ELSEVIER

European Journal of Operational Research 87 (1995) 203–213

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

Invited Review

The capacitated plant location problem

R. Sridharan

Department of Econometrics, Monash University, Clayton, Victoria, 3168, Australia

Accepted January 1995

Abstract

This paper provides a review of the various solution methods for the capacitated plant location problem. Heuristic and exact procedures, that have appeared in the literature, are covered. The review also examines two innovative concepts the Lagrangian heuristic and variable splitting, as proposed for the capacitated plant location problem.

Keywords: Distribution; Facilities; Location; Branch and bound; Heuristics; Lagrangian relaxation.

1. Introduction

There are many problems in Operations Management and Operations Research that require grouping parts to be processed by the same machine or clients to be serviced by the same service center.

For example, a basic scheduling problem occurs when several machines can be set up to perform a given operation. The parts requiring this operation must be partitioned into groups assigned to different machines. The problem is to find the minimum number of machines that must be set up for the given operation in order to process all the parts before a desired completion time. If the machines are not identical, it must be decided which subset of the machines should be set up.

Equipment replacement problems involve the selection of machines from a set of several possible alternative machines with different capacities and costs of purchase and operation. In addition to the type of machine, the selection also involves the replacement time. The objective is to minimize the total purchase and fixed cost of operating the machines (less its salvage value) and the total variable cost of production. The demand constraints need to be satisfied for each time period and the capacity constraints for each machine. Part of the decision problem is to identify the quantity produced by each machine in each time period.

The Star–Star Concentrator Location Problem (SSCLP) is a computer communication network design problem that involves connecting several remote terminal sites to a central site. Each terminal site is connected to the central site either directly or through a concentrator. A concentrator site is connected to the central site via high-speed lines. The concentrator sites are usually a subset of terminal sites and the capacity of a concentrator is defined in the problem. Each terminal must be connected, via low-speed lines, to a unique site, which is either a concentrator or

SSDI 0377-2217(95)00042-9

the central site. The fixed cost of installing a concentrator site and connecting it to the central site is given. The cost of connecting each terminal to a concentrator site is also given. The problem is to find a network that will minimize the total cost.

The Generalized Bin Packing Problem (GBPP) can be described as follows. We have a number of items with weights that have to be put in the bins, each item going into only one bin. There are a number of bins, each with a capacity and fixed cost, that can be used. The problem is to identify a subset of the bins to minimize the total fixed cost such that all the items can be assigned to the bins without exceeding their capacities.

All the decision problems that have been described above have the following common feature. In all these cases the decision involves two stages. In the first stage a choice of the subset of machines (or trucks or concentrators or bins) is made. In the second stage the parts (or clients or terminals or items) are assigned to the chosen subset of machines (or trucks or concentrators or bins).

The Capacitated Plant Location Problem (CPLP) also shares the same common features in terms of the underlying decision problem. In CPLP, there are a set of potential locations for plants with fixed costs and capacities, and a set of customers, with demands for goods supplied from these plants. The transportation cost per unit for goods supplied from the plants to all the customers is given. The problem is to find the subset of plants that will minimize the total fixed and transportation costs such that demand of all the customers can be satisfied without violating the capacity constraints of the plants. As described above, the two stages in the decision process can be identified for CPLP. In the first stage a choice is made of the subset of the plants to be opened and in the second stage the assignment of the customers to these plants is made.

When there is an additional restriction on CPLP that each customer be served only from a single plant, the Capacitated Plant Location Problem with Single Source constraints (CPLPSS) is obtained. It is immediate to see that CPLPSS also shares the same common features described above in terms of the underlying decision problem.

In this paper the Capacitated Plant Location Problem is reviewed. It can be shown that all the other problems described above are either special cases of CPLP or CPLPSS or closely related to these two problems.

2. The capacitated plant location problem

The location of plants such as warehouses or factories, is an inevitable strategic decision for most organizations as it has a direct bearing on the cost of supplying commodities to customers. Transportation costs often form a major portion of the price (or cost) of goods. Equally important to the organizations are the fixed costs involved in opening and operating a plant at any given location. Such location problems have been widely studied in the literature under the names of plant, warehouse, or facility location problems. When each potential plant has a capacity, that is, an upper bound, on the amount of demand that it can service, the problem is known as the capacitated plant location problem (CPLP). When the capacity assumption is not made, the problem is known as the simple or uncapacitated plant location problem (SPLP).

