GRASP with path-relinking for data clustering: a case study for biological data

Rafael M.D. Frinhani[1], Ricardo M.A. Silva[2,3], Geraldo R. Mateus[1], Paola Festa[4], and Mauricio G.C. Resende[5]

[1] Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brazil
[2] Universidade Federal de Lavras, Lavras, MG, Brazil
[3] Universidade Federal de Pernambuco, Recife, PE, Brazil
[4] University of Napoli Federico II, Napoli, Italy
[5] AT&T Labs Research, Florham Park, NJ, USA

Abstract. Cluster analysis has been applied to several domains with numerous applications. In this paper, we propose several GRASP with path-relinking heuristics for data clustering problems using as case study biological datasets. All these variants are based on the construction and local search procedures introduced by Nascimento et. al [22]. We hybridized the GRASP proposed by Nascimento et. al [22] with four alternatives for relinking method: forward, backward, mixed, and randomized. To our knowledge, GRASP with path-relinking has never been applied to cluster biological datasets. Extensive comparative experiments with other algorithms on a large set of test instances, according to different distance metrics (Euclidean, city block, cosine, and Pearson), show that the best of the proposed variants is both effective and efficient.

1 Introduction

Clustering algorithms aim to group data such that the most similar objects belong to the same group or cluster, and dissimilar objects are assigned to different clusters. According to Nascimento et. al [22], cluster analysis has been applied to several domains, natural language processing [2], galaxy formation [3], image segmentation [4], and biological data [7; 8; 9]. Surveys on clustering algorithms and their applications can be found in [5] and [6].

This paper presents a GRASP with path-relinking for data clustering based on a linearized model proposed by Nascimento et. al [22]:

$$\min \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} d_{ij} y_{ij} \tag{1}$$

subject to:

$$\sum_{k=1}^{M} x_{ik} = 1, \qquad i = 1, ..., N \tag{2}$$

$$\sum_{i=1}^{N} x_{ik} \geq 1, \quad k = 1, ..., M \tag{3}$$

$$x_{ik} \in \{0, 1\}, \quad i = 1, ..., N, \quad k = 1, ..., M \tag{4}$$

$$y_{ij} \geq x_{ik} + x_{jk} - 1, \quad i = 1, ..., N, \quad j = i + 1, ..., N, \quad k = 1, ..., M \tag{5}$$

$$y_{ij} \geq 0 \quad i = 1, ..., N, \quad j = i + 1, ..., N. \tag{6}$$

As described in [22], the objective function (1) aims to minimize the distance between the objects inside the same cluster, where d_{ij} denotes the distance between objects i and j; N denotes the number of objects; M denotes the number of clusters; x_{ik} is a binary variable that assumes value 1, if the object i belongs to the cluster k and 0, otherwise; and y_{ij} is a real variable that assumes the value 1, if the objects i and j belong to the same cluster.

While constraints (2) assure that object i belongs to only one cluster, constraints (3) guarantee that cluster k contains at least one object, and constraints (4) assure that the variables x_{ik} are binaries. Finally, constraints (5) and (6) guarantee that y_{ij} assumes the value 1, if both values of x_{ik} and x_{jk} are equal to 1.

The paper is organized as follows. In Section 2, we describe the GRASP with path-relinking procedure. Computational results are described in Section 3 and concluding remarks are made in Section 4.

2 GRASP with path-relinking for data clustering

GRASP, or greedy randomized adaptive search procedure, is a multi-start metaheuristic for finding approximate solutions to combinatorial optimization problems formulated as

$$\min f(x) \text{ subject to } x \in \mathcal{X},$$

where $f(\cdot)$ is an objective function to be minimized and $\mathcal{X}$ is a discrete set of feasible solutions. It was first introduced by Feo and Resende [7] in a paper describing a probabilistic heuristic for set covering. Since then, GRASP has experienced continued development [8; 23; 25] and has been applied in a wide range of problem areas [9; 10; 11].

At each GRASP iteration, a greedy randomized solution is constructed to be used as a starting solution for local search. Local search repeatedly substitutes the current solution by a better solution in the neighborhood of the current solution. If there is no better solution in the neighborhood, the current solution is declared a local minimum and the search stops. The best local minimum found over all GRASP iterations is output as the solution.

GRASP iterations are independent, i.e. solutions found in previous GRASP iterations do not influence the algorithm in the current iteration. The use of previously found solutions to influence the procedure in the current iteration can be thought of as a memory mechanism. One way to incorporate memory into GRASP is with path-relinking [13; 16]. In GRASP with path-relinking [18; 24], an elite set of diverse good-quality solutions is maintained to be used during each GRASP iteration. After a solution is produced with greedy randomized construction and local search, that solution is combined with a randomly selected solution from the elite set using the path-relinking operator. The best of the combined solutions is a candidate for inclusion in the elite set and is added to the elite set if it meets quality and diversity criteria.

