A Constructive Method and a Guided Hybrid GRASP for the Capacitated Multi-source Weber Problem in the Presence of Fixed Cost

Martino Luis[a], Said Salhi[b] and Gábor Nagy[b]
[a]Othman Yeop Abdullah Graduate School of Business,
Universiti Utara Malaysia, 06010 UUM Sintok, Kedah Darul Aman,
Malaysia
Email: martino@uum.edu.my
[b]The Centre for Logistics & Heuristic Optimisation,
Kent Business School, University of Kent at Canterbury, Canterbury
CT2 7PE, UK
Email: {S.Salhi, G.Nagy}@kent.ac.uk
Received: 15/04/2014; Accepted: 04/11/2014

ABSTRACT

This paper presents a new variant of the capacitated multi-source Weber problem that introduces fixed costs for opening facilities. Three types of fixed costs are considered and experimented upon. A guided constructive heuristic scheme based on the concept of restricted regions and a greedy randomized adaptive search procedure (GRASP) are proposed. The four known data sets in the literature, typically used for the uncapacitated multi-source Weber problem, are adapted by adding capacities and facility fixed costs and used as a platform to assess the performance of our proposed approaches. Computational results are provided and some research avenues highlighted.

Keywords: Continuous location, capacitated location, heuristics, GRASP, facility fixed cost.

1. INTRODUCTION

The continuous location-allocation problem (also known as the multi-source Weber problem) in the presence of capacity restrictions and fixed costs is studied. In this location-allocation problem, the number of facilities to locate

Corresponding author. Salhi Said <S.Salhi@kent.ac.uk>
This paper was presented at the EGH2013 Workshop, University of Essex, June 2013

can be either specified or unknown, and each facility has a capacity limit. Also, we need to serve a set of n customers at known locations with their respective demands, by finding the allocation of these customers to these facilities without violating the capacity of any of the facilities. The objective is to minimise the total sum of transportation and fixed costs associated with the opening of the new facilities.

The classical (*i.e.* uncapacitated) and the capacitated multi-source Weber problems both assume that the number of facilities to be located is known in advance, whereas in practice, determining the number of facilities is one of the main factors that is usually addressed at a strategic level as the establishment (opening) of a facility requires a massive investment. In this paper we investigate this capacitated multi-source Weber problem in the presence of fixed costs which we refer to for short as CMSWPF. To the best of our knowledge, this problem is new but also useful for practical applications.

The paper is structured as follows. In the next section, a brief literature review is given. Then, we present a mathematical model for the CMSWPF, followed by a discussion of the various types of fixed cost functions. Sections 5 and 6 presents our solution methods, namely the region rejection based heuristic and the GRASP heuristic; this is followed by a section on computational experiments. Finally, the last section summarizes our conclusions and highlights some research avenues that we believe to be worthwhile investigating in the future.

2. LITERATURE REVIEW

Our review is in two parts. First, we look at the capacitated multi-source Weber problem (CMSWP) without fixed costs and then at the uncapacitated multi-source Weber problem (MSWP) with fixed costs. We do not look at the MSWP without fixed costs, but refer the reader to Brimberg *et al.* (2008). Neither do we look at discrete location problems, for there adding fixed costs is a relatively simpler exercise – these only need to be determined for a finite set of candidate locations while in the continuous case a fixed cost function must be defined for an infinite domain.

Cooper (1972) observes that once the facilities are located, the CMSWP reduces to the classical Transportation Problem (TP). Exact and heuristic methods are presented. The latter is based on alternately solving the location-allocation problem and the TP until there is no epsilon (ε) improvement found in the total cost. This technique, known as ATL for short, is enhanced further by Cooper (1975) and is then adapted to solve the fixed charge problem in Cooper

(1976). Sherali and Shetty (1977) solve the rectilinear distance CMSWP using a convergent cutting plane algorithm originally derived from a bilinear programming problem by substituting decision variables by the difference of two non-negative variables. The problem was revisited 16 years later by Sherali and Tuncbilek (1992) who put forward a branch and bound algorithm to compute strong upper bounds. Sherali *et al*. (1994) study the rectilinear distance CMSWP as a mixed integer nonlinear programming formulation using a reformulation-linearization technique. Gong *et al*. (1997) propose a hybrid evolutionary method based on a genetic algorithm and Lagrange relaxation. Sherali *et al*. (2002) develop a branch and bound approach based on a partitioning of the allocation space for the capacitated Euclidean and ℓ_p distance MSWP. Aras *et al*. (2007a) propose three heuristic methods to tackle the CMSWP with Euclidean, squared Euclidean, and ℓ_p distances. Aras *et al*. (2007b) tackle the CMSWP with rectilinear, Euclidean, squared Euclidean, and ℓ_p distances by implementing simulated annealing, threshold accepting, and genetic algorithms. Zainuddin and Salhi (2007) deal with the Euclidean CMSWP by proposing a perturbation-based heuristic. Aras *et al*. (2008) adopt their earlier approaches to solve the CMSWP with rectilinear distance. Luis *et al*. (2009) solve the CMSWP by introducing the concept of region-rejection into their constructive heuristic. Mohammadi *et al*. (2010) design two genetic algorithms (GAs) one for the location problem and the other for the allocation of customers to those facilities. Very recently, Luis *et al*. (2011) present a novel guided reactive GRASP that combines adaptive learning with the concept of region-rejection.