The capacitated plant location problem, with n potential plants and m customers, can be formulated as a mixed integer program:

(P)

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

subject to

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

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

$$0 \le x_{ij} \le y_j \le 1, \quad i = 1, \ldots, m; \; j = 1, \ldots, n; \tag{4}$$

$$y_j = \{0, 1\}, \quad j = 1, \ldots, n; \tag{5}$$

and

$$\sum_{j=1}^{n} s_j y_j \geq \sum_{i=1}^{m} d_i, \tag{6}$$

where c_{ij} is the total cost of transportation from plant j to serve customer i; d_i is the demand of customer i; s_j is the capacity of plant j; f_j is the fixed cost associated with plant j; x_{ij} is the fraction of the demand of customer i supplied from plant j; $y_j = 0$ or 1, depending on whether plant j is closed or open; $I = \{1, \ldots, m\}$: the set of customers; $J = \{1, \ldots, n\}$: the set of potential locations.

The data consists of c_{ij}, d_i, s_j, and f_j, whereas x_{ij} and y_j are the variables.

The constraints (2) guarantee that the demand of every client is satisfied, and constraints (3) guarantee that each open plant does not supply more than its capacity, and that the clients are supplied only from open plants.

Here, the y_j's are the strategic variables since once the y_j's are fixed, the problem reduces to a transportation problem with capacities s_j and demands d_i. The constraint (6) specifies that the total capacity of open plants should be at least as large as the total demand of the customers. This constraint is actually derived by summing (3) over all the plants and using the equalities (2), and hence is a surrogate constraint. It is redundant in the original formulation, but it strengthens some of the relaxations. Also, the constraints $x_{ij} \leq y_j$ are redundant in the original formulation, but are very useful in relaxations. To see that $x_{ij} \leq y_j$ can be derived from the other constraints, note that (2) and $x_{ij} \geq 0$ imply that $x_{ij} \leq 1$ and using (3) and (5) the constraint $x_{ij} \leq y_j$ is obtained.

In the following sections the various approaches used in the literature to solve CPLP are reviewed.

3. Preview of solution methods for CPLP

The literature on CPLP is very rich; researchers have worked on both heuristic solution methods and exact algorithms to solve CPLP. While the exact algorithms can solve medium-sized problems, say 50 plants and 50 customers, within reasonable computer effort, heuristics are needed to solve problems with several hundred plants and customers.

The heuristics for CPLP are basically extensions of the heuristics used for the Simple Plant Location Problem (SPLP). One of the widely known heuristics for SPLP is the ADD procedure due to Kuehn and Hamburger (1963); another is a DROP procedured proposed by Feldman, Lehrer and Ray (1966). Both ADD and DROP procedures are *greedy* heuristics. For CPLP, ADD and DROP heuristics were tested by Khumawala (1974), Jacobsen (1983) and Domschke and Drexl (1985) among others. A detailed description of these heuristics is given in the following sections.

At the end of ADD or DROP procedures, the solution can be improved by making some perturbation. A typical example of such a method is the bump and shift routine of Kuehn and Hamburger (1963) for SPLP. Methods of this type are referred to as *interchange* heuristics and are studied in Jacobson (1983) and Khumawala (1974) for CPLP. Some of these interchange heuristics will be described in the following sections.

Heuristic methods can be useful in exact algorithms that require feasible solutions (Akinc and Khumawala, 1977). When heuristics are used, it is useful to have an upper bound on the gap between the heuristic and the optimal values.

Recently, a new type of heuristic, called the Lagrangian heuristic, has been proposed by several authors. In this type of heuristics, a Lagrangian relaxation of CPLP is solved, often with a surrogate constraint, with the upper bound being obtained either through a heuristic or an exact algorithm. The upper bound is found usually at the initial and the final steps, but with a suitable surrogate constraint, the upper bound can be updated in almost every Lagrangian iteration. A major advantage of the Lagrangian heuristic over the other heuristics is that an upper bound is obtained in conjunction with a lower bound; see Cornuejols et al. (1991) for details.

A variety of papers have been written on branch and bound procedures to obtain an optimal solution for CPLP. The earliest optimal solutions for CPLP have been attempted by Sa (1969) and Davis and Ray (1969). Subsequently many branch and bound based algorithms for CPLP have been developed see (Akinc and Khumawala, 1977; Christofides and Beasley, 1983; Ellwein and Gray, 1971; Geoffrion and McBride, 1978; Nauss, 1978; Van Roy, 1986; Nagelhout and Thompson, 1981).

Typically, a relaxation of the problem is solved at each node of the enumeration tree in a branch and bound procedure. This could be a linear programming relaxation (see, for example, Erlenkotter, 1978; Van Roy and Erlenkotter, 1982; Akinc and Khumawala, 1977; Sa, 1969; Davis and Ray, 1969; Ellwein and Gray, 1971; Geoffrion and Graves, 1974), or a Lagrangian relaxation (Geoffrion and McBride, 1978; Nauss, 1978; Christofides and Beasley, 1983; Van Roy, 1986). Geoffrion and Graves (1974) use the Benders decomposition procedure and Van Roy (1986) uses the Cross decomposition approach along with the Lagrangian relaxation. These methods are presented in some detail in the following sections.

