Journal of Computer and System Sciences ••• (••••) •••–•••

Contents lists available at SciVerse ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

# Three-objective subgraph mining using multiobjective evolutionary programming

Prakash Shelokar [a], Arnaud Quirin [a], Óscar Cordón [a,b,*]

[a] European Centre for Soft Computing, 33600-Mieres, Spain
[b] Department of Computer Science and Artificial Intelligence (DECSAI) and Research Centre on Information and Communication Technologies (CITIC-UGR), University of Granada, 18071-Granada, Spain

ARTICLE INFO

*Article history:*
Received 1 August 2012
Received in revised form 16 November 2012
Accepted 14 March 2013
Available online xxxx

*Keywords:*
Graph-based data mining
Frequent subgraph mining
Multiobjective optimization
Multiobjective graph mining
Multiobjective evolutionary programming
Subdue

ABSTRACT

The existing methods for graph-based data mining (GBDM) follow the basic approach of applying a single-objective search with a user-defined threshold to discover interesting subgraphs. This obliges the user to deal with simple thresholds and impedes her/him from evaluating the mined subgraphs by defining different "goodness" (i.e., multiobjective) criteria regarding the characteristics of the subgraphs. In previous papers, we defined a multiobjective GBDM framework to perform bi-objective graph mining in terms of subgraph support and size maximization. Two different search methods were considered with this aim, a multiobjective beam search and a multiobjective evolutionary programming (MOEP). In this contribution, we extend the latter formulation to a three-objective framework by incorporating another classical graph mining objective, the subgraph diameter. The proposed MOEP method for multiobjective GBDM is tested on five synthetic and real-world datasets and its performance is compared against single and multiobjective subgraph mining approaches based on the classical Subdue technique in GBDM. The results highlight the application of multiobjective subgraph mining allows us to discover more diversified subgraphs in the objective space.



## 1. Introduction

Many applications that contain complicated structures and relational objects rely on a graph-based data representation [1,2]. Some examples include scientific information analysis [3], bioinformatics [4], transportation networks [5], web data analysis [6], among others. Subgraph mining in graph-based data is the process of discovering subgraphs subject to some objective function. It usually involves applying some user-defined threshold, such as mining subgraphs whose frequency is above a specified threshold. For this task, several algorithms have been introduced in the graph-based data mining (GBDM) literature, starting with the classical heuristic search-based Subdue method [7] and being followed by some well-known exact search methods such as Gaston, gSpan, FSG, etc., [2,8]. Recently, evolutionary programming [9] has also been applied for frequent subgraph mining [10,11]. The proposal was basically an extension of Subdue and showed an improved performance over it. The performance improvement was a consequence of the use of global search instead of the beam search [12] with no backtracking as applied by the standard Subdue method in the subgraph search space.

Recently some important limitations of the existing approaches that operate by using simple user-defined constraints on the mined subgraphs have been highlighted in [5]. In addition, several authors [4,13–16] have noted that only employing

* Corresponding author at: European Centre for Soft Computing, 33600-Mieres, Spain.
*E-mail addresses:* prakash.shelokar@softcomputing.es (P. Shelokar), arnaud.quirin@softcomputing.es (A. Quirin), oscar.cordon@softcomputing.es, ocordon@decsai.ugr.es (Ó. Cordón).

0022-0000/$ – see front matter 
http://dx.doi.org/10.1016/j.jcss.2013.03.005

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005

the frequency-based subgraph discovery is not much interesting to properly solve some real-world applications. A good methodology should consider additional objectives as the complexity of the subgraphs being mined and their diversity. Taking into this consideration, a few approaches that perform multiobjective subgraph mining have been proposed [4,5, 17–20].

Multiobjective subgraph mining has been termed as *multiobjective GBDM* [4,17,20] or *skyline processing* [5]. The method proposed by Papadopoulos et al.'s [5], SkyGraph, employs two specific objectives, the order and the edge connectivity of the subgraph, to generate the Pareto-optimal subgraphs. However, the drawback of SkyGraph is that it is problem-specific, i.e., the algorithm design is characterized by the latter objective. Romero-Zaliz et al. [4] introduced the EMO-CC methodology (Evolutionary Multiobjective Optimization-based Conceptual Clustering) for the Gene Ontology domain. The method has solved a bi-objective problem using the support and the size of the mined subgraphs as objectives. However, EMO-CC has the important limitation of not being able to deal with general graphs, where a node may have several parents. Finally, MOSubdue (Multi-Objective Subdue), a Pareto dominance-based multiobjective subgraph mining algorithm, was developed by the authors in [17,21]. MOSubdue has performed multiobjective beam search using two objectives, support and size. It was also applied for a three-objective subgraph mining task by considering another objective, the density. MOSubdue is a general purpose multiobjective subgraph mining method, but it has the important limitation that its beam search does not allow backtracking in the subgraph search space.

Aiming to solve all these drawbacks, the authors also introduced in [18–20] a Multiobjective Evolutionary Propgramming (MOEP)-based approach to perform global search in the multiobjective subgraph solution space, thus allowing the user to obtain a good approximation to the Pareto-optimal subgraph set at a reasonable computational effort. An individual in the MOEP population is a subgraph in the input graph dataset. The input data is a set of connected relational graphs with or without cycles and directed or undirected edges. The individual is evaluated using two objectives, support and size of the subgraph. At any generation, parent individuals give rise to child individuals only through mutation, and subsequently the next generation is selected from the collection of parent and child individuals.

In this paper, we further extend the application of the latter MOEP-based GBDM method to solve a three-objective subgraph mining problem. An individual is evaluated using three objectives, namely, support, size, and diameter of the subgraph. The three-objective problem formulation is tested on two synthetic and three real-world datasets from the area of scientific information analysis [3]. The performance of MOEP is compared with single-objective Subdue, EP-Subdue, and MOSubdue algorithms. The comparison based on the different performance metrics (*C-metric* and *HVR-metric*) [22–24] shows superior performance of the multiobjective methods, and in particular of MOEP in the real-world graph datasets.

The paper is organized as follows. Section 2 defines the multiobjective subgraph mining problem. Section 3 describes the MOEP-based method for multiobjective subgraph mining. Section 4 discusses results of an experimental study and finally Section 5 provides conclusions and future works.

## 2. Multiobjective subgraph mining problem

In this section, we first give some basic definitions of the different objectives considered and then describe our multiobjective subgraph mining task.

### 2.1. Definitions

A labeled connected graph $G$ is denoted by a set of nodes $V(G)$ and a set of edges $E(G)$, where there is an edge $e_l$ between every pair of nodes $(v_i, v_j)$. Each node $v_i \in V(G)$ has a label from the node label set $L_V$, and each edge $e_l \in E(G)$ that connects two nodes $v_i, v_j$ has a label from the edge label set $L_E$. The edge $e_l$ can be directed or undirected. In this work, we consider a set of connected graphs $G = \{G_1, G_2, \ldots, G_n\}$.

