Yugoslav Journal of Operations Research
27 (2017), Number 3, 291–300
DOI: 10.2298/YJOR160517018B

USING INJECTION POINTS IN REFORMULATION LOCAL SEARCH FOR SOLVING CONTINUOUS LOCATION PROBLEMS

Jack BRIMBERG
Department of Mathematics and Computer Science, Royal Military College of Canada, Kingston, ON, K7K7B4, Canada
Jack.Brimberg@rmc.ca

Zvi DREZNER
Steven G. Mihaylo College of Business and Economics, California State University, Fullerton, CA 92634, USA
zdrezner@fullerton.edu

Nenad MLADENOVIĆ
Mathematical Institiute, Serbian Academy of Sciences and Arts, Knez Mihajlova 36, 11000 Belgrade, Serbia
nenad@mi.sanu.ac.rs

Said SALHI
Centre for Logistics & Heuristic Optimization, Kent Business School, University of Kent, Canterbury CT2 7PE, United Kingdom
S.Salhi@kent.ac.uk

Received: May 2016 / Accepted: October 2016

Abstract: Reformulation local search (RLS) has been recently proposed as a new approach for solving continuous location problems. The main idea, although not new, is to exploit the relation between the continuous model and its discrete counterpart. The RLS switches between the continuous model and a discrete relaxation in order to expand the search. In each iteration new points obtained in the continuous phase are added to the discrete formulation. Thus, the two formulations become equivalent in a limiting sense. In this paper we introduce the idea of adding 'injection points' in the discrete phase of RLS in order to escape a current local solution. Preliminary results are obtained on benchmark data sets for the multi-source Weber problem that support further investigation of the RLS framework.

Keywords: Continuous Location, Weber Oroblem, Formulation Space Search, Reformulation Descent, Variable Neighbourhood Search.

MSC: 90B85, 90C26.

1. INTRODUCTION

Continuous location problems generally require finding the location of a given number, say p, of new facility sites in $\mathbb{R}^N$, or a sub-region of $\mathbb{R}^N$, in order to serve a given set of existing facilities, also known as customers or demand (or fixed) points. In most applications, the facilities and customers are located in the Euclidean plane ($N = 2$). Extensions include modelling the entities as lines or areas instead of points. In all cases a distance function, such as the Euclidean, rectangular, or more generally the l_p norm, is required to calculate the distance between facilities and customers. For an overview of the continuous location literature, the interested reader is referred to Love et al.[13]. For a relatively recent survey of the multi-source Weber problem (also referred to as the continuous location-allocation problem), see Brimberg et al. [4].

Finding an optimal solution in continuous location problems is often a difficult proposition due to the existence of multiple local optima. For example, solving the multi-source Weber problem is equivalent to enumerating the Voronoi partitions of the customer set which is NP-hard [14]. Brimberg et al. [3] demonstrate its complexity on the well-known 50-customer problem in Eilon et al. [10] by using 10,000 random restarts of Cooper's algorithm for $p = 5, 10, 15$, to generate 272, 3008, and 3363, different local minima, respectively. The worst deviation from the optimal solution was found to be, respectively, 47%, 66% and 70%, while the optimal solution was obtained 690 times for $p = 5$, 34 times for $p = 10$, and only once for $p = 15$. Such relatively small instances are useful in demonstrating the tendency for the number of local minima to increase exponentially with problem size as defined by n and p. They also substantiate the need to use heuristic methods.

The classical procedure for solving the multi-source Weber problem, which is known as Cooper's algorithm, is based on a simple, yet clever, insight: if the locations of the new facilities are fixed, the allocation sub-problem is solved optimally by assigning customers to their nearest facilities; once the allocations are fixed, the location sub-problem is solved optimally as p independent single facility problems with convex objective functions, using for example the well-known Weiszfeld procedure [20]. Thus, Cooper's algorithm alternates between location and allocation steps until no further improvement is possible (a local optimum is reached). The original method proposed by Cooper began with a heuristic partitioning of the customer set. Various changes were later proposed including the use of random initial locations of the facilities ([19]). Although other local searches were devised, a multi-start version of Cooper's algorithm remained the state-of-the-art for several years until metaheuristics were applied to the problem, e.g., see [5,6]. Cooper's algorithm still maintains its usefulness as a local search within many of the current metaheuristic-based methods.