Algorithm 1 shows pseudo-code for a GRASP with path-relinking heuristic for the data clustering problem. The algorithm takes as input the dataset to be clustered and outputs the best clustering $\pi^* \in \chi$ found.

```
   Data   : Dataset to be clustered
   Result : Solution π* ∈ χ.
1  P ← ∅;
2  while stopping criterion not satisfied do
3      π′ ← GreedyRandomized(·) as described in [22];
4      if elite set P has at least ρ elements then
5          π′ ← LocalSearch(π′) as described in [22];
6          Randomly select a solution π⁺ ∈ P;
7          π′ ← PathRelinking(π′, π⁺);
8          π′ ← LocalSearch(π′) as described in [22];
9          if elite set P is full then
10             if c(π′) ≤ max{c(π) | π ∈ P} and π′ ≉ P then
11                 Replace the element most similar to π′ among all
                       elements with cost worst than π′;
12             end
13
14         else if π′ ≉ P then
15             P ← P ∪ {π′};
16         end
17
18     else if π′ ≉ P then
19         P ← P ∪ {π′};
20     end
21 end
22 return π* = min{c(π) | π ∈ P};
```

Algorithm 1: GRASP with path-relinking heuristic.

After initializing the elite set P as empty in line 1, the GRASP with path-relinking iterations are computed in lines 2 to 21 until a stopping criterion is satisfied. This criterion could be, for example, a maximum number of iterations, a target solution quality, or a maximum number of iterations without improvement. In this paper, we have adopted the maximum number of iterations without improvement (IWI) as stopping criterion of the GRASP-PR variants. During each iteration, a greedy randomized solution π' is generated in line 3. If the elite set P does not have at least ρ elements, then if π' is sufficiently different from all other elite set solutions, π' is added to the elite set in line 19. To define the term *sufficiently different* more precisely, let $\Delta(\pi', \pi)$ be defined as the minimum number of moves needed to transform π' into π or vice-versa. For a given level of difference δ, we say that π' is sufficiently different from all elite solutions in P if $\Delta(\pi', \pi) > \delta$ for all $\pi \in P$, which we indicate with the notation $\pi' \not\approx P$. If the elite set P does have at least ρ elements, then the steps in lines 5 to 16 are computed.

The local search described in [22] is applied in line 5 using π' as a starting point, resulting in a local minimum, which we denote by π'. Next, path-relinking is applied in line 7 between π' and an elite solution π^+, randomly chosen in line 6. Solution π^+ is selected with probability proportional to $\Delta(\pi', \pi^+)$. In line 8, the local search described in [22] is applied to π'. If the elite set is full, then if π' is of better quality than the worst elite solution and $\pi' \not\approx P$, then it will be added to the elite set in line 11 in place of some elite solution. Among all elite solutions having cost no better than that of π', a solution π most similar to π', i.e. with the smallest $\Delta(\pi', \pi)$ value, is selected to be removed from the elite set. Ties are broken at random. Otherwise, if the elite set is not full, π' is simply added to the elite set in line 15 if $\pi' \not\approx P$.

2.1 Path-relinking

Path-relinking was originally proposed by Glover [13] as an intensification strategy exploring trajectories connecting elite solutions obtained by tabu search or scatter search [14; 15; 16]. Starting from one or more elite solutions, paths in the solution space leading toward other elite solutions are generated and explored in the search for better solutions. To generate paths, moves are selected to introduce attributes in the current solution that are present in the elite guiding solution. Path-relinking may be viewed as a strategy that seeks to incorporate attributes of high quality solutions, by favoring these attributes in the selected moves.

Algorithm 2 illustrates the pseudo-code of the path-relinking procedure applied to a pair of solutions x_s (starting solution) and x_t (target solution). The procedure starts by computing the symmetric difference $\Delta(x_s, x_t)$ between the two solutions, i.e. the set of moves needed to reach x_t (target solution) from x_s (initial solution). A path of solutions is generated linking x_s and x_t. The best solution x^* in this path is returned by the algorithm. At each step, the procedure examines all moves $m \in \Delta(x, x_t)$ from the current solution x and selects the one which results in the least cost solution, i.e. the one which minimizes $f(x \oplus m)$, where $x \oplus m$ is the solution resulting from applying move m to solution x. The best move m^* is made, producing solution $x \oplus m^*$. The set of available moves is updated. If necessary, the best solution x^* is updated. The procedure terminates when x_t is reached, i.e. when $\Delta(x, x_t) = \emptyset$.

