J. Opl Res. Soc. Vol. 42, No. 7, pp. 579–585, 1991
Printed in Great Britain. All rights reserved
0160-5682/91 $3.50 + 0.00

A Lagrangian Heuristic for the Capacitated Plant Location Problem with Side Constraints

R. SRIDHARAN
Indian Institute of Management, Ahmedabad

In this paper we present a Lagrangian relaxation-based heuristic for solving the capacitated plant-location problem with side constraints. The side constraints are upper-bound constraints on disjoint subsets of the (0–1) variables. Computational results are reported for some problems, having been obtained both on a mainframe computer and on a personal computer.

Key words: location, allocation, heuristics

INTRODUCTION

The location of plants, such as factories or warehouses, is an important strategic decision for organizations. Transportation costs, which often form a major portion of the cost of goods supplied, are a function of the location of plants. The fixed costs of opening and operating a plant may also vary from one location to another. Such problems have been widely studied in the literature under the names of plant, warehouse, or facility location problems. When each potential location has a capacity, that is, an upper bound on the demand it can service, the problem is known as the capacitated plant location problem (CPLP). Cornuejols *et al.*,[1] Magnanti and Wong,[2] Wong,[3] Salkin[4] and Francis and Goldstein[5] provide excellent bibliographies on CPLP.

In this paper we present a Lagrangian relaxation approach for solving the capacitated plant location problem with side constraints. This extension can be used to solve the following variation of CPLP. Suppose we have a number of choices regarding the size of the plant that can be considered at a given location. Then we can consider this as a CPLP with a number of 'potential locations' computed as follows. Each choice of the size of a plant at a given location is considered as a potential location. For example, if there are three choices of plant sizes at a given location, then the number of potential locations is equal to three. But we cannot open more than one plant at a given location. Therefore, for each location we can have an upper-bound constraint that restricts to one the number of plants that can be opened there. Then the solution to this CPLP with the side constraints will give the solution to the CPLP where we make the choice of the size of the plant to be opened at any location. In general, the side constraints are upper-bound constraints on any combination of the (0–1) variables. We present a solution method for this problem based on a Lagrangian relaxation approach which is an extension of the procedure used by Christofides and Beasley[6] for CPLP. Christofides and Beasley[6] allow capacities of their plants to vary between lower and upper bounds while we consider a number of plants of different capacities at each location with side constraints to ensure that not more than one plant capacity is used at each location.

The paper is organized as follows. In the second section, a mathematical formulation of the problem is given. In the third, a Lagrangian relaxation of the problem is defined while the procedure to obtain the upper bound is described in the fourth. Then, in the fifth section a problem reduction procedure is given and in the sixth the subgradient optimization approach involved in the Lagrangian procedure is described. In the final section we provide some computational results and concluding remarks.

PROBLEM FORMULATION

The capacitated plant location problem with side constraints can be formulated as a mixed-integer programming problem as follows:

Correspondence: R. Sridharan, Indian Institute of Management, Vastrapur, Ahmedabad 380 015, India.

Let d_i be the demand of customer i,
s_j be the capacity of plant located at j,
c_{ij} be the cost of supplying all of the demand of customer i from a plant located at j,
P_L be a lower limit on the number of open plants,
P_U be an upper limit on the number of open plants,
f_j be the fixed cost associated with plant j,
x_{ij} be the fraction of demand of i supplied from plant j,
$I = \{1, \ldots, m\}$,
$J = \{1, \ldots, n\}$.

Define
$$y_j = \begin{cases} 1 & \text{if plant } j \text{ is open,} \\ 0 & \text{otherwise.} \end{cases}$$

Then,

$$Z = \min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} + \sum_{j=1}^{n} f_j y_j \tag{1}$$

s.t.

$$\sum_{j=1}^{n} x_{ij} = 1, i = 1, \ldots, m; \tag{2}$$

$$\sum_{i=1}^{m} d_i x_{ij} \leqslant s_j y_j, j = 1, \ldots, n; \tag{3}$$

$$0 \leqslant x_{ij} \leqslant y_j \leqslant 1, \text{ for every } i, j, \tag{4}$$

$$y_j = 0, 1; \tag{5}$$

$$P_L \leqslant \sum_{j \in J} y_j \leqslant P_U; \tag{6}$$

$$\sum_{j \in N_k} y_j \leqslant M_k, N_k \cap N_l = \phi; k, l = 1, \ldots, K; \tag{7}$$

where M_k is an upper bound on the number of plants that can be opened in the set $N_k \subseteq J$.

Constraint (6) is a surrogate constraint that strengthens the formulation of CPLP. This constraint improves the bounds obtained for the problem in the solution procedure. We also improve the bounds P_L and P_U during the solution procedure to provide tighter limits on the open plants.

