Section IV

Discrete and Network Location Problems

Annals of Operations Research, 18 (1989) 189–212

THE USE OF STATE SPACE RELAXATION FOR THE DYNAMIC FACILITY LOCATION PROBLEM

M. FRANTZESKAKIS [1] and C.D.T. WATSON-GANDY [2]

[1] *The School of Management, Imperial College, Exhibition Road, London SW7 2BX, UK*
[2] *School of Defence Management, Royal Military College of Science, Shrivenham, Swindon, Wilts SN6 8LA, UK (formerly at Imperial College)*

Abstract

Dynamic facility location is concerned with developing a location decision plan over a given planning horizon during which changes in the market and in costs are expected to occur. The objective is to select from a list of predetermined possible facility sites the locations of the facilities to use in each period of the planning horizon to minimise the total costs of operating the system. The costs considered here include not only transport and operation/maintenance charges but also relocation costs arising from the opening and closing of facilities as required by the plan.
The problem is formulated in terms of dynamic programming but for simplicity with restrictions on the numbers of facilities that can be opened in a given period. The problem was solved using both dynamic programming and a branch and bound approach using state space relaxation. These two approaches are contrasted with different data and with different assumptions to compare the influence of alternative factors on the computational efficiency of both solution methods.

1. Introduction

Most warehouse and facility location problems take place against a background of changing circumstances. With the passage of time, for example, markets will change, populations will grow or decline in different parts of the area of concern, freight rates will change, the transport infrastructure and road networks will change, and advances in technology will affect the cost structure. It follows, therefore, that over time the optimal number and location of the facilities required to service the demand will alter to cater for the changing situation. Two approaches have been taken to handle this problem.

The first approach is referred to as the *static* facility location problem. The static approach examines a single time period at a time giving essentially a series of snap shots over what is really a continually unfolding situation. A plan derived from a series of static snap shots may require that some facilities be opened and some may be closed from one picture to the next. Clearly it costs money to invest in new facilities and to close old ones. However the costs of opening and shutting facilities cannot be taken into account within a static model, and must be treated exogenously in a suboptimal and clumsy fashion.

The second approach is termed the *dynamic* facility location problem. The objective of the dynamic approach is to devise the lowest cost location plan over a given planning horizon. The dynamic model requires the demand data over the whole planning period, and, taking into account the costs of opening and closing facilities, will determine the best location strategy by balancing savings in service costs (transportation, facility maintenance, etc.) against the costs of changing the network of facilities.

A very considerable amount of work has been carried out in relation to facility location problems. The vast majority of this work has involved static models with differing characteristics, and only a limited number of papers have been published on dynamic models. This state of affairs is not because the dynamic problem is considered unimportant but because the dynamic models tend to depend on the static models for the background, and the latter are simpler to examine.

Dynamic models are principally concerned with planning the location and/or size of facilities over time and typically apply to plant, warehouse and public facility expansion. An interesting topology of the possible models is given by Shephard [19], and further review papers by Erlenkotter [5,6] and Luss [12]. The main division within the published literature concerns capacity restrictions. The work which started principally with Manne [13], examines the timing and size of capacity expansions rather than the introduction new plant. Other important papers using a similar restriction are those by Erlenkotter [4], Rao and Rutenberg [14], Jacobsen [9] and Fong and Srinivasan [7]. Alternatively some authors assume that capacity constraints are not important and that any new facility located will be large enough to cater for all the demand required of it during the whole planning period. Papers adopting this approach have been published by Klein and Kimpel [10], Ballou [1], Roodman and Schwartz [16], Wesolowsky and Truscott [22,23] and Van Roy and Erlenkotter [20].

As with static models a division also exists between public facilities, e.g. schools, fire stations, hospitals and medicare facilities, and private facilities, e.g. plant or warehouses. Papers describing the dynamic location of public facilities are given by ReVelle, Toregas and Falkson [15], Shilling [17,18], Gunawardane [8] and Chrissis et al. [2].

Some models examine only the case of increasing demand. These include Erlenkotter [4], ReVelle, Toregas and Falkson [15], Rao and Rutenberg [14], Jacobsen [9], and Schilling [18]. Roodman and Schwartz [16] examine the case of decreasing demand where facilities are to be phased out. The more general case will both open and close facilities as demand fluctuates.

Most models treat time as discrete and some care should be taken to avoid any confusion as to when within a time period the changes take place. Models employing continuous time formulations are given by Erlenkotter [4] and Rao and Rutenberg [14].

Integer programming has been used as a solution method by Roodman and

Schwartz [16], ReVelle, Toregas and Falkson [15], Wesolowsky and Truscott [23], Schilling [17,18], Gunawardane [8], Chrissis, Davis and Miller [2], and dual-based mixed integer programming by Van Roy and Erlenkotter [20]. Heuristic procedures have been employed by most authors examining capacity constrained models such as Manne [13], Erlenkotter [4], Jacobsen [9], Rao and Rutenberg [14], and Fong and Srinivasan [7].

Dynamic programming has been used by Ballou [1] and Wesolowsky and Truscott [23]. Dynamic programming has also been used as a basis for the heuristic procedures of Erlenkotter [4] and Wesolowsky and Truscott [23].

In this paper the discrete time uncapacitated dynamic facility location problem is examined. It is felt that although time is continuous, the discrete time approach offers a more convenient way of handling the costs of opening and closing facilities which are such an essential part of the problem. A branch and bound approach based on state space relaxation in dynamic programming is presented to solve the problem with the objective of minimising the discounted costs of operation and location over the given planning horizon. It is suggested these costs are discounted in order to account for the time value of money and to avoid those solutions which start with an over expensive pattern of facilities in the hope of making savings later.

The outline of this paper is as follows. Firstly, the problem is discussed in a little more detail and a dynamic programming formulation is presented. Then a branch and bound procedure is developed using a state space relaxation of the dynamic programming formulation. Finally some computational results are presented comparing dynamic programming and the new procedure.

2. The problem and dynamic programming formulation

In general terms the problem can be described as follows: Given a number of customers whose location and demand are assumed known for each period of the planning horizon and a number of potential facility sites available for use during each period, it is required to determine which sites should be used during each period so that the total discounted costs

$$C = \sum_{k=0}^{M} (\text{operating cost})(1+r)^{-k} + \sum_{k=0}^{M} (\text{relocation cost})(1+r)^{-k}$$

is minimised subject to the constraints:

(i) the demand of all customers must be met;

(ii) the number of facilities used in any period are within the limits pre-specified for that period.

Where $M =$ the number of equal periods in the planning horizon and $r =$ discount rate

The costs of operating refer to the cost of operating the entire system per period using whichever sites are appropriate. The costs of relocation are incurred if the facilities used differ from one period to the immediately following period.

The solution to the above problem can be simplified by the introduction of the concept of "Location Sets". A location set is defined here as a feasible combination of potential facility sites. For example, if there are three potential sites A, B and C, then there are 7 location sets.

$$\{A\}, \{B\}, \{C\}, \{AB\}, \{AC\}, \{BC\}, \{ABC\}$$

In this way instead of carrying out location allocation problem at each stage to determine the operating costs, it is only necessary to solve the customer to facility allocation for each location set. The latter is a relatively easy task. However, two preparation stages are required.

(i) All the feasible location sets must be generated from the list of potential sites for each period.

(ii) For each location set and each period it is necessary to determine the optimal operating costs.