The same continuous location problem may be formulated in discrete space by restricting the potential new facility sites to a specified finite set of points in the continuous space. A distance matrix containing distances between all facility-customer pairs of points (vertices) in the associated graph is constructed using the distance function given in the continuous model. If the candidate sites are chosen well, the constructed graph becomes an accurate representation of the original continuous problem. Thus, a good solution on the graph should also provide a good solution to the original problem. To illustrate, if we restrict the candidate facility sites to the given set of fixed points, the multi-source Weber problem converts to the well-studied (discrete) p-median problem. For a review of the latter problem, see e.g., Mladenović et al. [15].

Exploiting the relation between the discrete p-median model and the continuous Weber problem has been suggested as early as in the original work of Cooper ([7,8]). More recently, Hansen*et* al. [12] tested a heuristic that first solves the p-median problem exactly using a primal-dual algorithm by Erlenkotter [11], and then completes one step of "continuous-space adjustment" by solving the p continuous single facility problems identified in the first phase. Brimberg *e*t al. [5] test this heuristic among others, and conclude that computation time becomes a limiting factor on larger problem instances, due to the exact solution of the discrete model. More recently, Brimberg *e*t al. [2] extend the p-median based method of Hansen *e*t al. [12] as follows:

(i) A simple and fast vertex swap heuristic is used on the discrete problem instead of solving it exactly;

(ii) A Cooper-style alternating locate-allocate heuristic is used to subsequently descend from the discrete solution to a local minimum in continuous space (instead of stopping after one step of continuous adjustment).

Excellent computational results are obtained from a random multi-start version of this simple method.

Brimberg *e*t al. [2] also propose a new framework for exploiting the relation between a continuous location problem and its discrete counterpart which they term reformulation local search (RLS for short). The basic procedure is reviewed in the next section. Section 3 then introduces the idea of adding "injection points" in the discrete phase to extend the search whenever the basic procedure terminates. This new procedure is referred to as "augmented" RLS (ARLS). Various strategies for adding injection points are also discussed. Preliminary computational results are presented in section 4 where ARLS is used in the improvement step of an available Variable Neighborhood Search (VNS) heuristic. These results suggest that further development of ARLS is warranted. Conclusions and areas of future research are highlighted in section 5.

2. A REVIEW of REFORMULATION LOCAL SEARCH

We consider an unconstrained location problem of the general form

$$\min f(X_1, X_2, \ldots, X_p) \qquad (GLP)$$

where $X_i = (x_i, y_i) \in \mathbb{R}^2$ gives the unknown location of new facility $i, i = 1, \ldots, p$, and the objective function $f(.)$ represents some performance measure. To illustrate, consider the standard multi-source Weber problem, which is formulated as follows:

$$\min f(X_1, X_2, \ldots, X_p) = \sum_{j=1}^{n} w_j \min_{i=1,\ldots,p}\{l_2(X_i - A_j)\}. \qquad (MWP)$$

Here A_j denotes the known coordinates of customer j, $w_j > 0$, the demand at A_j, and $l_2(X_i - A_j)$ the Euclidean distance between the pair of points X_i and A_j, $i = 1, \ldots, p, j = 1, \ldots, n$. Thus, the objective function gives a sum of weighted distances from the demand points to their nearest facilities. In a practical setting this sum would be used to estimate the total cost of servicing the demand points. Another important illustration of (GLP) is given by the continuous weighted *p-center* problem:

$$\min g(X_1, X_2, \ldots, X_p) = \max_{j=1,\ldots,n}\{v_j \min_{i=1,\ldots,p}\{l_2(X_i - A_j)\}\}, \qquad (WCP)$$

where weights $v_j > 0$ are, for example, proportional to the populations of the fixed centres. The objective function $g(.)$ gives the maximum (weighted) distance between the demand points and their nearest facilities, and thus, represents a measure of the quality of service of the current solution. Most applications of the p-center problem consider the unweighted version ($v_j = 1, \forall j$) where all customers are treated equally and the goal is to give the best possible service to the farthest customer.