LOWER BOUNDS

The lower bound for (1) is obtained by solving a Lagrangian relaxation of the problem. Let u_i, $i = 1, \ldots, m$ be the Lagrange multipliers associated with the demand constraints (2). Then the Lagrangian dual program is obtained as

$$Z_D = \max_u Z_D(u),$$

where

$$Z_D(u) = \min \sum_i \sum_j (c_{ij} - u_i) x_{ij} + \sum_j f_j y_j + \sum_i u_i \tag{8}$$

s.t. (3), (4), (5), (6) and (7).

For a given u, the Lagrangian dual program can be solved very easily. This problem, without the constraints (6) and (7) for a given u, breaks up into continuous knapsack problems for each j. We can then write

$$Z_D(u) = \sum_j Z_D(u; j),$$

where

$$Z_D(u; j) = \min \sum_i (c_{ij} - u_i)x_{ij} + f_j \tag{9}$$

$$\sum_i d_i x_{ij} \leqslant s_j \tag{10}$$

$$x_{ij} \geqslant 0. \tag{11}$$

Let the value of $Z_D(u; j)$ be f'_j and let x^*_{ij} be the solution to (9) to (11).

We can now include constraints (6) and (7) to obtain a (0, 1) problem on the y_j variables. This problem KP_1 is

$$Z_{KP_1} = \min \sum_j f'_j y_j + \sum_i u_i \tag{12}$$

s.t. (5), (6) and (7).

The problem KP_1 is solved as follows:

Step 1 Sort f'_j in ascending order.

Step 2 Set y_j to one for the first P_L plants in the list taking into consideration the side constraints (7).

Step 3 If $P_L = P_U$, go to step 5.

Step 4 Keep setting y_j to one with due consideration to constraints (7) until either (a) P_U plants have been opened or (b) the f'_j values have become $\geqslant 0$.

Step 5 Let the set of values y_j chosen as above be y^*_j. The y^*_j that are not set at one are set at zero.

The optimal solution to the Lagrangian dual program is y^*_j, and x^*_{ij} found in (9) to (11) and Z_{KP_1} is equal to

$$Z_{KP_1} = \sum_j f'_j y^*_j + \sum_i u_i. \tag{13}$$

A lower bound for (1) is given by the objective value Z_{KP_1}. We compute an upper bound for (1) by solving a transportation problem which uses the open plants as given by KP_1. The determination of the initial and subsequent upper bounds is described in the next section.

UPPER BOUNDS

We can find a feasible solution to (1) by solving a transportation problem with the set of open plants given by y^*_j. (However, we note here that the y^*_j returned by the Lagrangian iteration may not always lead to a feasible solution to (1).) An improved feasible solution accelerates the chances of terminating the Lagrangian heuristic. We now give the procedure for finding the initial feasible solution and the improved feasible solutions.

The initial value of P_L is found by taking the first (subject to satisfying (7)) P_L number of plants with largest capacities such that the sum of their capacities just exceeds the total demand of all the customers. Then, with this set of plants being set open, we solve the transportation problem to give a feasible solution to (1). The objective of (1) is found by adding the transportation cost found above and the fixed costs of the open plants and this gives us the initial upper bound.

The improved feasible solutions are found by solving the transportation problem with the set of open plants being given by $y^*_j = 1$ in the Lagrangian procedure. Then the upper bound is found by adding the transportation cost and the fixed costs of plants with $y^*_j = 1$, and is updated if necessary. If the set y^*_j has already been found in an earlier Lagrangian iteration, then we may save computation effort by not recomputing the same upper bound. We have a procedure that stores up to five different sets of y^*_j for which we know the feasible solution already and if the newly found y_j^* is unique (compared to the five sets), then we resort to the upper-bound procedure. The set is then updated with the newly found y_j^* eliminating the oldest from the existing list. We have found that this procedure has significantly saved the total computation time.

A typical transportation problem (TP) that is solved to identify an upper bound is described in detail below.

Let

$$J^* = \{j : y_j \text{ fixed as one in the procedure}\}.$$

Then,

$$Z_{TP} = \min \sum_{i \in I} \sum_{j \in J^*} c_{ij} x_{ij}$$

$$\sum_{j \in J^*} x_{ij} = 1, \; i = 1, \ldots, m;$$

$$\sum_{i \in I} d_i x_{ij} \leqslant s_j, \; j \in J^*;$$

$$0 \leqslant x_{ij} \leqslant 1, \; i \in I, j \in J^*.$$

We also use a reduction test as given in Christofides and Beasley[6] for identifying tighter P_L and P_U bounds.

