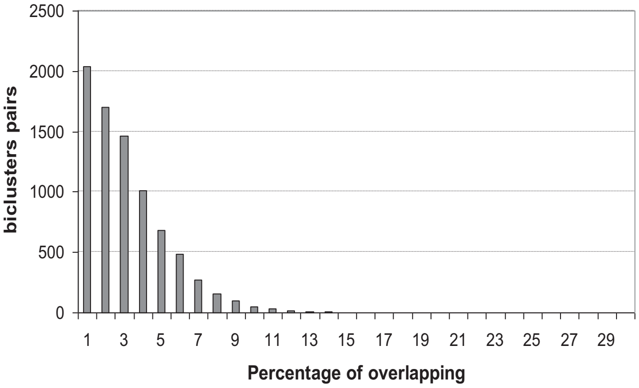

Fig. 9 – Histogram of percentage of overlapping among biclusters for GDS1116 dataset.

CC domains and the average number of GO terms per bicluster obtained by ChCh, ISA, OPSM, xMotifs and the proposed algorithm for both *GDS*1116 and *GDS*2914 datasets. Note that xMotifs can be only applied to datasets with less than 64 conditions, and therefore, the results for the *GDS*2914 dataset are just reported.

Fig. 10 represents the size of biclusters obtained by the ChCh, ISA and OPSM algorithms for *GDS*1116 dataset. Table 4 also shows the same than information than this figure.

Table 5 shows the quality of biclusters obtained by using the fitness function based on the FracGO measure for *GDS*1116 and *GDS*2914 datasets when the functional annotation file, which is an input of the proposed algorithm, is generated from databases different to GO such as KEEG pathways and InterPro. A description of this file for GO, KEEG pathways and InterPro databases is presented in Table 6. In particular, the size of the dataset, the functional annotation source, the number of genes, the number of terms and the average number of terms per gene are reported.

Table 7 shows the quality of biclusters obtained by using the fitness function based on other gene pairwise GO-based measures, concretely the SimGIC and SimUI measures. These measures are graph-based measures that consider the hierarchical nature of GO. SimGIC is a hybrid measure that uses IC in addition to GO graph structure while SimUI is only based on counting terms in the graph [31]. The source code provided by [46] has been used to integrate them in the proposed algorithm.

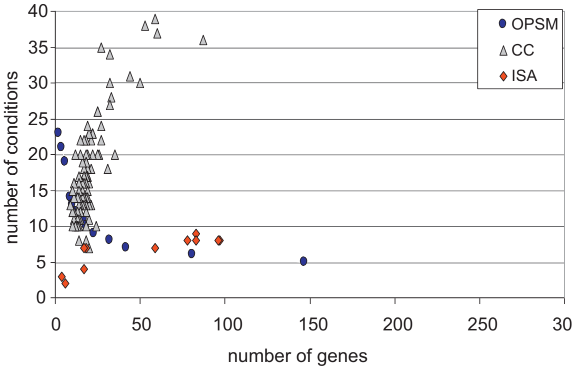

Fig. 10 – Size of biclusters obtained by CC, ISA and OPSM algorithms for GDS1116 dataset.

3.3. *Discussion of results*

The integration of biological knowledge improves the performance of the proposed algorithm with regard to the percentage of enriched biclusters. It can be observed from Table 2 that better biclusters are found for all configurations when the fitness function is based on FracGO and SimNTO measures for BP domain (see black bars in Figs. 3 and 4). Namely, a 100% of enriched biclusters are always obtained from the use of the FracGO measure in the fitness function for both datasets and 99%, 87% or 95% for *GDS*1116 dataset and 87%, 65% or 87% for *GDS*2914 dataset from the SimNTO measure. Although the BP domain is usually considered for comparative studies and enrichment analyses in the literature [6], it is important to note that these analyses are biased with respect to the FracGO measure due to the proper definition of this measure. The FracGO measure of a bicluster implies that this bicluster is enriched regarding the annotation file used as input of the algorithm. Hence, if BP domain is used as input of the algorithm and a criterion based on enrichment is used to evaluate the bicluster, a 100% of enriched biclusters is always expected to obtain with FracGO measure, as it can be observed in Table 2. In order to avoid this bias, the percentage of enriched biclusters for MF and CC domains is also used as evaluation criteria and not only BP. Note that SimNTO is based on an independent criterion to the enrichment. It can be observed that the performance of the algorithm for SimNTO and FracGO measures and for all parameter settings is better than when there is not biological integration ($f_3 = 0$) for *GDS*2914 dataset (see gray and white bars in Fig. 4). For *GDS*1116, it can be appreciated in Fig. 3 that the parameter setting 211 for SimNTO improves other configurations such as 210 and 220. On the other hand, the results for the 212 setting for FracGO are a bit lower than the results for the 210 configuration for MF and CC domains but they are in the same range of values (see horizontal line at 79% in this figure).