4. Heuristics for CPLP

Heuristics can normally handle large problems (with several hundred plants) and in many instances the solutions can be expected to be fairly close to the optimum value. The heuristics for CPLP are based primarily on the ones for SPLP and can be clasified under two basic approaches: the greedy and interchange heuristics. Another type of heuristic known as Lagrangian heuristic, recently proposed by several authors, is also presented in this section.

4.1. The greedy heuristics

There are two different greedy heuristics for CPLP, namely, the ADD procedures and the DROP procedures. These two procedures are described in this section. The following notation is used for ease of presentation. Let J0 be the subset of J for which $y_j = 0$; J$\emptyset$ be the subset of J for which y_j is yet undecided; and let J1 be the subset of J for which $y_j = 1$.

4.1.1. ADD procedures

The ADD procedure for the location problem was initially developed by Kuehn and Hamburger (1963) to solve the uncapacitated version of the problem. The extension of this procedure to the capacitated version is given in Jacobsen (1983). Here a formal description of the ADD procedure is given. Let $T^*(J, I)$ represent the optimal value of the transportation problem with source set J, and sink set I and data as in problem (1)–(5). A typical difficulty with the ADD procedure is that it always starts with an infeasible solution to the problem. This is tackled by having a super source with capacity equal to total demand and very high transportation costs to each of the demand locations. A description of the procedure is given below.

Start with no plants open. That is, all plants are in the set J$\emptyset$.

1. For each $j \in$ J$\emptyset$, compute the savings
$$\sigma_j = T^*(\mathrm{J1}, I) - T^*(\mathrm{J1} \cup \{j\}, I) - f_j.$$
2. Now identify the plant j^* that gives the maximum savings σ_{j^*} from
$$\sigma_{j^*} = \max_{j \in \mathrm{J}\emptyset} \{\sigma_j\}.$$
3. If $\sigma_{j^*} > 0$, transfer j^* to J1 and go back to Step 1. If $\sigma_{j^*} \le 0$, terminate the procedure with the set J1 of open plants, since no more savings can be made by adding another plant.

The above procedure requires the solution of |J$\emptyset$| transportation problems at each iteration. However, Khumawala (1974) and Jacobsen (1983) suggest procedures that avoid the need to solve so many transportation problems. Instead, they solve a continuous knapsack problem to obtain a bound on the savings. The bound computed by Khumawala (1974) (also see Akinc and Khumawala, 1977) is a lower bound on the savings, LBS, and is given below.

For $j \in$ J$\emptyset$, let
$$K_{ij} = \min_{k \in \mathrm{J1} \cup \mathrm{J}\emptyset, k \neq j} \{\max(c_{ik} - c_{ij}, 0)\}.$$

Then,

$$\mathrm{LBS}_j = \max \sum_i K_{ij}\gamma_i$$

subject to

$$\sum_i d_i\gamma_i \le s_j, \qquad 0 \le \gamma_i \le 1.$$

Khumawala (1974) observes that the bound LBS_j is very effective.

Jacobsen (1983) provides both a lower and an upper bound on the savings. The heuristic that uses the lower bound in his procedure is termed ADD-LO and the one that uses the upper bound is termed ADD-HI. Domschke and Drexl (1985) also use the same bounds. The lower bound on the savings obtained by adding plant j^* is given by $\Sigma_i\Sigma_{j\in \mathrm{J1}}\ x_{ij}\ \max(0, c_{ij} - c_{ij^*})$, which is the savings obtained by redirecting some of the demands to plant j^*. The upper bound on the savings for j^* is obtained when we substitute u_i^* for c_{ij} in the above expression where u_i^* is the dual variable for customer i in the solution to $T(\mathrm{J1}, I)$. These bounds have been found to be very useful both for solution quality and computation time.

4.1.2. Drop procedures

The DROP procedure was first used by Feldman, Lehrer and Ray (1966) for SPLP. It starts with all the plants in the set J1. In each step, a plant is dropped at a location where the largest savings is obtained. We formalize the procedure as follows:

1. For each plant $j \in \mathrm{J1}$, compute the savings
   $$\sigma_j = f_j + T^*(\mathrm{J1}, I) - T^*(\mathrm{J1}\setminus\{j\}, I).$$
2. Find the plant j^* that gives the maximum savings
   $$\sigma_{j^*} = \max_{j\in \mathrm{J1}}\{\sigma_j\}.$$
3. If $\sigma_{j^*} > 0$, transfer j^* from J1 to J0 and go to Step 1. If $\sigma_{j^*} \le 0$, terminate the procedure with the set J1 of open plants since we do not have positive savings by dropping any more plants.