Now let S denote a finite set of identified potential sites for the new facilities, and X a subset of p of these sites. For example, $S = \{A_1, ..., A_n\}$, where typically $n >> p$, has been recommended in earlier works as noted above. The discrete approximation of (GLP) is then given by:

$$\min_{X \subset S} f(X). \qquad (GLP)'$$

We also select local searches L_C and L_D to use to solve (GLP) and (GLP)', respectively. These search engines stop at a current solution if, and only if, a better solution cannot be found in the specified neighborhood of this point.

The RLS method given in Brimberg *et* al. [2] solves alternately (GLP) with L_C and (GLP)' with L_D using the obtained solution in the current phase (continuous/discrete) as the starting solution of the next phase (discrete/continuous). Thus, the RLS moves through a sequence of improving solutions until the current solution is locally optimal in both the continuous and discrete spaces. A novel aspect of RLS is that the set of candidate sites in (GLP)' grows each time as new solutions found in the preceding continuous phase are added to it. The steps are outlined below.

Algorithm 1: Basic Reformulation Local Search (RLS)

Step 1: Select an initial solution $X^0 = \{X_1^0, \ldots, X_p^0\}$, and initial set S for $(GLP)'$.

Step 2 (solving the continuous problem): $L_C(X^0) \rightarrow X^C$ (where $X^C \neq X^0$, only if $f(X^C) < f(X^0)$).

Step 3 (augmenting S): $S \leftarrow S \cup X^C$.

Step 4 (solving the discrete problem): $L_D(X^C) \rightarrow X^D$ (where $X^D \neq X^C$, only if $f(X^D) < f(X^C)$).

Step 5: If $X^D = X^C$, stop (final solution = X^D); else $X^0 \leftarrow X^D$ and return to step 2.

3. ADDING INJECTION POINTS in REFORMULATION LOCAL SEARCH

A possible weakness of basic RLS, observed in testing of this procedure on (MWP), is that it may converge to a local optimum after only a few switches between continuous and discrete spaces. To try to improve solution quality, we suggest the addition of "injection points" to the discrete model (GLP)'. The concept of adding injecting points to the current problem is also employed in genetic algorithms where new chromosomes are injected to the population from time to time either regularly or intelligently, see Salhi and Gamal [17] for solving a class of continuous location problems and Salhi and Petch [18] for dealing with multi-trip routing problems. Injection points may be added each time augmentation of set S occurs (step 3 in Algorithm 1), or we may start the process once the basic RLS procedure terminates. Denote these points by $Y_j, j = 1, ..., K$, where K (a new parameter) limits the number of injection points that will be added. These additional points may be generated in a number of different ways. Some possible strategies are discussed below.

1. Select a pair of demand points (or facilities, or a combination of the two) at random and choose the next injection point, $Y_j = \alpha A_{j_1} + (1-\alpha) A_{j_2}$ where α is a uniform random variable in the interval (0,1). Alternatively, choose $\alpha = 1/2$ to get the mid-point, or a value that maximizes the distance from the new injection point to its nearest facility. We may extend this approach to three or more vertices in the current set S. For example,

$$Y_j = \alpha_1 A_{j_1} + \alpha_2 A_{j_2} + \alpha_3 A_{j_3},$$

   where $\sum \alpha_i = 1$ and $\alpha_i \geq 0, \forall i$. To choose the centroid, set $\alpha_i = 1/m, \forall i$, where m is the number of vertices selected.

2. Run a local search for the continuous problem (which can be different from L_c) from some starting solution, and add some facility sites not currently in S. Various strategies such as greedy or random may be used for adding these points to S. Different strategies may also be devised and tested for selecting a starting solution for the local search. For example, we could select p facility sites randomly, or by a combination of greedy and random (e.g., see Drezner et al.[9]).

In general, g (another parameter) injection points may be inserted at a time. In the procedure below, we set $g = 1$ and begin the injection process after basic RLS (Algorithm 1) reaches a local minimum.