Moreover, the percentage of enriched biclusters when using the FracGO and SimNTO measures improves the results of classical biclustering algorithms. For *GDS*1116, Fig. 3 shows that SimNTO obtains better results than ChCh, ISA and OPSM for all domains. Otherwise, all configurations for FracGO improve ChCh and the 212 configuration obtains results in a similar range of values to those of ISA and OPSM for MF and CC domains (see horizontal line at 79% in the figure). For *GDS*2914, it can be observed in Fig. 4 that SimNTO clearly improves ChCh, ISA, OPSM and xMotifs for BP domain and FracGO improves all algorithms for BP, MF and CC domains. A high value for the GO terms per bicluster is obtained by the OPSM (20.06 and 28.64 for *GDS*1116 and *GDS*2914 datasets, respectively) due to the OPSM reports biclusters composed of a great number of genes.

The percentage of enriched biclusters for SimNTO measure is greater when emphasizing correlated patterns (221 configuration) than biological integration (212 configuration) (see circles in Figs. 3 and 4). This fact is because of the SimNTO measure is not based on the enrichment of genes but on an independent criterion such as the overlapping among GO

Table 5 – Results obtained by the fitness function based on FracGO from KEGG pathways and Interpro for both datasets. Each row shows a run that obtains 100 biclusters.

Dataset	Functional annotations from	Parameters (M_1, M_2, M_3)	Size	Enriched biclusters (%)			GO terms per biclus. (BP)	Time (s)
				BP	MF	CC		
GDS1116	KEGG	(2, 1, 1)	(73.4 × 18.1)	13	26	34	0.20	165.4
		(2, 1, 2)	(79.6 × 17.8)	13	26	27	0.21	175.6
		(2, 2, 1)	(49.8 × 3.1)	7	11	8	0.07	145.3
	InterPro	(2, 1, 1)	(14.4 × 16.2)	71	80	71	1.81	3613.3
		(2, 1, 2)	(30.4 × 17.0)	60	78	52	3.32	2380.3
		(2, 2, 1)	(43.3 × 3.3)	24	21	22	0.52	2243.8
GDS2914	KEGG	(2, 1, 1)	(69.4 × 13.3)	18	49	30	0.33	116.0
		(2, 1, 2)	(77.1 × 14.2)	24	46	23	0.48	126.6
		(2, 2, 1)	(42.6 × 3.0)	12	15	11	0.12	100.6
	InterPro	(2, 1, 1)	(17.5 × 9.1)	20	19	6	0.41	1795.3
		(2, 1, 2)	(15.5 × 9.6)	26	23	15	0.99	1825.8
		(2, 2, 1)	(39.2 × 3.1)	25	18	16	0.33	1850.6

Table 6 – Functional annotations from GO, KEGG pathways and InterPro for both datasets.

Dataset	Size	Functional annotation source	Number of genes	Number of terms	Avg. terms per gene
GDS1116	(882 × 131)	GO (BP domains)	632	245	10.6
		KEGG pathways	239	53	1.8
		InterPro	575	699	2.5
		GO (MF domains)	634	135	5.5
		GO (CC domains)	703	44	3.1
GDS2914	(975 × 36)	GO (BP domains)	658	256	10.1
		KEGG pathways	190	65	1.9
		InterPro	556	653	2.2
		GO (MF domains)	615	127	4.9
		GO (CC domains)	740	46	3.2

terms associated to genes. The best parameter setting for SimNTO measure is (2, 1, 1), that is, the same importance for M_2 and M_3 parameters. Thus, it can be concluded that an equilibrium between correlated patterns and biological integration is the best option for SimNTO measure.

