4OR-Q J Oper Res (2010) 8:195–211
DOI 10.1007/s10288-009-0110-3

RESEARCH PAPER

The small world of efficient solutions: empirical evidence from the bi-objective {0,1}-knapsack problem

Carlos Gomes da Silva · João Clímaco · Adiel Almeida Filho

Received: 12 November 2008 / Revised: 3 March 2009 / Published online: 15 October 2009

Abstract The small world phenomenon, Milgram (1967) has inspired the study of real networks such as cellular networks, telephone call networks, citation networks, power and neural networks, etc. The present work is about the study of the graphs produced by efficient solutions of the bi-objective {0,1}-knapsack problem. The experiments show that these graphs exhibit properties of small world networks. The importance of the supported and non-supported solutions in the entire efficient graph is investigated. The present research could be useful for developing more effective search strategies in both exact and approximate solution methods of {0,1} multi-objective combinatorial optimization problems.

Keywords Networks · Small world measures · {0, 1} Multi-objective combinatorial optimization problems

MSC classification (2000) 90C27

C. Gomes da Silva (✉)
Escola Superior de Tecnologia e Gestão de Leiria do Lena, Alto Vieiro, 2401-951 Leiria, Portugal
e-mail: cgsilva@estg.ipleiria.pt

C. Gomes da Silva · J. Clímaco
INESC-Coimbra, Rua Antero de Quental, 199, 3000-033 Coimbra, Portugal

J. Clímaco
Faculdade de Economia, Universidade de Coimbra, Av. Dias da Silva,165, 3004-512 Coimbra, Portugal
e-mail: jclimaco@inescc.pt

A. A. Filho
Federal University of Pernambuco, Cx. Postal 7462, Recife, PE, CEP 50.630-970, Brazil
e-mail: adieltaf@gmail.com

1 Introduction

Linear {0,1} multi-objective combinatorial optimization problems are very important from both a theoretical and a practical point of view. Solving these problems involves finding *efficient solutions*, that is, solutions such that there are no other feasible solutions that can improve the value of one criterion without degrading the value of at least one other criterion. However, in general, the generation of the entire set of efficient solutions for large-size linear {0,1} multi-objective combinatorial optimization problems is very difficult, despite significant efforts and improvements in research directions leading to either exact or approximate approaches.

One attempt to find the entire set of efficient solutions in multiple-objective combinatorial optimization problems in an efficient manner was to verify whether the well-established property of *connectedness* in multi-objective linear programming problems (Cohon 1978; Steuer 1986) remains valid. Based on some definition of adjacency of efficient solutions, (Ehrgott and Klamroth 1997, Gorski et al. 2006 and Ruzika 2008), provided a negative answer to this question for several multiple objective combinatorial problems, including the minimum spanning tree problem, the shortest path problem, linear assignment problem, and minimum cost flow problem. Gomes da Silva et al. (2004) showed the connectedness of the set of efficient solutions only for a particular model of the *bi-objective {0,1}-knapsack* problems, where the sum of the coefficients of each variable in the objectives functions is equal and all the variables have the same coefficient in the knapsack constraint.

A complementary line of research consists of studying the structure of the set of efficient solutions for small and average-size problems in order to capture important properties, which will hopefully be useful for understanding how the efficient solutions are related and contribute to the development of more effective methods of finding solutions in larger problems.

This paper studies the structure of the set of efficient solutions of the bi-objective {0,1}-knapsack problem, using a graph built with these solutions, checking whether these graphs exhibit the characteristics of the ones studied in the literature on *small worlds* popularized by Milgram (1967). Since the {0,1}-knapsack (Martello and Toth 1990) is a subproblem of larger problems, the conclusions obtained can be useful for those other more general problems. Our aim is to obtain a better knowledge of the organization of the graph of efficient solutions and, in this way, to inspire the development of more effective search algorithms. It may be noted that these studies can also contribute to connectedness research by suggesting different concepts of *adjacency*.

The small world networks have two defining properties (Albert and Barabási 2002; Dorogovtesv and Mendes 2003): a *short path* between any two nodes, despite the large sizes of the networks, and a high *degree of connection* between the neighbors of any node. *Random networks* also share the property of short path-length between any two nodes. In small world networks, the distribution of the degree of each node follows a power law with a negative exponent (scale-free networks), contrasting with a Poisson distribution followed by random graphs studied by Erdös and Rényi (1959). As a consequence of the power law distribution there is some asymmetry in the small world networks with the existence of a few highly connected nodes.

The study of characteristics of *networks* has attracted enormous interest in recent years, focusing on the world wide web (WWW), Internet, movie actor collaboration network, science collaboration graph, web of human sexual contacts, cellular networks, telephone call networks, citation networks, networks in linguistics, and power and neural networks. Many other studies can be found aiming at characterizing the topology of networks. For example, Albert et al. (1999) and Huberman and Adamic (2000) studied the WWW network. Considering the documents in the websites as nodes and the links (URLs) that points from a document to another as edges, in the first work it was found that the probability of a document having k outcoming or incoming links, $p(k)$, follows a power-law, $p(k) = k^{-\alpha}$, with $\alpha = 2.45$ and $\alpha = 2.1$ for the outcoming and incoming links, respectively. It was also shown characteristics that on average any two documents are separated by nineteen clicks, and that the average shortest path between any two documents increases logarithmically with the size of the network. The work by Huberman and Adamic (2000) also found a certain degree of organization in the web, as they showed that the web pages are distributed according to a power law distribution, revealing that the documents with any given number of links can be found without an exhaustive search of the web. Several more specialized studies on networks, enabled by increased computational capabilities, have appeared more recently. This is the case in the work by Newman and Girvan (2004) dedicated to the study of algorithms for discovering highly connected nodes in networks. Another case is the work by Amaral et al. (2000) in which statistical properties of various real-world networks are studied. Once again, the objective of these refined studies is to understand and visualize the structure of the networks.