The literature on the MSWP with fixed costs is very scarce; to our knowledge, it consists of just two papers. Brimberg and Salhi (2005) assume that fixed costs are zone-dependent, where zones are non-overlapping convex polygons. The paper mainly deals with locating a single facility. It is shown that the optimal solution falls either inside a zone with the cheapest cost, or on a zone edge. An exact algorithm, based on the above observation, is presented. For multiple facilities, discretizing the problem is suggested. Brimberg *et al*. (2004) use constant fixed costs and propose a multi-phase heuristic. Firstly, the corresponding discrete location problem, namely the simple plant location problem, is solved which gives a good approximation of the number of facilities needed. Then, Cooper's location-allocation method is used to relocate the facilities. Finally, a local search is carried out to see whether having a few more or a few less facilities may give a better solution, as the fixed cost of adding/removing a few facilities is balanced by the decreased/increased transportation cost.

3. MATHEMATICAL FORMULATION

Let

$\overline{M}$: An upper bound on the number of facilities to be located;

n : the number of fixed demand points (or customer points);

$z_i = 1$ if the i^{th} facility is opened, 0 otherwise, $i = 1, \ldots \overline{M}$;

x_{ij} : quantity assigned from facility i to customer j, $i = 1, \ldots \overline{M}$; $j = 1, \ldots, n$;

$d(X, Y)$: the Euclidean distance between locations X and Y, $X, Y \in \Re^2$;

$X_i = \left(X_i^1, X_i^2\right)$: coordinates of facility i where $X_i \in \Re^2$;

$a_j = \left(a_j^1, a_j^2\right)$: location of customer j where $a_j \in \Re^2$, ($j = 1, \ldots, n$);

$f(X)$: the fixed cost to open a facility at $X \in \Re^2$;

w_j : demand or weight of customer j ($j = 1, \ldots, n$);

b : a fixed capacity of a facility, where $b \in$ č;

This problem can be formulated as follows:

$$Minimise \ \sum_{i=1}^{\overline{M}} \sum_{j=1}^{n} x_{ij} d\left(X_i, a_j\right) + \sum_{i=1}^{\overline{M}} f\left(X_i\right) z_i \tag{1}$$

Subject to

$$\sum_{i=1}^{\overline{M}} x_{ij} = w_j, \ j = 1, \ldots, n \tag{2}$$

$$\sum_{j=1}^{n} x_{ij} \le b z_i, \ i = 1, \ldots \overline{M} \tag{3}$$

$$x_{ij} \ge 0, \ i = 1, \ldots \overline{M}; \ j = 1, \ldots, n \tag{4}$$

$$z_i = \{0, 1\}, \ i = 1, \ldots \overline{M} \tag{5}$$

The objective function (1) is to minimize the sum of the transportation costs and the fixed costs. Constraints (2) ensure the total demand of each customer is satisfied. Constraints (3) ensure capacity limits are not exceeded. Constraints (4) are the nonnegativity constraints and constraints (5) are the binary constraints. Note that $\sum_{i=1}^{\overline{M}} z_i$ represents the number of open facilities.

We note that if the fixed cost $f_i = 0$ for all sites, the problem becomes the capacitated MSWP. Also, if the value of b is large enough (say $b \ge \sum_{j=1}^{n} w_j$) the problem reduces to the classical MSWP.

4. THE CONSTRUCTION OF THE FACILITY FIXED COSTS

The set up or opening cost of a facility may be dependent on geographical (location) areas. In other words, some areas may have cheaper costs of establishing a facility whereas others may have exorbitant costs. For instance, government applies different taxes for urban, suburban, and remote regions or regional restrictions as some lands are under government protection such as forests, lakes, rivers. In this work, we investigate three types of facility fixed cost models that we consider to be practical though others could also be explored. In the following, we consider planar location with Euclidean distances.

4.1. A Constant Fixed Cost Function

The simplest model considers that the opening cost of a facility is a constant value irrespective of its location and its size (*i.e.* $f(X) = f$, $\forall X \in \Re^2$). For instance, if f is set to 0 then the problem reduces to the CMSWP. For a given value of M, this fixed cost may be removed from the objective function when solving the problem and then added at the end to the total cost as a constant Mf. This idea is simple and can be used for evaluating different values of M. This flexibility in assessing several values of M is useful in practice, given it provides several scenarios to the decision makers. One may try out different values of M, and then choose the solution with the smallest overall total cost.

4.2. A Zone-Based Fixed Cost Function

In this model, following Brimberg and Salhi (2005), we divide the plane into "mutually exclusive zones", and let the fixed cost be constant within each zone. This could be done in different ways; we chose to tessellate the plane into Voronoi regions (see Preparata and Shamos (1985) and Figure 1), with the customer locations as seed points. In effect, this relates to cost of locating a facility to its nearest customer. (We note that, following Brimberg and Salhi (2005), we define the fixed cost on the Voronoi edges to be the smaller of the two costs of the adjoining regions.) Such a model is applicable if the Voronoi regions represent different administrative regions, with different taxes or labour costs. (The customer can be thought of as the "capital city" of the region.) We note that our solution algorithm will not need to explicitly construct the Voronoi regions – a far from trivial task – as it merely calculates the fixed cost for a finite number of candidate locations during the search.

