AgEcon SEARCH
RESEARCH IN AGRICULTURAL & APPLIED ECONOMICS

*The World's Largest Open Access Agricultural & Applied Economics Digital Library*

# This document is discoverable and free to researchers across the globe due to the work of AgEcon Search.

# Help ensure our sustainability.

Give to AgEcon Search

AgEcon Search
http://ageconsearch.umn.edu
aesearch@umn.edu

*Papers downloaded from **AgEcon Search** may be used for non-commercial purposes and personal study only. No other use, including posting to another Internet site, is permitted without permission from the copyright owner (not AgEcon Search), or as allowed under the provisions of Fair Use, U.S. Copyright Act, Title 17 U.S.C.*

MONASH 7/93

MONASH
UNIVERSITY

AUSTRALIA

LIBRARY
DEC 13 1993
WITHDRAWN
GIANNINI FOUNDATION OF
AGRICULTURAL ECONOMICS

# THE CAPACITATED PLANT LOCATION PROBLEMS: A SURVEY

R. Sridharan

Working Paper No. 7/93

July 1993

DEPARTMENT OF ECONOMETRICS

ISSN 1032-3813

ISBN 0 7326 0375 7

# THE CAPACITATED PLANT LOCATION PROBLEMS: A SURVEY

R. Sridharan

Working Paper No. 7/93

July 1993

DEPARTMENT OF ECONOMETRICS

MONASH UNIVERSITY, CLAYTON, VICTORIA 3168, AUSTRALIA.

SR12

# THE CAPACITATED PLANT LOCATION PROBLEMS: A SURVEY

R. Sridharan
Department of Econometrics
Monash University, Clayton, Victoria, 3168, Australia

**Abstract**

In this paper we provide a comprehensive survey of a class of plant location problems. We cover both heuristic and exact procedures that have appeared in the literature. The survey also covers three decades of work on the capacitated plant location problem.

## 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. We have to satisfy the demand constraints 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 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. We have 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 we have described above belong to the same category in the following sense. In all these cases the decision involves two stages. In the first stage we make a choice of the subset of machines (or trucks or concentrators or bins). In the second stage we assign the parts (or clients or terminals or items) to the chosen subset of machines (or trucks or concentrators or bins).

The Capacitated Plant Location Problem (CPLP) also belongs to the above category in terms of the underlying decision problem. Here, we have a set of potential locations for plants with fixed costs and capacities, and 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, we can identify the two stages in the decision process for CPLP. In the first stage we make a choice of the subset of the plants to be opened and in the second stage we make the assignment of the customers to these plants.

When we make an additional restriction on CPLP that each customer be served only from a single plant we get the Capacitated Plant Location Problem with Single Source constraints (CPLPSS). It is immediate to see that CPLPSS also belongs to the category described above in terms of the underlying decision problem.

In this paper we will study the Capacitated Plant Location Problem and the Capacitated Plant Location Problem with Single Source constraints. We can show that all the other problems described above are either special cases of CPLP and 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). For a survey of the SPLP the reader is referred to Cornuejols, Nemhauser, and Wolsey [9], Krarup and Pruzan [39], and Thizy, Van Wassenhove and Khumawala [64].

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

$$\text{(P)} \qquad Z = \min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} + \sum_{i=1}^{n} f_j y_j \tag{1.1}$$

subject to

$$\sum_{i=1}^{m} x_{ij} = 1, \; i = 1, \ldots, m; \tag{1.2}$$

$$\sum_{j=1}^{m} d_i x_{ij} \leq s_j y_j \,, \; j = 1, \ldots, n; \tag{1.3}$$

$$0 \leq x_{ij} \leq y_j \leq 1, \; i = 1, \ldots, m; \quad j = 1, \ldots, n; \tag{1.4}$$

$$y_j = \{0,1\} \quad j = 1, \ldots, n; \tag{1.5}$$

and

$$\sum_{j=1}^{n} s_j y_j \geq \sum_{i=1}^{m} d_i \tag{1.6}$$

where $c_{ij}$ = the total cost of transportation from plant j to serve customer i;

$d_i$ = the demand of customer i;

$s_j$ = the capacity of plant j;

$f_j$ = the fixed cost associated with plant j;

$x_{ij}$ = 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, ..., m} : set of customers;

J = {1, ..., n} : 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 (1.2) guarantee that the demand of every client is satisfied, and constraints (1.3) 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 (1.6) 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 (1.3) over all the plants and using the equalities (1.2), and hence is a surrogate constraint. It is redundant in the original formulation, but it strengthens some of the relaxations. Also, the constraints $x_{ij} \le y_j$ are redundant in the original formulation, but are very useful in relaxations. To see that $x_{ij} \le y_j$ can be derived from the other constraints, note that (1.2) and $x_{ij} \ge 0$ imply that $x_{ij} \le 1$ and using (1.3) and (1.5) we get the constraint $x_{ij} \le y_j$.