All the above referred studies aim at improving understanding of the interaction between the components of complex networks, which is also pursued in the present work concerning the efficient solutions graph of the bi-objective {0,1}-knapsack problem.

The remainder of the paper is organised as follows. Section 2 describes the bi-objective {0,1}-knapsack problem used in this study. Section 3 presents some measures for assessing the topology of the network. Section 4 is about the computational experiments. Section 5 draws the main conclusions of the paper.

2 The bi-objective {0,1}-knapsack problem

The {0,1}-knapsack problem is one of the most important and most studied combinatorial problems. There are several applications of the {0,1}-knapsack problem that are well described in Martello and Toth (1990), in the areas of packing, investment selection, and budgeting. These applications are, in many situations, more realistic when models are extended to the multicriteria case. This paper is about the bi-objective {0,1}-knapsack problem, which can be mathematically formulated as follows:

$$\max z_1(x_1, \ldots, x_j, \ldots, x_n) = \sum_{j=1}^{n} c_j^1 x_j$$

$$\max z_2(x_1, \ldots, x_j, \ldots, x_n) = \sum_{j=1}^{n} c_j^2 x_j$$

$$s.t.: \quad (1)$$
$$\sum_{j=1}^{n} w_j x_j \leq W$$
$$x_j \in \{0, 1\}, j = 1, \ldots, n,$$

where c_j^i represents the value of item j on criterion $i, i = 1, 2, x_j = 1$ if item j ($j = 1, \ldots, n$) is included in the knapsack and $x_j = 0$ otherwise, w_j is the weight of item j and W is the overall knapsack capacity.

Formally, solving problem (1) consists of finding the set of efficient solutions. A feasible solution x is efficient if and only if there is no other feasible solution x' such that: $z_i\left(x'\right) \geq z_i\left(x\right), i = 1, 2$ and $z\left(x'\right) \neq z\left(x\right)$. Efficient solutions can be divided into *supported efficient solutions* and *non-supported efficient solutions*. Efficient supported solutions are much easier to find and can be obtained by solving single criterion problems $\max\{\lambda_1 z_1(x) + \lambda_2 z_2(x) | x \in X\}$, where $\lambda_1, \lambda_2 > 0, \lambda_1 + \lambda_2 = 1$ and X represents the feasible region of the problem. Efficient solutions that cannot be obtained in this way are called non-supported efficient solutions.

The bi-objective case has attracted enormous attention among researchers, with several exact and approximate resolution procedures being available. The *exact procedures* (Visée et al. 1998; Captivo et al. 2003) are based on traditional techniques such as branch-and-bound and dynamic programming, and can only be used for small to medium-size instances due to the exponential computational effort required by the search. In Klamroth and Wiecek (2000) a theoretical exact approach, based on dynamic programming, is also applied to this problem. *Approximate methods* (heuristic methods) can, a priori, be applied to solve large-size instances but the solutions can be of poor quality considering the proximity to exact solutions and the number of solutions found. In these procedures, repetitive combinations and modifications of solutions are used to discover subregions of the search space where possibly good solutions can be found. Zhang and Ong (2004) propose an approximate method that generates an approximation of the Pareto front by solving successive single objective linear programming relaxation problems with additional constraints on the values of the objective functions. Less demanding constraints led to an overall smaller number of solutions to be presented compared to more restrictive constraints. New competitive approximate approaches to the bi-objective {0,1}-knapsack problem were also proposed in Gomes da Silva (2005), Gomes da Silva et al. (2006, 2007). These approaches are based on the *scatter search* meta-heuristic. By using *smoothed analysis* (the specified inputs of the model are subjected to a random perturbation) Beier et al. (2007) proposed a tight bound on the expected number of efficient solutions for general bi-objective integer optimization problems, which is a very interesting information within a resolution procedure. More recent research works on the {0,1}-knapsack problem considering more than two objective functions are available: Aghezzaf and Naimi (2009), Bazgan et al. (2009a,b). The work by Bazgan et al. (2009b) is an exact approach based on dynamic programming using several complementary dominance relations in order to discard no promising partial solutions. Bazgan et al. (2009a) proposed a fully polynomial approximation scheme also based in the same ideas to solve an approximation of the problem. Aghezzaf and Naimi (2009) propose a new

recombination operator, which is a two-stage process preserving in the first stage the similar characteristics of the solutions to be combined and accepting the non-similar parts considering a fitness function. This approach is to be used in multi-objective evolutionary algorithms in order to enhance its performance. However, a common feature to these approximate approaches is that they have to face a difficult trade-off between the quality of approximation and the running time with respect to the increase in the number of variables in the problem, keeping the interest in the development of alternative resolution methods.

3 The topology of the network of efficient solutions

Let us consider the undirected graph (in the paper this undirected graph is also named a *network* as in the small world literature) of the efficient solutions set defined as follows. Let $G = (E, V)$ be an undirected graph, where E is the set of edges and V is the set of nodes. Each node represents one efficient solution. Thus, node v_i corresponds to the efficient solution x^i. An edge between two nodes $v_i, v_j \in V$ exists if and only if the distance between the efficient solutions x^i and x^j, $d\left(v_i, v_j\right)$, is less than or equal to a specified constant γ, that is, $d\left(v_i, v_j\right) \leq \gamma$, where γ is a previously defined constant, called here the *neighborhood radius* or *size*. The network is said to be connected if there is a path between every pair of nodes. Two subgraphs of G are also studied: $G_s = (E_s, V_s)$ and $G_{ns} = (E_{ns}, V_{ns})$, where $E_s, E_{ns} \subseteq E$ and $V_s, V_{ns} \subseteq V$. In the first subgraph, the nodes of V_s correspond only to the supported solutions, and E_s is the set of edges between these nodes. In this case, the network is called the *supported network*. In the second subgraph, the nodes of V_{ns} correspond only to the non-supported solutions, and E_{ns} is the set of edges between these nodes. The corresponding network is called the *non-supported network*.