The problem described above can be formulated as a dynamic programming problem as follows. A "forward" recursion is used as this would seem to be more easily appreciated. This formulation follows Watson-Gandy [21] who gives a multi-facility form of the model due to Ballou [1]. Let I be the location set operated in the current period k and J be the location set operated in the previous period $k-1$ where $J \in N_{k-1}$ and N_k is the set of location sets feasible for period k, $k = 1, \dots, M$. Then it is required to minimise the recursive equation.

$$F(I, k) = \min_{J \in N_{k-1}} \left(F(J, k-1) + m_{J,I} \right) + c_{I,k} \tag{1}$$

subject to

$$l_k \leqslant |I| \leqslant u_k \tag{2}$$

$$l_{k-1} \leqslant |J| \leqslant u_{k-1}. \tag{3}$$

Where

$c_{I,k}$ = the cost of operating from location set I in period k
$m_{J,I}$ = the cost of moving from location set J to location set I
l_k = smallest number of facilities allowed to operate in period k
$|I|$ = the number of facilities selected to operate in the current period
u_k = the largest number of facilities allowed to operate in period k.
$F(I, k)$ = the minimum total cost of operating an optimal policy from period 1 and ending with the operation of location set I in period k.

Equation (1) may be simply initialised by

$$F(I, 1) = c_{I,1} \tag{4}$$

for all I such that $l_1 \leqslant |I| \leqslant u_1$.

Note that other forms of initialisation of eq. (1) are possible. For example, it may be that the recursion will start with certain facilities already in place. This would occur if the institution concerned already operated a system.

Then the value of $F(I, k)$ is calculated in a recursive way for all location sets satisfying constraints (2) and (3) for all periods until the last period $k = M$ is reached. The optimal solution is that given by the minimum value of $F(I, M)$ and the optimal policy is derived by backtracking. Note that no mention has been made of the precise details of the operating costs. It is the intention of this paper to present an efficient algorithm to solve the dynamic facility location problem and detailed discussions of a complex costing exercise would distract from that purpose. Furthermore the calculation of the operating costs has no impact on the proposed method of solution. Indeed so long as the operating costs are available, the precise method of their determination is irrelevant to this paper. Nevertheless the concept of location sets is an extremely useful device in providing great flexibility in the application areas for the algorithm to be described. As long as the costs of operation are in the equivalent units to the cost of relocation, the problem tackled may be in either the public or private domain, and facilities, which are available only for part of the planning period, are easily catered for.

3. A branch and bound approach

Since the publication of the pioneering paper of Little et al. [11] for solving the travelling salesman problem, branch and bound techniques have been extensively used for solving combinatorial (NP-hard) optimisation problems. The dynamic depot location problem can be seen as a combinatorial optimisation problem being an extension of the NP-hard simple plant location problem.

In order to develop a branch and bound approach for the dynamic depot location problem it is necessary to define first a "relaxed" problem. This is usually done by relaxing one or more of the constraints of the original problem so that the solution to the relaxed problem provides a lower bound to the original problem. A tree search procedure is then established by solving the relaxed problem at each stage to provide the bounds that lead the search. Clearly the choice of the constraints to relax must lead to a relaxed problem which is quickly and easily solved but which also gives tight lower bounds in order to reduce the number of nodes to be explored in the tree. A relaxed problem will be formed here by using the concept of state-space relaxation first used by Christofides et al. [3].

Dynamic programming is a very suitable technique for many combinatorial problems. It is, however, well known to suffer from the "curse of dimensionality" as the state-space i.e. the number of possible states, grows. To illustrate this let us define M as the number of periods in the planning horizon and N the number of location sets. Then it is clear that in the formulation given here the computation time is proportional to MN^2; that is it is linear in the number of periods and of $O(N^2)$ with the number of location sets. This could be considered bad enough but clearly the number of location sets increases dramatically with the number of depot sites. For example, if there are 7 possible sites and it is permitted to locate up to 4 depots, then there are 98 location sets (all combinations of 7 taken in 4, 3, 2 and 1 ways). If there are 8 possible sites, the number of location sets increases to 162; with 9 possible sites, there are 255 location sets and so on. For realistically sized problems, if n is the number of sites, the number of location sets could be $n!$ and becomes very substantial indeed.

A significant reduction of the state space can be achieved if instead of considering the location sets themselves, only the number of depots that constitute the sets are considered. For the examples given above, this will result in a reduction from 98, 162 or 255 to only 4. To do this the transformation

For I take $|I| = s$ and for J takes $|J| = q$

is used and eqs. (1)–(3) are re-written as

$$F(s, k) = \min_q \left(F(q, k-1) + m_{q,s} \right) + c_{s,k} \tag{5}$$

$$\text{subject to } l_k \leqslant s \leqslant u_k \tag{6}$$

$$l_{k-1} \leqslant q \leqslant u_{k-1} \tag{7}$$

However, recursion (5) cannot be used as it stands because the values of $m_{q,s}$ and $c_{s,k}$ depend on the location sets I and J from which s and q are derived. However, as it is required to produce a lower bound, the substitution $\overline{m}_{q,s}$ for $m_{q,s}$ and $\bar{c}_{s,k}$ for $c_{s,k}$ may be chosen so as to satisfy

$$\overline{m}_{q,s} \leqslant m_{q,s} \text{ and } \bar{c}_{s,k} \leqslant c_{s,k} \tag{8}$$

$\overline{m}_{q,s}$ and $\bar{c}_{s,k}$ can be defined as follows

$$\overline{m}_{q,s} = \begin{cases} 0 & \text{if } q = s \\ (q-s)v_1 & \text{if } q > s \\ (s-q)v_2 & \text{if } s > q \end{cases} \tag{9}$$

Where v_1 = the cost of opening a depot, v_2 = the cost of closing a depot.

$$\text{and } \bar{c}_{s,k} = \min\{ c_{I,k} \} \tag{10}$$

$\overline{m}_{q,s}$ and $\bar{c}_{s,k}$ are the minimum possible values that $m_{q,s}$ and $c_{s,k}$ can take and so the inequalities (8) are satisfied. The relaxed problem may now be written as

$$F(s, k) = \min_q \left(F(q, k-1) + \overline{m}_{q,s} \right) + \bar{c}_{s,k} \tag{11}$$

$$\text{subject to } l_k \leqslant s \leqslant u_k \tag{12}$$

$$l_{k-1} \leqslant q \leqslant u_{k-1}. \tag{13}$$

Because the inequalities (8) are satisfied, the solution to the recursion (11) produces a lower bound to the original problem. The recursion (11) is also very much easier to solve than the original recursion (1) since the state space is substantially reduced.

The relaxed problem developed above is used to produce bounds in a branch and bound tree search procedure. This is done in the following way: Initially all the location sets available are assumed to operate for the first period and then bounds are calculated for each location set in turn by solving the relaxed problem for the remaining periods. This is done by initialising eq. (11) by setting

$$F(s, 1) = c_{I,1} \text{ for all } I \in N_1$$

where $s = |I|$ for the location set I in mind and then applying the recursive equation (11) for period $k = 2, \ldots, M$. This gives $|N_1|$ nodes with the bounds defined by the actual costs for period 1 plus a lower bound for the remaining periods. We branch from the node with the smallest lower bound. Let this node correspond to the operation of location set I^* during the first period. We now generate $|N_2|$ more nodes of the tree assuming that location set I^* was operating in the first period and each one of the N_2 location sets in the second period. Bounds for these nodes are calculated by solving the relaxed problem initialised by

$$F(s, 2) = (c_{I^*,1} + m_{I^*,I}) + c_{I,2} \text{ for all } I \in N_2$$

where $s = |I|$ and then applying (11) for the periods $k = 3, \ldots, M$. We continue to branch from the node with the smallest lower bound (excluding any which have already been branched from). In general, if the node with the minimum bound for period $k-1$ corresponds to location set I^*, then $|N_k|$ new nodes corresponding to the $|N_k|$ location sets are generated. The bounds are calculated from the relaxed recursive equation (11) initialised by

$$F(s, k) = (\text{actual operating and moving costs so far}) + m_{I^*,I} + c_{I,k} \tag{14}$$

where $s = |I|$ and $I \in N_k$, and applying the recursion (11) for $k = k + 1, \ldots, M$. The procedure comes to an end when the node with the currently lowest value corresponds to the final period.