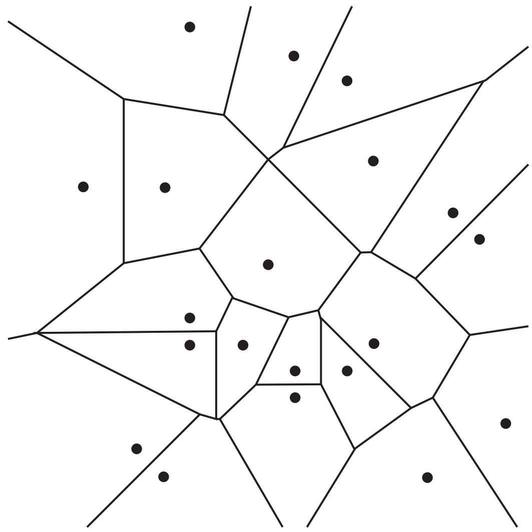

Figure 1. An illustration of Voronoi regions for 20 points in the plane.

4.3. A Continuous Fixed Cost Function

A disadvantage of the previous model is that the fixed cost function is not continuous; a small change in location (if it takes us over a Voronoi edge) may yield a large change in the establishment cost. Here, we propose a continuous function that can be seen as an extension of the previous model. First, we tessellate the plane into second-order Voronoi polygons (see Preparata and Shamos (1985)), with the customer locations as seed points. A second-order Voronoi polygon for seed points i and j consists of all points for which the two nearest seed points are i and j, see Figure 2 for an illustration. Within each region we define the fixed cost function based on the two nearest customers i and j. First, we associate a fixed cost f_i with every customer location (just like in the previous model). Then, we define the fixed cost for any point X in the region of i and j as:

$$f(X) = \frac{f_i\, d\left(X,a_j\right) + f_j\, d\left(X,a_i\right)}{d\left(X,a_j\right) + d\left(X,a_i\right)} \tag{6}$$

We can show that this is a continuous function. Clearly, within each second-order Voronoi region it is continuous; we just need to show continuity on the edges. Crossing a second-order Voronoi edge from the polygon of (i, j) to that of (i, k), $d(X, a_j)$ changes to $d(X, a_k)$ – but at this point the two are equal to each other, as the edge consists of points equidistant from j and k.

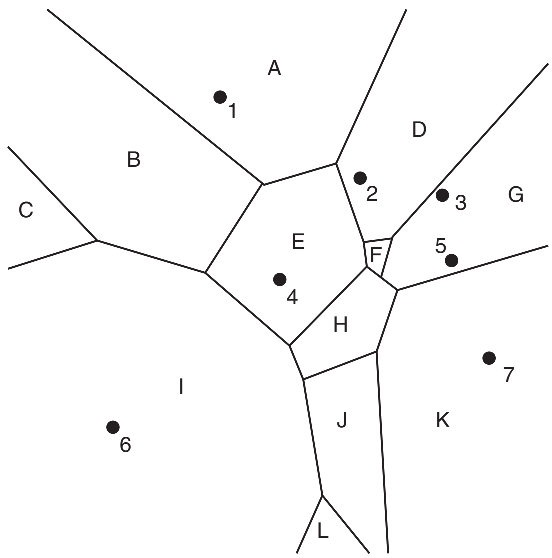

Figure 2. An illustration of second-order Voronoi regions for 7 points in the plane. (Numbers denote fixed points, letters denote zones; their correspondence is as follows: A:1,2; B:1,4; C:1,6; D:2,3; E:2,4; F:2,5; G:3,5; H:4,5; I:4,6; J:4,7; K:5,7; L:6,7.)

Luis (2008) also explored a model where for some zones of the plane the fixed cost is constant and for others it is continuously changing following the above function. However as the definition of the zones is somewhat awkward and the results were the same, we do not discuss that model here.

5. A CONSTRUCTIVE HEURISTIC FOR THE CMSWPF

First, we describe the earlier heuristic of Luis *et al*. (2009) that will form the basis of our CMSWPF algorithm. Then, we sketch a naïve way of solving the CMSWPF, followed by our algorithm proper. Some comments on details of the heuristic wrap up the section.

5.1. A Region-Rejection Heuristic for the CMSWP (without fixed costs)

Luis *et al*. (2009) put forward a scheme known as region rejection heuristic (RR) to solve the CMSWP. First, a customer site is randomly selected to become a facility location. Then, an area around this location is declared a "restricted region" within which no facilities may be located – this is to ensure facilities are not located too close to each other. Then, another customer is chosen at random from the set of customers not covered by a restricted region, and another restricted region is constructed around it. The process is repeated until the required number of facilities are located. The authors tested different shapes and sizes for the restricted regions. Best results were found when the shape of the region is a circle and its radius is dynamically adjusted after each iteration; from now on, only this version will be used.