THE SUBGRADIENT PROCEDURE

The subgradient procedure for solving problem (1) is described below.

Step 1 Solve the transportation problem with P_L open plants (as found in the section on upper bounds), and all customers. Adding their fixed costs to the transportation cost gives the initial value of the upper bound Z^{UB}. Initialize the Lagrange multipliers $u_i = \min_j c_{ij}$. Go to step 2.

Step 2 Solve the continuous knapsack $Z_D(u; j)$ for a given u for each j, and compute f'_j. Then, solve the knapsack problem KP_1 to obtain the set of plants to be opened. The objective of KP_1 gives us a lower bound Z_{LB}. Update the lower bound if necessary. Go to step 3.

Step 3 If needed, solve the transportation problem with the set of open plants identified in step 2. The transportation costs, along with the fixed costs of the open plants, give us an upper bound Z^{UB}. Update the upper bound if necessary. If $Z^{UB} = Z_{LB}$ stop; we have an optimal solution. If the iteration count is exceeded, stop. If the lower bound converges to a particular value, stop. Otherwise, go to step 4.

Step 4 Update the Lagrange multipliers u_i using the subgradient approach. If all subgradients are zero, stop. Otherwise, go to step 2.

The subgradients for u_i are computed as follows. Let x_{ij}^* and y_j^* be the optimal solutions to the Lagrangian problem. Then the subgradients $NU(i)$ for u_i are

$$NU(i) = \sum_{j \in J^*} x_{ij}^* - 1,$$

where j belongs to the set of open plants.

We then update the Lagrange multipliers as

$$u_i^{k+1} = u_i^k + t_k NU(i),$$

where

$$t_k = \lambda(Z^{UB} - Z_{LB})/\text{Norm}.$$

The norm is taken as the Euclidean norm.

We start with an intial λ value of 0.6 and halve the value every 12 iterations. The procedure is terminated after 200 iterations. The transportation problems are solved using the code developed by Srinivasan and Thompson.[7] Some computational results for this approach are provided in the next section.

RESULTS

The procedure was coded in FORTRAN 77 and run on a DEC 2060 time-sharing system at Carnegie-Mellon University. We tested this procedure on two different problem sets. The first problem set contained test problems described in Kim and Guignard[8] with some additional side constraints. The second problem set consisted of randomly generated problems. The parameters for the randomly generated problems were fixed as follows. The demands were generated from a uniform distribution in the interval [5, 35). The capacities were generated from a uniform distribution in the interval [10, 160). The fixed costs were generated using

$$f_j = U[0, 90) + U[100, 110)\sqrt{s_j},$$

where f_j is the fixed cost of plant j, s_j is the capacity of plant j and $U[a, b)$ stands for a value from the uniform distribution picked from the interval $[a, b)$. The transportation costs were computed by generating points in a unit square, computing the Euclidean distance between them and multiplying by 10. The way we compute the transportation costs ensures that we are solving geometric problems and therefore the duality gaps computed as given in this section are not subject to the well-known scaling effects. The tightness of the capacity constraints was such that

$$\sum_j s_j = 2 * \sum_i d_i.$$

In addition to those basic parameters, we also included some side constraints representing (7).

A summary of results for these problem sets will now be given. Table 1 presents the results for problem set 1 and Table 2 presents the results for the randomly generated problems which were solved on the DEC 2060 time-sharing system at Carnegie-Mellon University.

TABLE 1. *Results for problem set 1*

Problem	1	2	3	4	5	6	7
Size of customer × plants	4 × 5	6 × 5	8 × 5	10 × 10	10 × 10	15 × 10	35 × 20
Optimal solution	434	2,622	47,600	3,108	3,425	6,127	30,484
Lower bound	431	2,611	47,600	3,108	3,425	6,121	30,216
Duality gap (%)	0.69	0.42	0.00	0.00	0.00	0.10	0.08
CPU time (s)	0.342	0.706	0.053	0.319	0.379	2.779	7.314

Note: All times in DEC 2060 time-sharing system at Carnegie-Mellon.

The duality gap referred to in the following tables was computed as

$$\text{duality gap} = (Z^{UB} - Z_{LB})/Z^{UB} * 100.$$

As may be seen from Tables 1 and 2, this Lagrangian heuristic for the problem is very good. In problem set 1, none of the problems had a duality gap of more than 1%. Zero duality gaps were found in three of the seven problems. Also, by using a branch-and-bound algorithm for the CPLP, the upper bounds found in these problems were seen to be optimal. The optimality of problem

TABLE 2. *Results for random problems*

Problem	1	2	3	4	5	6
Size of customer × plants	25 × 25	25 × 25	25 × 25	50 × 50	50 × 50	50 × 50
Lower bound	7,028	7,482	7,587	14,741	14,515	14,461
Upper bound	7,129	7,501	7,632	14,794	14,625	14,536
Duality gap (%)	1.42	0.25	0.59	0.36	0.68	0.52
CPU time (s)	24.60	22.68	21.51	96.07	94.09	96.65

Note: All times in DEC 2060 times-sharing system at Carnegie-Mellon.

one, which has a different optimal value compared to CPLP, was verified by hand computation. There is a duality gap of less than 1% for all but one problem in the problem set containing randomly generated problems.