It is important to mention that the biclusters composed of a very low number of conditions could not reveal interesting information although they could be composed of a group of relevant genes. Therefore, although the percentage of enriched biclusters is a common criterion in order to compare biclustering techniques, other relevant criteria such as the size of the biclusters should be taken into account. From Figs. 5 and 6 and Table 2, it can be observed that the size of biclusters is similar when the biological relevance is equal or more important than finding interesting patterns by the correlation measure (parameters $M_2 = M_3 = 1$ or $M_2 = 1$ and $M_3 = 2$,

Table 7 – Results obtained by the fitness function based on others gene pairwise GO measures. Each row shows a run that obtains 100 biclusters.

Dataset	Fitness function		Size	Enriched biclusters (%)			GO terms per biclus. (BP)	Time (s)
	Measure f_3	Parameters (M_1, M_2, M_3)		BP	MF	CC		
GDS1116	1-SimGIC	(2, 1, 1)	(11.2 × 15.6)	100	100	99	4.5	1409.4
		(2, 1, 2)	(9.0 × 20.0)	100	100	100	2.0	419.0
		(2, 2, 1)	(19.6 × 4.2)	51	48	53	1.1	887.1
	1-SimUI	(2, 1, 1)	(10.6 × 15.6)	98	97	98	4.1	584.3
		(2, 1, 2)	(8.71 × 16.9)	94	90	93	3.1	275.5
		(2, 2, 1)	(10.2 × 4.6)	62	66	65	1.5	509.5
GDS2914	1-SimGIC	(2, 1, 1)	(23.0 × 9.1)	2	4	7	0.03	1572.2
		(2, 1, 2)	(14.8 × 8.8.x)	36	17	18	1.6	993.1
		(2, 2, 1)	(34.4 × 3.1)	10	8	9	0.1	993.1
	1-SimUI	(2, 1, 1)	(17.0 × 9.2)	11	5	11	0.2	348.2
		(2, 1, 2)	(8.4 × 9.8)	64	48	47	1.7	348.2
		(2, 2, 1)	(18.7 × 3.1)	21	8	19	0.6	536.4

respectively). However, it can be observed from Fig. 7 that the size of biclusters decreases when biclusters with shifting or scaling patterns are preferred (parameters $M_2 = 2$ and $M_3 = 1$). In particular, the number of conditions shows a drop, which is much more notable when not considering any biological knowledge (from three to five conditions) or in the case of the FracGO measure (three conditions). On the other hand, the biclusters obtained from the fitness function based on the FracGO measure have a number of genes significantly higher than that of the SimNTO measure.

Despite the percentage of enriched biclusters is 100% for the FracGO measure for BP domain, the average of the number of GO terms per bicluster is 1 (see Table 2). That is, the proposed algorithm finds biclusters composed of a group of genes sharing the same GO term when maximizing the FracGO measure. The reason can be that genes annotated in the upper levels of the GO hierarchy are being obtained. Contrarily, an average number of GO terms per bicluster of 5.74, 3.67 and 4.5 for *GDS*1116 dataset and 5.45, 2.94 and 5.39 for *GDS*2914 dataset are computed for the SimNTO measure. Thus, biclusters obtained from the SimNTO measure are composed of genes with a higher number of annotations in GO than that of the FracGO measure, and therefore, more information about biological process related to the genes composing of the biclusters is provided.

Due to the nature of the FracGO measure a high computational cost is expected. In fact, it can be observed in Table 2 that the computational cost of the FracGO measure is higher than that of the SimNTO measure. Note that given a set of genes forming the bicluster, a *p*-value for each GO term from the annotation file is computed to evaluate the quality of a bicluster. In addition, the evaluation of the biclusters that belong to the reference set is repeated during the search process by the different methods of the scatter search, in particular, the improvement method, the reference set update method and the build reference set method. Therefore, several hundred or even thousand evaluations can be required. It is important to note that the lower number of GO terms in the annotation file is, the better computational performance of the FracGO measure is.