As in the case of ADD procedures, here also we need to solve $|\mathrm{J1}|$ transportation problems in each iteration. Akinc and Khumawala (1977) and Jacobsen (1983) suggest procedures that provide a bound on the savings with much less computational effort. The bound computed by Akinc and Khumawala (1977) is as follows. Let v_i $(i = 1,\dots,m)$, and u_r, $r \in \mathrm{J1}$ be the optimum dual variables associated with the solution $T^*(\mathrm{J1}\setminus\{j\}, I)$. Define,

$$w_{ij} = v_i - c_{ij}.$$

Then, the savings obtained by closing plant j is,

$$\mathrm{UB}\Omega_j = \max \sum_i w_{ij}\delta_i$$

subject to

$$\sum_i d_i\delta_i \le s_j, \quad j \in J,$$

$$0 \le \delta_i \le 1, \quad i \in I.$$

As for the ADD procedure, Jacobsen (1983) provides both a lower bound and an upper bound on the savings. The DROP procedure that uses a lower bound is DROP-LO, and the one that uses the upper bound in his procedure is termed DROP-HI. For more details on the actual computation of these bounds refer to Jacobsen (1983).

4.2. Interchange heuristics

The ADD and DROP procedures belong to the category of greedy heuristics where once a decision is made it is not changed. Actually, some improvement on the greedy solution can be made by perturbation. The heuristics that attempt to make such improvements are referred to as the Interchange Heuristics. Two different methods that belong to the interchange heuristic category, (i) the Alternate Location Allocation (ALA), and (ii) the Vertex Substitution Method (VSM), are presented in this section.

4.2.1. Alternate location allocation

The earlier application of this method to location problems appear in Rapp (1962) and Cooper (1963). This method can be described as follows.

1. Start with a set of open plants, J1. All other plants are in J∅. Set JT = ∅. (JT is the set of plants that have been tried unsuccessfully in the interchange).
2. If JT = J1, stop. Else, let $j \in$ J1 − JT. Add j to JT.
3. Transfer j from J1 to J0. 'Reoptimize' using an ADD iteration. If the cost of the new solution is smaller than at the beginning of Step 3, then let J1 be defined by this new solution, J∅ = J − J1, J0 = ∅. Set JT = ∅ and go to Step 2. Otherwise, set J1, J0 and J∅ back to their value at the beginning of Step 3. Go to Step 2.

4.2.2. Vertex substitution method

Some application of VSM to SPLP can be found in Teitz and Bart (1968) and Cornuejols, Fisher and Nemhauser (1977). This procedure can be described as follows.

1. Start with a feasible solution. List J0, the set of closed plants. Set JT = ∅.
2. If JT = J1, stop. Else, let $j \in$ J1 − JT. Add j to JT.
3. Transfer j from J0 to J1. 'Reoptimize' by one DROP iteration. If the cost of the new solution is smaller than at the beginning of Step 3, then let J1 be defined by this new solution, J0 = J − J1. Set JT = ∅ and go to Step 2. Otherwise, set J1 and J0 back to what they were at the beginning of Step 3. Go to Step 2.

Some computational results for solving CPLP using ALA and VSM are given in Jacobsen (1983).

4.3. Lagrangian heuristics

The Lagrangian heuristics for CPLP involve a Lagrangian relaxation of (P). The details of Lagragian Relaxations of (P) are given in Section 5.2. A Lagragian relaxation of (P) provides a lower bound for (P), while the upper bound is computed using a suitable heuristic as in Section 4.1 or 4.2 or an exact algorithm. When constraint (6) is used along with a Lagrangian relaxation, it enables the computation of an upper bound in every Lagrangian iteration. Usually, the upper bound is found by a transportation algorithm. For details of this procedure, see Cornuejols et. al (1991), Guignard and Kim (1983), and Nauss (1978).

5. Exact methods for CPLP

An exact solution for CPLP can be obtained by using an enumerational tree. When the relaxed problem meets the constraints of (P), (P) is solved; otherwise a lower bound for (P) is obtained. Various relaxations that have been used for CPLP are described in this section.

There are many papers that use a branch and bound algorithm for CPLP. For CPLP, some examples of branching rules will be (i) to choose a node with the least lower bound, or (ii) to choose a node with least number of free variables. An example of separation will be to fix some y_j variables that are fractional (when the relaxations are LPs) to 0 or 1. In the branch and bound procedure we need to solve a relaxation of the problem P_j at each node n_j of the enumeration tree. Two such relaxations of CPLP are (i) linear programming relaxations, and (ii) lagrangian relaxations.