Since the combinatorial problem studied in the paper has {0,1} solutions, $x^i \in \{0, 1\}^n$, it is considered that $d\left(v_i, v_j\right)$ is the *Hamming distance* between solutions x^i and x^j, that is, $d\left(v_i, v_j\right) = \sum_{k=1}^{n} \left|x_k^i - x_k^j\right|$.

Since there is no justification for choosing a given neighborhood radius γ, we will consider several networks (one for each γ) with γ varying from two to the lowest neighborhood that gives rise to a connected network, that is, a network with a single *cluster*, i.e., a *connected component* containing all the nodes.

The most frequently used measures for assessing the characteristics of the networks are the *average degree of a node*, k, the *clustering coefficient*, C, and the average shortest path between any two nodes, l, also called the *path-length*. These measures are described below.

The degree of a node is equal to the number of neighbors of that node, that is, the number of edges emanating from that node, and reflects its connectivity. In random graphs, if k is lower than 1, the clusters are almost certainly trees or contain exactly one cycle, while with k greater than or equal to 1, the structure of the network completely changes towards the presence of a giant cluster containing all the nodes (Albert and Barabási 2002).

The clustering coefficient of a node is the ratio between the total number of edges connecting the nearest neighbors (nodes that are directly connected to it) and the total number of all possible edges between all those neighbors, that is,

$$C_i = \frac{2E_i}{k_i\,(k_i - 1)}$$

where k_i is the total number of nearest neighbors of node i and E_i is the number of edges that exist between those neighbors.

The average of the coefficients C_i is the clustering coefficient of the network (Albert and Barabási 2002). The clustering coefficient reveals how connected the neighbors of a node are. The path-length of the network is the average minimum number of edges between any two nodes of the network.

4 Experimental results

The computational experiments concern average-size problems with 100 binary variables ($n = 100$). The coefficients of the variables in problem (1) were randomly and uniformly generated within the range [1,100], using a uniform distribution, and the capacity of the knapsack was set to 50% of the sum of the coefficients of the capacity constraint. In order to obtain statistically reliable results 30 random instances were generated. Each of the 30 bi-objective problems was solved by using the exact method of Visée et al. (1998) and the supported and non-supported efficient solutions were identified to create the supported and non-supported networks.

4.1 The characteristics of the efficient networks

The minimum, average and maximum number of efficient solutions obtained with the above-described instances are presented in Table 1. On average, these networks are composed of about 125 nodes (efficient solutions), with the number of nodes of the non-supported networks being almost six times larger than the number for the supported networks. In the experiments, there are networks with a very smaller number of nodes and others with a much higher number of nodes, compared to the average number.

For illustration purposes, let us consider one instance. It has 73 efficient solutions, where 13 are supported and 60 are non-supported. Figures 1, 2 and 3 illustrate the

Table 1 Characteristics of the 30 networks used in the experiment

Set	Number of nodes		
	Min	Average	Max
Entire set	73	124.9	177
Supported set	12	18.5	25
Non-supported	57	106.4	156