It can be seen from (14) that the values associated with any terminal node (for which $k = M$) are the actual costs. Furthermore as the procedure is designed to ensure there is no repetition or cycling, the procedure will terminate in a finite number of operations.

4. Computational results

Both the dynamic programming and the branch and bound algorithms were programmed in FORTRAN 77 in order to compare their performance. In general it is expected that two particular parameters will show differences in the relative performance of the two algorithms. These parameters are

(i) the number of periods. Dynamic programming is linear with respect to the number of periods are required. However as the number of periods increases the higher is the chance of the branch and bound tree becoming bigger as the quality of the lower bounds in the early periods is likely to deteriorate with respect to the optimal solution.

(ii) The number of location sets. As the number of location sets increases the curse of dimensionality will effect the dynamic programming approach. It is expected that the branch and bound approach would show an increasing advantage.

Some other factors may also affect the performance of the branch and bound algorithm. Two factors in particular stand out. The first is the degree of "difficulty" of the problem. It is reasonable to assume that the more the customers' demands fluctuate the more nodes may be required to be examined in the branch and bound procedure. The second factor is the relative size of the relocation costs in relation to the other costs. It will be recalled that the effectiveness of the bounds derived from the relaxed problem is related to the value of the relocation costs. The relaxed problem assumes that one can move from one location set to another at minimum cost. However the larger the re-location costs the less effective the lower bound will become. Take, for example, the perhaps unlikely case of moving from one set of three facilities to another location set consisting of three entirely different facilities. In this case the

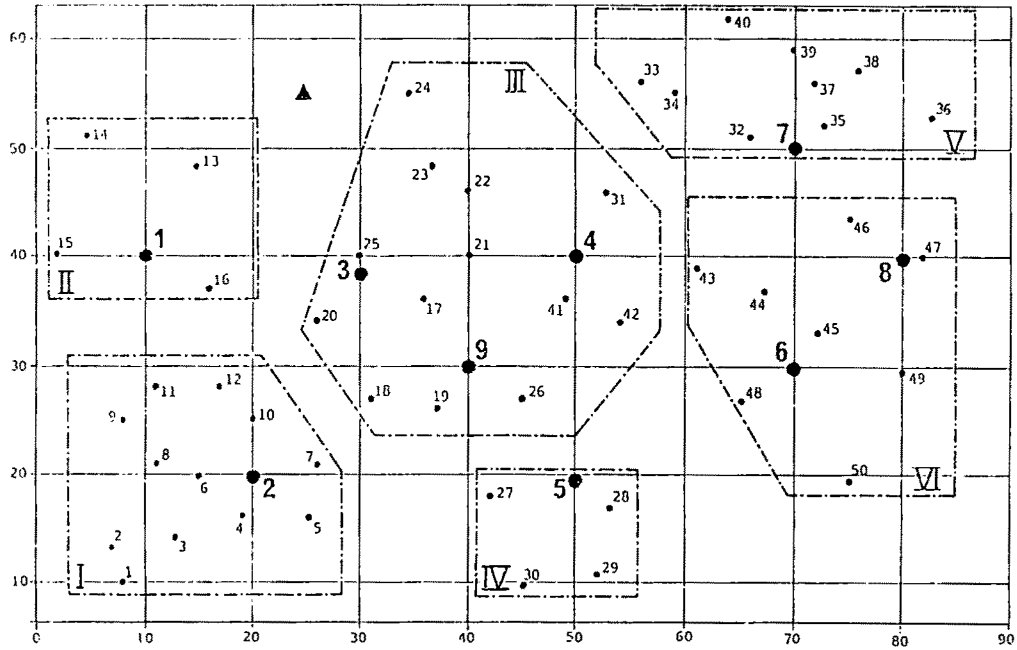


Fig. 1. Showing the customer and potential depot locations, and areas.

bound for the relocation costs is zero, whereas in reality the cost is three times the sum of the costs of opening and closing a depot.

To test these characteristics the following problem was devised representing the distribution problem of a manufacturing company. Fifty customers were generated and grouped into 6 sales areas, as shown in fig. 1. Note these areas are used as a device in the forecasting of future demand; it is difficult to predict with accuracy how the demand of individual customers will vary with time but easier to forecast the aggregated demand for an area. Figure 1 also shows 9 potential

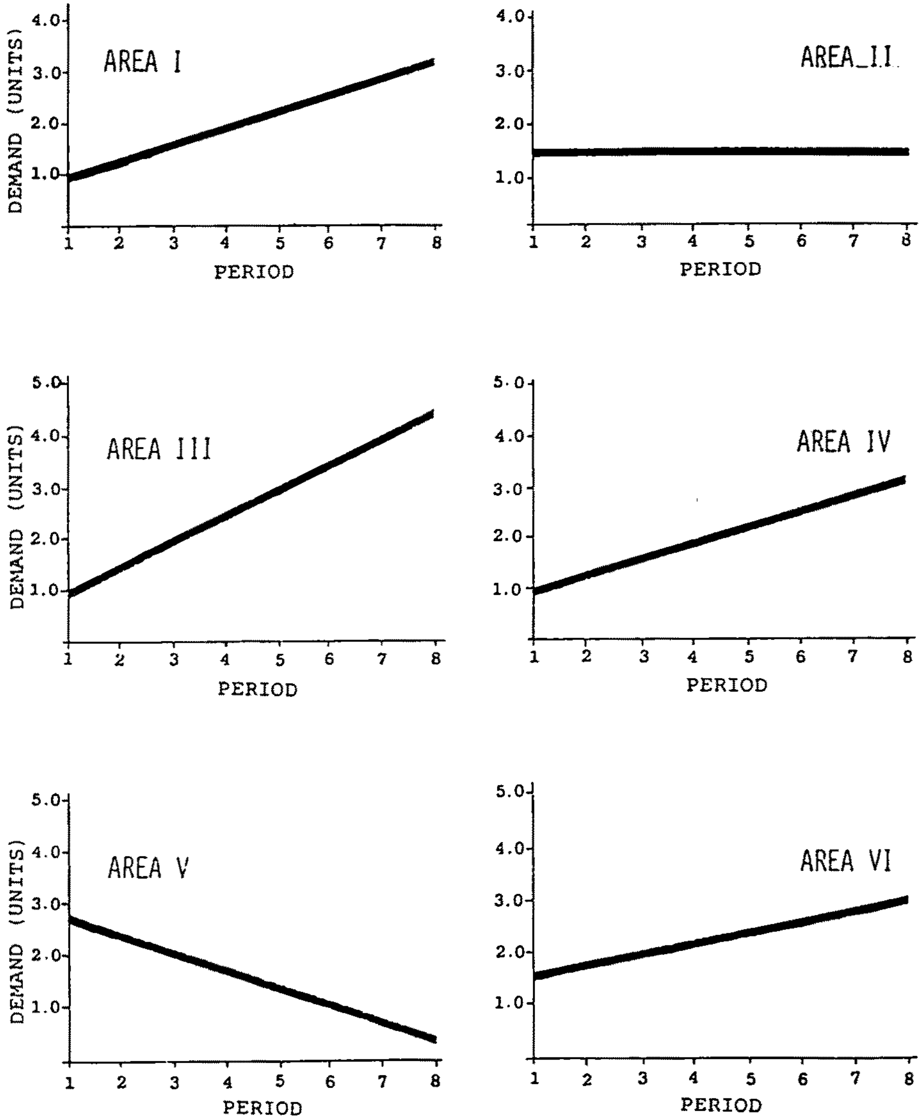


Fig. 2. The demand changes for data set 1.

