Preliminary Studies on Biclustering of GWA: A Multiobjective Approach

Khedidja Seridi[1,2], Laetitia Jourdan[1,2](✉), and El-Ghazali Talbi[1,2]

[1] INRIA Lille - Nord Europe, DOLPHIN Project-Team,
59650 Villeneuve d'Ascq Cedex, France
[2] Université Lille 1, LIFL, UMR CNRS 8022, 59655 Villeneuve d'Ascq Cedex, France
{laetitia.jourdan,khedidja.seridi}@inria.fr, talbi@lifl.fr

Abstract. Genome-wide association (GWA) studies aim to identify genetic variations (polymorphisms) associated with diseases, and more generally, with traits. Commonly, a Single Nucleotide Polymorphism (SNP) is considered as it is the most common form of genetic variations. In the literature, several statistical and data mining methods have been applied to GWA data analysis. In this article, we present a preliminary study where we examine the possibilities of applying biclustering approaches to detect association between SNP markers and phenotype traits. Therefore, we propose a multiobjective model for biclustering problems in GWA context. Furthermore, we propose an adapted heuristic and metaheuristic to solve it. The performance of our algorithms are assessed using synthetic data sets.

1 Introduction

Association mapping has recently become a popular approach to discover the genetic causes of many complex diseases. A genome wide association study (GWAs) is the examination process of different genetic variants (markers) in several individuals in the purpose of detecting eventual association between the variants and certain traits. GWAs particularly focus on associations between single-nucleotide polymorphisms (SNPs) and traits like major diseases. Once such genetic associations are identified, researchers can use the information to promote new strategies to detect, treat and prevent the diseases [2].

Regarding the considered phenotype's nature, GWA studies usually deal with two classes of data. In the first class, the data comprise the genetic informations of all or a large fraction of the diseased subjects (cases) that appear in the considered study base and then sampling a comparable number of healthy subjects (controls), ideally from the same study base, and potentially matched with the cases by some socio-demographic characteristics such as race, age and gender. Accordingly, the considered trait is a qualitative trait *i.e.* an individual is even a case or a control. In the second class, the addressed phenotype is a quantitative trait *i.e.* numerical values that can be ordered from highest to lowest such as height, weight, cholesterol level, etc. The analysis of the later form of data is known as *Quantitative Trait Locus* (QTL) analysis.

P. Legrand et al. (Eds.): EA 2013, LNCS 8752, pp. 106–117, 2014.
DOI: 10.1007/978-3-319-11683-9_9

By considering the entire genome, case/control data analysis is essentially based on seeking alleles of variants that are more frequent in people with the disease (cases). The found variant is then said to be *associated* with the disease.

Quantitative trait locus (QTL) analysis is a statistical method that links two types of information *i.e.* phenotypic data (quantitative trait) and genotypic data (usually markers), in an attempt to explain the genetic basis of variation in complex traits [5]. QTL analysis allows researchers in different fields such as agriculture, evolution, and medicine to link certain complex phenotypes to specific regions of chromosomes. The goal of this process is to identify the action, interaction, number, and precise location of these regions.

A QTL analysis starts by collecting phenotype and genotype data from a number of unrelated individuals in the same way as in a case-control study. However, in QTL studies there are no cases and no controls, just individuals with a range of phenotype values. After that, association between the traits and the different SNPs are detected using statistical method. The associations are commonly formulated as predictive models.

Generally, genome wide associations studies are performed using supervised methods such as logistic regression and discriminant analysis [1,9], Bayesian approaches [4], etc. Commonly, the treated data comprises two main informations for each individual: genotype informations and phenotype informations. Using a training data set, the study mainly consists in defining a predictive model and validate it through a test data set.

In this work we propose an unsupervised study of the GWA data with quantitative traits (QTL). By this study we aim to extract a subset of SNPs that have the same alleles for a sub set of individuals sharing similar traits. Actually, the considered data can be seen as a matrix $A = (X, (Y, Z)) = \{a_{ij}\}$ where each row i presents an individual, each column j represents either a SNP ($j \in Y$) or a trait ($j \in Z$) and an element a_{ij} presents the corresponding SNP's allele (if $j \in Y$) or the corresponding traits value (if $j \in Z$) (see Table 1). Thus, a bicluster $B = (I, (J, K))$ is a sub-matrix of $A = (X, (Y, Z))$ where $I \subset X$, $J \subset Y$ and $K \subset Z$.