Algorithm 2: Augmented Reformulation Local Search (ARLS)

Step 1: Select an initial solution $X^0 = \{X_1^0, \ldots, X_p^0\}$, an initial set S for $(GLP)'$, and a value for parameter K; set $j = 0$.

Step 2 (solving the continuous problem): $L_C(X^0) \rightarrow X^C$ (where $X^C \neq X^0$, only if $f(X^C) < f(X^0)$).

Step 3 (augmenting S): $S \leftarrow S \cup X^C$.

Step 4 (solving the discrete problem): $L_D(X^C) \rightarrow X^D$ (where $X^D \neq X^C$, only if $f(X^D) < f(X^C)$).

Step 5: If $X^D \neq X^C$, $X^0 \leftarrow X^D$ and return to step 2;
elseif $j < K$ obtain the next injection point Y_j,
set $S \leftarrow S \cup \{Y_j\}, j \leftarrow j + 1$, and return to step 4;
else stop.

Brimberg et al. [2] give a small example to illustrate that the addition of Weber points to set S in the discrete formulation of (MWP) can improve the solution quality by filling *'holes'* where no fixed points occur in the continuous space. We may use the same example and the same logic to motivate the process of adding injection points. If the initial solution in step 1 of basic RLS is restricted to a random selection of fixed points, algorithm 1 will always fail to find the optimal solution for this particular example. However, the addition of injection points will easily overcome this deficiency.

4. COMPUTATIONAL EXPERIMENTS

The objective of the computational tests is to validate the use of injection points within the RLS framework. Thus, only preliminary experimental results are reported here on four data sets commonly used to test new heuristics for the multi-source Weber problem (see Brimberg et al.[5]). These include the 50-customer problem in Eilon et al. [10], the 287-customer ambulance problem from

		deviation (%)				Running time			
	Best		VNS + ARLS			VNS	VNS + ARLS		
p	known	VNS	$g = 10$	$g = 3$	$g = 2$		$g = 10$	$g = 3$	$g = 2$
10	115339.03	0.00	0.00	0.00	0.00	5.42	1.81	1.91	1.80
20	63389.02	0.00	0.00	0.00	0.00	27.08	14.38	11.66	14.25
30	44705.19	0.01	0.01	0.01	0.01	24.70	39.55	26.33	39.73
40	35704.41	0.02	0.06	0.01	0.06	39.55	19.98	25.89	20.27
50	29338.01	0.00	0.00	0.11	0.00	27.19	27.48	15.14	33.42
60	24504.39	0.12	0.00	0.00	0.03	3.86	22.59	25.19	44.83
70	21465.44	0.04	0.03	0.03	0.04	34.05	16.22	22.69	11.39
80	19193.86	0.19	0.35	1.02	0.54	33.98	12.67	26.05	9.66
90	17514.42	0.52	0.66	0.48	0.69	42.84	17.89	49.23	9.25
100	16083.54	0.03	0.11	0.11	0.27	47.37	39.91	14.94	48.86

Table 1: ARLS results for $n = 654$ with $\alpha = 1/2$

Bongartz *e*t al. [1], and the 654- and the 1060-customer problems listed in the TSP library (see [16]).

To study the impact of injection points, two versions of VNS are compared. The first version is a basic VNS heuristic which uses relocation of facilities in the shaking operation, and basic RLS for the local search. The second version replaces the standard RLS with the new ARLS. Injection points are generated here using the first strategy only with two vertices randomly selected from S:

$$Y_j = \alpha A_{j_1} + (1 - \alpha) A_{j_2}.$$

Also, we initially set $\alpha = 1/2$ for simplicity. Results for $n = 50$ and $n = 287$ were very similar with both heuristics, so we start with $n = 654$ customers. Both methods use $t_{max} = 50$ secs as a stopping criterion and $k_{max} = 10$ for the maximum neighborhood size (shake) in Table 1.