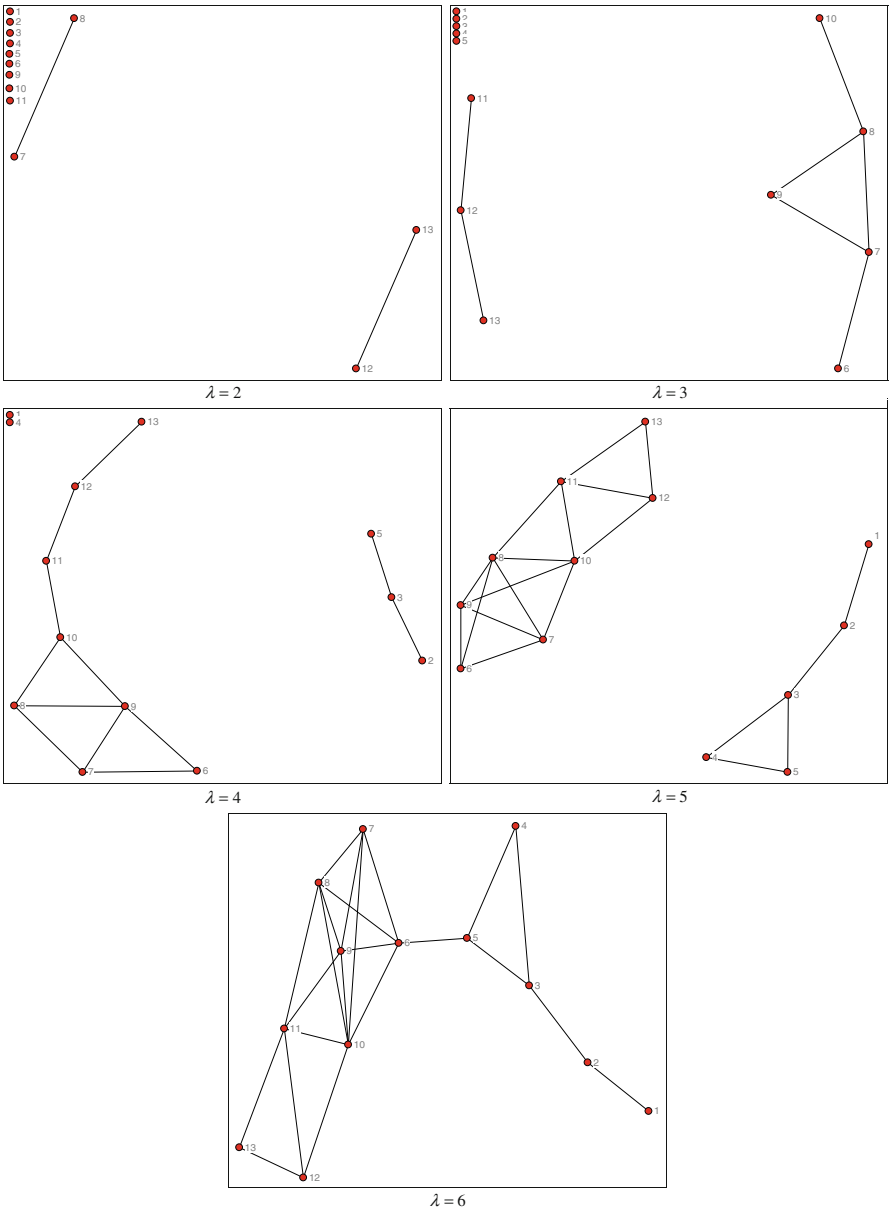

Fig. 1 An example of supported networks for different neighborhoods

configuration of supported, non-supported and entire set networks of that instance, for different neighborhoods until the networks become connected. The supported solutions are numbered sequentially from 1 to 13 and the remaining 60 are numbered from 14 to 73. In each set (supported and non-supported) solutions are sorted according

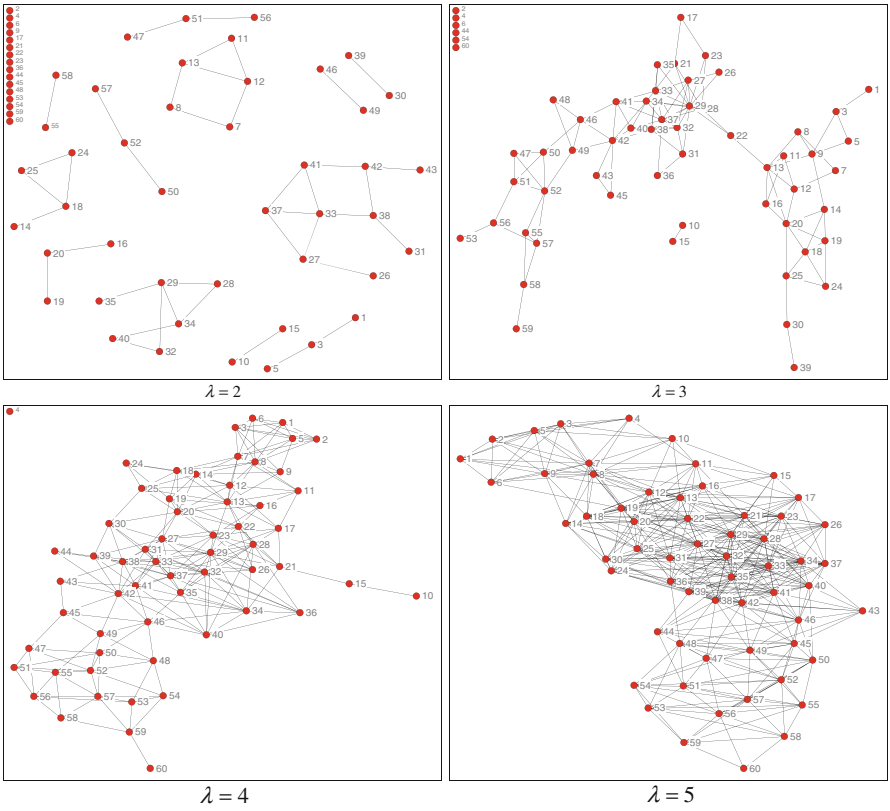

Fig. 2 An example of non-supported networks for different neighborhoods

to increasing values of objective function $z_1(x)$. The position of the points in these figures has no special meaning.

With the values presented in Tables 2, 3 and 4, it can be seen that the average number of edges for the entire networks and for the non-supported networks increases rapidly, in a non-linear way, with the neighborhood size. In the case of the supported networks, that relation is approximately linear. This means that the efficient solutions are quite close to each other according to the Hamming distance and small increases in the radius size generates a high number of newer edges. The non-supported nodes appear to be even closer to each other.

As the neighborhood radius increases, the number of clusters rapidly decreases (Tables 2, 3, 4). With $\gamma = 5$, a single giant cluster arises in the entire and non-supported networks of all the experiments. Hence, any two solutions are connected, that is, $d\left(x^i, x^j\right) \leq 5$, for every v_i, v_j. For supported networks the single cluster only occurs when γ equals 7.