Several variations of CPLP have been studied in the literature. One such variation requires that each customer is served from only one plant. In practice, this may allow the plants to keep a closer tab on supplies and demands. This problem is known as the 'single source problem and is discussed by several authors, see for example Nagelhout [46], Mulvey and Beck [45], Klincewicz and Luss [37], Barcelo and Casanovas [3], Neebe and Rao [51], DeMaio and Roveda [12] and Srinivasan and Thompson [62].

When one considers intermediate distribution facilities between plants and customers we get the multistage plant location problem. A cost operator method to solve the multistage problem is given in Nagelhout and Thompson [48]. If several commodities are produced at these plants, we have the multicommodity multistage location-allocation problem. Geoffrion and Graves [22] provide a solution technique based on the Benders decomposition method for this problem.

The problems we have described so far are all single period models. When the location decisions occur over time, the problem becomes a multiperiod location-allocation one. This problem has been studied by Eschenbach and Carlson [17], Elshaieb [14], and Van Roy and Erlenkotter [66].

Dearing and Newruck [11] introduced another variation of the problem which they called the bottleneck facility location problem (BFLP). BFLP has the same constraint set 1.2 to 1.5 above, but 1.1 is replaced by

$$\text{Min } [\text{Max } \{c_{ij}x_{ij}\} + a \sum_{j=1}^{n} f_j y_j] \qquad (1.1a)$$

The objective 1.1a is to minimize the bottleneck cost plus some fraction $\alpha$ $(0 \le \alpha \le 1)$, of the fixed costs.

Another related work in this area ia that of Jandy [29] on the fixed charge CPLP.

The capacitated plant location problem arises as a subproblem in several important applications, such as network design problems. For example, in their paper, Kochman and McCallum [38] propose facility location models for the problem of determining how to optimally serve the requirements for communication circuits between the United States and various European and Middle Eastern countries. Given a projection of future requirements, the problem is to plan for the economic growth of a communications network to satisfy these requirements. Both satellite and submarine facilities can be used. The objective is to find an optimal placement of cables (type, location, and timing) and the routing of individual circuits between demand points (over both satellites and cables) such that the total discounted cost over a T-period horizon is minimized. This problem was formulated as a multiperiod, capacitated facility location problem.

We will restrict our attention in this thesis to problems with linear economies of scale. For problems with nonlinear economies of scale the reader is referred to Kelly and Khumawala [33], Khumawala and Kelly [34], and Soland [59].

In the following sections we will survey the various approaches used in the literature to solve the problems CPLP and CPLPSS.

## 3. Preview of Solution Methods for CPLP

The literature on CPLP is very rich; see Magnanti and Wong [42], Francis and Goldstein [21], Salkin [57], and Wong [70] for bibliographies on CPLP. 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, one needs heuristics 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 [40]. Feldmand, Lehrer and Ray [18] propose a heuristic for SPLP called the DROP procedure. Both ADD and DROP procedures are <u>greedy</u> heuristics. For CPLP, ADD and DROP heuristics were tested by Khumawala [35], Jacobsen [28] and Domschke and Drexl [13] among others. We give a detailed description of these heuristics in the following sections and in chapter 3.

At the end of ADD or DROP procedures, one may still be able to improve the solution by making some perturbation. A typical example of such a method is the bump and shift routine of Kuehn and Hamburger [40] for SPLP. Methods of this type are referred to as <u>interchange</u> heuristics and are studied in references [28] and [35] 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 [1]. When heuristics are used, it is useful to have an upper bound on the gap between the heuristic and the optimal values. Corneujols, Fisher and Nemhauser [8] give bounds for the greedy and interchange heuristics for the uncapacitated version of the problem. Nemhauser, Wolsey and Fisher [52] generalize these bounds for submodular set functions. Therefore they are also valid for the capacitated plant location problem since its objective function is a submodular set function defined on the set J.