Summary results are presented in Table 1 for $n = 654$, and p =10, 20, ...,100. Each heuristic is run only once on each problem instance, and results are compared to the best known objective function values given in Brimberg *e*t al. [5], which were obtained after considerable testing with several state-of-the-art heuristics. It is interesting to observe that VNS+ARLS reaches the best known solutions in half the time or less of VNS for small values of p (=10, 20) and certainly well within the time limit of 50 secs. However, for large values of p, the addition of injection points besides increasing the size of the discrete subproblem may have a negative effect on the solution quality (see $p = 80, 100$). This can be expected since for large p, some of the facilities will coincide with customer locations in the optimal solution [12]. However, our injection points are generated not to coincide with customer locations as the latter are already included in the set of potential sites. This observation does not necessarily nullify the advantage of using injection points as some facilities will not coincide with customer sites for most values of $p < n$, in general.

We also tested the variant of ARLS where α is randomly taken from the interval (0,1). It appears that there is no difference in solution quality on average for random α and $\alpha = 1/2$. In Table 2 we give results of VNS and this variant of ARLS which we refer to as VNS+ for $n = 1060$. Here the tested version with random α uses $g = 4$ while the values of p vary from 10 to 150 with an increment of 10. Observe as in Table 1 that solution quality with ARLS is very good without finetuning of the parameters.

	Objective function			deviation (%)		Running time	
p	Best Known	VNS	VNS+	VNS	VNS+	VNS	VNS+
10	1249564.75	1249564.75	1249564.75	0.00	0.00	8.22	12.14
20	828685.63	828689.44	828690.06	0.00	0.00	134.44	241.95
30	638212.31	638231.63	638219.44	0.00	0.00	112.77	79.80
40	529660.12	529715.81	529788.31	0.01	0.00	187.23	68.50
50	453109.56	453191.91	453350.63	0.02	0.02	299.22	206.77
60	397674.53	397779.16	397737.81	0.03	0.02	290.23	96.66
70	357335.13	357373.56	357611.78	0.01	0.02	270.84	293.41
80	325971.25	326076.66	326067.00	0.03	0.03	104.94	34.17
90	302479.03	302889.69	302843.66	0.14	0.12	277.88	216.52
100	282536.44	283103.06	283022.50	0.20	0.17	109.63	96.41
110	264959.97	265483.13	265520.13	0.20	0.21	280.12	208.61
120	249050.48	249619.31	249707.58	0.23	0.26	67.72	297.34
130	235203.39	236059.08	235490.44	0.36	0.12	290.78	133.44
140	223062.63	223518.33	224032.66	0.20	0.43	213.63	297.44
150	212236.27	212705.41	212663.59	0.22	0.20	297.20	71.05

Table 2: ARLS results for $n = 1060$, random $\alpha \in (0, 1)$ and $g = 4$

The same basic observation from Table 1 appears to hold for Table 2, namely the injection points tend to increase the convergence speed for small values of p (say p up to 10% of n). Since relatively small values of p appear often in practical problems, we conclude that research in the direction of finding good strategies for injection points is useful.

5. CONCLUSIONS and SUGGESTIONS

In this study we extend the framework for reformulation local search (RLS) by examining the addition of injection points in the discrete phase. We refer to this new approach as Augmented reformulation local search (ARLS). Our preliminary empirical testing on existing data sets for the multi-source Weber problem proves to be interesting though the results are not overwhelmingly superior to the basic RLS. This is expected as the basic RLS already performs well, and the main objective is to put forward such new ideas that will hopefully be revisited in the aim to reach a more polished and finished product.

In this study we opted for a simple local search in both continuous and discrete phases, but more powerful local search operators from which we may select L_C

and/ or L_D can be used instead. For instance, the recently proposed local search operators by Drezner et al. [9] are worth examining. Clearly, the decision for L_C will affect the quality of the points (the X_i's) used to augment S in step 3 of RLS. By choosing a better operator L_C, we may expect the quality of the X_i's to improve, and as a consequence, better quality solutions will be obtained in the discrete phase. The choice of L_D will also affect the quality of these solutions as well as the number of iterations until the algorithm terminates.

Finally we note that the reformulation framework need not be restricted to local searches. Thus, the search operators L_C and/or L_D may be selected as more powerful searches using metaheuristics instead of local search. Design issues such as first or best improvement need to be considered in this more general "Reformulation Search" package.

From a practical perspective, although a specific model is used here for illustration purposes, the procedures are readily applied to other continuous location problems where the general objective is to locate p facilities in order to service a given set of n demand points in some 'optimal' way.

