J. Opl Res. Soc. Vol. 29, 10, pp. 989 to 995
Pergamon Press Ltd 1978. Printed in Great Britain

0030-3623/78/1001-0989$02.00/0

A Branch and Bound Algorithm for the p-Median Transportation Problem

ALAN W. NEEBE

School of Business Administration and Curriculum in Operations Research and Systems Analysis, University of North Carolina at Chapel Hill, NC 27514

The p-median transportation problem is to determine an optimal solution to a transportation problem having an additional constraint restricting the number of active supply points. The model is discussed as an example of a public sector location/allocation problem. A branch and bound procedure is proposed to solve the problem. Lagrangian relaxation is used to provide lower bounds. Computational results are given.

INTRODUCTION

CONSIDER a set of m supply points indexed by $I = \{1, \ldots, m\}$ each with known supply $a_i > 0$ of a homogeneous commodity, and a set of n customers indexed by $J = \{1, \ldots, n\}$ each with known demand $b_j > 0$ of that commodity. Let c_{ij}, $i \in I$, $j \in J$, be the unit transportation cost between supply point i and customer j. Then the p-median transportation problem (PMTP) is to determine a minimal cost transportation schedule subject to the restriction that at most p supply points be utilized ($1 \leq p \leq m$). This "p-median" constraint frequently occurs in public sector location models (see ReVelle *et al.*[1]) where the number of facilities (or in this case supply points) to be established is given. The constraint is particularly appropriate when the costs of establishing facilities do not vary between sites, and where a budget constraint limits the number of established facilities. Also, in public sector models the cost of establishing the facilities (money) is frequently in different units from the costs of operating the facilities (e.g. a measure of service to the public). Thus the location cost may be isolated from the allocation cost by restricting p, the number of established facilities. A given problem will be solved for a range of values of p.

The PMTP is related to two well-known facility location problems: the p-median problem (Garfinkel *et al.*[2]) and the capacitated warehouse location problem (CWLP) (Akinc and Khumawala[3]). In the p-median problem, given any set of p medians, the corresponding optimal assignment of customers to medians is easily obtained by assigning every customer to a closest (or least expensive) established median. However, in the PMTP, given any set of p supply points, a transportation problem must be solved to obtain the corresponding optimal allocation. A CWLP may be obtained from the PMTP by eliminating the p-median constraint and associating a fixed cost f_i with each supply point $i \in I$. This fixed cost is incurred if supply point i is established. Thus the CWLP may be thought of as a private sector location model where the location and allocation costs are in the same units and may be added. Note that an algorithm to solve the CWLP can be used, although indirectly, to solve the PMTP. This may be done by setting $f_i = f$ for all $i \in I$, and parametrizing the resulting problem on this constant fixed cost f. By increasing f from 0, optimal solutions to all PMTPs will be obtained sequentially, starting with $p = m$ and decreasing to $p = 1$ or until no feasible solution is possible. A simple search procedure may be used to solve a given PMTP. However, as will be seen, it is more efficient to solve the PMTP directly.

The PMTP may be formulated in the following fashion. Let x_{ij}, $i \in I$, $j \in J$, be the number of units shipped from supply point i to customer j. Define binary decision variables y_i, $i \in I$, such that $y_i = 1$ if supply point i is utilized, and $y_i = 0$ otherwise.

Then the PMTP is to determine

$$z^* = \text{minimum} \sum_{i\in I} \sum_{j\in J} c_{ij} x_{ij} \tag{1}$$

subject to

$$\sum_{j\in J} x_{ij} \leq a_i \qquad \text{for all } i \in I \tag{2}$$

$$\sum_{i\in I} x_{ij} = b_j \qquad \text{for all } j \in J \tag{3}$$

$$x_{ij} \leq \min\{a_i; b_j\} y_i \qquad \text{for all } i \in I, j \in J \tag{4}$$

$$\sum_{i\in I} y_i \leq P \tag{5}$$

$$x_{ij} \geq 0 \qquad \text{for all } i \in I, j \in J \tag{6}$$

$$y_i = 0, 1 \qquad \text{for all } i \in I. \tag{7}$$

Although it is possible to replace constraints (2) and (4) with the constraints

$$\sum_{j\in J} x_{ij} \leq a_i y_i \qquad \text{for all } i \in I \tag{8}$$

to obtain an integer-equivalent formulation, we will show reasons for not doing so.