In this study, we have used some of the commonly used preferences (or objectives) to evaluate a subgraph $S \in G$ which are given below as:

**Definition 1** *(Support of subgraph S).* The support or (frequency) of subgraph $S$ denoted by $sup(S)$ in the graph dataset $G$ is the cardinality of the set $\{G_i | S \subseteq G_i,\ i = 1, \ldots, n\}$.

**Definition 2** *(Size of subgraph S).* The size of subgraph $S$ denoted by $size(S)$ is the number of nodes and edges present in the subgraph $S$.

**Definition 3** *(Diameter of subgraph S).* The diameter of subgraph $S$ denoted by $dia(S)$ is the greatest distance between any pair of nodes. It is measured as the number of edges (or links) between the furthest nodes $(v_i, v_j)$ in the subgraph $S$. In this study it is evaluated using the Dijkstra's algorithm [25].

These objectives have been commonly applied in the frequent subgraph mining literature primarily to guide single-objective search methods by posing some threshold in the mining process [26–28].

The frequent subgraph mining problem is commonly modeled using a subgraph lattice [8]. Fig. 1 represents a subgraph lattice for a toy graph dataset $G = \{G_1, G_2, G_3\}$. The subgraph lattice in Fig. 1 models the search space in the dataset $G$ as

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005

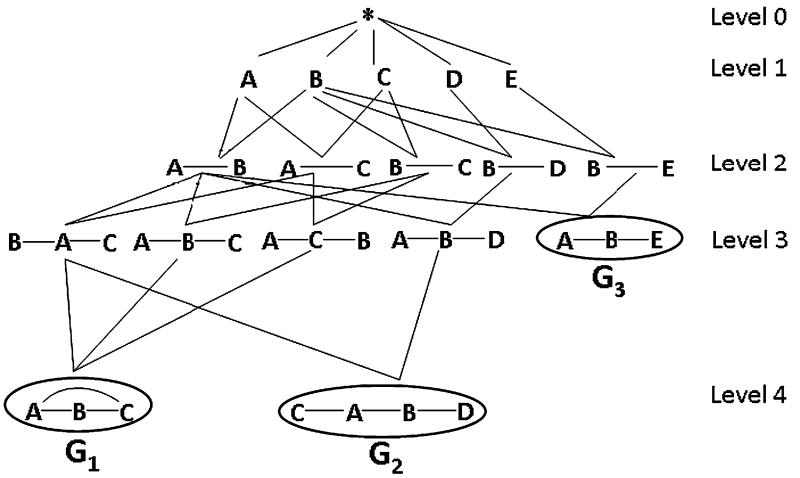


**Fig. 1.** A subgraphs lattice of the graph dataset $G = \{G_1, G_2, G_3\}$. (The figure based on a similar figure presented in Jiang et al. [29].)

follows. The top of the lattice, i.e., level 0, represents the empty subgraph labeled with *. The first level shows all possible single node subgraphs containing just one node with zero edges. The second level of the lattice lists subgraphs with one edge, and so on. At the bottom of the lattice, graphs in the dataset $G$ are shown. Frequent subgraph mining problem can be formulated as finding the embedding subgraphs in the lattice. For example, a subgraph in level 1 with just single node labeled as A can be embedded in two single edge subgraphs in the level 2 having other nodes labeled as B and C, respectively. In Fig. 1, the subgraph A–C in level 2 is a parent of the subgraph B–A–C in level 3 of the lattice as the subgraph B–A–C is different from the subgraph A–C by exactly one edge. Thus, the subgraph B–A–C is a child of the subgraph A–C. All the subgraphs of $G_i \in G$ are present in the lattice and every subgraph occurs only once in it.

### 2.2. Problem formulation

The multiobjective subgraph mining problem tackled in this contribution is defined considering a general definition of multiobjective optimization problems [22,30,31]. In our case, a solution is a subgraph $S$, a set of nodes and edges, and the solution space is the subgraph lattice (e.g. Fig. 1). The subgraph $S$ is evaluated considering $d$ different objectives on the subgraph's characteristics, such as the support, the size, etc., which are conflicting among them. For example, a subgraph with a high support is usually of a small size and *vice-versa*. As an example, consider two subgraphs A–B and B–A–C in the subgraph lattice space depicted in Fig. 1. The subgraph A–B in level 2 is of size three (two nodes and one edge) and has a support of four being embedded in as many subgraphs in level 3. On the other hand, the subgraph B–A–C in level 3 is of size five and has a support of two in level 4. Clearly, these objective functions, the support and the size of the subgraph, are competing in nature.

Formally, we give a problem definition for multiobjective GBDM as follows. Given a graph dataset $G$, mine the Pareto-optimal subgraphs representing all the connected subgraphs in $G$ defined by three user-defined objectives:

$$F(S) = \big(f_1(S), f_2(S), f_3(S)\big) \tag{1}$$

where

$$f_1 = \text{Max. support},\ sup(G, S) = \#N(G_i | S \subseteq G_i,\ i = 1, \ldots, n)$$

$$f_2 = \text{Max. size},\ size(S) = \#V(S) + \#E(S)$$

$$f_3 = \text{Min. diameter},\ dia(S) = \#\, max_{v_i, v_j \in V(S)}\, d(v_i, v_j)$$

subject to

$$S \in X \tag{2}$$

$$F(S) \in Y \tag{3}$$

where $\#N(.)$ is the number of graphs in $G$ which contains the subgraph $S$. $\#V(.)$ and $\#E(.)$ return the number of nodes and edges of the subgraph $S$, respectively. In the diameter function, $dia(S)$, the distance between any two vertices, $d(v_i, v_j)$, is the number of edges of the shortest path between $v_i$ and $v_j$. $X$ is the subgraph search space, and $Y$ is the objective space. The objectives used in this study are clearly conflicting in nature. For example, maximization of the size objective directs the search to find large subgraphs while minimization of the diameter objective aims to find small dense subgraphs.

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005

Solution to the problem in Eq. (1) is a set of optimal subgraphs in $X$, which represent different trade-offs in the objective space $Y$. To compare any pair of subgraphs, we apply the well-known concept of dominance [22,30,31]. For simplicity, we consider maximization of all the objectives. For this purpose, we convert a minimization of an objective function (i.e., diameter, in our case) into the maximization of another one by taking its opposite value. Suppose, we have two objective vectors $u = (u_1, \ldots, u_d)$, $v = (v_1, \ldots, v_d) \in Y$, corresponding to the subgraphs $S_1, S_2 \in X$, respectively. $u$ is said to dominate $v$ (denoted by $u \succcurlyeq v$) if $u$ is greater than or equal to $v$ in all objectives, and is strictly greater than $v$ in at least one objective, i.e., $\forall i \in \{1, 2, \ldots, d\}$: $u_i \geqslant v_i \wedge \exists j \in \{1, 2, \ldots, d\}$: $u_j > v_j$. This definition can also be applied for minimization or any condition of objectives.