Data : Starting solution x_s and target solution x_t
Result : Best solution x^* in path from x_s to x_t
Compute symmetric difference $\Delta(x_s, x_t)$;
$f^* \leftarrow \min\{f(x_s), f(x_t)\}$;
$x^* \leftarrow \mathrm{argmin}\{f(x_s), f(x_t)\}$;
$x \leftarrow x_s$;
while $\Delta(x, x_t) \neq \emptyset$ **do**
    $m^* \leftarrow \mathrm{argmin}\{f(x \oplus m) : m \in \Delta(x, x_t)\}$;
    $\Delta(x \oplus m^*, x_t) \leftarrow \Delta(x, x_t) \setminus \{m^*\}$;
    $x \leftarrow x \oplus m^*$;
    if $f(x) < f^*$ **then**
        $f^* \leftarrow f(x)$;
        $x^* \leftarrow x$;
    end
end

Algorithm 2: Path-relinking.

We notice that path-relinking may also be viewed as a constrained local search strategy applied to the initial solution x_s, in which only a limited set of moves can be performed and where uphill moves are allowed. Several alternatives have been considered and combined in recent implementations of path-relinking [1; 2; 3; 5; 26; 27; 29], among them:

– *forward relinking:* path-relinking is applied using the worst among x_s and x_t as the initial solution and the other as the target solution;
– *backward relinking:* the roles of x_s and x_t are interchanged, path-relinking is applied using the best among x_s and x_t as the initial solution and the other as the target solution;
– *mixed relinking:* two paths are simultaneously explored, the first emanating from x_s and the second from x_t, until they meet at an intermediary solution equidistant from x_s and x_t; and
– *randomized relinking:* instead of selecting the best yet unselected move, randomly select one from among a candidate list with the most promising moves in the path being investigated.

Figure 2.1 illustrates an example of path-relinking. Let x be a solution composed by clusters $A = \{2, 3, 7\}$, $B = \{4, 6\}$, and $C = \{1, 5\}$; and x_t the target solution with the clusters $A = \{6, 7\}$, $B = \{4, 5\}$, and $C = \{1, 2, 3\}$. Initially, $\Delta(x, x_t) = \{(2, A, C), (3, A, C), (5, C, B), (6, B, A)\}$, where (e, s, t) means a move of element e from cluster s to cluster t. After the best move $(2, A, C)$ from solution x is performed, x is updated with clusters $A = \{3, 7\}$, $B = \{4, 6\}$, and $C = \{1, 2, 5\}$. The process is repeated until x_t is reached.

3 Experimental results

In this section, we present results on computational experiments with the GRASP with path-relinking (GRASP-PR) heuristic introduced in this paper. First, we

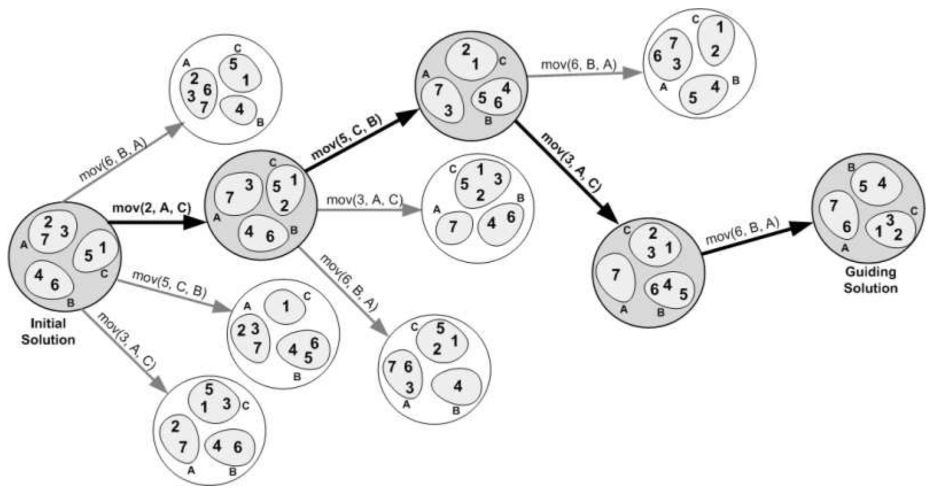

Fig. 1. A path-relinking example for data clustering.

describe our datasets. Second, we describe our test environment and determine an appropriated combination of values for the parameters of the heuristic. Finally, besides the GRASP-L algorithm introduced by Nascimento [22], we compare several GRASP-PR variants implementations with the three known clustering algorithms described in [22]: K-means, K-medians and PAM [17][6].

3.1 Datasets

We used the same five datasets from [22]: fold protein classification, named Protein [6], prediction of protein localization sites, named Yeast [21]; seven cancer diagnosis datasets, named Breast [4], Novartis [30], BreastA [31], BreastB [32], DLBCLA [20], DLBCLB [28] and MultiA [30]; and the benchmark Iris [12].

Table 1 shows the main characteristics of each dataset. The second column indicates the number of objects in each dataset. The third column shows the number of structures in the dataset and, in parenthesis, the number of clusters for each structure. The fourth column shows the number of attributes in the objects. Next, we describe in more details each of the datasets used.