5.1. Linear programming relaxation

Davis and Ray (1969), Sa (1969), Ellwein and Gray (1971), and Akinc and Khumawala (1977) use an LP relaxation of CPLP in their branch and bound algorithms. They relax the integer constraints on the y_j variables to reduce the problem to an LP and then use some branching rules to fix the y_j variables to 0 or 1. Sa (1969), Ellwein and Gray (1971), and Akinc and Khumawala (1977) work with the so called Weak Linear Programming relaxation (WLP). The WLP does not have the explicit constraints $x_{ij} \le y_j$. On the other hand, Davis and Ray (1969) use the Strong Linear Programming relaxation (SLP), that is, problem (1)–(5) with the integer constraints relaxed, in their procedure. Cornueojols et al. (1991) show theoretically that the lower bound obtained from SLP is usually stronger than the one obtained from WLP. Akinc and Khumawala (1977) claim to obtain a lower bound better than WLP by suitably modifying the capacity constraints of the 'free' plants available at any node of the enumeration tree. Baker (1982) also shows that the LP relaxation is strengthened by adding the disaggregated constraints of the type

$$\frac{\sum_i a_{ij} x_{ij}}{\sum_i a_{ij} \theta_{ij}} \leq y_j,$$

where a_{ij} are arbitrary coefficients, and θ_{ij} are optimal values of the variables x_{ij} in the solution to the continuous knapsack problem:

$$\max\left\{ \sum_i x_{ij} : 0 \leq x_{ij} \leq u_{ij},\ \sum_i x_{ij} \leq s_j \right\},$$

where $u_{ij} = \min\{d_i, s_j\}$.

Baker (1982) also gives some special cases where the constraints reduce to the type already known in the literature.

5.2. Lagrangian relaxation

The Lagrangian Relaxation is an approach used for solving the mixed integer and pure integer programming problems. An illustration of this approach is given by using the following problem as an example. Let

$$\begin{aligned} \text{(IP)} \quad & Z = \min\ cx \\ & \text{s.t.} \quad Ax = b, \\ & \qquad Dx \leq e, \\ & \qquad x \geq 0 \text{ and integral,} \end{aligned}$$

where x is $n \times 1$, b is $m \times 1$, e is $k \times 1$ and all other matrices have conformable dimensions.

Let that the constraints of (IP) be partitioned into two sets $Ax = b$ and $Dx \leq e$ so that the following Lagrangian problem can be easily solved. Let

$$\begin{aligned} (\mathrm{LR}(u)) \quad & Z_D(u) = \min\ cs + u(Ax - b) \\ & \text{s.t.} \quad Dx \leq e, \\ & \qquad x \geq 0 \text{ and integral,} \end{aligned}$$

where $u = (u_1, \ldots, u_m)$ is a vector of Lagrange multipliers.

Now the Lagrangian dual Z_D is defined to be

$$\text{(DP)} \quad Z_D = \max_u Z_D(u).$$

The best choice of u gives the optimal solution to the problem (DP). The value of u can be found by using different methods. The subgradient procedure is a very widely known procedure for updating the values of u. The subgradient method for updating the value of u is presented below. Given an initial value u^0 a sequence $\{u^k\}$ is obtained by the rule

$$u^{k+1} = u^k + t_k(Ax^k - b),$$

where x^k is an optimal solution to $(\mathrm{LR}(u^k))$ and t_k is a positive step size. The step size most commonly used in practice is

$$t_k = \lambda_k(Z^* - Z_D(u^k))/\text{Norm},$$

where Norm $= (Ax^k - b)^2$, $0 < \lambda_k \leq 2$, and Z^* is an upper bound on Z_D obtained by using a suitable heuristic or algorithm.

For CPLP, two different Lagrangian relaxations are possible. The first approach involves relaxing the demand constraints (2) and including them in the objective function with multipliers u_i. This approach was tried by Geoffrion and McBride (1978), Nauss (1978), Christofides and Beasley (1983) and Cornuejols et al. (1991). The second approach involves relaxing the capacity constraints (3) and including them in the objective function with multipliers v_j, and this has been tried by Van Roy (1986), Guignard and Kim (1983) and Cornuejols et al. (1991).

The lower bound at any node of the enumeration tree is obtained by solving either of the two above mentioned Lagrangian Relaxations of the problem. It has been pointed out by Nauss (1978), Guignard and Kim (1983) and Cornuejols et al. (1991) that addition of constraint (6) to the problem greatly improves the bound obtained in the relaxations. Christofides and Beasley (1983) also improve the bound they obtain in the relaxations by including constraints that are weaker than (6). Cornuejols et al. (1991) provide both a theoretical and a computational comparison of the Linear Programming and the Lagrangian Relaxations of the CPLP.