The subgraph $S \in X$ with objective vector $u$ is said to be Pareto-optimal with respect to the search space $X$ iff there is no subgraph $S' \in X$ with objective vector $u'$ that dominates $S$. For the multiobjective subgraph mining problem in Eq. (1), the Pareto-optimal set $\mathcal{P}$ is defined as:

$$\mathcal{P} := \left\{ S \in X \;\middle|\; \neg\exists\, S' \in X \; F(S) \preccurlyeq F\left(S'\right) \right\} \tag{4}$$

and the Pareto-optimal front $\mathcal{PF}$ associated with the Pareto-optimal set $\mathcal{P}$ is defined as:

$$\mathcal{PF} := \left\{ F(S) = \left( f_i(S), \ldots, f_d(S) \right) \;\middle|\; S \in \mathcal{P} \right\} \tag{5}$$

For the problem in Eq. (1), the algorithm produces a set of nondominated or Pareto subgraphs $P$ and the corresponding nondominated front $PF$. $PF$ is also called the Pareto, approximation, or efficient front.

## 3. Multiobjective evolutionary programming for subgraph mining

Recently a MOEP-based approach for a bi-objective subgraph mining was proposed in [18–20]. Two different MOEP methods were implemented based on the use of two different selection mechanisms for the individuals in the population at any generation. The first method, MOEP-NS, used NSGA-II's nondominated sorting (NS) approach [32], while the second method, MOEP-SO, applied the summation of objectives (SO) approach [33]. The comparison of results on several graph datasets has shown superior performance of MOEP-SO [20]. Therefore, in this study, we will apply MOEP-SO for our three-objective subgraph mining problem.

In MOEP-SO, an individual in the population $R$ is represented as a possible subgraph $S$ in the graph dataset $G$. $R$ is a set of subgraphs $S_1, S_i, \ldots, S_{|R|} \in X$, where $S_i$ is a connected subgraph within the graphical representation for all those subgraph instances in $G$ that match to the subgraph $S$. This graphical representation serves as a solution to the problem defined in Eq. (1). Thus, an individual in the population $R$ is always composed of a subgraph $S$ with its associated instances belonging to $G$. Consider an example depicted in Fig. 1, the subgraph A–B in level 2 has three instances in the input graph dataset $G$. The different steps of MOEP-SO algorithm are given as follows.

### 3.1. Initialization

Initially, the population $R$ contains randomly generated individuals. In this work, a simple procedure is applied to initialize the population. First, all one-edge subgraphs are created from unique label nodes in $G$. These subgraphs share the same values for two of the objectives, size (i.e., two nodes and one edge) and diameter (i.e., one link), but may have different values for the remaining objective, support. The subgraph lattice in Fig. 1 shows all one-edge subgraphs in level 2 obtained from five different node labels present in the input dataset $G$. The initial population contains subgraphs randomly selected from these one-edge subgraphs. More sophisticated initialization procedures can also be applied that may well represent different search space subgraphs.

### 3.2. Subgraph generation

To generate a child subgraph $S'$, a mutation operation is applied on a subgraph $S$ encoded in a parent individual in the population $R$. Mutation creates child instances by extending all instances of the parent subgraph $S$ in the dataset $G$ by an edge (and a node if no cycle is closed). For example, the subgraph lattice in Fig. 1 shows extending the parent subgraph A–B by an edge and node creates four children in level 3 of the subgraph lattice. A child instance is then randomly selected that becomes a child subgraph in graphical representation. All of the child instances belonging to $G$ that match this child subgraph become its new instances in $G$. This child subgraph must have at least two instances to qualify as a child subgraph of the parent subgraph $S$. Otherwise a new child instance of $S$ will be randomly selected to form a child subgraph. This is the most commonly used subgraph generation method in GBDM techniques [34,35]. Mutation is applied on each parent individual in $R$ to create the child population $Q$. All child subgraphs in $Q$ are evaluated using the three objectives in Eq. (1).

### 3.3. Subgraph selection

For the next generation, a new parent population $R$ is constructed by diversified selection of individuals from a temporary population $R \cup Q$. To this end, in the temporary population, a range is computed for each objective as the difference

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005

```
1. MOEP-SO (Graph G, pop size |R|, gen MaxGen, archive size |Archive|)
2. Initial Subgraph population, R = RandomSelection(one-edge subgraphs)
3. Evaluate subgraphs in R using objective functions in Eq. (1)
4. Nondominated subgraphs archive, Archive = {}
5. Archive = UpdateArchive(Archive, R) //nondominance criteria
6. while MaxGen > 0 do
7.    Child Subgraphs population, Q = {}
8.    for each parent p ∈ R
9.       Q = Q ∪ Mutation(p) //child generation
10.   Evaluate subgraphs in Q using objective functions in Eq. (1)
11.   Archive = UpdateArchive(Archive, Q )
12.   Combine two populations, R ∪ Q
13.   New population R = SubgraphSelection( R ∪ Q ) //summation of objectives method
14.   MaxGen = MaxGen − 1
15. end while
16. Return Archive // the nondominated subgraphs set
```

**Fig. 2.** The outline of MOEP-SO algorithm.

**Table 1**
Description of different graph datasets used.

| Dataset | #Graphs | #Nodes | #Edges | #Unique labels | MOEP run time (secs) |
|---|---|---|---|---|---|
| *random*1 | 100 | 2954 | 3009 | 7 | 1742 |
| *random*2 | 200 | 5876 | 6015 | 7 | 3075 |
| *US* | 10 | 2762 | 2769 | 294 | 1452 |
| *UK* | 10 | 2732 | 2748 | 292 | 787 |
| *Germany* | 10 | 2676 | 2702 | 284 | 1425 |

between the minimum and maximum values. Each range is then used to normalize the corresponding objective values. The fitness of an individual is computed as the summation of the normalized objective values. A minimization of fitness value is assumed. To select the diversified individuals from $R \cup Q$, each of the objectives is selected and its range is divided into 100 bins and some 80 percent of the bins are scanned. For each non-empty bin an individual with the smallest fitness value is chosen as an individual of the new population $R$.

### 3.4. External archive

MOEP-SO stores the nondominated subgraphs separately in an external archive *Archive* which is updated at the end of each generation. *Archive* is updated using the subgraph set $Q$ and the dominated subgraphs, if any, from *Archive* are removed. The output of MOEP-SO is the set of nondominated subgraphs collected in *Archive* after the algorithm run.

The outline of the MOEP-SO algorithm is given in Fig. 2. Inputs to MOEP-SO are graph dataset $G$, population size $|R|$, archive size $|Archive|$, and maximum number of generations *MaxGen* to perform the search.

