Where facility centers should be located?

A.D. López-Sánchez
Department of Economics, Quantitative Methods and Economic History
Pablo de Olavide University
Sevilla, Spain
adlopsan@upo.es

J. Sánchez-Oro
Department of Computer Science
Rey Juan Carlos University
Madrid, Spain
jesus.sanchezoro@urjc.es

A.G. Hernández-Díaz
Department of Economics, Quantitative Methods and Economic History
Pablo de Olavide University
Sevilla, Spain
agarher@upo.es

M. Laguna
Leeds School of Business
University of Colorado
Boulder, Colorado
laguna@colorado.edu

***Abstract*—Facility location problems aim to determine the best position to place facility centers. This task is not easy since many objectives may be taken into consideration to choose the best possibility and the locations can be substantially different. This paper presents a Scatter Search algorithm with Path Relinking to decide where facility centers should be located depending on the objective functions under consideration. The objectives considered in this study are to minimize the distance between demand points and their nearest facilities, to maximize the number of demand points covered by a facility center and to minimize the maximum distance between demand points and their nearest facilities. Then, this problem is addressed as a three-objective optimization problem and the algorithm proposed is compared against other competitors obtaining promising results according to three different performance metrics.**

***Index Terms*—Scatter Search, Path Relinking, Multi-objective optimization problems, Facility location problems**

I. Introduction

It is not an easy task to locate facility centers since many objectives may be taken into consideration to choose the best possibility. Depending on the objective and the constraints imposed by the decision-maker, the facility centers may be located in different places. Hence, the best location may be substantially different depending on the objective function considered for optimization. Specifically, this work is focused on three common objective functions: f_1, with the aim of minimizing the average distance between demands and their nearest facilities; f_2, focused on covering the maximum number of demand points; and f_3, which minimizes the maximum distance between demand points and their nearest facilities. But what happens when the decision-maker wants to optimize all the objectives at the same time? In such a case, the location problem becomes a multi-objective facility location problem (mo-FLP) in which the objectives are usually in conflict. That is, there is not a single solution that simultaneously optimizes all objectives, or, in other words, the value of one objective function cannot be improved without deteriorating the value of at least another objective function. Those solutions are known as efficient solutions, non-dominated solutions or Pareto optimal solutions.

The mo-FLP that optimizes f_1, f_2, and f_3, was recently studied by Karatas and Yakici, see [10]. The authors developed an hybrid algorithm, named ITER-FLOC, that combines branch & bound techniques and iterative goal programming. In particular, authors proposed a different formulation for each considered objective. Initially, lower and upper bounds for each objective are evaluated. Then, for each iteration of the ITER-FLOC algorithm, the location models are solved, verifying if the termination criterion has been achieved. If so, the algorithm ends, returning the Pareto front constructed. Otherwise, the lower and upper bounds of each objective are updated. Prior to executing the next iteration, the location models are updated with additional constraints.

They are able to generate the Pareto optimal solutions with high level of diversity and cardinality. However, the drawbacks are the requirement of preference information since goal programming is considered an *a priori* method and its slowness because it an exact algorithm that needs to be solve multiple times (one for each considered goal). Additionally, the method requires from several input parameters that can difficult the scalability of the algorithm for new datasets.

Here, the mo-FLP is addressed using a Scatter Search algorithm combined with Path Relinking (SSPR). The output of the algorithm is the approximation of the Pareto front containing efficient solutions. A variety of optimization problems has been solved and the computational results indicate that the Scatter Search algorithm is able to find the Pareto set in a simple run within short computational time.

This work is structured as follows. Section II describes the problem. Section III gives details of the Scatter Search algorithm with Path Relinking implemented to solve the problem under consideration. Section IV presents the computational results. Finally, Section V summarizes the paper and discusses

J. Sánchez-Oro is supported by the Spanish Ministry of "Economía y Competitividad", Grant Refs. TIN2015-65460-C2-2-P. A.D. López-Sánchez and A.G. Hernández-Díaz acknowledge support from the Spanish Ministry of Science and Innovation through Project ECO2016-76567-C4-1-R.

future work.