RESULTS ON PC/AT

A code has been written in Microsoft FORTRAN to run on the PC. Table 3 provides the result for some randomly generated problems that were solved on an IBM PC/AT with a 80286 coprocessor at the Indian Institute of Management, Ahmedabad. It can be seen that even on the IBM PC the computation times are relatively small.

TABLE 3. *Results for randomly generated problems on an IBM PC*

Problem number	1	2	3	4	5
Size: 25 × 8					
Lower bound	5,345	4,528	5,741	5,658	5,410
Upper bound	5,539	4,584	6,064	5,967	5,589
Duality gap (%)	3.52	1.22	5.33	5.18	3.20
CPU time (s)	94	92	92	99	98
Size: 25 × 16					
Lower bound	6,550	6,645	7,598	6,270	6,549
Upper bound	6,688	6,681	7,874	6,363	6,608
Duality gap (%)	2.06	0.54	3.51	1.46	0.89
CPU time	177	169	170	173	173
Size: 25 × 25					
Lower bound	7,847	7,291	7,256	7,730	7,426
Upper bound	7,934	7,367	7,310	7,856	7,579
Duality gap (%)	1.10	1.03	0.74	1.60	2.02
CPU time (s)	253	260	248	261	260
Size: 50 × 16					
Lower bound	9,649	9,388	8,994	10,121	9,403
Upper bound	9,822	9,621	9,053	10,236	9,679
Duality gap (%)	1.76	2.42	0.65	1.12	2.85
CPU time	337	330	323	332	334
Size: 50 × 33					
Lower bound	11,534	11,730	11,446	11,382	12,682
Upper bound	11,697	11,860	11,601	11,571	12,724
Duality gap (%)	1.39	1.10	1.34	1.63	0.33
CPU time (s)	670	682	678	679	689
Size: 50 × 50					
Lower bound	14,117	14,076	16,355	14,943	13,924
Upper bound	14,301	15,276	16,938	15,115	14,035
Duality gap (%)	1.29	7.86	3.44	1.14	0.79
CPU time (s)	1,024	1,011	1,011	1,026	1,031

Note: All times are in seconds on an IBM PC/AT with processing done with a 80286 coprocessor.

CONCLUSION

We conclude this paper by observing that we have a very efficient heuristic based on a Lagrangian relaxation for solving CPLP with side constraints. Beasley[9] also shows that Lagrangian heuristics are efficient for different types of location problems. As pointed out earlier in this paper, this extension of CPLP is useful in solving a decision problem that involves the choice of picking a plant of a particular size, from a given set of alternatives, at a location.

Acknowledgement—The author is grateful to the referees for their comments and suggestions that have added considerable value to this paper.

REFERENCES

1. G. CORNUEJOLS, R. SRIDHARAN and J. M. THIZY (1991) A comparison of heuristics and relaxations for the capacitated plant location problem. *Eur. J. Opl Res.* **50,** 280–297.

2. T. L. Magnanti and R. T. Wong (1984) Network design and transportation planning. *Transport. Sci.* **18,** 1–55.
3. T. L. Wong (1983) Recent research on location and network design problems: an annotated bibliography. *Proceedings, Summer School on Combinatorial Optimisation.* National Institute of Higher Education, Dublin, 4–15.
4. H. M. Salkin (1975) *Integer Programming.* Addison-Wesley, Reading, Mass.
5. R. L. Francis and J. M. Goldstein (1974) Location theory: a selective bibliography. *Opns Res.* **22,** 400–410.
6. N. Christofides and J. E. Beasley (1983) An algorithm for the capacitated plant location problem, *Eur. J. Opl Res.* **12,** 19–28.
7. V. Srinivasan and G. L. Thompson (1972) An operator theory of parametric programming for the transportation problem—I & II. *Nav. Res. Logist. Q.* **19,** 205–252.
8. S. Kim and M. Guignard (1983) A strong Lagrangian relaxation for capacitated plant location problems. W. P. No. 56, Wharton School, University of Pennsylvania.
9. J. E. Beasley (1990) Lagrangean heuristics for location problems. Working paper. Imperial College, London.