This paper is organized as follows. Section 2 presented the biclustering problem and a new multiobjective model for a biclustering problem applied to analyzing GWA data sets. An adapted heuristic and metaheuristic are proposed in

Table 1. Studied GWA data

	SNPs			Traits		
	S_1	...	S_A	T_1	...	T_B
A_1	a_{11}	...	a_{1A}	a_{1A+1}	...	a_{1M}
...	...	...	...	...	...	...
A_i	a_{i1}	...	a_{iA}	a_{iA+1}	....	a_{iM}
...	...	...	...	...	...	...
A_N	a_{N1}	...	a_{NA}	a_{NA+1}	....	a_{NM}

Sect. 3 to solve the proposed model. In Sect. 4, experimental analysis of the proposed approaches and results are presented. Finally Sect. 5 concludes the paper and presents perspectives.

2 Biclustering Method in Analyzing GWA Data

2.1 Biclustering

Biclustering or co-clustering is a well-known data mining method that has been widely applied in a broad range of domains such as marketing, psychology and bioinformatics. It consists in extracting submatrices $B = (I, J)$ $(I \subset X, J \subset Y)$ (called biclusters) with maximal size and respecting a certain coherence constraint. Depending on the addressed problem, biclusters of different types can be considered. The different biclusters types and some corresponding applications are described below.

1. Constant bicluster: all the biclusters elements have the same value.
2. Bicluster with constant rows/columns: the elements of each row (column) have the same value.
3. Bicluster with coherent values: the definition of this type of biclusters is a generalization of constant rows/columns biclusters. There exist two different models associated to this class of biclusters:
   (a) shifting model: where each row (and each column) can be obtained by adding an offset to an other row (column).
   (b) scaling model: where each row (and each column) can be obtained by multiplying an other row (column) by a factor.
4. Bicluster with coherent evolution: the elements of the bicluster behave similarly (correlated) independently of their numerical values.

When formulating a biclustering problem, a similarity (dissimilarity) measure is required in order to evaluate the extracted results. The measure is, commonly, related to the bicluster's type. In the case of microarray data analysis, the study aim to extract biclusters with coherent values or evolution (gene that present similar behavior under a sub set of conditions). Different multiobjective modeling for biclustering problem for microarrays data have been proposed [7,10–14] but none for the case of GWA data. Commonly, the proposed multiobjective models comprise: one or more function(s) to optimize the biclusters sizes, a function that optimizes biclusters coherences and a function to optimize the rows variances. In all of these models, a solution represents one bicluster. Regarding the size, most of the models maximize the ratio between the biclusters elements number and the microarray data elements. However, as the number of rows is generally more important than the number of columns, such functions may favor the maximization of rows number with regard to columns number. Thereby, in [7], authors proposed to maximize the number of rows and columns separately by using two objective functions. Concerning biclusters coherence, all the proposed models consider the Mean Squared Residue MSR [3] dissimilarity measure.

In [14] the MSR value is allowed to increase as it does not exceed the threshold δ. Regarding the rows fluctuations, all the existing models maximize the mean row variance. In [12] the coherence and fluctuation objectives are merged in one function by defining a function as the ratio between the MSR of the bicluster and its mean rows variance.

The MSR measure is well adapted to identify biclusters with coherent values. However, this measure can not be applied for GWA data as different biclusters type is required.

2.2 Multiobjective Problem Modeling

In this section, we propose a multiobjective model for a biclustering method applied to GWAs. In this study, we seek to extract biclusters with constant columns, which correspond to a set of individuals that share SNPs presenting the same alleles and the same traits. In order to extract such biclusters, two objectives have to be considered: maximizing the biclusters size (find maximal biclusters) and minimizing the average of columns variances. Actually, these two criteria are clearly independent and conflicting. In fact, a non perfect bicluster's coherence (columns constance) can be improved by removing a row or a column, *i.e.* by reducing its size. We can therefore deduce that the problem of biclustering in GWAs can be formulated as a multiobjective optimization problem. Thus, the proposed model is given by:

$$f_1(I,(J,K)) = \alpha \times \tfrac{|I|}{|X|} + \beta \times \tfrac{|J|}{|Y|} + \gamma \times \tfrac{|K|}{|Z|}$$
$$f_2(I,(J,K)) = Avar(I,(J,K)) = \tfrac{1}{|I] \times (|J|+|K|)} \textstyle\sum_{j \in J \bigcup K} \sum_{i \in I} (a_{ij} - a_{Ij})^2$$

Where f_1 (size) has to be maximized and f_2 (average variance) has to be minimized

3 Resolution Approaches

In this section we present two new approaches to solve the proposed model. The first approach is a greedy heuristic *Sbic* and the second approach is a multiobjective metaheuristic $SHMOBI_{ibea}$.