II. Location Problem Definition

Let $I = \{1 \dots m\}$ be the set of available locations to host a facility center and $J = \{1 \dots n\}$ the set of demand points that requires to be covered by a facility center. Each demand point $i \in I$ has an associated weight w_i that represents the cost of satisfying its necessity with a facility center. Additionally, let $d_{ij} \geq 0$ be the distance between a candidate location $i \in I$ and a demand point $j \in J$. The distance is evaluated as the length of the shortest path that connects x with y. A solution S for a facility location problem is then represented as the set of p candidate locations to host a facility center, and the objective of the problem is to find a solution with the optimum value with respect to one or more objective function.

This work is focused on optimizing three different objective functions simultaneously, becoming a multi-objective problem. The first considered objective function, f_1, is focused on minimizing the average weighted distance between the demand points and their nearest facility center, which can be found in the literature as the p-Median Problem (pMP). Given a solution S, the evaluation of f_1 is formally defined as:

$$f_1(S) \leftarrow \frac{1}{n} \sum_{j \in J} w_j \cdot d_{i^\star j}, \text{ where } i^\star \leftarrow \arg\min_{i \in S} d_{ij}$$

The second objective function, f_2, tries to maximize the number of demand points whose necessity is satisfied by the selected candidate locations, which is usually referenced as the Maximal Coverage Location Problem (MCLP). A demand point is covered by a facility center if the distance between them is smaller or equal than a predefined threshold r. More formally,

$$f_2(S) \leftarrow \left| \left\{ j \in J \; : \; d_{i^\star j} \leq r, \text{ where } i^\star \leftarrow \arg\min_{i \in S} d_{ij} \right\} \right|$$

The third objective function, f_3, is intended to minimize the maximum distance between the demand points and the facility centers, which result in the p-Center Problem (pCP). This objective function is formally defined as:

$$f_3(S) \leftarrow \max_{j \in J} d_{i^\star j}, \text{ where } i^\star \leftarrow \arg\min_{i \in S} d_{ij}$$

Some of the first studies dealing the previous location problems were proposed by [2], [8], [9].

III. Scatter Search algorithm with Path Relinking

Scatter Search (SS), first proposed by Glover, see [5], is a metaheuristic framework which generates, maintains, and transforms a reference set of solutions, *RefSet*. It has been successfully applied to a large variety of optimization problems [12], [13]. Figure 1 depicts the general scheme for Scatter Search.

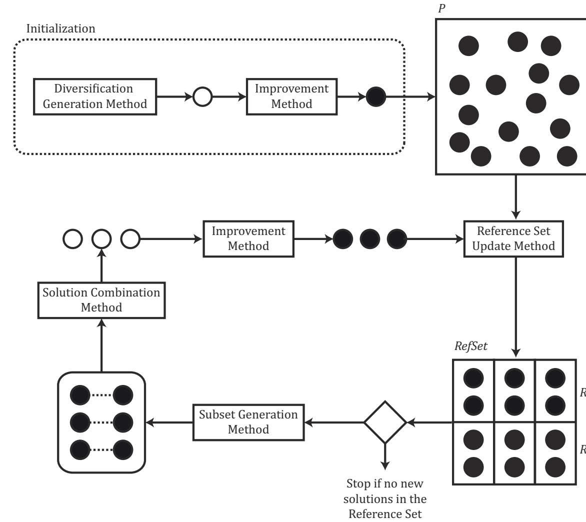

Fig. 1. Scatter Search algorithm.

The method firstly generates an initial population P. These solutions are generated using a diversification generation method (Section III-A). Then, a local optimum for each solution is found by an improvement method (Section III-B). It is important to remark that there are not repeated solutions in the initial population P.