Several constraints on the input annotation file are required in order to compute the SimNTO measure. In particular, this measure only is defined for an input file from the GO database, and moreover, the file must capture each one of the levels of GO and its tree structure must be contemplated. However, the FracGO measure can be used with annotation files from any database such as that are not related with an ontology structure as KEGG pathways or other functional annotation sources. Moreover, the FracGO measure does not need either the complete GO tree structure or to capture all levels of GO but only from a specific sublevel to other one when the file is generated from GO. This fact is essential because both measures or only the FracGO measure can be used depending on how the biological information is given. Table 5 shows a situation where only FracGO measure can be used. It can be seen in Table 6 that the number of genes annotated in GO for both datasets is significantly greater than that of the KEGG pathways and InterPro databases. It can be appreciated from Tables 2 and 5 how the number of genes annotated and the number of annotations per gene have an influence on the percentage of enriched biclusters obtained from the FracGO measure. In particular, the lower the number of genes annotated is, the lower the percentage of enriched biclusters is.

The SimGIC and SimUI measures that use a file with the GO tree structure in addition to the direct annotation file have been applied in order to compare the quality of the biclusters obtained. From Table 7, it can be observed that the biclusters obtained by the SimNTO have a number of GO terms per bicluster higher than that obtained by the SimGIC and SimUI. Moreover, the SimNTO obtains a greater percentage of enriched biclusters than other measures for all configurations and the (2, 2, 1) configuration for the *GDS*2914 and *GDS*1116 datasets, respectively. On the other hand, the computing time of SimNTO is several orders of magnitude lower than the computing time of SimGIC and SimUI. Thus, it can be concluded that SimNTO obtains better results than SimGIC and SimUI and it is simpler and faster. Although SimNTO does not use an additional file to compute the tree structure of GO, note that this one is considered through the way that the annotation file is built.

In short, as a consequence of the results and the computational cost of the FracGO measure, the SimNTO measure is the best measure when integrating the biological knowledge from GO annotation files that capture all direct annotations and all the associated parent terms for each gene, i.e., the annotations for all the levels of the ontology. The FracGO measure is appropriate when the GO annotation file captures only several sublevels from GO or when the annotation file is not related to an ontology as for example KEGG pathways or InterPro annotations. In fact, annotation files that do not propagate the annotation information to upper levels in GO are more adequate for the FracGO measure as the biclusters obtained from the FracGO measure are composed of sets of genes that share the same GO terms.

3.4. *Qualitative biological evaluation*

The gene enrichment analysis is the standard comparison methodology in the biclustering field [6,44]. However, an additional biological study is necessary to understand the biological relevance of the biclusters obtained by the proposed measures. Thus, SimNTO-based biclusters are biologically relevant and they capture meaningful information. However, FracGO-based biclusters are enriched biclusters but they only share a GO term (see Table 2). The hypothesis is that they are composed of genes annotated in the upper levels of the GO hierarchy, and therefore, with a very general and irrelevant information.

In this section a deep biological evaluation as in the reference [37] is presented. This study is based on the functional analysis by considering pathway mapping and statistical significance of gene enrichment in pathways. The resource used for mapping genes in pathways has been Reactome [47]. The first five biclusters according to the fitness function value for runs with *GDS*1116 dataset have been studied. Concretely, the five first biclusters for SimNTO and FracGO with 211 and 212 configuration parameters and for Corr with 210. Note that biclusters for SimNTO and FracGO 221 and for Corr 220 are composed of a low number of conditions and they are less

Table 8 – Mapping analysis provided by Reactome for bicluster 3 from the SimNTO measure and 211 configuration.

Pathway identifier	Pathway name	FDR
247749	Eukaryotic translation elongation	0.001
260795	Translation	0.001
232946	GTP hydrolysis and joining of the 60S ribosomal subunit	0.001
217188	Formation of a pool of free 40S subunits	0.001
257612	Eukaryotic translation termination	0.001
257951	Peptide chain elongation	0.001
188965	SRP-dependent cotranslational protein targeting to membrane	0.001
189183	Nonsense-mediated decay (NMD) independent of the exon junction complex (EJC)	0.001
189048	Nonsense-mediated decay (NMD)	0.001
189050	Nonsense-mediated decay (NMD) enhanced by the exon junction complex (EJC)	0.001
252688	L13a-mediated translational silencing of Ceruloplasmin expression	0.002
251703	Cap-dependent translation initiation	0.002
230274	Eukaryotic translation initiation	0.002
257608	Formation of the ternary complex, and subsequently, the 43S complex	0.040
233365	Metabolism of proteins	0.040
248935	Ribosomal scanning and start codon recognition	0.050

Table 9 – Mapping analysis provided by Reactome for bicluster 5 from the SimNTO measure and 212 configuration.