3.1 Sbic Heuristic

Sbic is a greedy heuristic that aims to extract relevant biclusters from GWA data matrix and that has been designed in a similar manner as Cheng and Churchs heuristic [3] widely used for microarray data. At each run, *Sbic* extracts one bicluster from the data matrix. *Sbic* deletes (adds) nodes that meet with some conditions in order to decrease the biclusters average columns variances and increase its size. The main steps of *Sbic* are given in Algorithm 1.

In multiple node deletion phase, *Sbic* starts by removing some nodes (rows and columns) in order to decrease the average columns variance. In columns

Algorithm 1. Sbic Algorithm

```
Input: Bicluster $(I, (J, K))$ /*which can be the whole data matrix*/
if$(Avar(I, (J, K)) > \delta)$
    MultipleNodeDeletion(I,J,K)
    if$(Avar(I, (J, K)) > \delta)$
        SingleNodeDeletion (I,J,K)
    endif
endif
MultipleNodeAddition(I,J,K)
```

dimension, the variance of each column is calculated. The columns that have the highest variance are deleted. This process will clearly decrease the whole average variances of the columns. Similarly, the average variance can also be decreased by applying the same process on the rows dimension. Indeed, rows with the highest contribution on the average columns variances are deleted. After that, if the bicluster's average variance still higher than δ the bicluster has to undergo the single node deletion processes. The main steps are illustrated in Algorithm 2.

Algorithm 2. Multiple node deletion

```
Input: Bicluster $(I, (J, K))$
Compute $a_{Ij}$, $Avar$ and $con_i = \frac{\sum_{j \in J}(a_{Ij} - a_{ij})^2 + \sum_{k \in K}(a_{Ik} - a_{ik})^2}{|J| + |K|}$ $i \in I$
if$(con_i > \gamma \times Avar)$
    Remove the rows $i \in I$
endif
Compute $a_{Ij}$, $Avar$ and $var_j$ $j \in J$
if$(var_j > \gamma \times Avar)$
    Remove the column $j \in J$
endif
Compute $a_{Ik}$, $Avar$ and $var_k$ $k \in K$
if$(var_k > \gamma \times Avar)$
    Remove the column $k \in K$
endif
```

In single node deletion, the nodes with the highest contribution on the average variance are iteratively deleted until the *Avar* reaches the desired value. The main steps are illustrated in Algorithm 3.

Once the *Avar* of the considered bicluster reaches the desired value, the algorithm tries to add other rows (columns) without increasing the *Avar*. For instance all the columns (not present yet in the bicluster) that have a variance lower than or equal to *Avar* are added to the bicluster. Furthermore, the expected contribution of each row i (con_i) in the biclusters *Avar* value is computed in order to decide whether the row can be added to the bicluster or not.

Algorithm 3. Single node deletion

```
Input: Bicluster $(I, (J, K))$
    while($Avar(I, (J, K)) > \delta$)
        Recompute $con_i$, $var_j$ and $var_k$.
        Find the node d (row or column) with the highest $var_d$ ($con_d$) .
        Delete d.
    endwhile
```

The main steps are illustrated in Algorithm 4.

Algorithm 4. Multiple node addition

```
Input: Bicluster $(I, (J, K))$
Compute $a_{Ij}$, $Avar$ and $con_i = \frac{\sum_{j \in J}(a_{Ij} - a_{ij})^2 + \sum_{k \in K}(a_{Ik} - a_{ik})^2}{|J|+|K|}$ $i \notin I$
if($con_i \leq Avar$)
    Add the rows $i$
endif
Compute $a_{Ij}$, $Avar$ and $var_j$ $j \notin J$
if($var_j \leq Avar$)
    Add the column $j$
endif
Compute $a_{Ik}$, $Avar$ and $var_k$ $k \notin K$
if($var_k \leq Avar$)
    Add the trait $k$
endif
```

Actually, *Sbic* is a deterministic algorithm. Thus, the same bicluster will be extracted if the starting matrix is always the same. In order to extract several biclusters from a data matrix $(X, (Y, Z))$ we propose to apply the *Sbic* over the whole data matrix to extract the first bicluster. After that, *Sbic* can be applied over a sub-matrix containing $p\%$ of the data's rows and columns selected randomly which will lead to discovering different bicluster at each run.