Traditional implementations of Scatter Search constructs the reference set *RefSet* with the $\beta/2$ best solutions of set P in terms of objective function value. This work adapt this criterion for a multi-objective approach, by dividing the *RefSet* in three subsets, $RefSet_1$, $RefSet_2$, and $RefSet_3$, with $|RefSet_1| = |RefSet_2| = RefSet_3 = \beta$. In particular, $RefSet_1$ initially contains the best $\beta/2$ solutions with respect to f_1, while $RefSet_2$ and $RefSet_3$ contains the best $\beta/2$ ones with respect to f_2 and f_3, respectively. The remaining $\beta/2$ solutions for each *RefSet* are selected as the most diverse solutions among the remaining solutions in P. Diversity between a solution S and the *RefSet* under construction is measured as the minimum distance between S and every solution in the *RefSet*, considering the distance between two solutions as the number of candidate locations that differs in both solutions.

Once the *RefSet* has been created, Scatter Search selects the subsets of reference solutions that will be combined with the aim of finding new trial solutions. The most common implementation of the subset generation method consists of combining all pairs of solutions included in the *RefSet*. Considering that we maintain three *RefSet*, this implementation would be rather computationally demanding. Instead, we generate all pairs of solutions (S, S') such that S and S' belong to different *RefSet* (i.e., $S \in RefSet_i$, $S' \in RefSet_j$, with $1 \leq i, j \leq 3 \wedge i \neq j$).

The solution combination method (Section III-C) is designed to combine all the subsets generated in the previous step in order to generate new solutions that become candidate for entering in the *RefSet*. Regarding that the moFLP considers three different objective functions, a local optimum is found with respect to each one of the by using the improvement method. Then, each improved solution is evaluated for being included in its corresponding *RefSet*. In particular, a solution enters in the *RefSet* if it is better than the worst solution already in the *RefSet*. It is worth mentioning that the size of the *RefSet* remains constant throughout the whole process. Therefore, the new solution must replace another one. Specifically, the new solution replaces the most similar solution already in the *RefSet* that presents a quality smaller or equal than it.

Scatter Search iterates until a stopping criterion is met. The algorithm proposed in this work stops when it has not been possible to include new solutions in any of the *RefSet*, returning the set of efficient solutions found during the search.

A. *Diversification Generation Method*

The diversification generation method (DGM) is designed for creating an initial set of solutions P which will become the source for creating the initial *RefSet*. On the one hand, P should contain high quality solutions in order to guide the search through promising regions of the search space. On the other hand, solutions in P must be diverse enough to provide different solutions for the combination stage, thus diversifying the search.

We propose three different constructive methods in this work, DGM_1, DGM_2, and DGM_3 each one of them focused on generating promising solutions for f_1, f_2, and f_3, respectively. With the aim of increasing the diversity of the set of solutions constructed, we propose a Greedy Randomized Adaptive Search Procedure (GRASP). GRASP is a multi-start methodology originally proposed by [4], that is conformed by two stages: construction and improvement. The former consists of a greedy, randomized, and adaptive construction of a solution while the latter is designed for finding a local optimum with respect to a predefined neighborhood.

The diversification generation method proposed in this work follows a traditional GRASP construction scheme. The method initially creates a candidate list *CL* with all the candidate locations available to host a facility. Then, each candidate location is evaluated with a greedy function that estimates the relevance of locating a facility in that candidate. For this problem, we propose a different greedy function for each objective function considered f_1, f_2, f_3. In particular, the greedy function value for each candidate location is calculated as the corresponding objective value (f_1, f_2, and f_3, respectively) if the location is assigned to host a facility. After evaluating all the candidates, the method calculates a threshold μ as follows:

$$\mu \leftarrow g_{\min} + \alpha \cdot (g_{\max} - g_{\min})$$

This threshold is used for constructing the restricted candidate list *RCL*, that contains the most promising candidates to host a facility. In particular, the *RCL* is conformed with all the candidate locations whose greedy function value is smaller or equal than threshold μ. For each iteration, a random element is selected from the *RCL* to host the next facility. Notice that $\alpha \in [0,1]$ is a parameter of the method that controls the randomness of the constructive procedure. On the one hand, if $\alpha = 0$ then the *RCL* would contain the candidates with the minimum greedy function value, being a totally greedy procedure. On the other hand, when $\alpha = 1$, the *RCL* contains all the candidate locations in *CL*, becoming a random procedure. Then, it is interesting to find a balance between diversification and intensification by varying the value of the α parameter. A new candidate is selected in each iteration following this strategy until p candidate locations already host a facility.

B. *Improvement Method*

The diversification generation method is designed to produce not only high quality solutions, but also diverse ones. The increase in the diversity of the solutions generated usually implies a decrease in the quality of those solutions. Therefore, it is interesting to use a local improving method designed to find a local optimum with respect to a previously defined neighborhood.

Regarding the algorithmic proposal for the moFLP, the solutions included in the set of initial solutions P can be further improved using a local optimizer. Scatter Search is a versatile methodology than allows using different types of optimizers, from local search methods to complete metaheuristics like Tabu Search or VNS, among others (see [REF AUTOCITA] for some successfull application of complete metaheuristics in the improving phase). For this problem, we propose a local search method designed to improve the quality of the initial solutions and of those resulting from the combination method. The computational effort required to evaluate the solutions makes the use of complete metaheuristics not suitable for the problem under consideration.

Prior to define the local improvement method, it is necessary to present the neighborhood of solutions considered by the method. We define the neighborhood $N(S)$ as all the solutions that can be reached from an initial solution S by performing a single *interchange* move, which consists of removing a selected location and replacing it with any non-selected facility location. More formally,

$$N(S) \leftarrow \{S' \ : \ S' \leftarrow S \setminus \{v\} \cup \{u\} \ \forall \ v \in S \ \forall \ u \in V \setminus S\}$$

Having defined $N(S)$, the local search method proposed visits all neighbor solutions in a random order and replaces the incumbent solution with the first neighbor solution with a better objective function value. It is worth mentioning that the local search proposed follows a first improvement approach in order to reduce the computational effort required to apply it. Specifically, if an interchange move results in an improvement with respect to the objective function being optimized, the move is performed, restarting the search with the new best solution. This strategy reduces the complexity of the search since the opposite strategy, best improvement, requires the complete exploration of the neighborhood in each iteration to select the best solution in the neighborhood.

In the framework of Scatter Search, the improvement method is applied in two different stages, see Figure 1. First of all, it is used for locally improving the solutions that are included in the initial population. It is worth mentioning that the set P is divided into three subsets (one for each objective function), so the local search method improves each solution with respect to the objective function considered for its construction.

The local search method is also applied to those solutions resulting from the combination stage. As it is described in Section III-C, the combination stage does not produce local optimum with respect to any neighborhood, so the resulting solutions can be further improved with a local optimizer. Specifically, for each solution derived from the combination stage, three local optima are found, one for each objective function of the moFLP, respectively. Then, each improved solution is evaluated to be included in the *RefSet*.

C. Solution Combination Method

The solution combination method is responsible for generating new solutions in each iteration of the Scatter Search algorithm by combining two or more solutions that are already in the *RefSet*. The combination can be performed following different strategies, from genetic operators to the generation of paths between solutions.

In this work we propose using Path Relinking (PR) [6], [7] as a combination method, which has been successfully applied in several recent works [1], [3]. Given an initial and a guiding solution, s_i and s_g, respectively, PR constructs a path of solutions that starts in s_i and finishes in s_g. The objective of PR is iteratively transform the initial solution into the guiding one. The transformation is achieved by adding attributes of the guiding solution into the initial one while removing those attributes of the initial solution that are not present in the guiding one, stopping when s_i becomes s_g.