Having found M facility locations, the next phase of the method is applying Cooper's method to improve on this solution. The Alternating Transportation-Location (ATL) method of Cooper (1972) takes a set of M open facilities as input. Then, a Transportation Problem (TP) is solved to find the allocation of customers to these facilities. We note that as the total capacity of the facilities Mb may be larger than the total customer demand $\sum_{j=1}^{n} w_j$, a dummy customer with a 0 transportation cost will be used; this dummy customer will only contribute to the search at the transportation problem phase and will not be considered at either the location or the allocation phases. For each cluster of customers, the location of its facility is found using the Weiszfeld algorithm (Weiszfeld and Plastria, 2009). We note that a customer may be allocated to more than one facility as its demand may be split. This now gives us a new set of facilities, for which the allocation is found again. We repeat the location and allocation phases until no improvement is found. In practice, the stopping criterion can be two successive non-improving iterations or an improvement that is below some very small threshold. In this paper, we always use the stopping criterion of a threshold of 0.0001. More details of Cooper's method can be found in Luis *et al*. (2009).

5.2. The Basic Idea of Extending the Method for CMSWP with Fixed Costs

A simple way of solving the CMSWPF would be to try out all values of M starting from 1 and solve the corresponding CMSWP, stopping when the total cost starts rising as the total cost is a unimodular function of M. However, there are three problems with this approach.

1. Small values of M may be infeasible. This can easily be corrected by starting the search with a lower bound, namely $LB = \left\lceil \left(\sum_{j=1}^{n} w_j \right) \Big/ b \right\rceil$.
2. Due to a heuristic rather than exact method being used, we may find a "false minimum", situations where the heuristic solution for some value of M is better than for M–1 and $M + 1$, but M is not in fact a minimum. We mitigate this in two different ways. Firstly, we stop our search only if in two successive searches the total costs go up.

Secondly, our heuristics are designed to operate in a multi-start fashion; it is unlikely that all runs would be trapped by false minima.

3. The procedure would be computationally expensive, requiring the CMSWP to be solved several times. Hence, once we find the initial solution for $M = LB$, we construct subsequent solutions by adding one facility at a time to existing solutions using an efficient implementation of the ADD heuristic originally proposed by Kuehn and Hamburger (1963). This is much faster than restarting some constructive algorithm from scratch for every value of M.

5.3. An Overview of the Proposed Constructive CMSWPF Algorithm

Our Region-Rejection algorithm is given here in the form of a pseudo-code.

Step 1. Set $M = LB$.
Find the solution to the CMSWP (no fixed costs), using the method of Luis *et al*. (2009). Add fixed costs to calculate the cost of the CMSWPF with M facilities, $C(M)$.

Step 2. Let F_j be the location of the nearest facility to customer j.
Randomly choose a subset S of fixed points not yet used as candidate locations and outside the restricted regions.

Step 3. For each $i \in S$, evaluate the function $\Delta_i = \sum_{j=1}^{n} \max[d(a_j, F_j) - d(a_j, a_i), 0]$.
Add $i = \text{argmax}_{i \in S}(\Delta_i)$ as a new facility and let $M = M + 1$.

Step 4. Adjust the radii of the restricted regions and add a restricted region around the new facility.
Use Cooper's method to solve the CMSWP, with the above M locations as input.
Add fixed costs to calculate the true total cost of the CMSWPF with M facilities, $C(M)$.

Step 5. If $M < LB + 2$, return to Step 2.
If $C(M-1) < C(M-2)$ or $C(M) < C(M-1)$, return to Step 2.
Let $M^* = M-2$ and record $C(M^*)$ as the cost of best solution found.

Step 6. Repeat Steps 1 to 5 K times and report the best overall solution found.

5.4. Further Details of the Algorithm

Step 1: We note that another constructive algorithm could have been used here to solve the CMSWP, but then we would need to create the restricted regions separately at the end of this step.

Step 2: We decided to set the cardinality of the set S to max{20, min(3*M*, 50)}. This, and all subsequent settings are based on, and found appropriate by, the experimentation of Luis (2008). We could have chosen to consider all remaining fixed points that do not fall into a restricted region. However, this would have slowed down the algorithm.

Step 3: The function Δ_i is an approximation for the saving in transportation costs if fixed point *i* is added. (In fact, it is the saving value of the corresponding uncapacitated discrete location problem.) To find the value of this saving properly, we would need to apply Cooper's method, which can be time-consuming. We only solve the TP once for each value of *M*.

Step 4: The adjustment scheme for region radii is done exactly the same way as in Luis *et al*. (2009).

Step 5: This expresses our stopping criterion: we stop after two (rather than one) non-improving iterations, to avoid being trapped in false minima.

Step 6: We chose to repeat the whole algorithm *K* times in order to find better solutions. A different multistart heuristic could have been created by repeating only Step 1 *K* times and use the best solution found as input to Step 2. This implementation would be much quicker, however it was observed in previous studies that there is a lack of correlation between the quality of $C(LB)$ and $C(M^*)$. In our experiments, we have set $K = \max\left(100, 5LB, \sqrt[3]{n}LB\right)$.

6. A GUIDED HYBRID GRASP FOR THE CMSWPF

We introduce a GRASP method that nonetheless retains the concept of region-rejection. First, we present GRASP in general and the method of Luis *et al*. (2011) in particular. Then, we explain the details of our Hybrid GRASP method for the CMSWPF.