In the following section we present the main components of $SHMOBI_{ibea}$ metaheuristic.

3.2 $SHMOBI_{ibea}$

$SHMOBI_{ibea}$ is based on $HMOBI_{ibea}$ [15] which is a multiobjective metaheuristic based on the evolutionary algorithm $MOBI_{ibea}$ [6] and $DMLS$ $(1 \cdot 1_{\succ})$ [8].

MOBI is a hybrid MOEA (Multi Objective Evolutionary Algorithm) for solving biclustering problem in the specific case of microarray data. It combines

IBEA with a local search inspired from Cheng and Churchs heuristic [3] which is dedicated for biclustering of microarray data. $MOBI_{ibea}$ [6] allows in the case of microarray data to extract biclusters of good quality.

DMLS (Dominance-based Multiobjective Local Search) are a general concept of multiobjective local searches using the concept of Pareto Optimality. At each generation, DMLS selects one or more non-visited solutions (solutions with non-explored neighborhood) from the *archive* and explores their neighborhoods. After that, the solutions are marked as visited. Different variants of DMLS exists depending on the number of selected solutions and on the exploration strategy. In this study, we will use $DMLS(1 \cdot 1_{\succ})$ where one solution is randomly selected and the exploration of its neighborhood stops when the first improving solution is found.

In this section, we propose $SHMOBI_{ibea}$ which is an adapted version of $HMOBI_{ibea}$ to SNP data. Several changes have been done to adapt $HMOBI_{ibea}$ to the specific case of SNPs. Therefore, we present a suitable solutions encoding and variation operators.

Solutions Encoding. In $SHMOBI_{ibea}$, we choose to represent a bicluster as a list compound of six parts: Each one of the first 3 parts of the chromosome is an ordered list of indexes corresponding to either rows, columns or traits; while parts 4 to 6 are just the cardinalities of those lists.

Example:
Given the data matrix presented in Table 2, the string {1 3 2 3 2 2 2 1} represents the following bicluster compound of two rows (1 and 3), two SNPs (2 and 3) and one trait (2):

$$\{1\ 3\ 2\ 3\ 2\ 2\ 2\ 1\} \Longrightarrow \begin{bmatrix} 2\ 1 & 0.3 \\ \\ 0\ 0 & -0.75 \end{bmatrix}$$

Variation Operators.

1. **Crossover**:
   A Single point crossover is used in the three first parts of the solution (rows part, columns part and traits part). Each part undergoes crossover separately. Let parents be chromosomes $P_1 = \{r_1 \ldots r_n\ c_1 \ldots c_m\ t_1 \ldots t_p\ r_{nb}\ c_{nb}\ t_{nb}\}$ and $P_2 = \{r'_1 \ldots r'_l\ c'_1 \ldots c'_k\ t'_1 \ldots t'_q\ r'_{nb}\ c'_{nb}\ t'_{nb}\}$ where $r_n \leqslant r'_l$.

Table 2. Example of SNPs and traits data matrix

SNPs			Traits	
1	**2**	**1**	12.5	**0.3**
0	1	2	10.75	1.2
1	**0**	**0**	10.33	–**0.75**

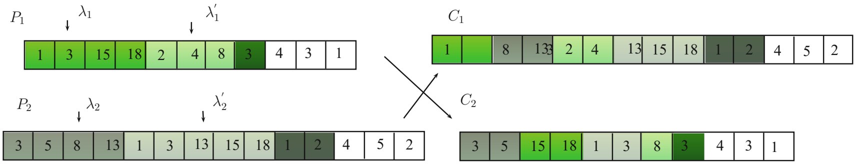

Fig. 1. An example of the crossover operator application.