3.2 Test environment and parameters for GRASP-PR heuristic

All experiments with GRASP-PR were done on a Dell computer with Core 2 Duo 2.1 GHz T8100 Intel processor and 3 Gb of memory, running Windows XP Professional version 5.1 2002 SP3 x86. The GRASP-PR heuristic was implemented

[6] K-means and K-medians implementations are available at `http://bonsai.ims.u-tokyo.ac.jp/~mdehoon/software/cluster/software.htm`.

Table 1. Characteristics of datasets used in the experiments.

Data Set	**#Objects**	**#Str(#Groups)**	**#Attrib**
Protein	698	2 (4,27)	125
Yeast	1484	1 (10)	8
Breast	699	2 (2,8)	9
Novartis	103	1 (4)	1000
BreastA	98	1 (3)	1213
BreastB	49	2 (2,4)	1213
DLBCLA	141	1 (3)	661
DLBCLB	180	1 (3)	661
MultiA	103	1 (4)	5565
Iris	140	1 (3)	4

in Java and compiled into bytecode with `javac` version 1.6.0.20. The random-number generator is an implementation of the Mersenne Twister algorithm [19] from the COLT[7] library.

The values of the parameters for GRASP-PR heuristic used for each dataset are shown in Table 2.

Table 2. Path-Relinking parameters. Pool size (PS), elements in pool before start PR (EPBS), symmetrical difference (SD), and Iterations without Improvement (IWI).

	Iris	Novartis	BrstA	BrstB1	BrstB2	DLBCLA	DLBCLB	MultA	Brst1	Brst2	Prt1	Prt2	Yeast
PS	3	5	4	3	3	5	5	5	3	6	5	5	7
EPBS	1	3	1	1	1	2	2	2	1	3	2	3	3
SD	4	70	4	30	30	100	100	70	4	550	450	450	1200
IWI	15	15	15	15	15	15	15	15	15	15	15	15	5

3.3 Numerical comparisons

We compare the three known clustering algorithms described in [22] (K-means, K-medians and PAM [17]) with the GRASP-L algorithm introduced by Nascimento [22] and the following five GRASP-PR variants implementations: GRASP, GRASP-PRf, GRASP-PRb, GRASP-PRm and GRASP-PRrnd. GRASP is our implementation of the GRASP-L algorithm. GRASP-PRf, GRASP-PRb, GRASP-PRm and GRASP-PRrnd correspond to the following relinking alternatives: forward, backward, mixed and greedy randomized, respectively. We used the same distance measurements for all of them.

The comparisons of the algorithms were based on the Corrected Rand index (CRand) proposed in [26] (Table 3). While GRASP-L, K-means and K-medians were run 100 times, GRASP-PRf, GRASP-PRb, GRASP-PRm and GRASP-

[7] COLT is a open source library for high performance scientific and technical computing in Java. See `http://acs.lbl.gov/~hoschek/colt/`.

PRrnd were run 30 times. All algorithms selected the partition with the best solution for each of the distance metrics.

With respect to the comparisons of the algorithms based on the Corrected Rand index (CRand) reported in Table 3, we observe that GRASP-PR variants found the best-quality solutions with all different dissimilarity measures, except for the Pearson correlation, for which there was a tie. In fact,

– using Euclidean metric as dissimilarity measure, GRASP-PRrnd found best results for 9 out of 10 datasets; GRASP-PRb and GRASP-PRm found best results for 8 datasets; GRASP-PRf and GRASP for 6, GRASP-L for 2, while K-means and K-medians found the best solution for only 1 and 2 datasets, respectively;
– using City Block metric as dissimilarity measure, GRASP-PRb, GRASP-PRrnd and GRASP-PRm for 8 out of 10 datasets; GRASP-PRf found best results for 7 datasets; GRASP for 6, GRASP-L and K-medians for 2, while K-means only for 1;
– using Cosine metric as dissimilarity measure, GRASP-PRrnd, GRASP-PRb, and GRASP-PRf found best results for 6 out of 10 datasets; GRASP-PRm for 5, GRASP for 4, and K-medians, PAM, and K-means for 4, 2, and 1, respectively.

4 Concluding remarks

In this paper, we propose four variants of GRASP with path-relinking (forward, backward, mixed, and randomized) for data clustering problem. The algorithms were implemented in Java and extensively tested. Computational results from several instances from the literature demonstrate that the heuristic is a well-suited approach for data clustering.

5 Acknowledgment

The research of R.M.A. Silva was partially supported by the Brazilian National Council for Scientific and Technological Development (CNPq), the Foundation for Support of Research of the State of Minas Gerais, Brazil (FAPEMIG), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brazil (CAPES), and Fundação de Apoio ao Desenvolvimento da UFPE, Brazil (FADE).