Observing the relative size ($cluster\ size/number\ of\ nodes \times 100$) of the largest cluster in the networks, it can be seen that the gap between the minimum and the maximum value is large, until γ reaches the value of 4. With $\gamma = 4$, the largest cluster

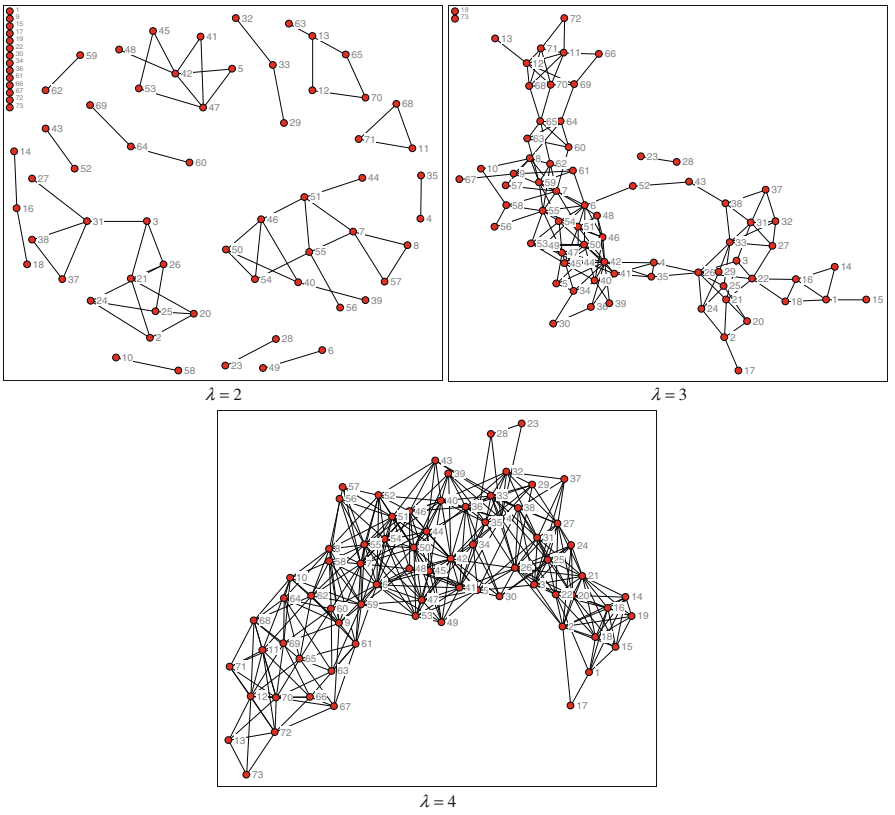

Fig. 3 An example of entire networks for different neighborhoods

Table 2 Cluster features for the entire network

Edges (E)		Number of clusters			Relative size of the largest cluster		
γ	Average	Min	Average	Max	Min	Average	Max
2	97.2	28	53.8	97	5.8	13.5	39.8
3	242.4	4	9.1	7	40.8	78.9	97.7
4	651.2	1	1.3	2	98.2	99.7	100
5	1121.7	1	1	1	100	100	100

contains more than 98% of the nodes in all the networks of the experiments, meaning that there are only a few nodes which remain disconnected, revealing that the associated efficient solutions have a higher hamming distance to the rest of the efficient solutions. In the supported networks, similar results are only obtained with a higher value of γ. The smallest value γ which generates a connected network and the high percentage of nodes in the largest cluster with lower values of γ is very interesting for searching algorithms, since it raises the possibility of achieving any efficient solution

Table 3 Cluster features for the supported network

Edges (E)		Number of clusters			Relative size of the largest cluster		
γ	Average	Min	Average	Max	Min	Average	Max
2	3.4	10	15.2	22	8.0	14.5	29.4
3	8.6	7	10.6	15	14.3	25.8	44.0
4	18.2	2	4.8	8	23.8	48.5	93.8
5	27.8	1	2.0	5	50.0	83.3	100
6	32.7	1	1.2	2	57.1	94.3	100
7	37.8	1	1	1	100	100	100

Table 4 Cluster features for the non-supported network

Edges (E)		Number of clusters			Relative size of the largest cluster		
γ	Average	Min	Average	Max	Min	Average	Max
2	61.2	28	56.5	96	5.1	11.4	36.7
3	154.2	1	15.4	30	33.8	62.4	100
4	444.3	1	1.7	4	95.0	98.9	100
5	822.6	1	1	1	100	100	100

from any other efficient solution, or at least a significant part of the efficient solutions. This means that efficient solutions are easily linked (it is only necessary to commute a small number of variables of one efficient solution to obtain another efficient solutions), which is very important since it could open the possibility of transition among efficient solutions, as is well-known in multi-objective linear programming problems. These results are not contrary to the conclusion of the connectedness studies presented in the Introduction. Indeed, they provide a possibility of connectedness based on different concepts of adjacency. Obviously, a further effort is required to find a proper definition of connectivity between solutions.

It is interesting to investigate what happens to the network when the nodes V_s or V_{ns} are removed. In order to evaluate the impact of removing each set of nodes, we consider the connected network with all the nodes, and then each set of nodes is removed. The elimination of the supported nodes (about 15% of the total number of nodes) does not seem to compromise the connectedness of the network. In the majority of the networks, 76.7%, the size of the largest cluster remains above 99%, and in 66.7% of the networks the largest cluster is still composed of all the nodes. The minimum size of the largest cluster in the modified network contains 95% of the nodes. This shows that almost all the non-supported nodes have a path linking them, meaning that non-supported solutions can still be reached from other non-supported solutions. In terms of future algorithms these results mean that the previous computation of supported solutions may not be necessary for finding non-supported solutions. Indeed, the current general approaches put the initial computational effort into the computation of the set of supported solutions. Our results show that this may not be the best strategy.