Acknowledgement: This research has been supported in part by a Natural Sciences and Engineering Research Council of Canada Discovery Grant (NSERC #205041-2014), the UK Research Council EPSRC (EP/ I009299/1) and the Spanish Ministry of Economy and Competitiveness (research project #MTM2015-70260-P). The work of N. Mladenovic is partly supported by the Ministry of Education, Science and Technology of Serbia, project number 174010 and by the Ministry of Education and Science, Republic of Kazakhstan (Institute of Information and Computer Technologies, project no. 0115PK00546).

REFERENCES

[1] I. Bongartz, P.H. Calamai, and A.R. Conn, "A projection method for l_p norm location-allocation problems", *Mathematical Programming*, 66 (1994) 283–312.

[2] J. Brimberg, Z. Drezner, N. Mladenović, and S. Salhi, "A new local search for continuous location problems", *European Journal of Operational Research*, 232 (2014) 256–265.

[3] J. Brimberg, P. Hansen, and N. Mladenović, "Attraction probabilities in variable neighbourhood search", *4OR- Quarterly Journal of Operations Research*, 8 (2010) 181–194.

[4] J. Brimberg, P. Hansen, N. Mladenović, and S. Salhi, "A survey of solution methods for the continuous location-allocation problem", *International Journal of Operations Research*, 5 (2008) 1–12.

[5] J. Brimberg, P. Hansen, N. Mladenović, and E.D. Taillard, "Improvement and comparison of heuristics for solving the uncapacitated multisource Weber problem", *Operations Research*, 48(3) (2000) 444–460.

[6] J. Brimberg and J. Hodgson, "*Heuristics for Location Models*", in *Foundations of Location Analysis* (eds.)H.A. Eiselt, Vladimir Marianov, Springer, Berlin, 2011, 335-355.

[7] L. Cooper. "Location-allocation problem", *Operations Research*, 11 (1963) 331–343.

[8] L. Cooper, "Heuristics methods for location-allocation problems", *SIAM Review*, 6 (1964) 37–53.

[9] Z. Drezner, J. Brimberg, N. Mladenović, and S. Salhi, "New local searches for solving the multi–source Weber problem", *Annals of Operations Research*, 246, (2016) 181–203.

[10] S. Eilon, C.D.T. Watson-Gandy, and N. Christofides, *Distribution Management*, Hafner, New York, 1971.

[11] D. Erlenkotter, "A dual-based procedure for uncapacitated facility location", *Operations Research*, 26 (1978) 992–1009.

[12] P. Hansen, N. Mladenović, and E. Taillard, "Heuristic solution of the multisource weber problem as a p-median problem", *Operations Research Letters*, 22 (1998) 55–62.

[13] R. F. Love, J. G. Morris, and G. O. Wesolowsky, *"Facilities Location: Models and Methods"*, North-Holland, New York, 1988.

[14] M. Megiddo and K.J. Supowit, "On the complexity of some common geometric location problems", *SIAM Journal on Computing*, 13 (1984) 182–196.

[15] N. Mladenović, J. Brimberg, P. Hansen, and J.A. Moreno-Perez, "The p-median problem: A survey of metaheuristic approaches", *European Journal of Operational Research*, 179 (2007) 927–939.

[16] G. Reinelt, "TSLIB-a travelling salesman library" *ORSA J Computing*, 3 (1991) 376–384.

[17] S. Salhi and M.D.H. Gamal, "An algorithm based approach for the uncapacitated continuous location-allocation problem", *Annals of Operations Research*, 123 (2003) 203–222.

[18] S. Salhi and R.J. Petch, "A GA based heuristic for the vehicle routing problem with multiple trips", *Journal of Mathematical Modelling and Algorithms*, 6 (2007) 591–613.

[19] A.J. Scott, *"Combinatorial Programming, Spatial Analysis and Planning"*, Methuen, London, 1971.

[20] E. Weiszfeld, "Sur le point pour lequel la somme des distances de n points donnés est minimum", *Tohoku Mathematical Journal*, 43 (1937) 355–386.