warehouse sites and the location of the factory or manufacturing plant. The factory is essentially irrelevant to the discussion here but important in the calculation of the costs. Goods will be trunked from the factory to the warehouses,

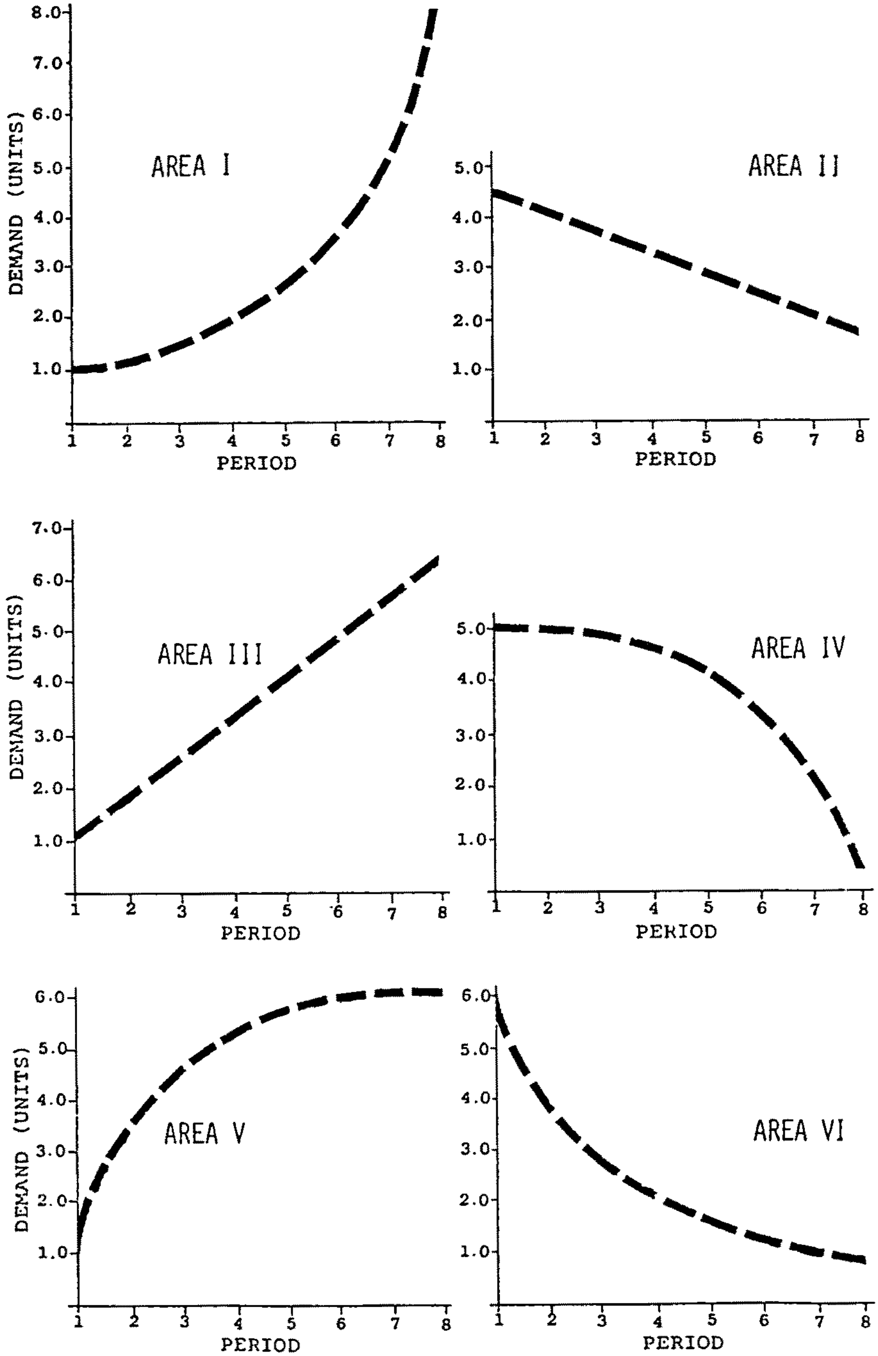


Fig. 3. The demand changes for data set 2.

and these costs, as well as the costs of goods storage within the warehouses, must be considered in the total cost equation. To compare the degree of difficulty 2 data sets were created which differ in the way the customer demand develops in each of the 6 areas. Data set 1 has linear changes and no severe fluctuations but data set 2 has non-linear and significant demand changes over time. The demands for the two data sets are illustrated in figs. 2 and 3 respectively.

Several problems were derived from the above data to investigate the effect of the factors that have been identified. The number of periods was varied from 2 to 8 while the number of location sets remained constant. Then keeping the number of periods constant the number of location sets were varied from 28 to 255. These variations were achieved by varying the number of potential sites (from 7 to 9) and changing the number of depots which are allowed to operate during each period. The allowed number is bounded by l_k, which is always 1, and u_k which has been varied from 2 to 5. Note that the bounds are kept constant with each individual problem. Finally using data set 2 and a problem with 5 periods and 98 location sets, the relocation costs were increased in steps from 0 to practically infinity. Here the ratio of closing costs to opening costs was kept constant (at 1 : 1.6). The results of these experiments are discussed below. Note that the

Table 1(i)
Results for data set 1 (98 location sets).

Number of periods	DP time *	BB		
		Time *	Nodes	Levels
2	0.347	0.017	195	2
3	0.363	0.041	293	3
4	0.390	0.073	391	4
5	0.420	0.123	489	5
6	0.449	0.166	587	6
7	0.482	0.225	685	7
8	0.507	0.317	979	10

Table 1(ii)
Results for data set 2 (98 location sets).

Number of periods	DP time *	BB		
		Time *	Nodes	Levels
2	0.345	0.031	391	4
3	0.372	0.214	1273	13
4	0.406	0.333	1371	14
5	0.432	0.533	1763	18
6	0.454	1.167	3327	34
7	0.477	1.770	4185	43
8	0.495	2.139	4241	44

* CP seconds for a CDC 6000 computer system.