Table 5 Average degree of the nodes

Entire network				Non-supp. network			Supp. network		
γ	Min	Aver.	Max	Min	Aver.	Max	Min	Aver.	Max
2	0.99	1.55	2.43	0.67	1.15	1.96	0.12	0.36	0.67
3	2.70	3.87	4.71	1.73	2.88	3.57	0.43	0.89	1.44
4	8.19	10.35	12.43	5.72	8.24	10.33	1.00	1.92	2.88
5	14.86	18.99	20.92	11.58	15.95	19.04	1.33	2.93	3.92
6	–	–	–	–	–	–	2.33	3.73	4.63
7	–	–	–	–	–	–	4.13	4.33	4.75

The elimination of the non-supported nodes (85% of the total number of nodes) has a significantly different impact. In the remaining network the largest cluster can have a low percentage of the total number of nodes but also the entire set of nodes. The distribution of the relative size of the largest cluster is almost uniform in the range [23.8, 100]. These results suggest that in this case the impact on the connectivity is data-dependent. Thus, one can have supported nodes that are linked through non-supported nodes and, in this way, kept apart when those nodes are removed. It seems that for some instances the euclidean distance in the objective space is transferred for the decision space, and for other instances, the two spaces have different organizations. This issue is investigated in Gomes da Silva and Clímaco (2008).

These results concerning the relative importance of supported and non-supported efficient solutions show that trying to find the set of efficient solutions by searching neighborhoods of supported solutions (the easiest to obtain) may be not the best strategy to follow. Note that this strategy is the most frequently used one in solving many multi-objective combinatorial problems.

4.2 Average degree of a node

An important measure for the networks is the average degree of their nodes. The average degree of a node evaluates its connectivity. Table 5 presents the average, maximum, and minimum of the average degree of the nodes. The results show that for $\gamma = 2$, the entire networks and the non-supported networks contain disjoint pairs of linked nodes while the supported networks are composed of isolated nodes. As γ increases, the average degree also increases in a non-linear way for the entire networks and non-supported networks. In the supported networks, the average degree increases linearly with γ (see Fig. 4).

For the connected networks the average degree of a node differs significantly between supported and non-supported networks. In the supported networks, each node has an average of 3.7 neighbors, which is less than the number of neighbors of the nodes of the non-supported networks: 11.9. For the entire network the value is 12.9.

The degree distribution is also very important to characterize the network topology, since it can be used to evaluate how the nodes are connected (Albert and Barabási

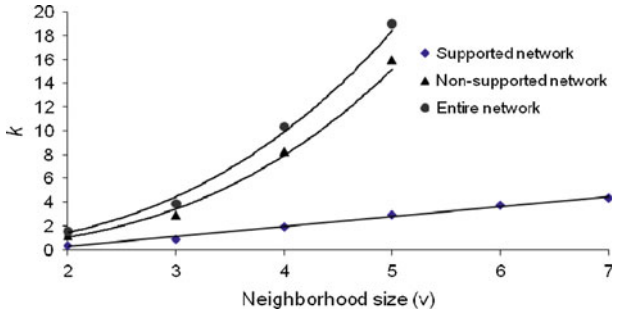

Fig. 4 Average degree functions

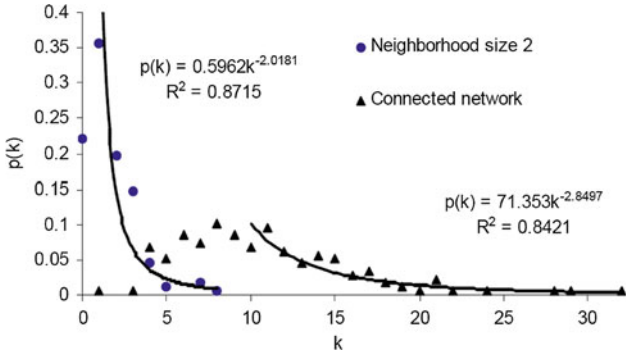

Fig. 5 Distribution of the degree of nodes

2002). In a random graph, this distribution follows a Poisson distribution. In the experiments, a typical degree distribution is presented in Fig. 5. In this figure the distributions of k when $\gamma = 2$ and when the network of the efficient solutions of a given instance of the knapsack problem is connected are presented. Despite the different distributions, both share a tail approximated by a power law with a negative exponent, which reveals the existence of a few highly linked nodes (in relative terms). Using the linear estimation on the log-log model form, the estimated models and their main statistics' p-value and R^2 (R^2 means the proportion of the total variation of the explanatory variable explained by the model and the p-values for the parameters give the lowest level of probability that leads to the rejection of statistical insignificance of the parameters) are presented below. The normality of the residuals was verified and not statistically excluded.

Neighborhood size equal to 2:

$$p(k) = 0.5962k^{-2.0181} (R^2 = 0.8715;\ p\text{-values} : 0.3444; 0.0021)$$

Connected network:

$$p(k) = 71.353k^{-2.8497} (R^2 = 0.8421;\ p\text{-values} : 0.00034; 2.13\text{E} - 07)$$

In the first model, a decrease of (2.0181 ± 0.8909) % in the probability of existence of a node is expected for a 95% confidence level, with an increase of 1% in the degree of the nodes $\left(\frac{dp(k)}{dk} \times \frac{k}{p(k)}\right)$. In the connected network, the decrease is very similar and is about (2.85 ± 0.6792) %.

To illustrate the differences between the two networks, let us consider the probability that a node has five direct neighbors. One can see that in the network with a neighborhood size equal to 2, this probability is 0.023, while in the connected network this probability corresponds to almost 17 direct neighbors.

Using the model, the probability of a node being linked to all the possible 123 neighbors (the average number of possible neighbors of a node, see Table 1) is only $71.353 \times 123^{-2.8497} = 4.31e^{-4}$.

Highly connected nodes mean that there are some efficient solutions which are very similar, in terms of the Hamming distance, to many other efficient solutions. As a consequence, once these solutions are found, many more can be obtained following the direct links of those solutions, that is, by commuting the value of at most γ variables. The nature of the problem can give some insight regarding where to explore a solution for other interesting neighbors. For example, in the bi-objective $\{0,1\}$-knapsack it was found (Gomes da Silva et al. 2008) that the exploration of a neighborhood around the fractional variables of the optimal solution of weighted sums of the criteria functions of the linear relaxation of the problem can lead to many other efficient solutions.