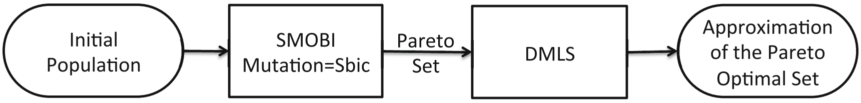

Fig. 2. General scheme of *SHMOBI*

The crossover in the rows part is performed as follows: The crossover point in P_1 (λ_1) is generated as a random integer in the range $2 \leqslant \lambda_1 \leqslant r_n$. the crossover point in P_2 $\lambda_2 = r'_j$ where $r'_j \geqslant \lambda_1$ and $r'_{j-1} \leqslant \lambda_1$. In the same way, the crossover in the columns part and traits part is performed. The parts 4–6 are not involved directly in the crossover and are computed after it.

For example, consider the parents P_1 and P_2 presented in Fig. 1. Suppose the 3^{rd} gene index and the 2^{nd} condition index of P_1 are selected, so: $\lambda_1 = 15$ and $\lambda'_1 = 5$ then $\lambda_2 = 16$ and $\lambda'_2 = 6$, which results on the offspring C_1 and C_2.

2. **Mutation**:
   We replace the mutation operator by the *Sbic* heuristic.

When generating random biclusters, it may happen that irrelevant rows and columns get included in spite of their values lying far apart. Therefore, we start by randomly generating a population where the irrelevant rows and columns of each bicluster are deleted using the *Sbic* heuristic. The resulting population is used as the initial population for $SMOBI_{ibea}$. After that, the $DMLS(1 \cdot 1_{\succ})$ is applied for each solution of $SMOBI_{ibea}$'s archive (Pareto approximation). The main steps of $SHMOBI_{ibea}$ are illustrated in Fig. 2.

4 Experiments and Results

In this section we present the experimental protocol in assessing the performance of the presented algorithms over synthetic data sets.

4.1 Data Sets

In order to assess the performance of the proposed algorithms, we use synthetic data sets to investigate the ability of our algorithms to extract implanted biclusters. In this purpose, we randomly generate different data sets of size:

Set1(100, (1000, 3)) which corresponds to 100 rows 1000 SNPs columns and 3 traits columns and Set2(100, (10000, 3)) which corresponds to 100 rows 10000 SNPs columns and 3 traits columns. For each data set we implant 1 (called Set1-1 et Set2-1) and 5 biclusters (called Set1-5 and Set2-5) with size 10 rows 50 SNPs columns. In each case, the biclusters may involve all (Set*-A) or some of the traits (Set*-T).

4.2 Comparison Criteria

In order to assess the performance of the proposed biclustering algorithm, we use the following two ratios:

$$\theta_{Shared} = \frac{S_{cb}}{Tot_{size}} \times 100 \tag{1}$$

Where S_{cb} is the portion size of bicluster correctly extracted and Tot_{size} is the total size of the implanted bicluster.

$$\theta_{NotShared} = \frac{S_{ncb}}{Tot^{'}_{size}} \times 100 \tag{2}$$

Where S_{ncb} is the portion size of bicluster not correctly extracted and $Tot^{'}_{size}$ is the total size of the extracted bicluster.

The ratio θ_{Shared} (resp. $\theta_{NotShared}$) expresses the rate of shared (resp. not shared) biclusters volume with real biclusters. In fact, when θ_{Shared} (resp. $\theta_{NotShared}$) is equal to 100 % the algorithm extracts the correct (resp. not correct) biclusters. A perfect solution has θ_{Shared} =100 % and $\theta_{NotShared}$=0 % respectively. That is, the exact number of rows and columns of implanted biclusters.

4.3 Parameters

Concerning the models parameters, we set α, β and γ to 0.5, 0, 0.5 respectively. In fact, given the nature of data, SNPs columns present low variance compared to trait columns. Hence, a big number of SNP columns will be added for each bicluster undergoing the *Sbic* heuristic. Therefore, we favor biclusters having low average variance and low SNPs columns to be selected in the search process and this by setting $\beta = 0$.

In the other hand, all algorithms parameters have been set experimentally. For the *Sbic* we set α to 1.5, δ to 0.15 and $\%p$ to 50 %. The algorithm is run 20 times in order to extract 20 biclusters. The first run uses all the data matrix. The remaining runs starts by sub-matrices where the rows and the columns are chosen randomly. When selecting rows, more chance is given to rows not present yet in the previously extracted biclusters.