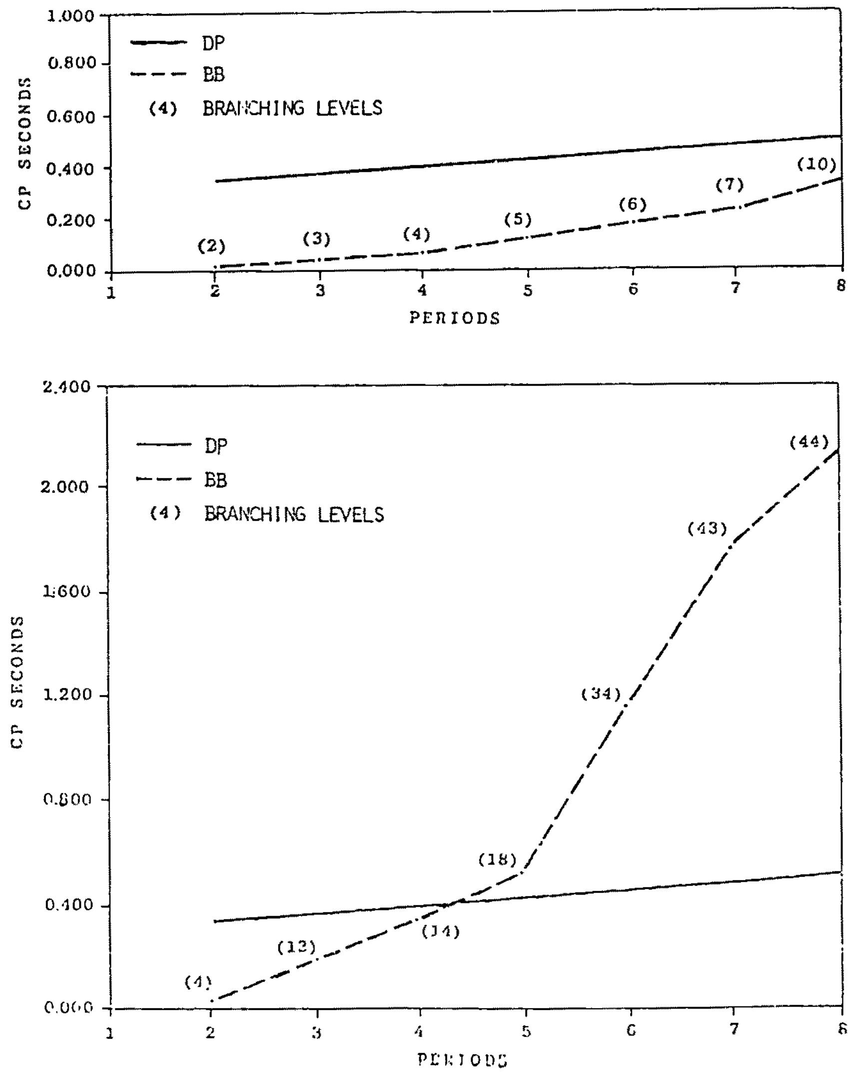


Fig. 4(i). Computer times for data set 1 with 98 location sets. (ii). Computer times for data set 2 with 98 location sets.

computer times given refer only to the running of the algorithms; the times for the initialisation of the problems, which could be substantial in a large problem, are common to both programs and are excluded. We will use the notation (i) and (ii) attached to figure and table numbers to refer to data sets 1 and 2 respectively.

1. *Number of periods*. Tables 1(i) and 1(ii) give the computer times of this experiment and these times are also shown plotted in figs. 4(i) and 4(ii). It can be seen from these that the dynamic programming approach is linear in the number of periods and is not effected by the degree of difficulty of the problem. The branch and bound approach clearly is effected by the degree of difficulty.

2. *Number of location sets*. The results for this experiment are tabulated in tables 2(i) and 2(ii) and illustrated in figs. 5(i) and 5(ii). Here it can be clearly seen how an increase in the state space has a deplorable effect on the dynamic programming approach. The larger the problem the better the state space relaxation method performs especially in the case of data set 1.

3. *Relocation costs*. The computer times given by the branch and bound method for this experiment are tabulated in table 3(ii) and illustrated in fig. 6(ii). Note that changes in relocation costs do not effect the dynamic programming approach at all. These figures show, as expected, that the higher the relocation costs the more effort is required in the branch and bound approach until the costs become so high that no relocation is worthwhile.

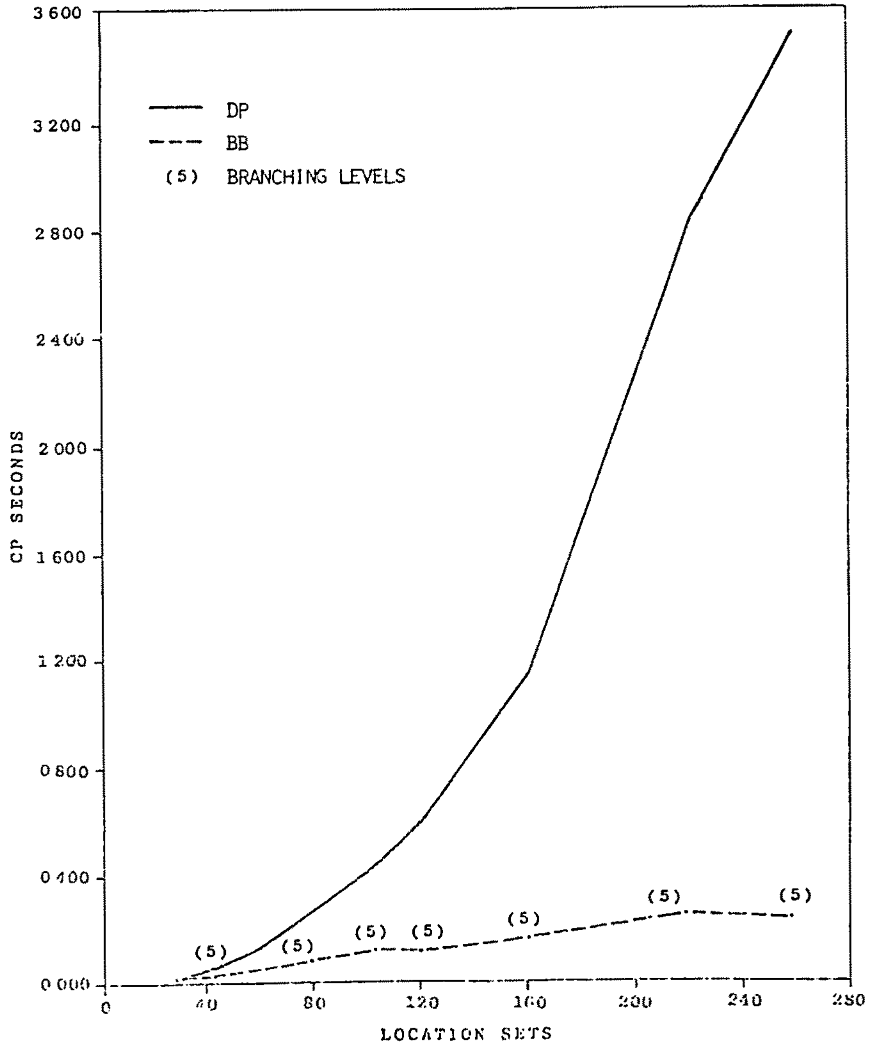


Fig. 5(i). Computer times for data set 1 with 5 periods.