6.1. A Guided Hybrid GRASP for the CMSWP (without fixed costs)

The greedy randomized adaptive search procedure (GRASP for short) is a multi-start heuristic technique consisting of constructive and a local search phases to tackle hard combinatorial optimization problems (see Resende and Ribeiro (2010)). In the first phase of GRASP, known as the construction phase, a feasible initial solution is built one at a time. The construction of these feasible solutions is based on the creation of a restricted candidate list (RCL) made up of good attributes including those of the best solution. The choice of this list is a crucial issue in GRASP and can be based on two aspects. It may be size-based, where the best |RCL| elements are selected. Alternatively, it may be attribute-based: those candidates whose solution quality is better than a certain threshold are chosen. From this set RCL, one element is chosen one at a time either randomly or following a certain selection rule (pseudo-randomly) until the full solution is completed. The second phase or the improvement phase is a standard local search applied to explore the neighbourhood of the constructed solution in order to find a better solution. The two phases are reiterated several times either independently or using some learning scheme and the best overall local optimum is then selected as the final result.

Luis *et al*. (2011) propose a hybrid heuristic that combines guided reactive GRASP with region-rejection when constructing the restricted candidate list (RCL). Only fixed points that lie without the restricted regions can be selected for inclusion in the RCL. Both size-based and attribute-based aspects are taken into account when creating RCL. Four different types of GRASP are tried out. The variant that combined region-rejection with dynamically adjusted radius and a learning process within GRASP, where the learning process governs the parameter α, was found to be the best. Here, we use this best variant.

6.2. A Guided Hybrid GRASP for the CMSWP (with fixed costs)

Our GRASP algorithm has a similar structure to our Region-Rejection algorithm, retaining the ADD methodology and the concept of restricted regions. The differences are only in Steps 1 and 3. In Step 1, we use the GRASP method of Luis *et al*. (2011) instead of the constructive method of Luis *et al*. (2009) to find an initial solution for $M = LB$.

In Step 3, instead of adding the "best" location from S (the randomly chosen set of non-restricted customer locations), we add a "good" location pseudo-randomly as follows. First, we create the Restricted Candidate List (RCL), where RCL$\subseteq$S. This will contain all elements i of S for which $\Delta_i \le \min_{j \in S}\Delta_j + \alpha(\max_{j \in S}\Delta_j - \min_{j \in S}\Delta_j)$ and also the best (with respect to Δ_i) r elements of S. The size of the candidate list depends on the parameters α $(0 \le \alpha \le 1)$ and r $(1 \le r \le |S|)$. We allow α to vary dynamically as a linear function with a learning process and let $r = \max(5, \lceil M/2 \rceil)$. Finally, we choose pseudo-randomly an element of RCL to be the next location – we let customer i ($i \in$RCL) to become our next facility location with probability $p_i = \dfrac{\Delta_i}{\sum_{j \in RCL} \Delta_j}$.

7. COMPUTATIONAL EXPERIMENTS

The proposed methods were coded in C++, compiled with Salford FTN95 compiler, and run on a PC with an Intel 1.5 GHz Pentium M processor and 1.3 GB RAM. We tested our approaches on three classes of instances. To evaluate the performance of our proposed methods, we adapted the four well known data sets from Brimberg *et al*. (2000) with the addition of facility fixed costs and capacity. These data sets vary from 50 to 1060 customers and use the Euclidean distance measure.

As the above data sets do not have a capacity of the facilities, we created the capacity values ourselves; these are given in Table 1. Every facility is set to have the same capacity irrespective of location. We tested our proposed methods using three types of fixed costs. Type I is a constant-based fixed cost where the fixed cost is set to be a constant number. In Type II and Type III, the fixed costs were generated from discrete uniform distributions in the range of [2, 8] for 50 fixed points, [50, 400] for 287 fixed points, [1000, 10000] for 654 fixed points, and [10000, 100000] for 1060 fixed points, see Luis (2008) for more details. For each data set, we present three instances for the 50 fixed points and five instances for the other data sets. The instances are available from the authors upon request.

Table 1. Results for the Case of a Constant Fixed Cost Function.