Pathway identifier	Pathway name	FDR
232946	GTP hydrolysis and joining of the 60S ribosomal subunit	0.014
217188	Formation of a pool of free 40S subunits	0.014
257951	Peptide chain elongation	0.014
188965	SRP-dependent cotranslational protein targeting to membrane	0.014
247749	Eukaryotic translation elongation	0.014
189183	Nonsense-mediated decay (NMD) independent of the exon junction complex (EJC)	0.014
189048	Nonsense-mediated decay (NMD)	0.014
189050	Nonsense-mediated decay (NMD) enhanced by the exon junction complex (EJC)	0.014
188483	S6K1 signaling	0.014
252688	L13a-mediated translational silencing of Ceruloplasmin expression	0.014
251703	Cap-dependent translation initiation	0.019
230274	Eukaryotic translation initiation	0.020
260795	Translation	0.032
188482	mTORC1-mediated signaling	0.036
188481	S6K1-mediated signaling	0.036

interesting to study. The five SimNTO 211 and 212 biclusters present pathways mapping, only two biclusters for Corr 220 present information but there is not any information for FracGO 211 and 212 biclusters. Tables 8 and 9 show the mapping analysis reported by Reactome for the third and fifth bicluster for SimNTO 211 and 212, respectively, and Table 10 the information for the fourth bicluster for Corr 220. Each row in these tables shows the pathway identifier and name and the false discovery rate for each pathway found for the bicluster. The complete information for all tables is provided as Supplementary information.

Most of overrepresented pathways in four of the five biclusters studied for SimNTO and 112 configuration are: eukaryotic translation Initiation, cap-dependent translation initiation and L13a-mediated translational silencing of Ceruloplasmin expression. These signaling pathways with significant FDR are signaling by ribosomal protein of the large subunit (RPL9A and RPL9B) and RPL8A also known as ribosomal protein of the large subunit required for processing of 27SA3 pre-rRNA to 27SB pre-rRNA during assembly of large ribosomal subunit. Analogously, the pathways in four of the five biclusters for SimNTO and 212 configuration are: S6K1 signaling, S6K1-mediated signaling and mTORC1-mediated signaling. These pathways are signaling by protein component of the small (40S) ribosomal subunit homologous to mammalian ribosomal protein S6 (RPS6A, RPS6B and POCX37). In the case of the biclusters obtained by the Corr measure and 220 configuration, significant signaling pathways information is found only for two biclusters. Contrarily what happens with the results for SimNTO, the results for Corr cannot be considered impressive. In the functional analysis, SimNTO biclusters have shown more mapping genes in pathways than Corr biclusters. This fact supports the hypothesis that the integration of biological information based on the measure SimNTO improved the results obtained by the proposed biclustering algorithm.

Moreover, the most interesting information is that Reactome does not report any information for biclusters obtained when using the FracGO measure. The reason is that the FracGO

Table 10 – Mapping analysis provided by Reactome for bicluster 4 from the Corr measure and 210 configuration.

Pathway identifier	Pathway name	FDR
231079	Gene expression	0.002
233365	Metabolism of proteins	0.010
188483	S6K1 signaling	0.012
188482	mTORC1-mediated signaling	0.039
188481	S6K1-mediated signaling	0.039

protein ubiquitination
protein
ubiquitination
metabolism
metabolism

Fig. 11 – GO term clusters reported by Revigo from bicluster 1 for the FracGO measure and 211 configuration. Only two general GO terms are reported: protein-ubiquitination and metabolism.

measure captures biclusters where genes are together in a same GO term, and as a consequence, their genes are annotated in the upper levels of the GO hierarchy. Note that the annotation files used as input parameter are generated capturing all the associated parent terms for each gene. Gene Term Linker [48] and Revigo [49] tools have been used to confirm that the biclusters based on the FracGO measure show very general GO information. Firstly, the first tool filters irrelevant GO information by identifying metagroups of genes with coherent biological significance. Secondly, Revigo summarizes a list of these GO terms by finding representative subsets of terms using a clustering procedure that removes redundant terms. Fig. 11 shows the significant enriched GO term in the first bicluster obtained by the FracGO measure and 211 configuration. All enriched GO terms have been clustered in two very general terms in GO: metabolism and protein ubiquitation. After having applied this pipeline analysis, Gene Term Linker and Revigo, all reported enriched GO terms are clustered in general GO terms. Therefore, the assumption that FracGO-based biclusters are composed of genes annotated in the upper levels of the GO hierarchy is confirmed. The FracGO measure finds biclusters with a general information in GO, and hence, these biclusters are composed of genes that are not related as group with any pathway. The complete information for all figures can be read as Supplementary information.