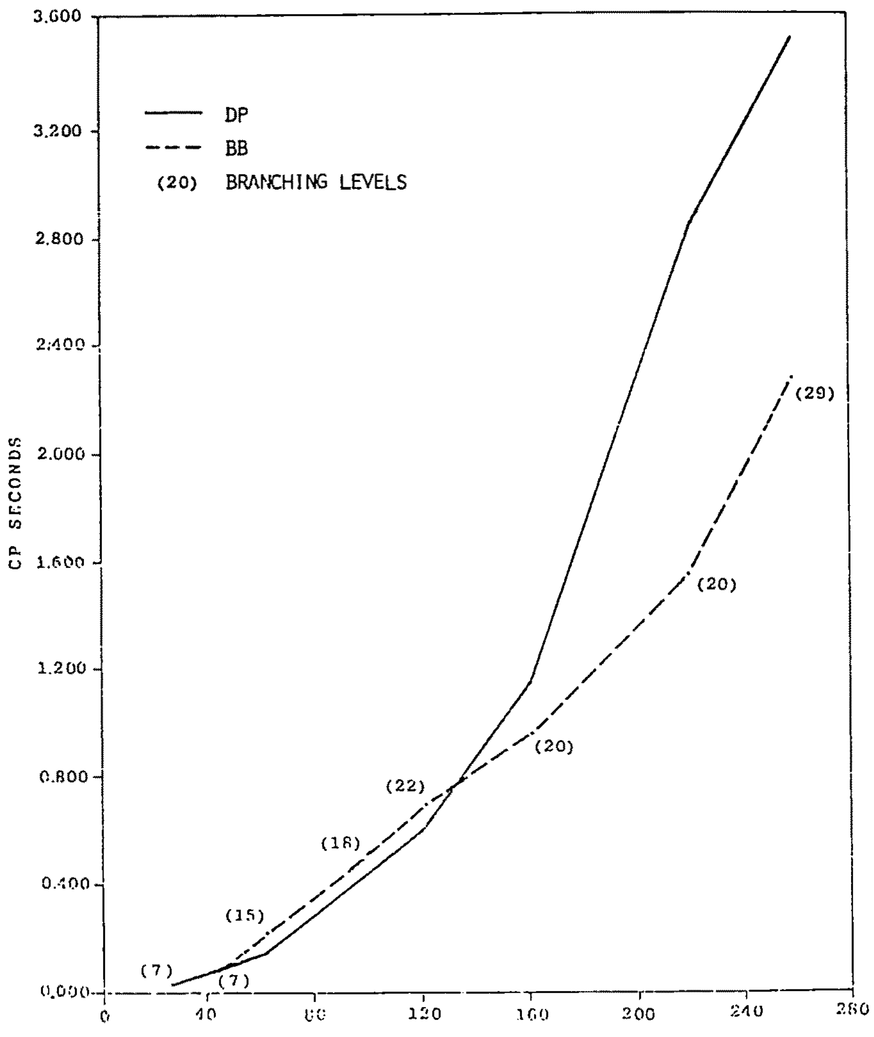


Fig. 5(ii). Computer times for data set 2 with 5 periods.

Finally as an indication of the effectiveness of the bounds calculated we illustrate four trees from the experiments. The trees shown in figs. 7a and 7b show differences with a change in the number of periods; those trees shown in figs. 8a and 8b show differences in the number of location sets. Note that each apparent node on the tree, which we call a "branching level", actually consists, not of 1 node but of N nodes (where N is the number of permitted location sets) and node 1 represents the N_1 nodes for period 1. Although this cannot be stated as a general rule, almost all the trees we developed followed this pattern. The reasons may be various but can be summarised in the statement that the pattern

Table 2(i)
Results for data set 1 (5 periods).

Depot sites	Max. depots in one period	Location sets	DP time *	BB		
				Time *	Nodes	Levels
7	2	28	0.023	0.022	140	5
9	2	41	0.063	0.032	215	5
7	3	63	0.151	0.061	315	5
7	4	98	0.420	0.120	489	5
9	3	125	0.630	0.110	572	5
8	4	162	1.205	0.176	725	5
8	5	218	2.282	0.268	923	5
9	4	255	3.546	0.241	1037	5

Table 2(ii)
Results for data set 2 (5 periods).

Depot sites	Max. depots in one period	Location sets	DP time *	BB		
				Time *	Nodes	Levels
7	2	28	0.025	0.031	196	7
9	2	41	0.062	0.051	309	7
7	3	63	0.147	0.217	945	15
7	4	98	0.432	0.533	1763	18
9	3	125	0.627	0.698	2873	23
8	4	162	1.193	0.959	3029	20
8	5	218	2.877	1.576	3907	20
9	4	255	3.574	2.333	6870	29

* CP time in seconds on a CDC 6000 computer.

Table 3(ii)
Computing times for the BB method (data set2) with 5 periods and 98 location sets.

Relocation costs		Time *	Nodes	Levels
Closing	Opening			
0.0	0.0	0.124	489	5
1.0	1.6	0.160	587	6
2.0	3.2	0.250	881	9
3.0	4.8	0.359	1175	12
4.0	6.4	0.502	1576	16
** 5.0	8.0	0.533	1763	18
6.0	9.6	0.537	1665	17
7.0	11.2	0.500	1567	16
8.0	12.8	0.501	1567	16
9.0	14.4	0.426	1371	14
10.0	16.0	0.428	1371	14
16.0	25.6	0.424	1371	14
999.0	999.0	0.424	1371	14

* CP time in seconds on a CDC 6000 computer.
** Actual values used in the earlier results.

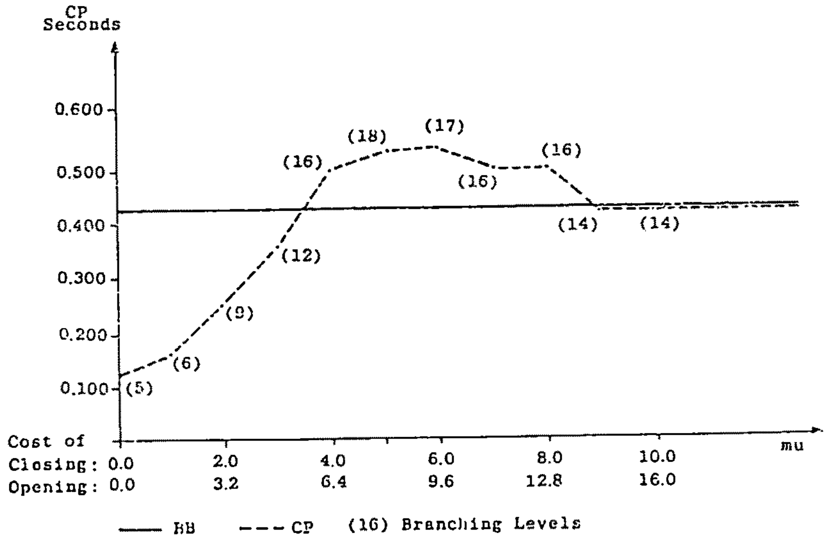
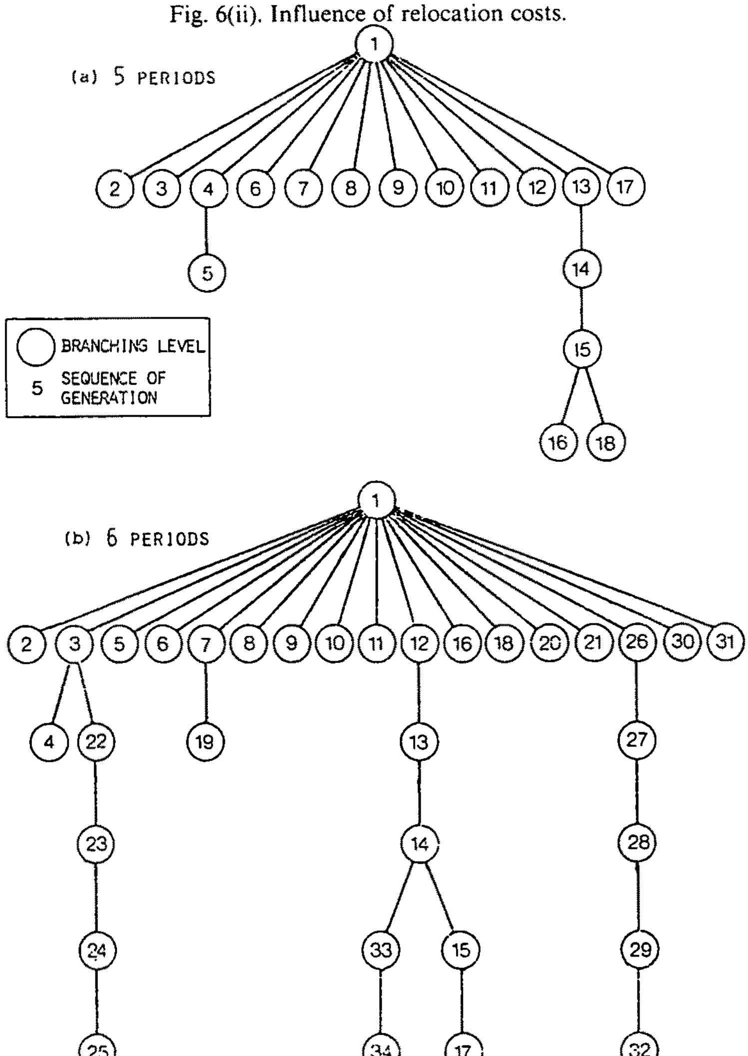


Fig. 6(ii). Influence of relocation costs.