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 [56], and Davis and Ray [10]. Subsequently many branch and bound based algorithms for CPLP have been developed, see Akinc and Khumawala [1], Christofides and Beasley [6], Ellwein and Gray [15], Geoffrion and McBride [24], Nauss [49], Van Roy [67] and Nagelhout and Thompson [48].

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 [16], Van Roy and Erlenkotter [66], Akinc and Khumawala [1], Sa [56], Davis and Ray [10], Ellwein and Gray [15], and Geoffrion and Graves [22], or a Lagrangian Relaxation, Geoffrion and McBride [24], Nauss [50], Christofides and Beasley [6], and Van Roy [65]. Geoffrion and Graves [22] use the Benders decomposition procedure and Van Roy [65] uses the Cross decomposition approach along with the Lagrangian relaxation. We will look at these methods more closely in the following sections.

Nagelhout and Thompson [48] use an implicit enumeration approach that does not use any relaxation of the problem. Instead they use some lower bounds obtained from the submodular property of the objective function, together with some fathoming rules, to implicitly enumerate the solutions to the problem P. The movement in the search tree from one feasible solution to another is facilitated by applying cost operators, (for a description of the cost operator technique refer to Srinivasan and Thompson [60, 61], to P. The cost operators are used to fix plants open or closed which helps in pruning the search tree.

## 4. Heuristics for CPLP

Heuristics can normally handle large problems (with serveral 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.

### 4.1 The Greedy Heuristics

There are two different greedy heuristics for CPLP, namely, the ADD procedures and the DROP procedures. We will look at these two procedures in this section. For ease of presentation we will denote by

$J0$ : the subset of J for which $y_j = 0$

$J_\phi$ : the subset of J for which $y_j$ is yet undecided

$J2$ : 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 [40] to solve the uncapacitated version of the problem. The extension of this procedure to the capacitated version is given in Jacobsen [28]. Here we give a formal description of the ADD procedure. 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 (1.1) to (1.5).

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

1. For each $j \in J_\phi$, compute the savings

$$\sigma_j = T^*(J1, I) - T^*(J1 \cup \{j\}, I) - f_j.$$

2. Now identify the plant j* that gives the maximum savings $\sigma_{j*}$ from, $\sigma_{j*} = \max_{j \in J\phi}\{ \sigma_j \}$.

3. If the savings is positive, that is, $\sigma_{j*} > 0$, 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 $| J\phi |$ transportation problems at each iteration. However, Khumawala [35], and Jacobsen [28] 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 [35] (also see Akinc and Khumawala [1] is a lower bound on the savings, LBS, and is given below.

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

Then,

$$LBS_j = \max \Sigma_i K_{ij} \gamma_i$$

subject to

$$\Sigma_i d_i \gamma_i \le s_j$$

$$0 \le \gamma_i \le 1.$$

Khumawala [35] observes that the bound $LBS_j$ is very effective.

Jacobsen [28] provides both a lower and an upper bound on the savings. He calls the heuristic that uses the lower bound ADD-LO and the one that uses the upper bound ADD-HI. Domschke and Drexl [13] also use the same bounds. The lower bound on the savings obtained by adding plant j* is given by $\Sigma_i \Sigma_{j \in J1} x_{ij} \max(0, c_{ij} - c_{ij*})$, 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(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 Feldmand, Lehrer and Ray [18] for the uncapacitated plant location problem. 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 $j \in J1$, compute the sayings

   $$\sigma_j = f_j + T^*(J1,I) - T^*(J1 \setminus \{j\},I).$$

2. Find the plant j* that gives the maximum savings

   $$\sigma_{j*} = \max_{j \in J1} \{ \sigma_j \}.$$

3. If $\sigma_{j*} > 0$, transfer i* 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 $|J\sigma|$ transportation problems in each iteration. But, as before, Akinc and Khumawala [1], and Jacobsen [28] suggest procedures that provide a bound on the savings with much less computational effort. The bound computed by Akinc and Khumawala [1] is as follows. Let

$v_i$ $(i = 1,\ldots,m)$, and $u_r$, $r \in J1$ be the optimum dual variables associated with the solution $T^*(J1\setminus\{j\},I)$. Define,

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

Then, the savings obtained by closing plant j is,

$$UB\Omega_j = \max \Sigma_i w_{ij}\delta_i$$

$$\text{subject to}$$

$$\Sigma_i d_i \delta_i \le s_j,\ j \in J$$

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

As for the ADD procedure, Jacobsen [28] provides both a lower bound and an upper bound on the savings. The DROP procedure that uses a lower bound is called DROP-LO, and the one that uses the upper bound is called DROP-HI. For more details on the actual computation of these bounds refer to Jacobsen [28].