## 4. Experimental study

The performance of the MOEP-SO algorithm for the tackled three-objective subgraph mining task (as defined in Section 2.2) is analyzed by means of unary and binary metrics [22–24] and visual representations of the obtained *PF* approximations. Five different graph datasets are considered in the experimentation developed (see Section 4.1). For comparison purposes, we also apply single-objective Subdue [7,34] and EP-Subdue [10,11] using three different objective functions to produce aggregated PFs on several graph datasets (see Section 4.2). Besides, MOSubdue [17] is applied for a broader performance study (see Section 4.3). All the considered methods are implemented in C and all experiments are performed on an Intel Core Quad at 2.66 GHz, with 4 GB RAM, running CentOS 5.5, using the parameter values reported in Section 4.4. Section 4.5 collects the obtained results and the analysis developed.

### 4.1. Graph datasets used

The performance evaluation study is conducted using two synthetic and three real-world datasets. Table 1 summarizes a few characteristics of the employed datasets, such as the number of nodes, the number of edges, etc. The datasets are of different sizes and consist of varying degrees of nodes and unique labels.

The first two datasets, *random*1 and *random*2, were synthetically generated using the random graph generator available at Subdue's website.[1] The graph generation program takes several parameters to generate the output graph. The basic

[1] http://ailab.wsu.edu/subdue/datasets/subgen.tar.gz.

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005

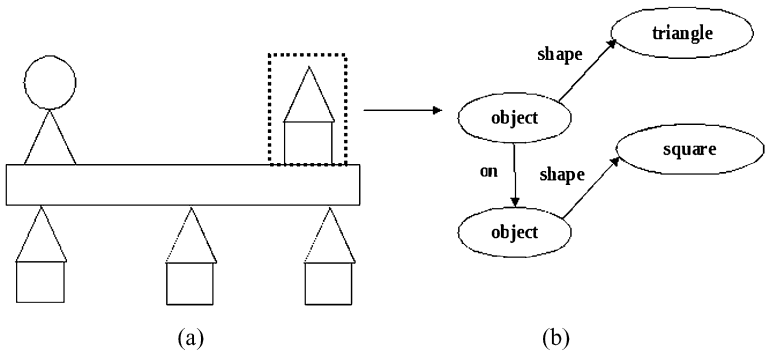


**Fig. 3.** A substructure to be embedded in random graphs. (a) shows a drawing of the substructure, and (b) highlights a graph representation for a portion of the substructure.

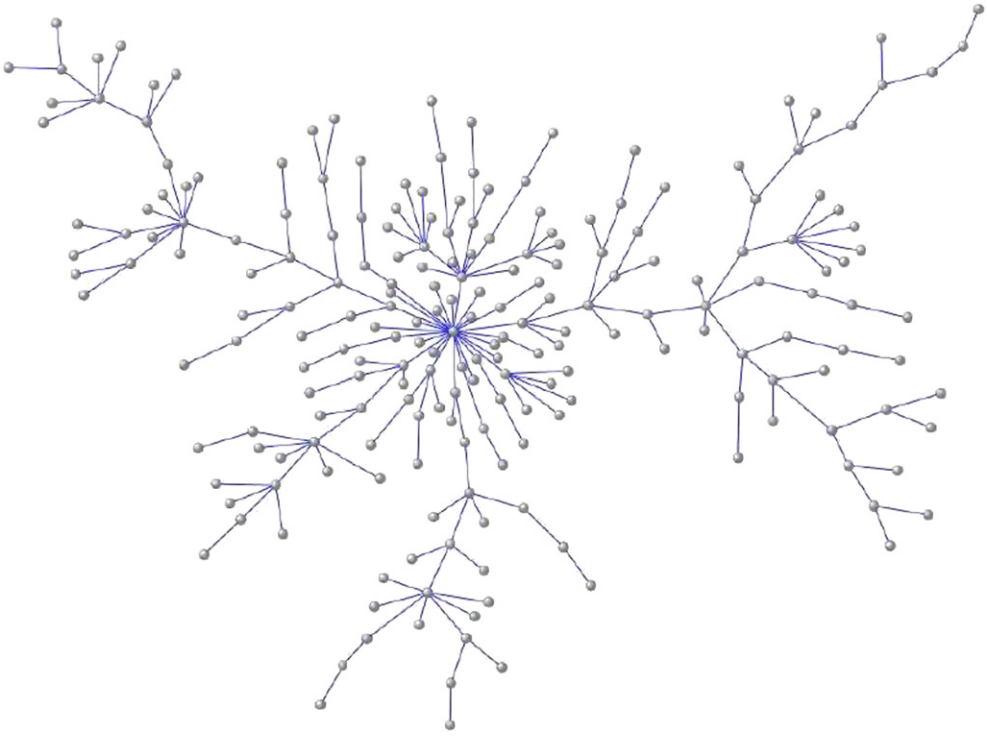

**Fig. 4.** A scientogram of the European scientific domain in year 2002 (category names, are not shown to improve the readability).

parameters are a substructure to be embedded in the output graph and the number of nodes and edges in the final graph. A sample substructure to be embedded is depicted in Fig. 3(a) and Fig. 3(b) shows the graph representation for a highlighted part of the substructure in Fig. 3(a). The graph generation program used different random numbers to generate graphs with the average number of nodes and edges equal to 60 and keeping other parameters settings to default.

The remaining three real-world datasets were generated from the world scientograms database [3]. The scientograms database is built following de Moya-Anegón et al.'s methodology [36] to design visual science maps (scientograms) for huge scientific publications collections. The rough considered data have been extracted from the Scimago Journal & Country Rank portal[2] and comprise a set of 36 millions documents indexed in Elsevier Scopus from 1996 to 2008 over 73 countries [36]. The nodes of the graphs correspond to Elsevier SCOPUS-SJR[3] co-citation categories. Only the salient relationships between categories are kept, capturing the essential underlying intellectual structure of the studied scientific domain, using the Pathfinder social networks algorithm [37] to prune the graphs. An example of one such scientogram is shown in Fig. 4. Recently, this database has been extensively analyzed in [3] to propose an automatic Subdue-based approach for

[2] http://www.scimagojr.com/.
[3] http://www.scopus.com.

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005

the identification and the comparison of scientific structures within scientograms. In our experimental study, we have used three datasets compiled for United States (*US*), United Kingdom (*UK*), and Germany over the period of 10 years from 1996 to 2005.

### 4.2. Nondominated subgraph generation using single-objective methods

Two single-objective search methods for frequent subgraph mining were implemented. They are briefly described along with the parameter settings in the following subsections.