The research of R.M.D. Frinhani was partially supported by CAPES-MINTER Program between the Federal Universities of Minas Gerais and Lavras, Brazil.

We would like to thank Mariá C.V. Nascimento, Franklina M. B. Toledo, and André C.P.L.F. de Carvalho about the information related to paper [22].

Table 3. Summary of CRand results for GRASP-PRrnd, GRASP-PRm, GRASP-PRb, GRASP-PRf, GRASP, GRASP-L, K-means, K-medians and PAM algorithms. M is the number of clusters for the best CRand found. Times are given in seconds on a Core 2 Duo 2.1 GHz T8100 Intel processor (`javac` compiler version 1.6.0.20). Times for GRASP-L, K-means, K-medians and PAM algorithms are not reported in [22].

	GRASP-PRrnd			**GRASP-PRm**			**GRASP-PRb**			**GRASP-PRf**			**GRASP**			**GRASP-L**		**KMEANS**		**KMEDIANS**		**PAM**	
	M	cRand	Time	M	cRand	Time	M	cRand	Time	M	cRand	Time	M	cRand	Time	M	cRand	M	cRand	M	cRand	M	cRand
EUCLIDEAN																							
Protein	4	0.297	71.156	4	0.297	63.624	4	0.294	60.594	4	0.294	61.672	4	0.294	55.234	4	**0.322**	7	**0.322**	7	0.313	6	0.250
	11	**0.169**	107.328	11	0.168	306.197	11	0.168	130.249	11	0.168	130.249	11	**0.169**	121.547	11	0.168	17	0.139	25	0.134	13	0.098
Breast	2	**0.878**	16.344	2	**0.878**	19.781	2	**0.878**	19.843	2	**0.878**	19.343	2	**0.878**	18.625	2	0.877	2	0.803	2	0.782	2	0.828
	15	0.016	172.422	15	0.016	137.857	15	0.016	131.202	15	0.016	152.203	15	0.016	312.563	15	0.015	18	-0.010	17	**0.036**	5	0.012
Yeast	9	0.151	1689.766	9	0.153	1410.047	9	0.153	1492.132	9	0.150	849.363	9	0.151	1738.641	9	0.150	7	0.170	8	**0.173**	8	0.143
Novartis	4	**0.950**	7.124	4	**0.950**	6.921	4	**0.950**	7.045	4	**0.950**	7.344	4	**0.950**	6.344	4	0.921	4	0.946	4	0.946	4	0.897
BreastA	2	0.682	5.782	2	**0.723**	5.844	2	0.682	5.891	2	0.682	6.188	2	0.682	6.172	2	0.682	2	0.654	2	0.654	2	0.543
BreastB	2	**0.694**	1.875	2	**0.694**	1.906	2	**0.694**	1.906	2	**0.694**	1.985	2	**0.694**	1.968	2	0.626	3	0.502	4	0.500	2	0.388
	2	**0.322**	1.890	2	**0.322**	2.031	2	**0.322**	1.922	2	**0.322**	1.984	2	0.321	1.968	2	0.314	3	0.286	3	0.260	2	0.187
DLBCLA	4	**0.447**	9.531	4	0.431	10.187	4	**0.447**	8.249	4	0.408	8.297	4	0.408	11.750	4	0.408	4	0.309	5	0.365	4	0.276
DLBCLB	4	**0.519**	11.437	4	**0.519**	21.661	4	**0.519**	16.390	4	0.509	13.219	4	0.509	12.468	4	0.481	2	0.420	3	0.424	3	0.391
MultiA	4	**0.874**	32.629	4	**0.874**	31.562	4	**0.874**	32.359	4	**0.874**	33.859	4	**0.874**	29.937	4	**0.874**	6	0.765	5	0.682	4	0.765
Iris	3	**0.757**	0.281	3	**0.757**	0.312	3	**0.757**	0.312	3	**0.757**	0.391	3	**0.757**	0.391	3	0.756	3	0.730	3	0.744	3	0.730
CITY BLOCK																							
Protein	5	**0.310**	81.937	5	**0.310**	85.748	5	**0.310**	57.161	5	0.309	50.812	5	**0.310**	44.562	5	0.293	8	0.223	7	0.229	3	0.192
	9	0.180	77.328	9	0.176	254.432	9	**0.185**	164.155	9	**0.185**	164.155	9	0.178	76.937	9	0.166	17	0.158	28	0.141	19	0.084
Breast	2	**0.877**	14.406	2	**0.877**	17.672	2	**0.877**	17.502	2	**0.877**	17.234	2	**0.877**	16.531	2	**0.877**	2	0.770	2	0.765	2	0.807
	19	0.016	134.782	19	0.015	295.462	19	0.016	336.701	19	0.015	210.172	19	0.016	237.203	19	0.013	19	-0.009	10	**0.023**	13	0.010
Yeast	7	0.161	1432.047	7	0.159	953.766	7	0.160	1374.019	7	0.161	1630.917	7	0.161	706.266	7	0.157	7	**0.181**	6	0.167	7	0.152
Novartis	4	**0.950**	2.874	4	**0.950**	2.796	4	**0.950**	2.749	4	**0.950**	2.796	4	**0.950**	2.516	4	0.921	4	0.946	4	0.921	4	0.947
BreastA	2	**0.723**	1.875	2	**0.723**	1.889	2	**0.723**	1.750	2	**0.723**	1.922	2	0.722	1.890	2	0.682	2	0.583	2	0.618	4	0.560
BreastB	4	0.329	1.000	4	0.366	1.343	4	0.281	1.250	4	0.288	2.250	4	0.328	2.125	4	0.228	3	**0.563**	2	0.561	2	0.388
	7	**0.368**	3.172	7	0.344	1.828	7	0.293	1.390	7	0.328	1.265	7	0.293	1.140	7	0.159	3	0.328	3	0.284	2	0.187