We propose a branch and bound strategy for solving the PMTP. In constructing the branch and bound tree, a supply point i is defined to be "open", "closed", or "free" depending, respectively, on whether supply point i is established ($y_i = 1$), is not established ($y_i = 0$), or whose status is yet to be determined ($y_i \in \{0, 1\}$). K_1, K_0 and K_2 denote the sets containing the indices of open, closed and free supply points. Initially, K_1 and K_0 are empty, while $K_2 = I$. Thereafter, the algorithm proceeds by selecting an active node in the branch and bound tree, according to a given node selection rule. From this node, a free supply point i is selected, according to a given branching decision rule. Two new nodes are created, corresponding to $y_i = 0$ and $y_i = 1$. A suggested node selection rule and a suggested branching decision rule are given later. A lower bound on the objective value of any node in the tree is given by

$$\sum_{j\in J} b_j \cdot \min_{i\in K_1 \cup K_2} \{c_{ij}\}. \tag{9}$$

This lower bound, while easy to compute, unfortunately is not very strong, but we describe a Lagrangian relaxation for the PMTP which may be used to obtain an improved lower bound.

A node need not be branched on further, and is thus a terminal node, if any of the following three conditions holds:

(1) $|K_1| = p$. In this case since p supply points have been selected, a transportation problem using those supply points in K_1 must be solved. Provided this transportation problem is feasible, an upper bound on the optimal objective value is obtained.
(2) $|K_0| = m - p$. In this case since $m - p$ supply points have been closed, a transportation problem using those supply points in $K_1 \cup K_2$ must be solved. If this transportation problem is feasible, an upper bound is obtained.
(3) The lower bound is greater than or equal to the current minimum upper bound. In this case the node is terminal since no better solutions will be found further down this section of the tree.

The algorithm continues branching until all nodes in the tree are terminal.

A LAGRANGIAN RELAXATION

In this section we will describe a Lagrangian relaxation which may be used to obtain an improved lower bound at any node in the branch and bound tree. Lagrangian relaxation is a dual-based technique which has proved successful in solving a number of related combinatorial problems. For example, the technique has been used by Ross and Soland[4] in their algorithm for the generalized assignment problem, by Cornuejols *et al.*[5] in their procedure for the p-median problem, and by Nauss[6] in his algorithm for the CWLP.

The PMTP at any node in the branch and bound tree is the original problem (1)–(7) in addition to the appropriate constraints

$$\begin{aligned} y_i &= 0 \quad \text{for all } i \in K_0 \\ y_i &= 1 \quad \text{for all } i \in K_1. \end{aligned} \tag{10}$$

For fixed K_0 and K_1, let z^* (K_0, K_1) be the optimal objective value to the PMTP (1)–(7), (10).

Let $\lambda = (\lambda_1, \ldots, \lambda_n) \geq 0$ be a vector of Lagrange multipliers associated with the constraints (3). If the objective function (1) is augmented with these constraints, a relaxed form of the PMTP is given by the following Lagrangian problem:

$$L(\lambda) = \text{minimum} \sum_{i \in I} \sum_{j \in J} (c_{ij} - \lambda_j) x_{ij} + \sum_{j \in J} \lambda_j b_j$$

subject to (2), (4)–(7), (10).

For fixed λ, the Lagrangian problem may be solved by inspection: decomposing on the index i, and relaxing constraints (5) and (10) for the present, we are left with $m - |K_0|$ subproblems. Subproblem $i \in K_1 \cup K_2$ is of the form

$$\rho_i^* = \text{minimum} \sum_{j \in J} (c_{ij} - \lambda_j) x_{ij} \tag{11}$$

subject to

$$\sum_{j \in J} x_{ij} \leq a_i \tag{12}$$

$$x_{ij} \leq \min\{a_i; b_j\} \cdot y_i \qquad \text{for all } j \in J \tag{13}$$

$$x_{ij} \geq 0 \qquad \text{for all } j \in J \tag{14}$$

$$y_i = 1. \tag{15}$$

N.B. the corresponding subproblem with $y_i = 0$ replacing (15) is trivial since in this case $x_{ij} = 0$ for all $j \in J$ and the objective value is 0. For a given i, let $\{x'_{ij}\}$ be an optimal solution to (11)–(15). This solution is easily obtained by ranking the $(c_{ij} - \lambda_j)$ in non-decreasing order, and adding the x'_{ij} to the solution in this order, always making x'_{ij} as large as possible consistent with (12) and (13). Continue in this fashion until either (a) constraint (12) is satisfied as an equality or (b) the next smallest $(c_{ij} - \lambda_j)$ is positive.