Many search algorithms can benefit from these experimental results. The development of exact methods specialized for finding efficient solutions that are connected to many other efficient solutions could be very interesting, since those solutions could be used to achieve many other interesting ones. Additional research is required to understand how and why these solutions exist and to characterize them.

4.3 Clustering coefficient

The clustering coefficient, C, evaluates the average connectivity among the neighbors in the network. In small world networks, this coefficient is relatively high, contrasting with its low value in random networks. In random networks, the clustering coefficient is given by $C_{\text{rand}} = k/N$, where N is the number of nodes (Albert and Barabási 2002).

Table 6 presents the clustering coefficient for the three networks, for different values of γ and also for equivalent random networks (with a similar number of nodes and edges, obtained by the average number of edges and the average number of nodes of the instances used). In the table, valid observations refer to the number of networks where it was possible to compute the clustering coefficient for at least one node. The average values are high in all the three networks, in which more than 50% of the possible links among the neighbors of a given node are present. The increase in the clustering coefficient, as γ increases, and because the average degree also increases, means that a node becomes attached to new nodes which also become connected to the neighbors of that node.

Considering the connected networks, the clustering coefficients were computed and are 0.533, 0.515 and 0.616 for the entire, non-supported and supported networks,

Table 6 Clustering coefficient

Entire network					Non-supp. network				Supp. network			
γ	Min	Aver.	Max	C_{rand}	Min	Aver.	Max	C_{rand}	Min	Aver.	Max	C_{rand}
2	0.207	0.407	0.564	0.012	0.055[a]	0.357	0.630	0.011	1[b]	1	1	0.005
3	0.305	0.405	0.506	0.031	0.151	0.358	0.528	0.027	0.185[c]	0.496	0.750	0.009
4	0.439	0.517	0.622	0.083	0.391	0.485	0.625	0.077	0.167[d]	0.511	0.845	0.017
5	0.556	0.582	0.608	0.163	0.484	0.553	0.643	0.158	0.167[e]	0.560	0.765	0.026
6	–	–	–		–	–	–		0.379	0.641	0.725	0.217
7	–	–	–		–	–	–		0.663	0.704	0.729	0.247

[a] 29 valid observations
[b] 1 valid observations
[c] 11 valid observations
[d] 26 valid observations
[e] 29 valid observations

respectively. In equivalent random networks, the expected clustering coefficients are 0.1036, 0.1120 and 0.1977, respectively.

4.4 Average shortest path

The average shortest path, l, is an important measure in small world networks. In random networks, the average path-length is given by $l_{rand} = \ln(N) / \ln(k)$ (Albert and Barabási 2002). We have computed, for the connected networks, the shortest path between all the pairs of nodes and their average length presented in Table 7 (γ was set in each problem in order to generate a connected network). It was found that the average shortest paths between any two nodes are 3.62, 3.49, and 3.58 for the supported, the non-supported, and the entire network, respectively. These values are higher than the ones expected from equivalent random graphs, but in any case, the average shortest path between any two nodes has less than two additional links compared with equivalent random graphs. On average, any two efficient solutions are separated by less than four edges. In the non-supported networks any two random nodes are separated by a lower number of intermediary nodes.

Figure 6 presents the distribution of the path-length of the entire, non-supported and supported connected network of the instance of the experiments with the maximum number of nodes (21 supported nodes and 156 non-supported nodes). The average path-length of this network is equal to 4.31. The distribution of the path-length for

Table 7 Average shortest path

Entire network				Non-supp. network				Supp. network			
Min	Aver.	Max	l_{rand}	Min	Aver.	Max	l_{rand}	Min	Aver.	Max	l_{rand}
2.590	3.584	4.476	1.885	2.488	3.488	4.921	1.883	2.508	3.621	5.423	2.250

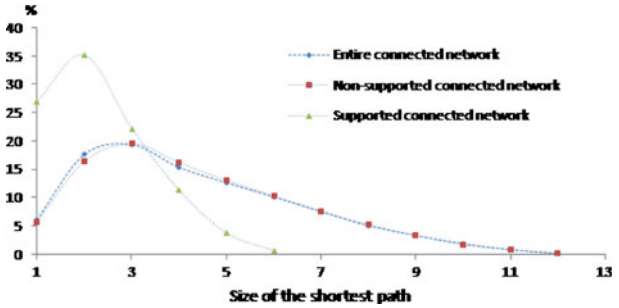

Fig. 6 Distribution of the path-length

supported and non-supported networks is quite different. The most important fact is that, in the supported case, the size of the path-length decreases more rapidly.

The relevance of the small shortest path in our networks can be expressed as in Albert et al. (1999) for the web network: as a consequence of the study, an intelligent agent who can interpret the links and follow only a relevant one can find the desired information by navigating the web. In our case, future search algorithms have to incorporate the ability to detect interesting transition paths among solutions, since the experimental results provide evidence of their existence.

5 Conclusions

Our experiments show that the network of efficient solutions shares the properties of small world networks: a small shortest path-length between any pair of nodes and a high clustering coefficient. Additionally, the degree of the nodes is reasonably well approximated by a power law with a negative exponent. It was also observed that the supported networks are quite different from the non-supported ones. In the supported networks the number of clusters decreases more slowly, a giant cluster is harder to achieve in terms of the neighborhood size, the relative size of the largest cluster is more dependent on other nodes, and the increase in the number of edges follows a linear function when the neighborhood size increases while in the non-supported networks this increase is much faster. It was also found that in most of the cases the elimination of supported nodes does not significantly affect the connectedness of the connected network. On the contrary, if the non-supported nodes are removed, several clusters emerge, destroying the connectivity.