#### 4.2.1. Single-objective Subdue method

Subdue [7,34] is a classical method in GBDM. It performs a constrained beam search [12] in the subgraph search space by defining a *beam-width* parameter. To evaluate a subgraph, Subdue uses a measure based on the MDL principle [38], which assumes the best subgraph is the one that minimizes the description length of the input graph when compressed by the subgraph [7]. An implementation of the Subdue algorithm is publicly available at Subdue's website.[4] In this work, we have only modified the subgraph evaluation function of the Subdue algorithm. Thus, instead of the MDL measure, now Subdue evaluates a subgraph using three different objective functions defined in Eq. (1). To obtain a nondominated set on the graph dataset $G$, Subdue was executed with each of the three subgraph evaluation functions independently. The final output of Subdue was an aggregation of three different outputs with removal of repeated subgraphs, if any, and the application of nondominance criteria to return the nondominated subgraphs.

#### 4.2.2. Single-objective EP-Subdue method

EP-Subdue [10,11] is a simple improvement to the constrained beam search of Subdue by maintaining a population of subgraphs. Unlike Subdue, at any generation, this EP-Subdue method utilizes a population of subgraphs in order to explore different regions of the subgraph search space. Except the beam search, EP-Subdue utilizes the remaining implementation of Subdue. In this work, we have implemented EP-Subdue by modifying the Subdue implementation available at Subdue's website (see footnote 4). To produce a nondominated set on the graph dataset $G$, a single run of EP-Subdue per objective was carried out. The final output of EP-Subdue was an aggregation of three different outputs with the removal of repeated subgraphs, if any, and the application of nondominance criteria to return the nondominated subgraphs. In EP-Subdue, the algorithm parameter is subgraph population size.

### 4.3. Multiobjective Subdue method

MOSubdue [17,21] is one of the first general-purpose multiobjective GBDM methods in the specialized literature. MOSubdue performs a multiobjective subgraph selection to guide Subdue's beam search. Two different implementations of MOSubdue were proposed in [17], where the first one is based on NSGA-II's NS procedure was purely deterministic while the second also incorporated crowding [21] performing stochastic search. The comparison of results of the two variants of MOSubdue revealed a superior performance of MOSubdue with stochastic search [17]. Therefore, in this work, we have used MOSubdue with stochastic search approach to solve 3-objective problem defined in Eq. (1). Like Subdue, MOSubdue has *beam-width* as the only algorithmic parameter.

### 4.4. Parameter settings

Subdue is a deterministic heuristic search method. Hence, it is applied once on each dataset. Subdue's beam search parameter, *beam-width*, was set equal to 5 after a preliminary experimentation. The output of Subdue was set to return a maximum of 100 best subgraphs corresponding to each of the three subgraph evaluation functions. These outputs were combined and a maximum of 100 nondominated subgraphs were returned as the final output of Subdue.

EP-Subdue is a stochastic search method. Hence, it was run 10 times on each dataset with different random seeds. The algorithm parameter, subgraph population was set to 100 and the output of EP-Subdue was set to return a maximum of 100 best subgraphs corresponding to each of the three subgraph evaluation functions. Three different outputs were combined and a maximum of 100 nondominated subgraphs were returned as the final output of EP-Subdue.

As MOSubdue performs stochastic search, it was executed 10 times independently on each dataset. The *beam-width* parameter was set to 5 after a preliminary experimentation. The output of MOSubdue was set to return a maximum of 100 nondominated subgraphs.

MOEP-SO is a pure stochastic search approach for multiobjective subgraph mining. The parameter settings were population $|R| = 100$, external Pareto archive $|Archive| = 100$. Like MOSubdue, MOEP-SO was executed 10 times independently on each dataset.

All the considered methods have stored the nondominated subgraphs externally with the maximum limit of 100. For all the datasets used, none of the methods in any of their executions could produce a number of nondominated subgraphs

[4] http://ailab.wsu.edu/subdue/software.

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005

**Table 2**
The *HVR-metric* values for the nondominated sets found by different methods. The numbers in the parentheses represent the standard deviation.

| Dataset | Subdue | EP-Subdue | MOSubdue | MOEP-SO |
|---|---|---|---|---|
| *random*1 | 0.7421 | 0.6955(0.01) | **0.9933(0.0)** | 0.9456(0.0) |
| *random*2 | 0.7446 | 0.6904(0.01) | **0.9902(0.0)** | 0.9522(0.0) |
| *US* | 0.3166 | 0.2291(0.02) | 0.4219(0.10) | **0.8446(0.04)** |
| *UK* | 0.3636 | 0.2580(0.03) | 0.4785(0.11) | **0.8616(0.10)** |
| *Germany* | 0.4190 | 0.2678(0.02) | 0.4641(0.07) | **0.8291(0.05)** |
| **Average** | 0.5171(0.47) | 0.4281(0.24) | 0.6696(0.55) | **0.8866(0.56)** |

that surpass the maximum archive size limit of 100. For safer side, all the methods used crowding measure [32] to prune the nondominated set whenever it exceeds the limit. However, when datasets have a large number of nondominated and repetitive subgraphs, a larger limit on the archive size could be used.

In this study, both Subdue and MOSubdue were run till exhaustion, i.e., until no subgraph growth possible, on each dataset. To have a fair comparison between the EP-based methods and MOSubdue-II, MOEP-SO and EP-Subdue have used a fixed run time as given in Table 1. This run time was determined from the average run time of 10 different executions of MOSubdue for each dataset.

### 4.5. Experimental analysis

The performance comparison study of different algorithms for multiobjective optimization is more complex than in the case of single-objective optimization. To this end, different unary and binary metrics are proposed in the EMO community [22–24]. The unary metric computes some score for Pareto front approximation that reflects a certain quality aspect. Although unary metrics let us determine the absolute, individual quality of the Pareto front approximation, they cannot be used for comparing the nature of different Pareto front approximations. To do so, the binary metrics are introduced which let us compare in pairs the different Pareto front approximations. Therefore, it is common to apply both types of metrics for the performance study [22–24]. In this work, we have utilized the *HVR-metric* and the *C-metric* which are the most commonly used unary and binary metrics, respectively, in the EMO-literature [22–24]. The *HVR-metric*, the hypervolume ratio, is to compare the nondominated subgraph set $P$ produced by an algorithm with respect to the Pareto-optimal subgraph set $\mathcal{P}$. The *HVR-metric* is computed as the ratio of the areas/hypervolumes enclosed by the nondominated front *PF* and the true Pareto-optimal front $\mathcal{PF}$. For the set $P$, the *HVR*-metric value is better when it tends to one. In our experimental study, the set $\mathcal{P}$ is not known beforehand for any of the employed datasets. Therefore, we have used a pseudo-optimal nondominated subgraph set generated from the aggregation of the sets $P$ produced by every method in every run. The *C-metric* uses dominance criteria to compare in pairs the nondominated sets produced by different algorithms. The $C$-metric is also better when it tends to one. It is computed in pair $C(Z', Z'')$ as the fraction of the nondominated set $Z'$ that covers the nondominated set $Z''$ [23]:

$$C(Z', Z'') = \frac{|\{\forall S'' \in Z''; \ \exists S' \in Z' : \ S' \succcurlyeq S''\}|}{|Z''|} \tag{6}$$

where $S' \succcurlyeq S''$ indicates that the subgraph $S'$ dominates or covers the subgraph $S''$ in a maximization problem. A value of $C(Z', Z'') = 1$ means that all the subgraphs in $Z''$ are dominated or covered by the subgraphs in $Z'$.