Concerning $SMOBI_{ibea}$, we experimentally set the initial population size to 400. The mutation and crossover operators parameters are set to 0.2 and 0.5

respectively. The algorithm stops after a fixed time depending on the data set size. For Set1 data sets the execution time is set to 500 s, and 700 s for Set2 data sets. The same time is allocated to $SHMOBI_{ibea}$ algorithm where 90 % of the execution time is accorded to $SMOBI_{ibea}$ and the remaining 10 % to $DMLS(1 \cdot 1_{\succ})$.

We apply our algorithms on the considered data sets and for each algorithm we select the closest biclusters to the implanted ones. Thereafter, we calculate θ_{Shared} and $\theta_{NotShared}$ for each bicluster. For instances where several biclusters are implanted, we report the average θ_{Shared} and $\theta_{NotShared}$ of the extracted biclusters.

4.4 Results

In this section, we compare the efficiency of *Sbic*, $SMOBI_{ibea}$ and $SHMOBI_{ibea}$ in extracting the implanted biclusters. The comparison is done with regard to θ_{Shared}, $\theta_{NotShared}$ and the rate of found biclusters.

Tables 3 and 4 present the obtained results for the different instances corresponding to one and five implanted biclusters respectively. A detailed observation of the found solutions show that, in most cases, the not correctly biclusters extracted portions are mainly composed of extra columns (SNPs).

In Table 3 we can observe that all the approaches can find the implanted bicluster. However, $SHMOBI_{ibea}$ find the best results with the highest θ_{Shared} and lowest $\theta_{notShared}$. For instance, in the case of data Set1-1-A where all the traits are involved in the bicluster, $SHMOBI_{ibea}$ extracts the bicluster with only $\theta_{notShared} = 24.24\,\%$. Actually, $SMOBI_{ibea}$ is able to find the implanted bicluster. However, the $\theta_{NotShared}$ of the extracted bicluster is very high. This result demonstrates the role of $DMLS(1, 1_{\succ})$ in fine-tuning the found results.

Table 3. Comparative results when extracting one bicluster. *SMOBI* stands for $SMOBI_{ibea}$, *SHMOBI* for $SHMOBI_{ibea}$.

Data	θ_{Shared}			$\theta_{NotShared}$			Rate of found biclusters		
	Sbic	*SMOBI*	*SHMOBI*	*Sbic*	*SMOBI*	*SHMOBI*	*Sbic*	*SMOBI*	*SHMOBI*
Set1-1-A	78.6 %	100 %	100 %	86.6 %	80.07 %	24.24 %	100 %	100 %	100 %
Set2-1-A	100 %	100 %	100 %	86.07 %	86.73 %	57.01 %	100 %	100 %	100 %
Set1-1-T	60.0 %	100 %	100 %	78.05 %	92.46 %	76.36 %	100 %	100 %	100 %
Set2-1-T	30 %	90 %	100 %	81.41 %	95.12 %	67.12 %	100 %	100 %	100 %

Table 4. Comparative results when extracting five biclusters. *SMOBI* stands for $SMOBI_{ibea}$, *SHMOBI* for $SHMOBI_{ibea}$.

Data	θ_{Shared}			$\theta_{NotShared}$			Rate of found biclusters		
	Sbic	*SMOBI*	*SHMOBI*	*Sbic*	*SMOBI*	*SHMOBI*	*Sbic*	*SMOBI*	*SHMOBI*
Set1-5-A	50.62 %	52.5 %	85.92 %	86.37 %	85.61 %	60.9 %	60 %	80 %	80 %
Set2-5-A	63.33 %	64.16 %	85.83 %	92.87 %	91.11 %	92.26 %	60 %	60 %	60 %
Set1-5-T	41.66 %	64.88 %	62.61 %	87.27 %	82.04 %	82.44 %	40 %	80 %	80 %
Set2-5-T	45 %	75 %	85.83 %	98.21 %	94.5 %	93.47 %	40 %	40 %	60 %

Similarly, Table 4 shows that $SHMOBI_{ibea}$ outperforms *Sbic* and $SMOBI_{ibea}$ in finding the implanted biclusters. Actually, $SHMOBI_{ibea}$ finds more biclusters than the other approaches with higher θ_{Shared}. However, the $\theta_{NotShared}$ value of the biclusters extracted using all the approaches are relatively high. This can be explained by the huge number of SNPs columns in the data set.

Concerning running times, they are of 500 s for small instances (Set1-*) and 700 s for large instances (Set2-*).