Now, to satisfy constraints (5) and (10), let E be the index set of the $p - |K_1|$ smallest ρ_i^*'s from the set K_2. Then an optimal solution $\{y_i^*\}$, $\{x_{ij}^*\}$ to the Lagrangian problem is given by

$$y_i^* = \begin{cases} 1 \text{ if } i \in K_1 \cup E \\ 0 \text{ otherwise} \end{cases}$$

$$x_{ij}^* = \begin{cases} x'_{ij} \text{ for all } j \in J \text{ if } i \in K_1 \cup E \\ 0 \text{ for all } j \in J \text{ otherwise.} \end{cases}$$

Finally

$$L(\lambda) = \sum_{i \in K_1 \cup E} \rho_i^* + \sum_{j \in J} \lambda_j b_j.$$

Observe that $\{y_i^*\}$ $\{x_{ij}^*\}$ is not necessarily feasible to the PMTP (1)–(7), (10) since constraints (3) are not necessarily satisfied. Provided $\lambda \geq 0$, it is well known (by "weak duality") that $L(\lambda) \leq z^*$ (K_0, K_1) the optimal objective value to the corresponding PMTP. The goal is to find the "best" lower bound, which is now accomplished by solving the dual problem:

$$L^* = \text{maximum } L(\lambda)$$
$$\text{subject to } \lambda \geq 0.$$

The objective is to find the right λ vector. For this we use the subgradient procedure proposed by Held *et al.*[7] Assume the Lagrangian problem has been solved for a given λ. For all $j \in J$ let

$$g_j = \sum_{i \in I} x_{ij}^* - b_j.$$

Held *et al.* define $g = (g_1, \ldots, g_n)$ as a subgradient, and suggest using it as a direction to alter λ. Note that if

$$c_{(1),j} = \min_{i \in I} \{c_{ij}\}$$

is the smallest cost corresponding to customer j, clearly λ_j is at least equal to $c_{(1),j}$. The general subgradient procedure is as follows:
Step 1. Initialize λ.
Step 2. Given λ, solve the Lagrangian problem to obtain the solution $\{y_i^*\}$, $\{x_{ij}^*\}$.
Step 3. If $g_j = 0$ for all $j \in J$, stop (the optimal solution to the PMTP (1)–(7), (10) has been found). Otherwise, go to Step 4.
Step 4. For all $j \in J$, let

$$\lambda_j = \max\{\lambda_j - Q_j g_j; c_{(1),j}\},$$

where Q_j is some positive constant. Go to Step 2.

Note that this iterative procedure may be terminated at any time to produce a lower bound. For Step 1 in the initial node of the tree, set $\lambda = [c_{(1),1}, \ldots, c_{(1),n}]$. Thereafter, at any node, set λ equal to that λ vector which produced the greatest lower bound in its predecessor node.

Held *et al.* suggest letting

$$Q_j = \frac{2[\bar{L} - L(\lambda)]}{\sum_{j=1}^{n} g_j^2} \text{ for all } j \in J,$$

where $\bar{L}$ is any upper bound on L^*. They discuss convergence properties of the subgradient procedure, note that the sequence of values $L(\lambda)$ is not necessarily monotonically increasing, and demonstrate that the stepsize rule can be tailored to fit the type of problem being solved. We found that their stepsize rule, while possessing interesting theoretical properties, was computationally ineffective. By examining the behaviour of several test problems, we obtained good results by letting $Q_j = c/b_j$, where c is a given constant. Note that this rule is not arbitrary, since g_j/b_j is the proportion by which customer j was oversupplied or undersupplied by the solution $\{x_{ij}\}$. In this way the change in λ_j is proportional to this amount.

One can see now why the PMTP was formulated with (2) and (4) rather than (8). If the smallest $(c_{ij} - \lambda_j)$, say $(c_{ik} - \lambda_k)$, is non-positive, then the solution to the ith subproblem is $x_{ik}' = a_i$. Since a_i is usually larger than b_j, this solution is usually far from optimal to the original PMTP.

THE BRANCH AND BOUND PROCEDURE

The node selection rule

At the point when the next active node has to be considered, one possibility is to branch on that node having the smallest lower bound. This rule, in further investigating a promising node, tends to discover good feasible solutions (and thus good upper bounds) relatively early in the tree search.

The branch selection rule

For the active node identified by the node selection rule, let λ^* be that λ vector producing the best (i.e. largest) lower bound, and let $\{y_i^*\}$, $\{x_{ij}^*\}$ and $\{\rho_i^*\}$ be the corresponding Lagrangian solution and set of objective values. We need to select some "free" supply point, say k, and create two new nodes. The first node restricts $y_k = 0$, while the second restricts $y_k = 1$. One possibility is to select k such that

$$\rho_k^* = \min_{i \in K_2} \{\rho_i^*\}.$$

This rule tends to maximize the increase in the new lower bound of the node corresponding to $y_k = 0$ and could hasten its termination. Note that the Lagrangian procedure need not be applied to the node corresponding to $y_k = 1$. This is because the free variable y_k has been temporarily set equal to 1 to produce the solution $\{y_i^*\}$. Thus the Lagrangian procedure already has been applied to this node.