Table 2 presents the mean and standard deviation of the *HVR-metric* values of the nondominated set approximations achieved by the different algorithms except Subdue for each dataset. The *HVR-metric* values reported for the deterministic Subdue method are corresponding to a single run of the algorithm on each of the datasets. Table 2 also provides overall performance of each of the methods by averaging the different *HVR-metric* values over the five datasets.

Fig. 5 shows the assessment of different algorithms in pairs using the *C-metric*. For an ordered algorithm pair $(A, B)$, there is a sample of 10 *C-metric* values according to the 10 runs performed. Each value is computed on the basis of nondominated sets achieved by $A$ and $B$ with the same initial population. Note that, in case of deterministic Subdue, a single run was performed on each dataset. Here, box-plots are used to visualize the distribution of these samples.

For illustrative purpose, Figs. 6 and 7 show the plots of the approximation set *PF* to the set $\mathcal{PF}$ achieved by different algorithms on the *US* and *UK* datasets, respectively. The plots corresponding to Subdue are the approximations generated by the three runs of the algorithm, each with different objective; while those corresponding to the remaining methods are the aggregation of the results of 10 different runs with a different random seed. The fused outputs of stochastic methods are only used for graphical representation. For the sake of a better visual representation, the *PF* plots are grouped into two figures per dataset. They are grouped according to the average *HVR*-metric values, considering a figure for the worst performing and another for the best performing algorithms.

The *HVR-metric* values in Table 2 indicate that the single-objective search methods (Subdue and EP-Subdue) on all the datasets have produced the worst approximations to $\mathcal{PF}$ as compared to those achieved by the multiobjective search methods (MOSubdue and MOEP-SO). This conclusion is further reinforced from the average value of the *HVR-metric* over all the datasets, which is much lower for Subdue (0.5171) and EP-Subdue (0.4281) as compared to both multiobjective search

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005

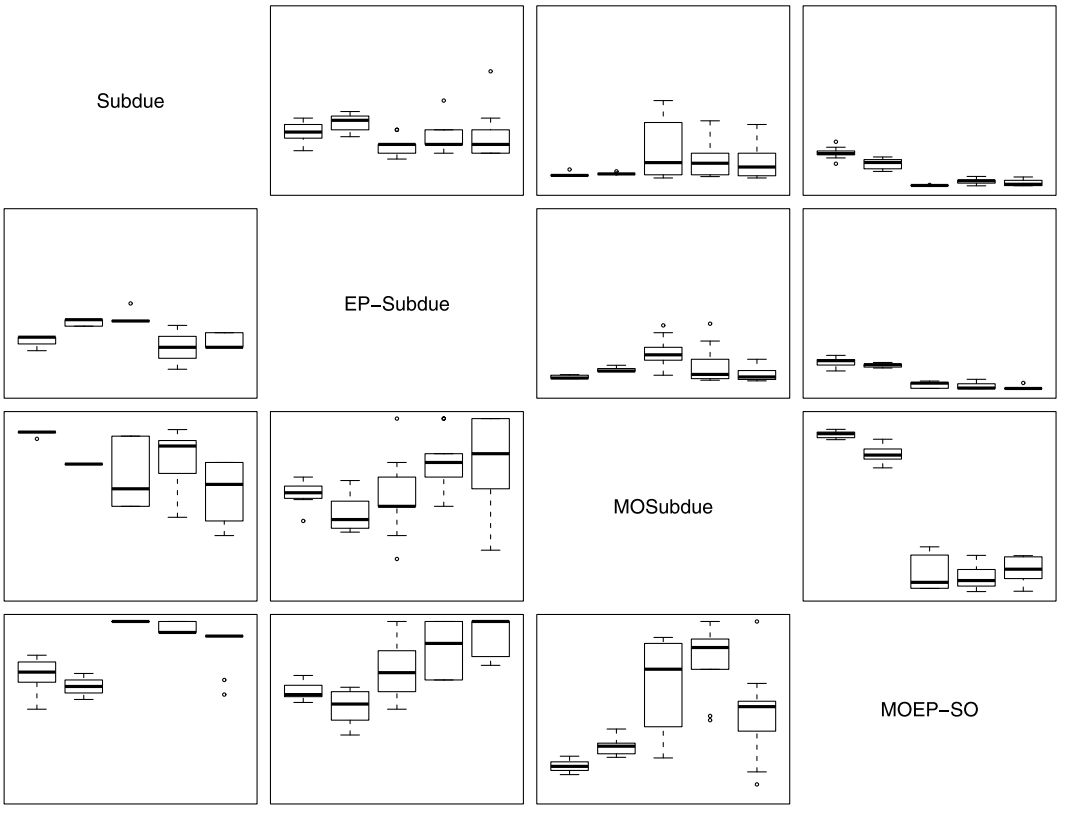


**Fig. 5.** Box-plots based on the $C$-metric computed for the different methods considered. Each rectangle contains 5 box-plots representing the distribution of the $C$-metric values for a certain ordered pair of algorithms. The leftmost box-plot relates to *random*1 dataset, the rightmost to *Germany* dataset.

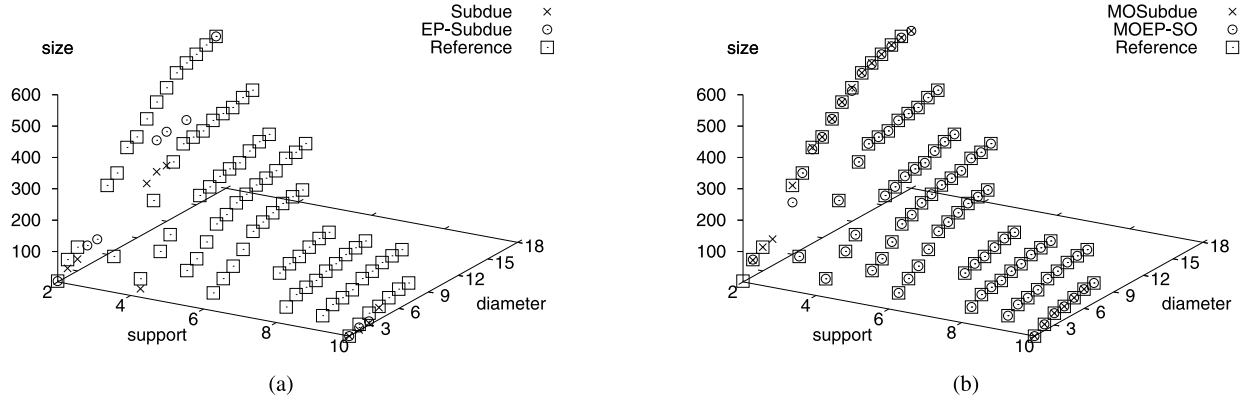


**Fig. 6.** The nondominated set approximations produced by different algorithms on the *US* dataset. The pseudo Pareto-optimal front $\mathcal{PF}$ is also shown as a reference.

methods. This confirms the incorporation of multiobjective search strategy enables the algorithm to explore more efficiently the multiobjective subgraph search space.