DLBCLA	3	**0.838**	1.875	3	**0.838**	1.999	3	**0.838**	1.875	3	**0.838**	1.954	3	**0.838**	1.937	3	0.800	3	0.805	3	0.784	3	0.406
DLBCLB	2	**0.701**	2.703	2	**0.701**	2.797	2	**0.701**	2.797	2	**0.701**	2.843	2	**0.701**	2.640	2	0.700	2	0.690	2	0.690	3	0.350
MultiA	4	0.899	9.888	4	**0.924**	11.141	4	0.899	10.890	4	0.899	11.015	4	0.899	10.406	4	0.899	4	0.851	4	0.875	5	0.750
Iris	3	**0.818**	0.250	3	**0.818**	0.281	3	**0.818**	0.281	3	**0.818**	0.359	3	**0.818**	0.343	3	**0.818**	3	0.717	3	0.717	3	0.772
COSINE																							
Protein	4	**0.350**	102.668	4	0.348	89.419	4	0.348	98.421	4	0.342	81.656	4	0.348	71.235	4	0.349	7	0.320	6	0.304	6	0.247
	12	0.170	135.000	12	0.170	141.794	12	**0.173**	269.374	12	**0.173**	269.374	12	0.170	291.391	12	0.166	20	0.134	21	0.125	15	0.091
Breast	3	0.294	28.282	3	0.294	32.812	3	0.294	32.297	3	0.294	31.796	3	0.294	31.610	3	0.293	4	0.258	3	0.306	3	**0.332**
	8	0.021	75.859	8	0.021	77.403	8	0.021	92.515	8	0.022	82.703	8	0.021	90.610	8	0.020	2	0.027	8	**0.052**	3	0.021
Yeast	9	0.137	1103.942	9	0.137	972.313	9	0.137	680.172	9	0.137	988.547	9	0.136	716.172	9	0.135	9	0.138	6	0.132	7	**0.146**
Novartis	4	**0.950**	12.559	4	**0.950**	12.328	4	**0.950**	12.045	4	**0.950**	11.734	4	**0.950**	10.860	4	0.920	4	0.919	4	0.919	4	0.745
BreastA	2	0.687	12.125	2	0.687	10.996	2	0.687	10.921	2	0.687	10.485	2	0.687	10.453	2	0.686	2	0.691	2	**0.691**	2	0.664
BreastB	2	**0.694**	3.016	2	**0.694**	2.875	2	**0.694**	2.891	2	**0.694**	2.688	2	**0.694**	2.687	2	0.626	2	0.561	3	0.502	4	0.443
	2	**0.322**	3.000	2	**0.322**	2.875	2	**0.322**	2.875	2	**0.322**	2.687	2	0.321	2.687	2	0.314	2	0.269	3	0.264	4	0.239
DLBCLA	4	0.607	14.406	4	0.619	16.64	4	0.607	11.844	4	0.607	12.078	4	0.607	11.406	4	0.605	5	0.642	4	**0.678**	3	0.547
DLBCLB	4	0.500	22.968	4	0.500	21.109	4	0.500	20.921	4	0.500	20.031	4	0.500	20.031	4	0.502	3	0.501	3	**0.623**	5	0.385
MultiA	4	**0.831**	54.303	4	**0.831**	54.468	4	**0.831**	54.046	4	**0.831**	48.532	4	**0.831**	45.609	4	0.805	4	0.718	7	0.731	6	0.716
Iris	3	**0.942**	0.296	3	**0.942**	0.344	3	**0.942**	0.359	3	**0.942**	0.406	3	**0.942**	0.391	3	0.941	3	0.904	3	0.941	3	0.904
PEARSON																							
Protein	4	**0.345**	124.032	4	**0.345**	127.592	4	**0.345**	129.281	4	**0.345**	139.625	4	**0.345**	120.219	4	0.344	7	0.313	7	0.306	6	0.245
	12	0.164	363.957	12	0.172	251.089	12	0.168	211.186	12	0.168	211.186	12	**0.174**	382.015	12	0.167	20	0.129	27	0.136	14	0.096
Breast	3	0.311	38.109	3	0.311	41.327	3	0.311	41.609	3	0.311	42.141	3	0.311	39.360	3	0.284	2	**0.441**	2	0.368	2	0.289
	11	0.016	95.922	11	0.016	197.056	11	0.017	93.452	11	0.016	110.844	11	0.017	130.156	11	0.017	9	0.015	19	**0.024**	6	0.015
Yeast	9	0.138	1010.289	9	0.138	877	9	0.138	662.390	9	0.138	660.406	9	0.138	510.969	9	0.131	8	0.135	8	0.133	7	**0.145**
Novartis	4	**0.950**	20.621	4	**0.950**	20.156	4	**0.950**	20.357	4	**0.950**	23.422	4	**0.950**	21.265	4	0.920	4	0.919	4	0.919	4	0.746
BreastA	2	0.692	20.563	2	0.692	18.904	2	0.692	19.188	2	0.692	21.734	2	0.692	21.609	2	0.692	2	**0.705**	2	**0.705**	2	0.635
BreastB	2	**0.766**	4.734	2	**0.766**	5.562	2	**0.766**	6.453	2	**0.766**	5.546	2	**0.766**	5.562	2	0.694	3	0.502	3	0.529	3	0.445
	2	0.281	4.842	2	0.322	5.016	2	0.281	4.735	2	0.281	5.547	2	0.279	5.219	2	**0.355**	4	0.289	3	0.283	3	0.227
DLBCLA	4	0.604	20.562	4	0.604	20.577	4	0.604	20.687	4	0.604	23.140	4	0.607	19.578	4	0.585	4	0.605	4	**0.684**	4	0.586
DLBCLB	2	0.585	36.750	2	0.585	33.796	2	0.585	34.093	2	0.585	39.641	2	0.585	39.641	2	0.527	3	**0.665**	3	0.561	3	0.545
MultiA	4	**0.829**	93.395	4	**0.829**	92.655	4	**0.829**	93.093	4	**0.829**	102.156	4	**0.829**	87.079	4	0.828	4	0.718	9	0.691	4	0.705
Iris	3	0.886	0.500	3	0.886	0.64	3	0.886	0.656	3	0.886	0.781	3	0.886	0.796	3	0.886	3	0.886	3	**0.941**	3	0.886