Guignard and Kim (1987) and Barcelo et al. (1991) have tried a new type of relaxation, known as variable splitting. This can be described in general for mixed (or pure) integer programs as follows. A new set of variables $w_{ij} = x_{ij}$ is introduced. Then, (1)–(6) can be rewritten as

$$\min \alpha \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} + \beta \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} w_{ij} + \sum_{j=1}^{n} f_j y_j$$

subject to

$$\sum_{j=1}^{n} w_{ij} = 1, \quad \text{for } i = 1,\dots,m;$$

$$\sum_{i=1}^{m} d_i x_{ij} \le s_j y_j, \quad \text{for } j = 1,\dots,n;$$

$$x_{ij} = w_{ij}, \quad \text{for } i = 1,\dots,m;\ j = 1,\dots,n;$$

$$0 \le x_{ij} \le y_j \le 1, \quad \text{for } i = 1,\dots,m;\ j = 1,\dots,n;$$

$$y_j = \{0, 1\}, \quad \text{for } j = 1,\dots,n;$$

and

$$\sum_{j=1}^{n} s_j y_j \ge \sum_{i=1}^{m} d_i,$$

where α and β are parameters summing to one. A Lagrangian relaxation is obtained by relaxing the constraints $w_{ij} = x_{ij}$. Cornuejols et al. (1991) show that this new relaxation does not provide lower bounds stronger than those already known for CPLP. However, computationally this relaxation may provide good lower bounds in fewer iterations.

As was shown in Section 4, the Lagrangian Relaxations can also be used as a stand-alone hueristic procedure. There are also other methods in the literature for CPLP that can be used either alone or in a branch and bound procedure. These methods are described in Sections 6, 7 and 8.

6. Dual ascent method

Erlenkotter (1978) proposed the dual ascent method for solving the SPLP. This method was also developed independently, around the same time, by Bilde and Krarup (1977). Erlenkotter (1978) adds another procedure called the dual adjustment procedure to do some fine tuning. Computationally this method is excellent for SPLP. This method can be adopted for CPLP with some minor modifications, see Guignard and Spielberg (1979).

A description of the dual ascent procedure for CPLP is presented in this section. Consider the LP relaxation of CPLP and its dual as given below.

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} + \sum_{j=1}^{n} f_j y_j \tag{10}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \forall i;$$

$$-\sum_{i=1}^{m} d_i x_{ij} + s_j y_j \ge 0 \quad \forall j;$$

$$-y_j \ge -1 \quad \forall j;$$

$$y_j \ge 0 \quad \forall j.$$

The dual of the above problem is

$$\max \sum_{i=1}^{m} u_i - \sum_{j=1}^{n} t_j \tag{11}$$

subject to

$$u_i - d_i v_j \le c_{ij} \quad \forall i, j;$$

$$s_j v_j - t_j \le f_j \quad \forall j;$$

$$v_j, t_j \ge 0 \quad \forall j.$$

In the above problem v_j and t_j can be written as

$$v_j = \max\{0, (u_i - c_{ij})/d_i\} \quad \forall j,$$

$$v_j = \{(u_i - c_{ij})/d_i\}^+ \quad \forall j,$$

and

$$t_j \ge s_j v_j - f_j \quad \forall j,$$

$$t_j = (s_j v_j - f_j)^+ \quad \forall j,$$

$$t_j = \left[(s_j/d_i)(u_i - c_{ij})^+ - f_j\right]^+ \quad \forall j.$$

With the above expressions for v_j and t_j the dual problem in the condensed form can be written as

$$\max_u \sum_{i=1}^{m} u_i - \sum_{j=1}^{n} \left[(s_j/d_i)(u - c_{ij})^+ - f_j\right]^+, \tag{12}$$

u_i unrestricted $\forall i$.

If

$$\sum_{j=1}^{n} \left[(s_j/d_i)(u_i - c_{ij})^+ - f_j \right]^+ > 0,$$

then some u_i can be decreased without decreasing the objective function (12). Also, if $u_i < \min_j c_{ij}$, then some u_i can be increased without decreasing the objective function (12). These results provide another condensed dual:

$$\max_u \sum_{i=1}^{m} u_i, \tag{13}$$

$$(s_j/d_i)(u_i - c_{ij})^+ - f_j \le 0 \quad \forall j, \tag{14}$$

$$u_i \ge \min_j c_{ij} \quad \forall i. \tag{15}$$

The dual ascent method develops a set of feasible $\{u_i^t\}$ that is optimal or near optimal for the condensed dual formulation (13)–(15). The procedure begins with any dual-feasible solution $\{u_i^t\}$, say $u_i^t = \min\, c_{ij}^t$, and repeatedly cycles through the demand points i one at a time to increase u_i to the next higher value of c_{ij}. Every time some u_i^t is increased, the corresponding slack variables for constraints (14) are decreased until u_i^t is blocked from increasing by one or more slack variables with values zero. Each increase of u_i^t increases the value of the dual objective function (13). This procedure terminates when no u_i^t can be increased further.

The dual ascent procedure has performed very effectively for SPLP, and an extension of this method to the capacitated version is provided in Guignard and Spielberg (1979).

7. Benders decomposition

The method can be viewed as follows. Consider the mixed integer programming problem (P)

$$\text{(P)} \quad \min_{X \in S} cx, \qquad Ax \ge b,$$

where

$S = \{x = (x_1, x_2) : x_1 \ge 0,\ x_2 \text{ integer}\}$.