COMPUTATIONAL RESULTS

The branch and bound algorithm was written in FORTRAN IV, compiled under level H, and run on an IBM 370/165. An efficient, primal transportation code written by Srinivasan and Thompson[8] was modified and used as a subroutine. The algorithm was tested on a total of five test problems for various capacities a_i and for various values of p. The first two test problems are from Ellwein and Gray[9] while the remaining problems are derived from the Kuehn and Hamburger[10] problems. All of the problems are further described by Ellwein and Gray[9] and by Sa.[11]

Good results were obtained by repeating the subgradient procedure 10 times at every node. In our suggested stepsize rule, the results were obtained by setting $c = 1.5$ for the Ellwein and Gray problems, and $c = 1.0$ for the Kuehn and Hamburger problems. This stepsize rule was reasonably robust, as values for c half or twice as much worked nearly as well.

TABLE 1. ELLWEIN AND GRAY[9] PROBLEM 2; $m = 15$, $n = 45$, WEAK LOWER BOUND = 115595

Capacity a_i	Medians p	Total nodes created	Max. nodes active	Total transportation problems solved	$L(\lambda^*)$	Run time (sec)
1000–5000	11	308	44	14	158518	9.25
	12	144	17	14	152458	4.74
	13	65	14	14	152162	3.44
	14	29	15	7	152162	2.01

TABLE 2. ELLWEIN AND GRAY[9] PROBLEM 3; $m = 15$, $n = 45$, WEAK LOWER BOUND = 115595

Capacity a_i	Medians p	Total nodes created	Max. nodes active	Total transportation problems solved	$L(\lambda^*)$	Run time in (sec)
1500–7500	6	113	25	3	158851	4.91
	7	51	15	2	135340	2.22
	8	69	9	6	123372	2.32
	9	65	11	9	116112	2.64
	10	197	41	18	115595	6.84
	11	315	82	39	115595	12.92
	12	275	82	45	115595	13.07
	13	129	18	27	115595	6.22
	14	29	15	8	115595	2.35

TABLE 3. KUEHN AND HAMBURGER[10] PROBLEM PLANT AT INDIANAPOLIS; $m = 16$, $n = 50$, WEAK LOWER BOUND = 837.92

Capacity a_i	Medians p	Total nodes created	Max. nodes active	Total transportation problems solved	$L(\lambda^*)$	Run time (sec)
15000	6	45	10	10	919.20	3.80
	7	49	9	7	903.51	3.53
	8	19	9	2	887.40	1.95
	9	19	10	1	874.37	1.63
	10	25	11	2	864.16	1.96
	11	27	12	2	855.66	1.96
	12	27	13	2	850.82	2.13
	13	43	14	2	845.76	2.56
	14	37	13	3	841.89	2.31
	15	31	6	2	838.78	1.53
10000	6	61	11	10	937.84	3.87
	7	23	8	3	914.68	2.22
	8	31	10	4	898.33	2.92
	9	27	10	2	883.28	2.19
	10	23	11	2	871.97	1.96
	11	25	12	2	864.03	1.93
	12	25	13	1	858.85	1.79
	13	39	14	2	853.69	2.47
	14	43	20	3	848.78	2.81
	15	31	16	2	846.82	2.09

Computational results are given in Tables 1–5. For all problems we give the values of m and n and, for the initial node, the value of the "weak" lower bound as determined by (9). We indicate values of a_i, p, the total number of nodes created in the branch and bound tree, the maximum number of nodes that were active, and the total number of transportation problems that were solved. As one measure of the efficiency of the subgradient procedure, we indicate the value of $L(\lambda^*)$ for the initial node of the branch and bound tree. Finally we give the total computer c.p.u. time in seconds required to solve the problem.