There exist several strategies for combining solutions in the context of Path Relinking: Random Path Relinking, Greedy Path Relinking, or Greedy Randomized Path Relinking, among others. Most of the greedy variants require the exploration of all the alternative solutions in each step of the path, in order to select the most promising solution to continue the path. However, these strategies are usually very time consuming, increasing the required computing time to execute the algorithm. We have selected the Random Path Relinking variant in order to accelerate the proposed algorithm, thus increasing the diversification of the search.

Starting from the initial solution s_i, Random Path Relinking generates a random solution in the neighborhood defined in Section III-B that inserts a new candidate location which is already in the guiding solution, removing one of the candidate locations currently in s_i that does not belongs to s_g. Notice that after a certain number of iterations, the initial solution would become the guiding one, since in each iteration the initial solution will have an additional candidate location in common with s_g.

IV. Computational Results

In this section the numerical results are shown in order to prove the superiority of the Scatter Search algorithm with Path Relinking in comparison to the algorithm proposed by Karatas and Yakici in [10].

Since the original instances are not available, we request the code from the previous work in order to have a fair comparison. Additionally, 20 new instances of sizes were generated following the instructions of the previous work. The instances are divided into three sets depending on their size: small, those with 20 candidate locations and 50 demand points; medium, with 50 candidate locations and 100 demand points; and large, with 200 candidate locations and 400 demand points. Table I summarizes the following parameters of the instances generated:

- m: number of candidate locations to host a facility
- n: number of demand points
- p: number of candidate locations that must be selected
- r: radius in which a facility is covering a demand point

We refer the reader to [10] for a more detailed description on the instance structure.

TABLE I
Parameters setting for the instances.

Parameter	Small	Medium	Large
m	20	50	200
n	50	100	400
p	5	10	15
r	20	15	10

All the algorithms proposed in this work have been implemented using Java 8 and the experiments were performed on an Intel Core i7 920 (2.67 GHz) with 8 GB RAM. It is worth mentioning that the previous algorithm has been also executed in the same computer in order to have a fair comparison.

Table II shows average results of both the Scatter Search with Path Relinking algorithm proposed (SSPR) and the best previous method (ITER-FLOC) the 20 instances. Regarding the multi-objective nature of the problem under consideration, we have considered using the following metrics: coverage, C; hypervolume, HV; epsilon indicator, Eps; and CPU time. Results in Table II shows the superiority of the SSPR algorithm.

If we focus on the coverage metric, it can be hold that the proportion of solutions covered by the SSPR algorithm is larger than the proportion of solution covered by the ITER-FLOC algorithm. Furthermore, the SSPR scales better thatn the ITER-FLOC algorithm, as it can be seen with the increase of the solutions covered by SSPR when increasing the size of the instance, achieving a 100% of coverage when analyzing the set of large instances.

Regarding the hypervolume (larger values are better) and the epsilon indicator (smaller values are better), we can conclude that SSPR consistently obtains better results thatn the ITER-FLOC algorithm. Again, the larger the instance set, the better the results of SSPR when compared against ITER-FLOC. This behavior suggests that SSPR is a more adequate algorithm for real-life problems with a large number of candidate locations and demand points.

Finally, if we consider the computing time, we can see that ITER-FLOC is equivalent or even faster when considering small or medium instances, but the performance does not scale good with the size of the instance. Therefore, when solving the largest instances the proposed SSPR algorithm is considerably faster than the ITER-FLOC approach.

TABLE II
AVERAGE RESULTS.