Bibliography

[1] Aiex, R.: Uma investigação experimental da distribuição de probabilidade de tempo de solução em heurísticas GRASP e sua aplicação na análise de implementações paralelas. Ph.D. thesis, Department of Computer Science, Catholic University of Rio de Janeiro, Rio de Janeiro, Brazil (2002)

[2] Aiex, R., Binato, S., Resende, M.: Parallel GRASP with path-relinking for job shop scheduling. Parallel Computing 29, 393–430 (2003)

[3] Aiex, R., Resende, M., Pardalos, P., Toraldo, G.: GRASP with path relinking for the three-index assignment problem. INFORMS J. on Computing 17(2), 224–247 (2005)

[4] Bennett, K.P., Mangasarian, O.: Robust linear programming discrimination of two linearly inseparable sets. Optimization Methods and Software 1(1), 23–34 (1992)

[5] Binato, S., Faria Jr., H., Resende, M.: Greedy randomized adaptive path relinking. In: Sousa, J. (ed.) Proceedings of the IV Metaheuristics International Conference. pp. 393–397 (2001)

[6] Ding, C., Dubchak, I.: Multi-class protein fold recognition using support vector machines and neural networks. Bioinformatics 17(4), 349–358 (2001)

[7] Feo, T., Resende, M.: A probabilistic heuristic for a computationally difficult set covering problem. Operations Research Letters 8, 67–71 (1989)

[8] Feo, T., Resende, M.: Greedy randomized adaptive search procedures. J. of Global Optimization 6, 109–133 (1995)

[9] Festa, P., Resende, M.: GRASP: An annotated bibliography. In: Ribeiro, C., Hansen, P. (eds.) Essays and Surveys on Metaheuristics, pp. 325–367. Kluwer Academic Publishers (2002)

[10] Festa, P., Resende, M.: An annotated bibliography of GRASP – Part I: Algorithms. International Transactions on Operational Research 16, 1–24 (2009)

[11] Festa, P., Resende, M.: An annotated bibliography of GRASP – Part II: Applications. International Transactions on Operational Research (2009)

[12] Fisher, R., et al.: The use of multiple measurements in taxonomic problems. Annals of Eugenics 7, 179–188 (1936)