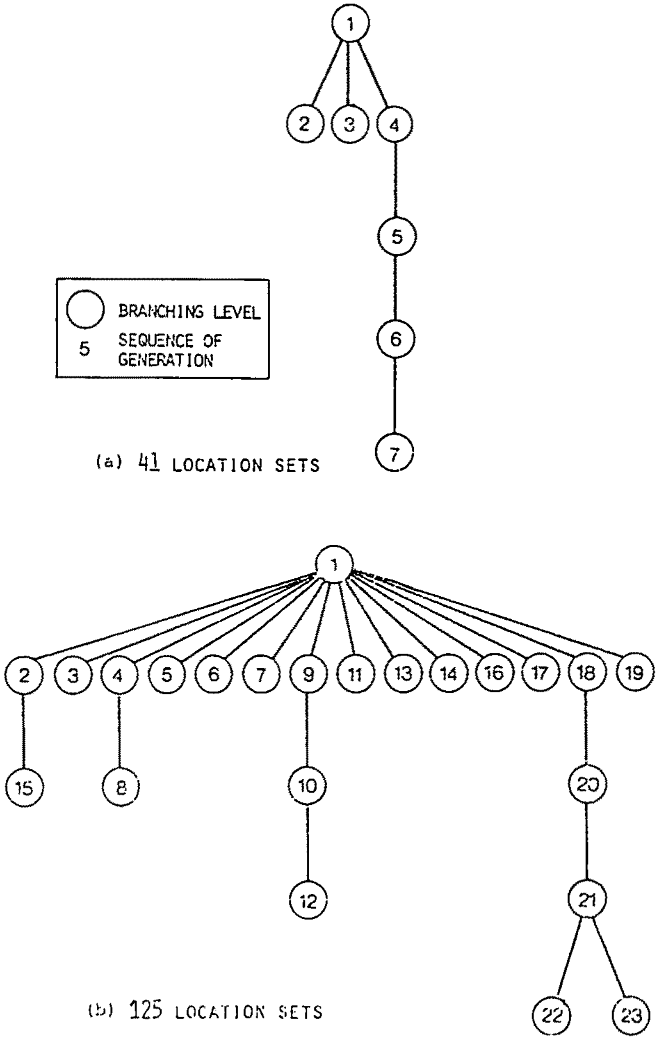


Fig. 8. Trees for data set 2 (5 periods).

of the costs in the problems we examined, were such that certain location sets could be identified as of no worth very early in the search.

5. Conclusions

We have developed and described an approach to solving the dynamic facility location problem using state space relaxation in dynamic programming. The procedure has been programmed and compared with a classical dynamic programming approach using an example from the distribution industry. Both techniques are shown to have advantages in performance under differing circumstances but in realistically sized problems there will be a large number location sets–which is where our proposal technique is most effective.

References

[1] R. Ballou, Dynamic warehouse location analysis, Journal of Marketing Research 5 (1968) 271.
[2] J.W. Chrissis, R.P. Davis and D.M. Miller, The dynamic set covering problem, Appl. Math. Modelling 6 (1982) 2.
[3] N. Christofides, A. Mingozzi and P. Toth, State space relaxation procedures for the computation of bounds to routing problems, Networks 11 (1981) 145.
[4] D. Erlenkotter, Capacity planning for large multilocation systems: approximate and incomplete dynamic programming approaches, Mgmt. Sci. 22 (1975) 274.
[5] D. Erlenkotter, Bibliography on dynamic location models, Discussion paper No. 55, Management Science Study Centre, Graduate School of Management, University of California, Los Angeles, CA, 1975.
[6] D. Erlenkotter, A comparative study of approaches to dynamic location problems, Eur. J. Oper. Res. 6 (1981) 133.
[7] C.O. Fong and V. Srinivasan, The multiregion dynamic capacity expansion problem: an improved heuristic, Mgmt. Sci. 32 (1986) 1140.
[8] G. Gunawardane, Dynamic versions of set covering type public facility location problems, Eur. J. Oper. Res. 10 (1982) 190.
[9] S.K. Jacobsen, Heuristic solutions to dynamic plant location problems, in: *Advances in Operational Research*, ed. M. Roubens (North Holland, Amsterdam 1977).
[10] M. Klein and K. Kimpel, Application of linearly constrained non-linear optimisation to plant location and sizing, Journal of Industrial Engineering 18 (1967) 90.
[11] J.D.C. Little, K.G. Murty, D.W. Sweeney and C. Karel, An algorithm for the travelling salesman problem, Ops. Res. 11 (1963) 979.
[12] H. Luss, Operations research and capacity expansion problems: a survey, Ops. Res. 30 (1982) 907.
[13] A.S. Manne (ed), *Investments for Capacity Expansion: Size, Location and Time-phasing* (The MIT Press, Cambridge, Mass. 1967).
[14] R.C. Rao and D.P. Rutenberg, Multilocation plant sizing and timing, Mgmt. Sci. 23 (1977) 1187.
[15] C. ReVelle, C. Toregas and L. Falkson, Applications of set-covering problem, Geographical Analysis 8 (1976) 65.
[16] G.M. Roodman and L.B. Schwarz, Optimal and heuristic facility phase-out strategies, AIIE Trans. 7 (1975) 177.
[17] D.A. Schilling, Dynamic location modelling for public-sector facilities: a multicriteria approach, Decision Sciences 11 (1980) 714.
[18] D.A. Schilling, Strategic facility planning: the analysis of options, Decision Sciences 13 (1982) 1.
[19] E. Shephard, A conceptual framework for dynamic location-allocation analysis, Environment and Planning A, 6 (1974) 547.
[20] T.J. van Roy and D. Erlenkotter, A dual-based procedure for dynamic facility location, Mgmt. Sci. 28 (1982) 1091.
[21] C.D.T. Watson-Gandy, Warehouse/depot location, in: *Marketing Logistics and Distribution Planning*, eds. M. Christopher and G. Wills (George Allen and Unwin, London, 1972).
[22] G. Wesolowsky and W. Truscott, The multiperiod location-allocation problem with re-location of facilities, Mgmt. Sci. 22 (1975) 57.

[23] G. Wesolowsky and W. Truscott, Dynamic location of multiple facilities for a class of continuous problems, AIIE Trans. 8 (1976) 76.

Appendix

In this appendix a small example is given to contrast the two different methods and demonstrate the calculations involved in the branch and bound method. A 3 period problem with 3 potential sites (A, B and C) is considered. The restrictions on the problem are that not more than 2 sites should be used in period 1 and not more than 3 sites in periods 2 and 3. This gives $l_1 = l_2 = l_3 = 1$ and $u_1 = 2$; $u_2 = u_3 = 3$. There are 7 potential location sets

{A}, {B}, {C}, {AB}, {AC}, {BC}, {ABC} of which the last {ABC} is not feasible for period 1.