Number of Custo-mers (n)	Capacity (b)	Fixed Cost (f_i)	Lower bound (LB)	Region-Rejection				GRASP			
				Total Cost using LB	M^*	Total Cost using M^*	CPU (secs)	Total Cost using LB	M^*	Total Cost using M^*	CPU (secs)
50	10	5	5	101.52	7	**89.47**	3	101.52	7	**89.47**	8
	5	3	10	78.17	12	**71.76**	5	78.17	12	**71.76**	29
	3	2	17	59.81	19	**59.68**	7	59.81	19	**59.68**	93
287	1264	350	5	12235.46	10	10285.31	115	12235.38	10	**10204.84**	179
	632	250	10	10935.16	14	9107.84	154	10935.00	14	**9061.22**	383
	422	200	15	10014.11	18	8738.95	180	9995.19	21	**8520.12**	1517
	316	100	20	7745.38	27	6377.04	587	7745.36	28	**6131.37**	4291
	211	50	30	5688.03	38	4676.11	1333	5688.03	38	**4580.47**	11185
654	131	10000	5	371969.69	11	210132.86	300	371969.69	11	**210132.86**	563
	66	8000	10	244716.80	15	201612.79	485	244716.80	15	**201552.52**	1381
	33	5000	20	207357.31	24	181535.31	1050	207357.31	25	**180527.07**	7264
	22	3000	30	168830.49	34	**151644.56**	2070	168830.49	34	**151644.56**	22204
	17	1000	40	91359.35	45	**83643.39**	4790	91359.35	45	**83643.39**	43276
1060	212	100000	5	2371544.64	8	**2224272.48**	355	2371544.64	8	**2224272.48**	1134
	106	80000	10	2088424.42	11	**2067868.60**	583	2088424.42	11	**2067868.60**	1105
	53	50000	20	1856628.50	20	**1856628.50**	1024	1856628.50	20	**1856628.50**	3526
	36	30000	30	1563406.53	30	**1563406.53**	2163	1563406.53	30	**1563406.53**	10345
	27	10000	40	1362247.52	40	1362247.52	3638	1362086.12	40	**1362086.12**	22661

Tables 1 to 3 summarize the obtained results using Region-Rejection and GRASP for the three types of fixed cost functions namely constant, zone-based and continuous. **Bold** numbers represent the minimum costs found.

Based on these results, GRASP gives better or equal solutions for all the instances when compared to region-rejection results, which is in line with the findings of Luis *et al*. (2011) for the case of no fixed costs. In the case of the constant fixed cost, both methods produce similar solutions, see Table 1. For example, when n = 50, all the instances show the same total costs for both methods. This is because the fixed cost is constant, therefore it can be removed from the objective function. In fact, both methods produce the same final facility configurations. For the case of the zone-based and the continuous fixed cost, Region-Rejection is found to be inferior when compared to the results found by GRASP in most instances. For instance, in the zone-based case (see Table 2) with

Table 2. Results for the Case of the Zone-based Fixed Cost Function.

Number of Customers (n)	Capacity (b)	Lower bound (LB)	Region-Rejection				GRASP			
			Total Cost using LB	M^*	Total Cost using M^*	CPU (secs)	Total Cost using LB	M^*	Total Cost using M^*	CPU (secs)
50	10	5	93.71	9	82.05	4	93.71	9	**81.92**	16
	5	10	90.19	12	83.76	5	91.10	12	**83.00**	29
	3	17	99.45	18	98.94	5	99.32	19	**98.50**	94
287	1264	5	11735.46	12	8413.99	189	11735.46	12	**8131.69**	281
	632	10	10196.04	13	8373.43	132	10195.80	16	**8220.61**	650
	422	15	9392.15	18	8692.08	190	9162.67	20	**8396.55**	1226
	316	20	9315.88	23	9161.27	292	9283.84	23	**8990.75**	1508
	211	30	10859.66	31	10679.37	295	10771.38	31	**10643.24**	2355
654	131	5	347987.58	12	153443.79	491	346971.99	14	**147199.56**	1319
	66	10	220716.80	15	148612.37	516	220716.80	15	**139552.52**	1370
	33	20	192381.16	23	176031.92	852	192381.16	22	**173048.47**	3382
	22	30	212859.47	32	199178.43	1269	212859.47	32	**197831.42**	9934
	17	40	215268.27	40	215268.27	1277	209655.23	40	**209655.23**	10328
1060	212	5	2221544.65	9	1635855.70	489	2221544.65	11	**1558082.23**	2858
	106	10	1675180.02	13	1611733.66	1205	1675106.66	12	**1571835.08**	1767
	53	20	1690831.25	21	1662551.25	1696	1569320.06	20	**1569320.06**	3553
	36	30	1912378.86	32	1821326.68	5280	1897549.25	32	**1765815.94**	23390
	27	40	2041853.24	41	**2012700.96**	6039	2041853.24	41	**2012700.96**	35732

Table 3. Results for the Case of a Continuous Fixed Cost Function.

Number of Customers (n)	Capacity (b)	Lower bound (LB)	Region-Rejection				GRASP			
			Total Cost using LB	M^*	Total Cost using M^*	CPU (secs)	Total Cost using LB	M^*	Total Cost using M^*	CPU (secs)
50	10	5	95.05	8	**81.09**	4	95.05	8	**81.09**	13
	5	10	91.68	12	81.77	4	91.68	13	**81.52**	32
	3	17	97.44	17	97.44	5	97.33	17	**97.33**	62
287	1264	5	11738.38	13	8439.36	201	11738.38	12	**8180.17**	294
	632	10	10310.82	14	8591.78	155	10306.11	13	**8516.38**	290
	422	15	9542.66	16	8972.53	104	9307.46	18	**8571.86**	752
	316	20	9373.61	23	9203.13	310	9348.14	26	**8907.67**	2976
	211	30	10911.57	32	10699.59	422	10801.66	32	**10607.88**	3318
654	131	5	347582.96	12	154280.76	499	346332.01	14	**149929.33**	1341
	66	10	221651.96	16	155544.30	592	221651.96	15	**149183.10**	1388
	33	20	209650.61	22	185436.68	606	209650.64	23	**182340.27**	4526
	22	30	224259.94	31	218516.54	866	224657.43	34	**214481.92**	16466
	17	40	240465.69	40	240465.69	1349	239132.80	40	**239132.80**	10401
1060	212	5	2154044.58	11	1600478.55	647	2154044.58	11	**1569610.77**	2866
	106	10	1673323.58	12	1613478.77	868	1673323.58	12	**1606542.46**	1798
	53	20	1683940.67	20	1683940.67	521	1672401.91	20	**1672401.91**	3591
	36	30	1964836.78	30	**1964836.78**	2162	1964836.78	30	**1964836.78**	10417
	27	40	2184908.54	40	**2184908.54**	3646	2184908.54	40	**2184908.54**	22787