4. Conclusions

A scatter search-based biclustering algorithm that integrates biological information has been proposed in this paper. The data input of the proposed algorithm are the gene expression matrix and a direct annotation file linking genes with sets of biological terms extracted from a biological repository or database. The Gene Ontology, KEGG pathways and InterPro databases have been used as source for generating these files in this work. Two different biological measures, FracGO and SimNTO, have been proposed to integrate this information by means of its addition to the fitness function to be optimized to evaluate the quality of the biclusters in the scatter search. The measure FracGO is based on the biological enrichment and SimNTO is based on the overlapping among GO annotations of pairs of genes. Experimental results from the application of the proposed algorithm for two datasets have been reported and discussed showing a better performance when biological knowledge is integrated and better biclusters than that of the classical biclustering algorithms.

The main motivation has been to use a standard biological validation criterion in biclustering [6,44] as mechanism to integrate biological knowledge. This criterion is based on the percentage of enriched biclusters that is calculated using direct annotation files. If these files are GO files which have been generated by propagating annotations to upper levels in the GO hierarchy, the experimental results show that FracGO-based biclusters only share general GO terms and they do not capture relevant biological information. In this case, the SimNTO measure, which is based on the *term overlap* defined in [32] and uses annotation files as input, is faster and simpler than other GO semantic measures and solves this problem. Several experiments show that SimNTO-based biclusters capture relevant biological information that present pathways mapping in Reactome, for example. It is important to note that SimNTO captures the GO hierarchical structure if the annotation files contain all parent terms for each term. As a summary, the differences between the FracGO and SimNTO measures depend on the data source and how the annotation file has been built in the case GO is used.

Future work will be focused on the study of other biological measures to handle microRNA/mRNA data and how to integrate information from different biological sources. Some improvements in the search procedure of the proposed algorithm will be also analyzed such as the setting configuration of inner parameters.

Conflicts of interest

We declare that we have not any actual or potential competing financial interests.

Acknowledgments

We would like to thank Spanish Ministry of Science and Innovation, Junta de Andalucía and University Pablo de Olavide for the financial support under projects TIN2011-28956-C02-02, P12-TIC-1728 and APPB813097, respectively.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.cmpb.2015.02.010.

REFERENCES

[1] F. Markowetz, R. Spang, Inferring cellular networks – a review, BMC Bioinform. 8 (2007) S5.

[2] K. Rhrissorrakrai, K. Gunsalus, Mine: module identification in networks, BMC Bioinform. 12 (2011) 192.

[3] T. Nepusz, H. Yu, A. Paccanaro, Detecting overlapping protein complexes in protein–protein interaction networks, Nat. Methods 9 (2012) 471.

[4] J. Morgan, J. Sonquistz, Problems in the analysis of survey data, and a proposal, J. Am. Stat. Assoc. 58 (1963) 415–434.

[5] Y. Cheng, G. Church, Biclustering of expression data, in: Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology, vol. 8, 2000, pp. 93–103.

[6] A. Prelic, S. Bleuler, P. Zimmermann, A. Wille, P. Buhlmann, W. Gruissem, L. Hennig, L. Thiele, E. Zitzler, A systematic comparison and evaluation of biclustering methods for gene expression data, Bioinformatics 22 (2006) 1122–1129.

[7] S. Madeira, A. Oliveira, Biclustering algorithms for biological data analysis: a survey, IEEE Trans. Comput. Biol. Bioinform. 1 (2004) 24–45.

[8] A. Tanay, R. Sharan, R. Shamir, Biclustering algorithms: a survey Handbook of Computational Molecular Biology, vol. 9, Edited by: Aluru S. Chapman & Hall/CRC Computer and Information Science Series, 2005, pp. 26–31.

[9] S. Busygin, O. Prokopyev, P. Pardalos, Biclustering in data mining, Comput. Oper. Res. 35 (2008) 2964–2987.