In summary, all problems attempted were solved in a reasonable amount of computer time. Our computational results compare favourably with Akinc and Khumawala's

TABLE 4. KUEHN AND HAMBURGER[10] PROBLEM PLANT AT INDIANAPOLIS; $m = 25$, $n = 50$, WEAK LOWER BOUND = 652.04

Capacity a_i	Medians p	Total nodes created	Max. nodes active	Total transportation problems solved	$L(\lambda^*)$	Run time (sec)
15000	6	45	9	7	805.70	5.64
	8	97	11	18	762.56	9.41
	10	197	36	21	733.05	13.58
	12	151	19	17	711.89	9.06
	14	61	13	4	697.28	4.18
	16	69	15	3	681.88	3.42
	18	77	19	5	672.05	4.49
	20	51	18	2	663.07	3.44
	22	51	22	2	656.39	3.68
	24	49	2	1	652.04	1.56
10000	6	61	12	12	832.57	7.08
	8	111	24	17	777.34	11.13
	10	187	32	26	740.46	16.42
	12	127	23	14	716.45	10.82
	14	55	18	3	700.54	5.17
	16	89	24	5	684.23	6.54
	18	77	19	6	674.99	4.87
	20	67	21	3	666.17	4.30
	22	73	23	3	659.64	5.49
	24	49	25	2	655.27	4.18

TABLE 5. KUEHN AND HAMBURGER[10] PROBLEM PLANT AT JACKSONVILLE; $m = 25$, $n = 50$, WEAK LOWER BOUND = 942.87

Capacity a_i	Medians p	Total nodes created	Max. nodes active	Total transportation problems solved	$L(\lambda^*)$	Run time (sec)
15000	6	49	14	5	1114.90	6.04
	8	33	10	4	1061.81	4.02
	10	37	16	3	1028.18	4.11
	12	113	16	13	1010.03	7.83
	14	93	17	10	993.00	7.90
	16	116	39	8	976.29	12.91
	18	111	25	6	965.22	7.34
	20	95	21	5	956.50	5.41
	22	45	5	1	948.90	2.15
	24	49	2	1	942.87	2.01
10000	6	93	18	18	1133.00	12.56
	8	211	23	32	1074.32	16.68
	10	77	21	7	1038.01	6.41
	12	283	21	32	1013.53	14.69
	14	181	29	20	995.86	12.84
	16	259	36	24	980.73	15.31
	18	137	24	9	968.43	7.78
	20	97	28	5	959.62	6.44
	22	61	29	2	951.81	5.20
	24	49	25	3	946.19	4.34

results for the CWLP using the same data sets. The comparison is even more favourable when one considers that the PMTP algorithm cannot utilize the powerful "delta" and "omega" tests which were used to solve the CWLP. To illustrate, Akinc and Khumawala's algorithm required 12.55 sec of IBM 370/165 time to solve Ellwein and Gray's problem 2, and 4.17 sec to solve problem 3. On the other hand, depending on the capacity a_i, their algorithm required from 0.23 to 38.65 sec to solve Kuehn and Hamburger's $m = 16$, $n = 50$ problem. Finally, after 2 min of run time, their algorithm still could not solve two out of four versions of Kuehn and Hamburger's $m = 25$, $n = 50$ "plant at Indianapolis" problem.

REFERENCES

1. C. ReVelle, D. Marks and J. C. Liebman (1970) An analysis of private and public sector location models. *Mgmt Sci.* **16,** 692–707.
2. R. S. Garfinkel, A. W. Neebe and M. R. Rao (1974) An algorithm for the *m*-median plant location problem. *Trans. Sci.* **8,** 217–236.
3. U. Akinc and B. M. Khumawala (1977) An efficient branch and bound algorithm for the capacitated warehouse location problem. *Mgmt Sci.* **23,** 585–594.
4. G. T. Ross and R. M. Soland (1975) A branch and bound algorithm for the generalized assignment problem. *Math. Prog.* **8,** 91–103.
5. G. Cornuejols, M. L. Fisher and G. L. Nemhauser (1977) Location of bank accounts to optimize float: an analytic study of exact and approximate algorithms. *Mgmt Sci.* **23,** 789–810.
6. R. M. Nauss (1975) An efficient algorithm for the capacitated facility location problem. Working Paper, University of Missouri—St. Louis.
7. M. Held, P. Wolfe and H. P. Crowder (1974) Validation of subgradient optimization. *Math. Prog.* **6,** 62–88.
8. V. Srinivasan and G. L. Thompson (1971) A FORTRAN V code for the primal transportation algorithm. Graduate School of Industrial Administration, Carnegie–Mellon University.
9. L. B. Ellwein and P. Gray (1971) Solving fixed charge location–allocation problems with capacity and configuration constraints. *AIIE Trans.* **3,** 290–297.
10. A. A. Kuehn and M. J. Hamburger (1963) A heuristic program for locating warehouses. *Mgmt Sci.* **9,** 643–666.
11. G. Sa (1969) Branch-and-bound and approximate solutions to the capacitated plant-location problem. *Ops Res.* **17,** 1005–1016.