[13] Glover, F.: Tabu search and adaptive memory programing – Advances, applications and challenges. In: Barr, R., Helgason, R., Kennington, J. (eds.) Interfaces in Computer Science and Operations Research, pp. 1–75. Kluwer (1996)

[14] Glover, F.: Multi-start and strategic oscillation methods – Principles to exploit adaptive memory. In: Laguna, M., Gonzáles-Velarde, J. (eds.) Computing Tools for Modeling, Optimization and Simulation: Interfaces in Computer Science and Operations Research, pp. 1–24. Kluwer (2000)

[15] Glover, F., Laguna, M.: Tabu Search. Kluwer (1997)

[16] Glover, F., Laguna, M., Martí, R.: Fundamentals of scatter search and path relinking. Control and Cybernetics 39, 653–684 (2000)

[17] Kaufman, L., Rousseeuw, P.: Finding groups in data: an introduction to cluster analysis. WileyBlackwell (2005)
[18] Laguna, M., Martí, R.: GRASP and path relinking for 2-layer straight line crossing minimization. INFORMS Journal on Computing 11, 44–52 (1999)
[19] Matsumoto, M., Nishimura, T.: Mersenne Twister: A 623-dimensionally equidistributed uniform pseudo-random number generator. ACM Transactions on Modeling and Computer Simulation 8, 3–30 (1998)
[20] Monti, S., Savage, K., Kutok, J., Feuerhake, F., Kurtin, P., Mihm, M., Wu, B., Pasqualucci, L., Neuberg, D., Aguiar, R., et al.: Molecular profiling of diffuse large B-cell lymphoma identifies robust subtypes including one characterized by host inflammatory response. Blood 105(5), 1851–1861 (2005)
[21] Nakai, K., Kanehisa, M.: Expert system for predicting protein localization sites in gram-negative bacteria. Proteins: Structure, Function, and Bioinformatics 11(2), 95–110 (1991)
[22] Nascimento, M., Toledo, F., de Carvalho, A.: Investigation of a new GRASP-based clustering algorithm applied to biological data. Computers & Operations Research 37(8), 1381–1388 (Aug 2010)
[23] Resende, M., Ribeiro, C.: Greedy randomized adaptive search procedures. In: Glover, F., Kochenberger, G. (eds.) Handbook of Metaheuristics, pp. 219–249. Kluwer Academic Publishers (2002)
[24] Resende, M., Ribeiro, C.: GRASP with path-relinking: Recent advances and applications. In: Ibaraki, T., Nonobe, K., Yagiura, M. (eds.) Metaheuristics: Progress as Real Problem Solvers, pp. 29–63. Springer (2005)
[25] Resende, M., Ribeiro, C.: Greedy randomized adaptive search procedures: Advances and applications. In: Gendreau, M., Potvin, J.Y. (eds.) Handbook of Metaheuristics, pp. 281–317. Springer Science+Business Media, 2nd edn. (2010)
[26] Ribeiro, C., Rosseti, I.: A parallel GRASP for the 2-path network design problem. Lecture Notes in Computer Science 2004, 922–926 (2002)
[27] Ribeiro, C., Uchoa, E., Werneck, R.: A hybrid GRASP with perturbations for the Steiner problem in graphs. INFORMS Journal on Computing 14, 228–246 (2002)
[28] Rosenwald, A., Wright, G., Chan, W., Connors, J., Campo, E., Fisher, R., Gascoyne, R., Muller-Hermelink, H., Smeland, E., Staudt, L.: The use of molecular profiling to predict survival after chemotherapy for diffuse Large-B-cell lymphoma. The New England Journal of Medicine 346(25), 1937–1947 (2002)
[29] Rosseti, I.: Heurísticas para o problema de síntese de redes a 2-caminhos. Ph.D. thesis, Department of Computer Science, Catholic University of Rio de Janeiro, Rio de Janeiro, Brazil (July 2003)
[30] Su, A., Cooke, M., Ching, K., Hakak, Y., Walker, J., Wiltshire, T., Orth, A., Vega, R., Sapinoso, L., Moqrich, A., et al.: Large-scale analysis of the human and mouse transcriptomes. Proceedings of the National Academy of Sciences 99(7), 4465–4470 (2002)
[31] Van't, V., Laura, J., Hongyue, D., Vijver, M.V.D., He, Y., Hart, A., et al.: Gene expression profiling predicts clinical outcome of breast cancer. Nature 415(6871), 530–536 (2002)

[32] West, M., Blanchette, C., Dressman, H., Huang, E., Ishida, S., Spang, R., Zuzan, H., Olson, J., Marks, J., Nevins, J.: Predicting the clinical status of human breast cancer by using gene expression profiles. Proceedings of the National Academy of Sciences of the United States of America 98(20), 11462 (2001)