5 Conclusion

In this article, we have presented a preliminary study on using a biclustering method to analyze GWA data. Actually, GWA data consists in two types of information *i.e.* phenotype data (traits) and genotype data (genetic variations). Commonly, SNPs are considered as they present the most frequent form of genetic variations. The analysis of such data consists in finding eventual associations between traits and SNPs combinations. Therefore, we propose a multiobjective modeling for biclustering in order to extract samples (individuals) sharing similar traits and having same alleles for a SNPs combination. The corresponding biclusters are constant columns biclusters.

The extracted biclusters may bring out existing associations between the considered SNPs and traits. Moreover, the extracted biclusters may provide important informations that can be used in further GWA studies. Given the huge number of SNPs, we propose to solve this problem using a hybrid metaheuristic $SHMOBI_{ibea}$. The efficiency of $SHMOBI_{ibea}$ have been assessed using synthetic data sets of different sizes and different implanted biclusters numbers. Further studies will be carried out in real data sets provided by the company *Genes Diffusions*[1].

References

1. Binder, H., Tina, M., Holger, S., Klaus, G., Michael, S., Jan, H., Katja, I., Martin, S.: Cluster-localized sparse logistic regression for SNP data. Stat. Appl. Genet. Mol. Biol. **11**(4), 13 (2012)
2. Bush, W.S., Moore, J.H.: Chapter 11: Genome-wide association studies. PLoS Comput. Biol. **8**(2), e1002822 (2012)
3. Cheng, Y., Church, G.M.: Biclustering of expression data. In: Proceedings of the 8th ISMB, pp. 93–103. AAAI Press, Menlo Park (2000)
4. Ding, L., Baye, T.M., He, H., Zhang, X., Kurowski, B.G., Martin, L.J.: Detection of associations with rare and common SNPs for quantitative traits: a nonparametric Bayes-based approach. BMC Proc. **5**(Suppl 9), S10 (2011)
5. Douglas, F., Trudy, M.: Introduction to Quantitative Genetics, 4th edn. Prentice Hall, Englewood Cliffs (1996)
6. Seridi, K., Jourdan, L., Talbi, E.-G.: Multi-objective evolutionary algorithm for biclustering in microarrays data. In: IEEE Congress on Evolutionary Computation, pp. 2593–2599. IEEE (2011)

[1] http://www.genesdiffusion.com/default.aspx

7. Lashkargir, M., Monadjemi, S.A., Dastjerdi, A.B.: A new biclustering method for gene expersion data based on adaptive multi objective particle swarm optimization. In: Proceedings of the 2009 Second International Conference on Computer and Electrical Engineering. ICCEE '09, vol. 01, pp. 559–563. IEEE Computer Society, Washington, DC, USA (2009)
8. Liefooghe, A., Humeau, J., Mesmoudi, S., Jourdan, L., Talbi, E.-G.: On dominance-based multiobjective local search: design, implementation and experimental analysis on scheduling and traveling salesman problems. J. Heuristics **18**(2), 317–352 (2012)
9. Lin, H., Desmond, R., Bridges, S.L., Soong, S.: Variable selection in logistic regression for detecting SNP-SNP interactions: the rheumatoid arthritis example. Eur. J. Hum. Genet. **16**(6), 735 (2008)
10. Liu, J., Chen, Y.: Dynamic biclustering of microarray data with MOPSO. In: 2010 IEEE International Conference on Granular Computing. GrC 2010, pp. 330–334. IEEE Computer Society, San Jose, CA, USA, 14–16 August 2010
11. Liu, J., Li, Z., Hu, X., Chen, Y.: Biclustering of microarray data with MOSPO based on crowding distance. BMC Bioinf. **10**(Suppl 4), S9 (2009)
12. Liu, J., Li, Z., Hu, X., Chen, Y.: Multi-objective ant colony optimization biclustering of microarray data. In: Granular Computing, pp. 424–429 (2009)
13. Liu, J., Li, Z., Liu, F., Chen, Y.: Multi-objective particle swarm optimization biclustering of microarray data. In: IEEE International Conference on Bioinformatics and Biomedicine, pp. 363–366 (2008)
14. Mitra, S., Banka, H.: Multi-objective evolutionary biclustering of gene expression data. Pattern Recogn. **39**(12), 2464–2477 (2006)
15. Seridi, K., Jourdan, L., Talbi, E.-G.: Hybrid metaheuristic for multi-objective biclustering in microarray data. In: CIBCB, pp. 222–228. IEEE (2012)