[10] S. Bergmann, J. Ihmels, N. Barkai, Iterative signature algorithm for the analysis of large-scale gene expression data, Phys. Rev. E 67 (2003) 1–18.

[11] A. Ben-Dor, B. Chor, R. Karp, Z. Yakhini, Discovering local structure in gene expression data: the order-preserving submatrix problem, J. Comput. Biol. 10 (2003) 373–384.

[12] T. Murali, S. Kasif, Extracting conserved gene expression motifs from gene expression data, in: Proceedings of Pacific Symposium on Biocomputing, 2003, pp. 77–88.

[13] J. Aguilar-Ruiz, Shifting and scaling patterns from gene expression data, Bioinformatics 21 (2005) 3840–3845.

[14] H. Banka, S. Mitra, Evolutionary biclustering of gene expressions, Ubiquity 7 (2006) 1–12.

[15] F. Divina, J. Aguilar-Ruiz, A multi-objective approach to discover biclusters in microarray data, in: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, ACM Press, New York, USA, 2007, pp. 385–392.

[16] J. Liu, Z. Li, X. Hu, Y. Chen, Biclustering of microarray data with MOSPO based on crowding distance, BMC Bioinform. 10 (2009) S9.

[17] C.A. Gallo, J.A. Carballido, I. Ponzoni, Microarray biclustering: a novel memetic approach based on the PISA platform, in: Proceedings of the 7th European Conference on Evolutionary Computation, Machine Learning and Data Mining – EvoBIO 2009, 2009, pp. 44–55.

[18] W. Ayadi, M. Elloumi, J.-K. Hao, A biclustering algorithm based on a bicluster enumeration tree: application to DNA microarray data, BioData Min. 2 (2009) 9.

[19] W.-H. Yang, H. Yan, D.-Q. Dai, Finding correlated biclusters from gene expression data, IEEE Trans. Knowl. Data Eng. (2010) 568–584.

[20] G. Li, Q. Ma, H. Tang, A.H. Paterson, Y. Xu, Qubic: a qualitative biclustering algorithm for analyses of gene expression data, Nucleic Acids Res. 37 (2009) e101.

[21] A. Bhattacharya, R.K. De, Bi-correlation clustering algorithm for determining a set of co-regulated genes, Bioinformatics 25 (2009) 2795–2801.

[22] A. Bhattacharya, R.K. De, Bi-correlation clustering algorithm for determining a set of co-regulated genes, Bioinformatics 25 (2009) 2795–2801.

[23] T. Yun, G.-S. Yi, Biclustering for the comprehensive search of correlated gene expression patterns using clustered seed expansion, BMC Genomics 14 (2013) 144.

[24] T. Zeng, J. Li, Maximization of negative correlations in time-course gene expression data for enhancing understanding of molecular pathways, Nucleic Acids Res. 38 (2010) e1.

[25] J.A. Nepomuceno, A. Troncoso, J. Aguilar-Ruiz, Biclustering of gene expression data by correlation-based scatter search, BioData Min. 4 (2011) 3.

[26] J.L. Flores, I. Inza, P. Larrañaga, B. Calvo, A new measure for gene expression biclustering based on non-parametric correlation, Comput. Methods Programs Biomed. 112 (2013) 367–397.

[27] J. Bryan, Problems in gene clustering based on gene expression data, J. Multivar. Anal. 90 (2004) 44–66.

[28] M. Verbanck, S. Le, J. Pages, A new unsupervised gene clustering algorithm based on the integration of biological knowledge into expression data, BMC Bioinform. 14 (2013) 42.

[29] F. Azuaje, H. Wang, H. Zheng, F. Leonard, M. Rolland-Turner, L. Zhang, Y. Devaux, D. Wagner, Predictive integration of gene functional similarity and co-expression defines treatment response of endothelial progenitor cells, BMC Syst. Biol. 5 (2011) 46.

[30] R. Luque-Baena, D. Urda, M.G. Claros, L. Franco, J. Jerez, Robust gene signatures from microarray data using genetic algorithms enriched with biological pathway keywords, J. Biomed. Inform. 49 (2014) 32–44.

[31] C. Pesquita, D. Faria, H. Bastos, A. Ferreira, A. Falcao, F. Couto, Metrics for go based protein semantic similarity: a systematic evaluation, BMC Bioinform. 9 (2008) S4.