The operating costs for these location sets have been calculated for each period and are given in table A-1 below
The cost of opening a facility is 10 mu and the cost of closing a facility is 8 mu. These values are, for convenience, taken as the same for all periods. Table A-2 shows the costs involved of transferring from one location set to another.
The problem will be solved first using dynamic programming. Here table A-3 shows the costs for the first period.
Table A-4 (A-5) shows the costs for the second (third) period. Note that the calculations for $F(5, 2)$ are given in full, whereas only the minimum values are given for the other values of I.
From table A-5 the minimum cost strategy costs 115 mu and is derived from operating either set 5 {AC} or set 7 {ABC} in the third period and set 5 in the second and first periods.

Table A-1
Operating costs (mu) for each location set per period.

Location set	Facility	Period		
		1	2	3
1	A	35	45	68
2	B	38	48	70
3	C	40	50	65
4	A, B	48	37	50
5	A, C	40	30	45
6	B, C	45	35	40
7	A, B, C	–	50	35

Table A-2
The relocation costs from one set to another.

		From Location Set:						
		1	2	3	4	5	6	7
To location set:	1	0	18	18	18	8	26	16
	2	18	0	18	8	26	8	16
	3	18	18	0	26	8	8	16
	4	10	10	28	0	18	18	8
	5	10	28	10	18	0	18	8
	6	28	10	10	18	18	0	8
	7	20	20	20	10	10	10	0

Table A-3
The first period.

I	$F(I, 1)$
1	35
2	38
3	40
4	48
5	40
6	45

Table A-4
The second period.

I	J	$F(J, 1)$	$m_{J,I}$	$\text{Min}(F(J, 1)+m_{J,I})$	$c_{I,2}$	$F(I, 2)$
1	1	35	0	35	45	80
2	2	38	0	38	48	86
3	3	40	0	40	50	90
4	1	35	10	45	37	82
	1	35	10			
	2	38	28			
	3	40	10			
5	4	48	18	40	30	70
	5	40	0			
	6	45	18			
6	6	45	0	45	35	80
7	5	40	10	50	50	100

Table A-5
The third period.

I	J	$F(J, 2)$	$m_{J,I}$	$\text{Min}(\text{Min}(F(J, 2)+m_{J,I}))$	$c_{I,3}$	$F(I, 3)$
1	1	70	8	78	68	146
2	2	86	0	86	70	156
3	5	70	8	78	65	143
4	4	82	0	82	50	132
5	5	70	0	70	45	115
6	6	80	0	80	40	120
7	5	70	10	80	35	115

Branch and bound

The full tree for the problem solved by the proposed method is shown in fig. A-1 which uses the following notation. The large figure within the circle is I, the location set to be used at that period (or level). The small figures underneath the node in brackets is the lower bound value. The small figure to the left of the node is the node number. The calculations for two nodes, node 5 in period 1 and node 7 in period 2, will be given now to illustrate the method. However, first table A-6 gives the minimum operating costs for a given number of depots. These are extracted from table A-1. For example, the minimum operating costs for 1 depot in period 1 is min $\{35, 38, 40\} = 35$.

Similarly table A-7 gives the minimum relocation costs

Node 5: For node 5 location set 5 (that is {AC} with 2 facilities) is to be operated in the first period.

Therefore $F(2, 1) = c_{5,1} = 40$ mu.

This is the value of $F(s, 1)$ for period 1.

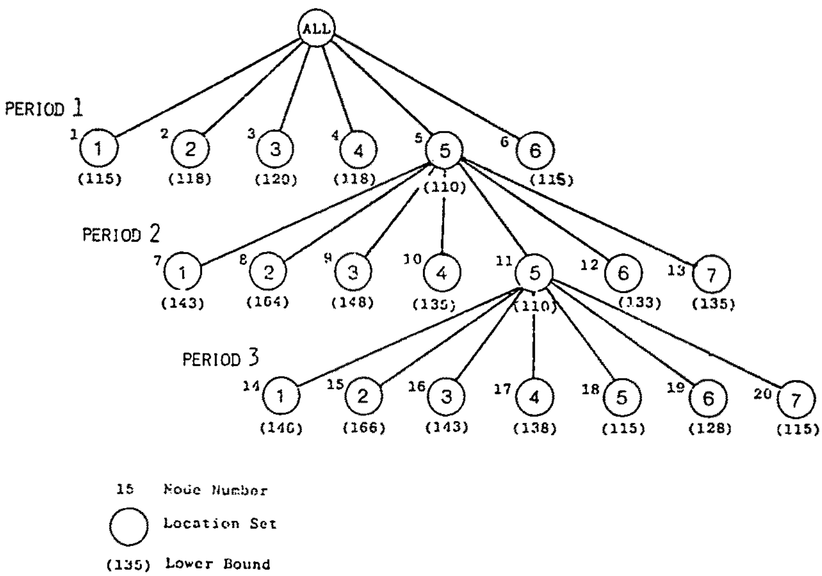


Fig. A-1. Branch and bound tree search.

Table A-6
The minimum operating costs (mu).

Number of depots	Period		
	1	2	3
1	35	45	65
2	40	30	40
3	–	50	35

Table A-7
Minimum relocation costs.

		To:		
		1	2	3
From:	1	0	10	20
	2	8	0	10
	3	16	8	0

The recursion (11) is now followed to give the bounds for node 5 as shown in Table A-8. For period 2 q has the value of 2 and $F(q, 1)$ is 40 from period 1. s takes the values 1, 2 or 3 facilities and the minimum operating costs $c_{s,2}$ are 45, 30 and 50 respectively. To move from $q = 2$ to $s = 1$ means closing one facility at a cost of 8 mu.
Hence $F(1, 2) = 40 + 8 + 45 = 93$.
The minimum value of $F(s, 2)$ is 70 mu with $s = 2$.
Finally the lower bound for node 5 is found in period 3 to be 110 mu with $s = 2$
Node 7: The calculations for node 7 can be made in a similar manner to that shown above. Node 7 is in period 2 and required that location set 5 {AC} be operated in period 1 and location set 1 {A} in period 2. This gives for period 2.

$$F(1, 2) = c_{5,1} + m_{5,1} + c_{1,2} = 40 + 8 + 45 = 93.$$

Following recursion (11) for period 3 gives the lower bound 143 mu as can be seen from table A-9.

Table A-8
The bounds for Node 5.

PERIOD 1

s	$F(s, 1)$
2	40

PERIOD 2

s	q	$F(q, 1)$	$\overline{m}_{q,s}$	$\bar{c}_{s,2}$	$F(s, 2)$
1	2	40	8	45	93
2	2	40	0	30	70
3	2	40	10	50	100

PERIOD 3

s	q	$F(q, 2)$	$\overline{m}_{q,s}$	$\bar{c}_{s,3}$	$F(s, 3)$
1	2	70	8	65	143
2	2	70	0	40	110
3	2	70	10	35	115

Table A-9
The bounds for node 7.

PERIOD 2

s	$F(s, 2)$
1	93

PERIOD 3

s	q	$F(q, 2)$	$\overline{m}_{q,s}$	$\bar{c}_{s,3}$	$F(s, 3)$
1	1	93	0	65	158
2	1	93	10	40	143
3	1	93	20·	35	148

It can be seen that the lower bounds for the third period ($M = 3$) are the actual values and the optimum solution is given by node 18 or node 20 and is exactly similar to the solution given by the dynamic programming approach.

Acknowledgements

The authors would like to express their gratitude to the referees for their helpful comments.
M. Frantzeskakis became the Flockvale Distribution Limited 1986 Student of the Year for the work involved in this paper.