Due to the known complexity associated with the generation of the exact entire set of efficient solutions, several researchers are developing approximated procedures, mainly using meta-heuristics methods. In the research into both exact and approximate methods, the present work can inspire new search strategies, since the search can be restricted to a small fraction of the search space using small neighborhood sizes (γ), as the experimental results show. Particularly, the usefulness of detailed studies on the connectedness of the efficient solutions in the development of new heuristics dedicated to specific $\{0,1\}$ multi-objective combinatorial optimization problems is

foreseeable. For example, knowing the expected values for the clustering coefficient or for the path-length, one could define neighborhoods of solutions where a given value for the clustering coefficient is preserved, or composed by solutions that preserve a given value for the clustering coefficient or path-length. Future research will focus on additional experiments on the bi-objective {0,1}-knapsack, the assessment of small word measures for other multi-objective combinatorial problems and the development of new heuristics and exact search procedures based on the small world properties for multi-objective {0,1}-knapsack and other combinatorial problems. Last but not least, why should the set of efficient solutions of {0,1} multi-objective combinatorial optimization problems have the characteristics of small world networks, as confirmed by the experiments, usually found in real complex systems? We think this was expectable because multi-objective models also represent, at least partially, the complexity of real problems, considering explicitly conflicting criteria.

Acknowledgments The authors would like to thank to the editor of the 4OR and to the anonymous referees for their comments and suggestions.

References

Aghezzaf B, Naimi M (2009) The two-stage recombination operator and its application to the multi-objective 0/1 knapsack problem: a comparative study. Comp Oper Res 36:3247–3262

Albert R, Barabási A (2002) Statistical mechanics of complex networks. Rev Mod Phys 74:47–97

Albert R, Jeong H, Barabási A (1999) Diameter of the world-wide web. Nature 401:130

Amaral L, Scala A, Barthélémy M, Stanley H (2000) Classes of small world networks. Proc Natl Acad Sci 97(21):11149–11152

Bazgan C, Hugot H, Vanderpooten D (2009a) Implementing an efficient fptas for the 0-1 multi-objective knapsack problem. Eur J Oper Res 198(1):47–56

Bazgan C, Hugot H, Vanderpooten D (2009b) Solving efficiently the 0-1 multi-objective knapsack problem. Comp Oper Res 36:260–279

Beier R, Röglin H, Vöcking B (2007) The smoothed number of pareto optimal solutions in bicriteria integer optimization. Lect Notes Comput Sci 4153:53–67

Captivo M, Clímaco J, Figueira J, Martins E, Santos JL (2003) Solving multiple criteria 0,1-knapsack problems using a labeling algorithm. Comp Oper Res 30:1865–1886

Cohon J (1978) Multi-objective programming and planning. Academic Press, London

Dorogovtesv S, Mendes F (2003) Evolution of networks—from biology nets to the internet and WWW. Oxford University Press, Oxford

Ehrgott M, Klamroth K (1997) Connectedness of efficient solutions in multicriteria combinatorial optimization. Eur J Oper Res 97(1):159–166

Erdös P, Rényi A (1959) On random graph. Publ Math 6:290–297

Gomes da Silva C (2005) Finding the set of non-dominated solutions in the bi-objective 0,1-knapsack: an approach with meta-heuristics and exact methods. PhD thesis, Faculty of Economics, University of Coimbra

Gomes da Silva C, Clímaco J (2008) Finding communities in the efficient solutions network. Technical report, Institute for Systems and Computers Engineering, Coimbra, Portugal

Gomes da Silva C, Clímaco J, Figueira J (2004) Geometrical configuration of the pareto frontier of bi-criteria 0,1-knapsack problems. Technical report, Institute for Systems and Computers Engineering, Coimbra, Portugal

Gomes da Silva C, Clímaco J, Figueira J (2006) A scatter search method dedicated to the bi-objective knapsack problems. Eur J Oper Res 169(2):373–391

Gomes da Silva C, Clímaco J, Figueira J (2008) Core problems in bi-objective 0-1 knapsack problems. Comp Oper Res 35(7):2292–2306

Gomes da Silva C, Figueira J, Clímaco J (2007) Integrating partial optimization with scatter search for solving bi-objective 0,1-knapsack problems. Eur J Oper Res 177:1656–1677

Gorski J, Klamroth K, Ruzika S (2006) Connectedness of efficient solutions in multiple objective combinatorial optimization. Technical report, Department of Mathematics, University of Kaiserslautern
Huberman B, Adamic L (2000) Growth dynamics of the world-wide web. Nature 401:131
Klamroth K, Wiecek M (2000) Dynamic programming approaches to the multiple criteria knapsack problem. Nav Res Logist 47(1):57–76
Martello S, Toth P (1990) Knapsack problems—algorithms and computer implementations. Wiley, New York
Milgram S (1967) The small world problem. Psychol Today 1(1):60–67
Newman M, Girvan M (2004) Finding and evaluating community structure in networks. Phys Rev E 69:026113.1–026113.15
Ruzika S (2008) On multiple objective combinatorial optimization. PhD thesis, Department of Mathematics, University of Kaisreslautern
Steuer R (1986) Multiple criteria optimization, theory, computation and application. Wiley, New York
Visée M, Teghem J, Pirlot M, Ulungu E (1998) Two phases method and branch and bound procedures to solve the bi-objective knapsack problem. J Glob Optim 12:139–155
Zhang CW, Ong HL (2004) Solving the biobjective zero-one knapsack problem by an efficient lp-based heuristic. Eur J Oper Res 159:545–557