The comparison of *HVR-metric* values in the case of the multiobjective search-based methods show that MOEP-SO is the best performer on the three real-world datasets, while MOSubdue is superior on the two synthetic datasets. Overall, MOEP-SO has achieved the best average *HVR-metric* value of 0.8866 as compared to that of 0.6696 attained by MOSubdue. However, on the two synthetic datasets MOEP-SO has shown somewhat inferior performance as against MOSubdue.

Further comparing the *C-metric* values of the different methods, Fig. 5 also confirms that the *PF* approximations achieved by the multiobjective search-based methods have more coverage over those obtained by the single-objective search-based methods. The comparison between MOEP-SO and MOSubdue reveals that MOEP-SO has outperformed MOSubdue in terms of coverage on the three real-world datasets. As against, MOSubdue has again attained the best coverage on the two synthetic datasets.

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005

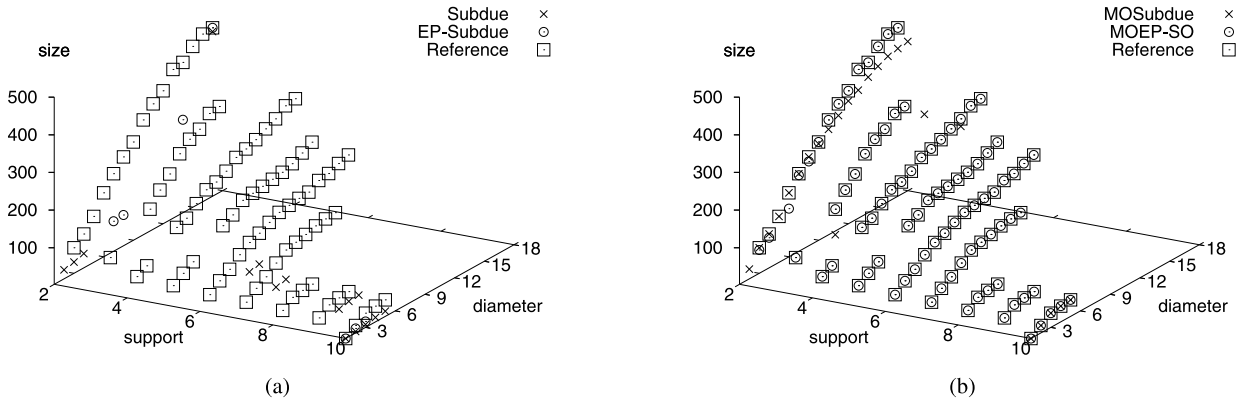


**Fig. 7.** The nondominated set approximations produced by different algorithms on the *UK* dataset. The pseudo Pareto-optimal front $\mathcal{PF}$ is also shown as a reference.

In addition, the graphical representations in Figs. 6 and 7 show the superior performance of MOEP-SO and MOSubdue methods. Comparing the plots of MOEP-SO and MOSubdue methods, it can be seen how MOEP-SO attained a wider spread of subgraphs on the set $\mathcal{P}$.

Finally, we performed an in depth analysis of the subgraph generation process of MOEP-SO on the synthetic datasets in order to get some insights into the inferior performance of MOEP-SO on these datasets. At the beginning of the search process, a parent subgraph has several repetitive instances in a graph of the dataset. As against, the definition of the objective support assumes just one occurrence (and no repetition) in any graph of the dataset. The different repetitive instances of the subgraph bring huge redundancy in the mutation operation for a child generation. The current definition of the objective support fails to take into account this redundancy during subgraph selection. Thus, a new definition of the objective support is needed to apply MOEP-SO efficiently and effectively in such scenarios. Note that, in the real-world scientogram datasets, a parent subgraph has no repetitive instances in a graph of the dataset and thus there is no such additional redundancy in the mutation operation for a child generation.

## 5. Conclusions

This contribution has successfully shown the application of MOEP for a three-objective subgraph mining problem. The performance of the proposed multiobjective GBDM method has been tested on five datasets and compared against single- and multiobjective Subdue-based methods. On all the datasets, the multiobjective subgraph mining methods have shown superior performance over the single-objective ones. The results confirm the application of multiobjective subgraph mining can discover more diversified subgraphs in the objective space. Overall, MOEP-SO is the best performer followed by MOSubdue. Nevertheless, we should remark that MOSubdue outperformed MOEP-SO on the two synthetic graph datasets. To this end, a new definition of the objective support is required to handle the subgraph selection pressure in the presence of subgraphs with repetitive instances in a graph of the input dataset. Future studies will also include a few more formulations of the multiobjective subgraph mining problem, a performance study on large scale datasets, and the development of a genetic algorithm for multiobjective subgraph mining.

## Acknowledgments

This work has been partially supported by the Spanish Ministry of Science and Innovation (MICINN) under project TIN2009-07727, including EDRF fundings. The first author acknowledges the partial support received from MICINN under the Juan de la Cierva programme JCI-2010-07626.

## References

[1] C. Aggarwal, H. Wang (Eds.), Managing and Mining Graph Data, Springer, 2010.
[2] D. Cook, L. Holder (Eds.), Mining Graph Data, Wiley, London, 2007.
[3] A. Quirin, Ó. Cordón, B. Vargas-Quesada, F. de Moya-Anegón, Graph-based data mining: A new tool for the analysis and comparison of scientific domains represented as scientograms, J. Informetr. 4 (2010) 291–312.
[4] R.C. Romero-Zaliz, C. Rubio-Escudero, J.P. Cobb, F. Herrera, Ó. Cordón, I. Zwir, A multiobjective evolutionary conceptual clustering methodology for gene annotation within structural databases: A case of study on the gene ontology database, IEEE Trans. Evol. Comput. 12 (2008) 679–701.
[5] A. Papadopoulos, A. Lyritsis, Y. Manolopoulos, SkyGraph: An algorithm for important subgraph discovery in relational graphs, Data Min. Knowl. Discov. 17 (2008) 57–76.
[6] A.J. Lee, Y.-A. Chen, W.-C. Ip, Mining frequent trajectory patterns in spatial–temporal databases, Inform. Sci. 179 (2009) 2218–2231.
[7] D.J. Cook, L.B. Holder, Substructure discovery using minimum description length and background knowledge, J. Artificial Intelligence Res. 1 (1994) 231–255.