n = 287 and b = 1264, GRASP finds M = 12 and a total cost of 8131.69 whereas Region-Rejection yields a cost of 8413.99, yielding a deviation of about 3.5%. In the continuous case (see Table 3) with n = 654 and b = 66, GRASP produces a total cost of 149183.10 with M = 15 but Region-Rejection gives a total cost of 155544.30 with M = 16, a deviation of approximately 4.3%. It can be observed that GRASP outperforms Region-Rejection in almost of instances. However, it is worth noticing that Region-Rejection is relatively faster than GRASP.

We can observe that using two, rather than one, increasing moves has turned out to be a worthwhile modification to avoid false minima. For example, let us look at the case of the zone-based fixed cost with n = 654 customers and b = 131, this is given in Table 4 and is also illustrated in Figure 3. We can see that had we stopped at M = 8 because $C(8) < C(9)$, we would have paid a penalty of 34%. This behaviour may obviously not have been presented if an

Table 4. An illustration of a "false minimum": zone-based fixed cost for n = 654 and b = 131.

M	**Total Cost**
5	346971.99
6	248542.66
7	221940.87
8	**197533.39**
9	198977.91
10	162695.53
11	159132.86
12	153499.32
13	149598.38
14	**147199.56**
15	148057.77
16	149269.27

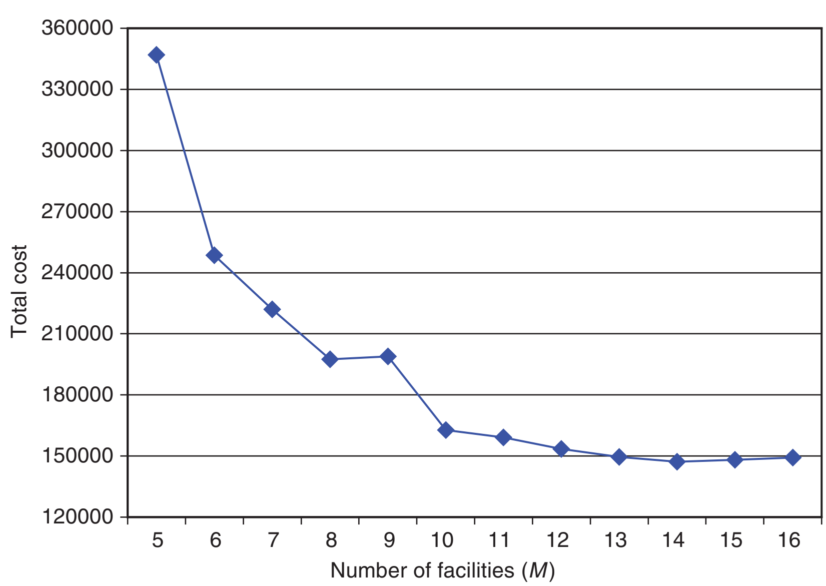

Figure 3. A graphical illustration of a "false minimum".

exact method was used instead. One way to reduce the risk of such behaviour would be to have three increasing moves as our stopping criterion. Another would be to re-solve the problems in the neighbourhood of a suspected minimum using a more powerful local search or an exact method.

8. CONCLUSIONS AND FUTURE RESEARCH

This paper studies a new variant of the classical multi-source Weber problem with the presences of capacity restrictions and fixed costs for opening facilities. Two approaches using the concept of region-rejection heuristic and a guided GRASP are adopted to tackle this problem. A scheme to reduce the risk of "false minima" is also put forward and shown to be useful. Three types of fixed costs are investigated: a constant fixed cost, a Voronoi zone–based and a continuous function based on second-order Voronoi polygons. The proposed schemes were evaluated using some adaptations of the well-known instances taken from the MSWP literature. Comparative results were produced using our proposed heuristics which can then be used for future benchmarking.

Research avenues that we consider to be worth investigated in the future include other forms of fixed costs. We could consider a fixed cost function that contain terms inversely proportional to its distance to the customers. (It is usually cheaper to locate facilities further away for inhabited areas; such model may also be useful in obnoxious facility location.) Alternatively, the cost of a facility could be dependent on its throughput. In particular, we could consider facilities of different sizes; the larger the facility's capacity the larger its establishment cost. It may also be worthwhile to explore other powerful meta-heuristics, based on VNS or TS, or a hybrid of the two, as these proved to be effective in tackling hard combinatorial and global optimisation problems.