	C(SSPR,ITER-FLOC)	C(ITER-FLOC,SSPR)
Small	**0.61**	0.17
Medium	**0.82**	0.00
Large	**1.00**	0.00
	HV(SSPR)	HV(ITER-FLOC)
Small	**0.28**	0.07
Medium	**0.52**	0.12
Large	**0.81**	0.20
	Eps(SSPR)	Eps(ITER-FLOC)
Small	**0.44**	0.78
Medium	**0.14**	0.70
Large	**0.00**	0.50
	CPU(SSPR)	CPU(ITER-FLOC)
Small	**6.41**	7.25
Medium	72.66	**19.74**
Large	**1920.53**	5920.34

V. Conclusions

A population-based metaheuristic with a method for combining solutions have been proposed for a multi-objective facility location problem (mo-FLP) which considers three different objectives of interest in real-life problems: to minimize the average distance between demands and their nearest facilities, to maximize the total number of demand points covered, and to minimize the maximum distance between demand points and their nearest facilities.

The computational experiments shows how Scatter Search with Path Relinking is a suitable algorithm for solving large scale instances, performing better in both quality and computing time than the best previous algorithm found in the state of the art. The experiments have been performed in the same computer in order to have comparable results, concluding that SSPR outperforms the best previous method considering all the metrics presented.

Acknowledgment

J. Sánchez-Oro is supported by the Spanish Ministry of "Economía y Competitividad", Grant Refs. TIN2015-65460-C2-2-P and TIN2014-54806-R.

A.D. López-Sánchez and A.G. Hernández-Díaz acknowledge support from the Spanish Ministry of Science and Innovation through Project ECO2016-76567-C4-1-R.

References

[1] V. Campos, R. Martí, J. Sánchez-Oro, A. Duarte, "GRASP with path relinking for the orienteering problem", Journal of the Operational Research Society, vol. 65, pp. 1800–1813, 2014.
[2] R. Church, and C.R. Velle, "The maximal covering location problem," Papers in Regional Science, vol. 32, pp. 101–118,1974.
[3] A. Duarte, J. Sánchez-Oro, M.G.C. Resende, F. Glover, R. Martí, "Greedy randomized adaptive search procedure with exterior path relinking for differential dispersion minimization", Information Sciences, vol. 296, pp. 46–60, 2015.
[4] T. Feo, M.G.C. Resende. "A probabilistic heuristic for a computationally difficult set covering problem", Operations Research Letters, vol. 8, pp. 67–71, 1989.
[5] F. Glover, "Heuristics for integer programming using surrogate constraints," Decision Sciences, vol. 8, pp. 156–166, 1977.
[6] F. Glover"Tabu search and adaptive memory programming – advances, applications, and challenges," Kluwer Academic Publishers, 1996.
[7] F. Glover, and M. Laguna, "Tabu Search," Kluwer Academic Publishers, 1997.
[8] S.L. Hakimi, "Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph," Operations Research, vol. 12, pp. 450–459, 1964.
[9] S.L. Hakimi, "Optimum Distribution of Switching Centers in a Communication Network and Some Related Graph Theoretic Problems," Operations Research, vol. 13, pp. 462–475, 1965.
[10] M. Karatas, and E. Yakc, "An iterative solution approach to a multi-objective facility location problem," Applied Soft Computing, vol. 62, pp. 272–287, 2018.
[11] R. Martí, M. Laguna, and F. Glover. "Principles of scatter search," European Journal of Operational Research, vol. 169, pp. 359–372, 2006.
[12] J. Sánchez-Oro, M. Laguna, R. Martí, and A. Duarte. "Scatter Search for the Bandpass Problem", Journal of Global Optimization, vol. 66(4), pp. 769–790, 2016
[13] J. Sánchez-Oro, A. Martínez-Gavara, M. Laguna, A. Duarte, and R. Martí. "Variable neighborhood scatter search for the incremental graph drawing problem", Computational Optimization and Applications, vol. 68(3), pp. 775–797, 2017
[14] J.J. Sylvester, "A Question in the Geometry of Situation," Quarterly Journal of Pure and Applied Mathematics, vol. 1, 1857.