[8] I. Fischer, T. Meinl, Graph based molecular data mining – An overview, in: W. Thissen, P. Wieringa, M. Pantic, M. Ludema (Eds.), Proc. IEEE Int. Conf. Systems, Man & Cybernetics, vol. 5, pp. 4578–4582.
[9] D. Fogel, System Identification through Simulated Evolution: A Machine Learning Approach to Modeling, Ginn Press, 1991.
[10] S. Bandyopadhyay, U. Maulik, D.J. Cook, L.B. Holder, Y. Ajmerwala, Enhancing structure discovery for data mining in graphical databases using evolutionary programming, in: Int. Conf. Florida Artificial Intelligence Research Society (FLAIRS), 2002, pp. 232–236.
[11] U. Maulik, Hierarchical pattern discovery in graphs, IEEE Trans. Syst. Man Cybern. C 38 (2008) 867–872.
[12] B.T. Lowerre, The HARPY speech recognition system, PhD thesis, Carnegie Mellon University, Pittsburgh, 1976.
[13] D. Cook, L. Holder, S. Su, R. Maglothin, I. Jonyer, Structural mining of molecular biology data, IEEE Eng. Med. Biol. 20 (2001) 67–74.
[14] E. Ruspini, I. Zwir, Automated generation of qualitative representations of complex object by hybrid soft-computing methods, in: S. Pal, A. Pal (Eds.), Pattern Recognition: From Classical to Modern Approaches, World Scientific Company, 2001, pp. 453–474.
[15] I. Zwir, R. Romero-Zaliz, E. Ruspini, Automated biological sequence description by genetic multiobjective generalized clustering, in: F. Valafar (Ed.), Techniques in Bioinformatics and Medical Informatics, in: Ann. New York Acad. Sci., vol. 980, 2002, pp. 65–82.
[16] R. Romero-Zaliz, I. Zwir, E. Ruspini, Generalized analysis of promoters (GAP): A method for DNA sequence description, in: C.A. Coello, G.B. Lamont (Eds.), Applications of Multi-Objective Evolutionary Algorithms, vol. 1, World Scientific Company, 2004, pp. 427–450.
[17] P. Shelokar, A. Quirin, Ó. Cordón, MOSubdue: A Pareto dominance-based multiobjective Subdue algorithm for frequent subgraph mining, Knowl. Inf. Syst. 34 (2013) 75–108.
[18] P. Shelokar, A. Quirin, Ó. Cordón, Subgraph mining in graph-based data using multiobjective evolutionary programming, in: Proc. IEEE Conf. Evolutionary Computation (CEC'11), 2011, pp. 1730–1737.
[19] P. Shelokar, A. Quirin, Ó. Cordón, MOEP-SO: A multiobjective evolutionary programming algorithm for graph mining, in: Int. Conf. Intelligent System Design and Application (ISDA'11), 2011, pp. 219–224.
[20] P. Shelokar, A. Quirin, Ó. Cordón, A multiobjective evolutionary programming framework for graph-based data mining, Inform. Sci. (2013), in press, http://dx.doi.org/10.1016/j.ins.2013.02.014.
[21] P. Shelokar, A. Quirin, Ó. Cordón, A multiobjective variant of the Subdue graph mining algorithm based on the NSGA-II selection mechanism, in: Proc. IEEE Conf. Evolutionary Computation (CEC'10), 2010, pp. 463–470.
[22] C.A. Coello, G.B. Lamont, D.A.V. Veldhuizen, Evolutionary Algorithms for Solving Multi-Objective Problems, Springer, Berlin, 2007.
[23] E. Zitzler, L. Thiele, K. Deb, Comparison of multiobjective evolutionary algorithms: Empirical results, IEEE Trans. Evol. Comput. 8 (2000) 173–195.
[24] E. Zitzler, L. Thiele, M. Laumanns, C. Fonseca, V. da Fonseca, Performance assessment of multiobjective optimizers: An analysis and review, IEEE Trans. Evol. Comput. 7 (2003) 117–132.
[25] E. Dijkstra, A note on two problems in connexion with graphs, Numer. Math. 1 (1959) 269–271.
[26] H. Hu, X. Yan, Y. Huang, J. Han, X. Zhou, Mining coherent dense subgraphs across massive biological networks for functional discovery, Bioinformatics 21 (2005) i213–i221.
[27] T. Falkowski, A. Barth, M. Spiliopoulou, Dengraph: A density-based community detection algorithm, in: IEEE/WIC/ACM Int. Conf. Web Intelligence, IEEE Computer Society, Los Alamitos, CA, USA, 2007, pp. 112–115.
[28] N. Shrivastava, A. Majumder, R. Rastogi, Mining (social) network graphs to detect random link attacks, in: Proc. IEEE Conf. Data Engineering (ICDE'08), 2008, pp. 486–495.
[29] C. Jiang, F. Coenen, M. Zito, A survey of frequent subgraph mining algorithms, Knowl. Eng. Rev. 28 (2013) 75–105.
[30] V. Chankong, Y.Y. Haimes, Multiobjective Decision Making Theory and Methodology, North-Holland, Amsterdam, 1983.
[31] T. Gal, T. Stewart, T. Hanne (Eds.), Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory and Applications, Kluwer Academic, Dordrecht, 1999.
[32] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2002) 182–197.
[33] B. Qu, P. Suganthan, Multi-objective evolutionary algorithms based on the summation of normalized objectives and diversified selection, Inform. Sci. 180 (2010) 3170–3181.
[34] D. Cook, L. Holder, Graph-based data mining, IEEE Intell. Syst. 15 (2000) 32–41.
[35] X. Yan, J. Han, gSpan: Graph-based substructure pattern mining, in: Proc. IEEE Conf. Data Mining (ICDM'02), 2002, pp. 721–724.
[36] B. Vargas-Quesada, F. de Moya-Anegón, Visualizing the Structure of Science, Springer-Verlag New York, Secaucus, 2007.
[37] A. Quirin, Ó. Cordón, V.P. Guerrero-Bote, B. Vargas-Quesada, F. de Moya-Anegón, A quick MST-based algorithm to obtain Pathfinder networks, J. Am. Soc. Inf. Sci. Technol. 59 (2008) 1912–1924.
[38] J. Rissanen, Stochastic Complexity in Statistical Inquiry Theory, World Scientific Company, River Edge, 1989.

Please cite this article in press as: P. Shelokar et al., Three-objective subgraph mining using multiobjective evolutionary programming, J. Comput. System Sci. (2013), http://dx.doi.org/10.1016/j.jcss.2013.03.005