Benders decomposition will successively solve the subproblem (SP), or rather its dual, for different values of x_2:

$$\text{(SP)} \quad c^2 x_2 + \min_{x_1 \ge 0} c^1 x_1, \qquad A^1 x_i \ge b - A^2 x_2.$$

At each iteration, a new set of x_2 values is determined by an integer program called the Benders master problem. The constraints of this master problem are generated from the solutions of the dual of (SP) found in the previous iterations. This iterative procedure is repeated until optimality is verified. Geoffrion and Graves (1974) use the Benders decomposition method to solve the multicommodity distribution design problem which is a generalization of CPLP with single source constraints.

8. Cross decomposition

Two methods are used for solving mixed integer programs, one based on primal decomposition which yields a Benders decomposition algorithm, and the other based on dual decomposition which yields the Lagrangian relaxation method. Each of these methods has its own advantages to exploit the structure of primal or dual problems. The cross decomposition approach proposed by Van Roy (1986) exploits simultaneously the structure of both the primal and dual problems. The basic idea underlying cross decomposition, as given by Van Roy (1986) is to use both the subproblems in one single decomposition procedure. The procedure is as follows. Consider

$$\text{(MIP)} \quad \min_{X \in S} cx, \qquad Ax \ge b,$$

where

$S = \left\{x = (x_1, x_2)^T : x_1 \ge 0,\ x_2 \ge 0 \text{ and integer}\right\}$.

The primal subproblem (SP) is defined to be

$$\text{(SP)} \quad c^2 x_2 + \min\ c^1 x_1, \qquad A^1 x_1 \ge b - A^2 x_2, \qquad x_1 \ge 0.$$

And the dual subproblem (SD) is defined to be

$$\text{(SD)} \quad \min_{x \in S} cx + u_2(b_2 - A_2 x),$$
$$A_1 x \geq b_1,$$

where u_2 are the Lagrange multipliers for the constraints of variables x_2. The procedure is given below.

1. *Initialize*. Select initial values u_2^0 for the Lagrangian multipliers and set up the corresponding dual subproblem.
2. *Solve the dual subproblem* (SD). Either stop, or go to Step 4, or set up the primal subproblem corresponding to the optimal solution of the current (SD) and go to Step 3.
3. *Solve the primal subproblem* (SP). Either stop, or go to Step 4, or set up the dual subproblem corresponding to the optimal dual solution of the current (SP) and go to Step 2.
4. *Master problem*. Find new values either for the Lagrange multipliers or the primal variables that are held fixed in (SD) or (SP). Set up the corresponding subproblem and go respectively to Step 2 or 3.

Cross decomposition is efficient and gives better bounds than the earlier reported results. Guignard and Kim (1983) use the approach of Van Roy for the formulation of CPLP strengthened by (6).

9. Reduction tests

In this section, reduction tests are given that have been used in the literature to fix plants as either open or closed. Reduction tests are very useful as they reduce the dimension of the problem, thereby improving the computational efficiency.

The following two tests are for fixing a free plant open.

(1) Let T(J1) represent the optimal value of the transportation problem with the set J1 of open plants. The plants that do not belong to set J1 are assumed to be closed. Then set $y_j = 1$, $j \in$ J0 if

$$T[J - \mathrm{J0} - \{j\}] - T[J - \mathrm{J0}] \geq f_j.$$

That is, if the increase in the transportation costs by not including the plant j is greater than or equal to its fixed cost, then fix plant j open.

(2) Another reduction test that is computationally less expensive than the one given above is as follows. Define

$$\lambda_{ij} = \min_{k \in J - \mathrm{J0} - \{j\}} (c_{ik}),$$

that is, λ_{ij} is the minimum transportation cost of customer i served by plants other than j. Let

$$Z_j = \max \sum_i (\lambda_{ij} - c_{ij}) x_{ij}$$

subject to

$$\sum_i d_i x_{ij} \leq s_j,$$

$$0 \leq x_{ij} \leq 1 \quad \text{for all } i.$$

If $Z_j \geq f_j$, then $y_j = 1$.

The following two tests are for fixing plants closed.

(1) Set $y_j = 0$ for $j \in$ J0 if

$$T(\mathrm{J1}) - T(\mathrm{J1} \cup \{j\}) \leq f_j.$$

That is, if the savings obtained in the transportation costs by including j is less than or equal to its fixed cost, then fix plant j closed.

(2) Another reduction test that is computationally less expensive is as follows. Define

$$\lambda_{ij} = u_i - c_{ij},$$

where u_i's are the dual variables associated with the demand constraints. Let

$$Z_j = \max \sum_i \lambda_{ij} x_{ij}$$

subject to

$$\sum_i d_i x_{ij} \leq s_j,$$

$$0 \leq x_{ij} \leq 1 \quad \text{for all i.}$$

If $Z_j \leq f_j$, then $y_j = 0$.