[32] M. Mistry, P. Pavlidis, Gene ontology term overlap as a measure of gene functional similarity, BMC Bioinform. 9 (2008) 327.

[33] F. Azuaje, Bioinformatics and Biomarker Discovery: Omic Data Analysis for Personalized Medicine, Wiley-Blackwell, 2010.

[34] K. Lo, A. Raftery, K. Dombek, J. Zhu, E. Schadt, R. Bumgarner, K. Yeung, Integrating external biological knowledge in the construction of regulatory networks from time-series expression data, BMC Syst. Biol. 6 (2012) 101.

[35] C. Huttenhower, K.T. Mutungu, N. Indik, W. Yang, M. Schroeder, J.J. Forman, O.G. Troyanskaya, H.A. Coller, Detailing regulatory networks through large scale data integration, Bioinformatics 25 (2009) 3267–3274.

[36] P.E.A. Dao, Inferring cancer subnetwork markers using density-constrained biclustering, Bioinformatics 26 (2010) 625–631.

[37] G. Pio, M. Ceci, D. D'Elia, C. Loglisci, D. Malerba, A novel biclustering algorithm for the discovery of meaningful biological correlations between micrornas and their target genes, BMC Bioinform. 14 (2013) S8.

[38] J.A. Nepomuceno, A. Troncoso, J.S. Aguilar-Ruiz, Evolutionary metaheuristic for biclustering based on linear correlations among genes, in: SAC 2010: Proceedings of the 2010 ACM Symposium on Applied Computing (SAC), Sierre, Switzerland, March 22–26, 2010, 2010, pp. 1143–1147.

[39] G.F. Berriz, O.D. King, B. Bryant, C. Sander, F.P. Roth, Characterizing gene sets with FuncAssociate, Bioinformatics 19 (2003) 2502–2504.

[40] R. Marti, M. Laguna, Scatter Search. Methodology and Implementation in C, Kluwer Academic Publishers, Boston, 2003.

[41] J.A. Nepomuceno, A.T. Lora, J.S. Aguilar-Ruiz, An overlapping control-biclustering algorithm from gene expression data, in: Ninth International Conference on Intelligent Systems

Design and Applications, ISDA 2009, Pisa, Italy, November 30–December 2, 2009, 2009, pp. 1239–1244.

[42] R. Edgar, M. Domrachev, A.E. Lash, Gene expression omnibus: NCBI gene expression and hybridization array data repository, Nucleic Acids Res. 30 (2002) 207–210.

[43] I. Medina, J. Carbonell, L. Pulido, S.C. Madeira, S. Goetz, A. Conesa, J. Tárraga, A. Pascual-Montano, R. Nogales-Cadenas, J. Santoyo, F. García, M. Marbà, D. Montaner, J. Dopazo, Babelomics: an integrative platform for the analysis of transcriptomics, proteomics and genomic data with advanced functional profiling, Nucleic Acids Res. 38 (2010) W210–W213.

[44] K. Eren, M. Deveci, O. Kucuktunc, U.V. Catalyurek, A comparative analysis of biclustering algorithms for gene expression data, Brief. Bioinform. 14 (2013) 279–292.

[45] S. Barkow, S. Bleuler, A. Prelic, P. Zimmermann, E. Zitzler, Bicat: a biclustering analysis toolbox, Bioinformatics 22 (2006) 1282–1283.

[46] H. Caniza, A.E. Romero, S. Heron, H. Yang, A. Devoto, M. Frasca, M. Mesiti, G. Valentini, A. Paccanaro, Gossto: a stand-alone application and a web tool for calculating semantic similarities on the gene ontology, Bioinformatics 30 (2014) 2235–2236.

[47] R. Haw, H. Hermjakob, P. D'Eustachio, L. Stein, Reactome pathway analysis to enrich biological discovery in proteomics data sets, Proteomics 11 (2011) 3598–3613.

[48] C. Fontanillo, R. Nogales-Cadenas, A. Pascual-Montano, J. De Las Rivas, Functional analysis beyond enrichment: non-redundant reciprocal linkage of genes and biological terms, PLoS ONE 6 (2011) e24289.

[49] F. Supek, M. Bosnjak, N. Skunca, T. Smuc, Revigo summarizes and visualizes long lists of gene ontology terms, PLoS ONE 6 (2011) e21800.