REFERENCES

[1] Aras, N., Altınel, I.K. and Orbay, M., 2007a. New Heuristic Methods for the Capacitated Multi-Facility Weber Problem. *Naval Research Logistics* 54, 21–32.

[2] Aras, N., Yumusak, S. and Altınel, I. K., 2007b. Solving the Capacitated Multi-Facility Weber Problem by Simulated Annealing, Threshold Accepting and Genetic Algorithms. In Doerner, K.F., Gendreau, M., Greistorfer, P., Gutjahr, W. J., Hartl, R.F. and Reimann, M., editors. *Metaheuristics: Progress in Complex Systems Optimization*, Springer, 91–112.

[3] Aras, N., Orbay, M. and Altinel, I.K., 2008. Efficient Heuristics for the Rectilinear Distance Capacitated Multi-Facility Weber Problem. *Journal of the Operational Research Society* 59, 64–79.

[4] Brimberg, J., Hansen, P., Mladenović, N. and Taillard, E. D., 2000. Improvements and Comparison of Heuristics for Solving the Uncapacitated Multisource Weber Problem. *Operations Research* 48, 444–460.

[5] Brimberg, J., Mladenović, N. and Salhi, S., 2004. The Multi-Source Weber Problem with Constant Opening Cost. *Journal of the Operational Research Society* 55, 640–646.

[6] Brimberg, J. and Salhi, S., 2005. A Continuous Location-Allocation Problem with Zone-Dependent Fixed Cost. *Annals of Operations Research* 136, 99–115.

[7] Brimberg, J., Hansen, P., Mladenović, N. and Salhi, S., 2008. A Survey of Solution Methods for the Continuous Location-Allocation Problem. *International Journal of Operations Research* 5, 1–12.

[8] Cooper, L., 1964. Heuristic Methods for Location-Allocation Problems. *SIAM Review* 6, 37–53.

[9] Cooper, L., 1972. The Transportation-Location Problem. *Operations Research* 20, 94–108.

[10] Cooper, L., 1975. The Fixed Charge Problem – I: A New Heuristic Method. *Computers and Mathematics with Applications* 1, 89–95.

[11] Cooper, L. 1976. An Efficient Heuristic Algorithm for the Transportation-Location Problem. *Journal of Regional Science* 16, 309–3015.

[12] Gong, D., Gen, M., Yamazaki, G. and Xu, W., 1997. Hybrid Evolutionary Method for Capacitated Location-Allocation Problem. *Computers and Industrial Engineering* 33, 577–580.

[13] Kuehn, A.A. and Hamburger, M.J., 1963. A Heuristic Program for Locating Warehouses. *Management Science* 9, 643–666.

[14] Luis, M., 2008. Meta-heuristics for the Capacitated Multi-Source Weber Problem, *Ph.D. Thesis*, University of Kent, UK.

[15] Luis, M., Salhi, S. and Nagy, G., 2009. Region-Rejection Based Heuristics for the Capacitated Multi-Source Weber Problem, *Computers and Operations Research* 36, 2007–2017.

[16] Luis, M., Salhi, S. and Nagy, G., 2011. A Guided Reactive GRASP for the Capacitated Multi-Source Weber Problem, *Computers and Operations Research* 38, 1014–1024.

[17] Mohammadi, N., Malek, M.R., and Alesheikh, A. A., 2010. A New GA Based Solution for Capacitated Multi Source Weber Problem, *International Journal of Computational Intelligence Systems* 3, 514–521.

[18] Preparata, F. and Shamos, M. I., 1985. *Computational Geometry*. Springer, New York.

[19] Resende, M.G.C. and Ribeiro, C.G., 2010. Greedy Randomized Adaptive Search Procedures: Advances, Hybridizations and Applications. In Gendreau, M. and Potvin, J-Y., Editors. *Handbook of Metaheuristics* (2nd Edition), Springer, NY, p. 283–319.

[20] Rosing, K.E., 1992. The Optimal Location of Steam Generators in Large Heavy Oil Fields. *American Journal of Mathematical and Management Sciences* 12, 19–42.

[21] Sherali, H.D. and Shetty, C. M., 1977. The Rectilinear Distance Location-Allocation Problem. *AIIE Transaction* 9, 136–143.

[22] Sherali, H.D. and Tuncbilek, C.H., 1992. A Squared-Euclidean Distance Location-Allocation Problem. *Naval Research Logistics* 39, 447–469.

[23] Sherali, H.D., Ramachandran, S. and Kim, S., 1994. A Localization and Reformulation Discrete Programming Approach for the Rectilinear Distance Location-Allocation Problem. *Discrete Applied Mathematics* 49, 357–378.

[24] Sherali, H.D., Al-Loughani, I. and Subramanian, S., 2002. Global Optimization Procedures for the Capacitated Euclidean and ℓ_p Distance Multifacility Location-Allocation Problem. *Operations Research* 50, 433–448.

[25] Weiszfeld, E. and Plastria, F., 2009. On the Point for Which the Sum of the Distances to *n* Given Points is Minimum. *Annals of Operations Research* 167, 7–41.

[26] Zainuddin, Z.M. and Salhi, S., 2007. A Perturbation-Based Heuristic for the Capacitated Multisource Weber Problem. *European Journal of Operational Research* 179, 1194–1207.