The above reduction tests have been used by various authors to improve the efficiency of their procedures.

11. Conclusion

In this paper both the heuristic and exact solution methods for capacitated plant location problems were surveyed. As is evident from this paper, a number of solution methods have been tried by various authors. An interesting feature of CPLP is that the structure of this problem has enabled innovative solution methods being proposed for solving it, notably the concept of a Lagrangian heuristic and the concept of variable splitting. As was shown in the introduction, a number of decision problems can be obtained as special cases of CPLP and hence can be solved using the techniques described in this survey.

References

Akinc, U., and Khumawala, M. (1977), "An efficient branch and bound algorithm for the capacitated warehouse location problem", *Management Science* 23/6, 585–594.

Baker, B.M. (1982), "Linear relaxations of the capacitated warehouse location problem", *Journal of the Operational Research Society* 33, 475–479.

Barcelo, J., Fernandex, E., and Jörnsten, K.O. (1991), "Computational results from a new Lagrangian relaxation algorithm for the capacitated plant location problem", *European Journal of Operational Research* 53, 38–45.

Bilde, O., and Krarup, J. (1977), "Sharp lower bounds and efficient algorithms for the simple plant location problem", *Annals of Discrete Mathematics* 1, 79–97.

Christofides, N., and Beasley, J.E. (1983), "An algorithm for the capacitated warehouse location problem", *Journal of the Operations Research Society* 12/1, 19–28.

Cooper, L. (1963), "Location–allocation problems", *Operations Research* 11, 331–343.

Cornueojols, G., Fisher, M.L., and Nemhauser, G.L. (1977), "Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms", *Management Science*, 789–810.

Cornuejols, G., Sridharan, R., and Thizy, J.M. (1991), "A comparison of heuristics and relaxations for the capacitated plant location problem", *European Journal of Operational Research*, 50, 280–297.

Davis, P.R., and Ray, T.L. (1969), "A branch and bound algorithm for the capacitated facilities location problem", *Naval Research Logistics Quarterly* 16.

Domschke, W., and Drexl, A. (1985), "ADD-heuristics' starting procedures for capacitated plant location models", *European Journal of Operational Research* 21, 47–53.

Ellwein, L.B., and Gray, P. (1971), "Solving Fixed Charge Location–Allocation Problem with capacity and configuration constraints", *AIIE Transactions* 3/4, 290–98.

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

Feldman, E., Lehrer, F.A., and Ray, T.L. (1966), "Warehouse locations under continuous economies of scale", *Management Science* 2.

Geoffrion, A.M., and Graves, G.W. (1974), "Multicommodity distribution system design by Benders decomposition", *Mathematical Programming* 2, 82–114.

Geoffrion, A.M., and McBride, R. (1978), "Lagrangean relaxation applied to capacitated facility location problems", *AIIE Transactions,* 40–47.

Guignard, M., and Spielberg, K. (1979), "A direct dual method for the mixed plant location problem with some side constraints", *Mathematical Programming* 17/2, 198–228.

Guignard, M., and Kim, S. (1983), "A strong Lagrangian relaxation for capacitated plant location problems", University of Pennsylvania, The Wharton School, Dept. of Statistics, Technical Report No. 56.

Guignard, M., and Kim, S. (1987), "Lagrangian decomposition: A model yielding stronger bounds", *Mathematical Programming* 39, 215–228.

Jacobson, S.K. (1983), "Heuristics for the capacitated plant location model", *European Journal of Operational Research* 12, 253–261.

Khumawala, B.M. (1974), "An efficient heuristic procedure for the capacitated warehouse location problem", *Naval Research Logistics Quarterly* 21/4, 609–623.

Kuehn, A.A., and Hamburger, M.J. (1963), "A heuristic program for locating warehouses", *Management Science* 9, 643–666.

Nagelhout, R.V., and Thompson, G.L. (1981), "A cost operator approach to multistage location–allocation", *European Journal of Operational Research* 6, 149–161.

Nauss, R.M. (1978), "An improved algorithm for the capacitated facility location problem", *Journal of the Operational Research Society*, 1195–1201.

Rapp, Y. (1962), "Planning of exchange locations and boundaries", *Ericsson Technics* 2, 1–22.

Sa, G. (1969), "Branch and bound and approximate solutions to the capacitated plant location problem", *Operations Research* 17/6, 1005–1016.

Teitz, M.B., and Bart, P. (1968), "Heuristic methods for estimating the generalized vertex median of weighted graph", *Operations Research* 16/5, 955–961.

Van Roy, T.J., and Erlenkotter, D. (1982), "Dual-based procedure for dynamic facility location", *Management Science* 28, 10.

Van Roy, T.J. (1986), "A cross decomposition algorithm for capacitated facility location", *Operations Research* 34, 145–163.