### 4.2 Interchange Heuristics

The ADD and DROP procedures that we looked at belong to the category of greedy heuristics where once a decision is made it is not changed. Actually we may be able to improve on the greedy solution by making some changes in the solution. The heuristics that attempt to make such improvements are referred to as the Interchange Heuristics. We will look at two different methods that belong to the interchange heuristic category, (i) the Alternate Location Allocation (ALA), and (ii) the Vertex Substitution Method (VSM).

### 4.2.1 Alternate Location Allocation

The earlier application of this method to location problems appear in Rapp [53], and Cooper [7]. The general outline of this method can be described as follows.

1. Start with a set of open plants, J1. All other plants are in J$\phi$. Set JT = $\phi$. (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$\phi$ = J - J1, J0 = $\phi$. Set JT = $\phi$ and go to step 2. Otherwise, set J1, J0 and J$\phi$ 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 [63], and Cornuejols, Fisher and Nemhauser [8]. This procedure can be described as follows.

1. Start with a feasible solution. List J0, the set of closed plants. Set JT = $\phi$.

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 = $\phi$ 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 [28].

## 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, we have solved P; otherwise we obtain a lower bound for P. Various relaxations that have used for CPLP are described in this section.

There are many papers that use a branch and bound algorithm for CPLP, see, for instance, Sa [56], Davis and Ray [10], Ellwein and Gray [15], Akinc and Khumawala [1], Geoffrion and McBride [24], Nauss [50], Christofides and Beasley [6], and Van Roy [68]. 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 $T_j$ 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. We will look at these relaxations now.

### 5.1 Linear Programming Relaxation

Davis and Ray [10], Sa [56], Ellwein and Gray [15], and Akinc and Khumawala [1] 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 at 0 or 1. Sa [56], Ellwein and Gray [15], and Akinc and Khumawala [1] 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 [10] use the Strong Linear Programming relaxa6tion (SLP), that is, problem (1.1) - (1.5) with the integer constraints relaxed, in their procedure. As we will see in the next chapter the lower bound obtained from SLP is usually stronger than the one obtained from WLP. However, Akinc and Khumawala [1] 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 [2] also shows that the LP relaxation is strengthened by adding the disaggregated constraints of the type

$$\frac{\Sigma_i a_{ij} x_{ij}}{\Sigma_i a_{ij} \theta_{ij}} \le y_j$$

where $<a_{ij}>$ are arbitrary coefficients, and $<\theta_{ij}>$ are optimal values of the variables $x_{ij}$ in the solution of the continuous knapsack problem:

$$\max \{\Sigma_i a_{ij} : 0 \le x_{ij} \le u_{ij}, \Sigma_i x_{ij} \le s_j\}$$

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

Baker [2] also gives some special cases where the constraints reduce to the type already known in the literature.

### 5.4.2 Lagrangian Relaxation

The Lagrangian Relation is an approach used for solving the mixed integer and pure integer programming problems, see Fisher [19], and Geoffrion [23] for a very nice description of this technique. We will illustrate this approach by using the following problem as an example. Let

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

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

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

$$Z_D(u) = \min cs + u(Ax - b) \qquad \text{(LR(u))}$$

$$Dx \le e$$

$$x \ge 0 \text{ and integral}$$

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

Now the Lagrangian dual $Z_D$ is defined to be

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

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. Details of subgradient procedure are explained in chapters 2 and 5. We will now present the subgradient method for updating the value of u. Given an initial value $u^o$ 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 $LR(u^k)$ 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 $\text{Norm} = (Ax^k - b)^2$

and $0 < \lambda_k \le 2$

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 (1.2) and including them in the objective function with multipliers $u_i$. This approach was tried by Geoffrion and McBride [24], Nauss [50], and Christofides and Beasley [6]. The second approach involves relaxing the capacity constraints (1.3) and including them in the objective function with multipliers $v_j$, and this has been tried by Van Roy [68], and Guignard and Kim [26].

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 [50], and Guignard and Kim [27] that addition of constraint (1.6) to the problem greatly improves the bound obtained in the relaxations. Christofides and Beasley [6] also improve the bound they obtain in the relaxations by including constraints that are weaker than (1.6). We give the relationship among all these relaxations in the next chapter.

The Lagrangian relaxation can also be used to do a parametric analysis of the problem. Such an analysis is possible by observing that each time a Lagrangian calculation is made for a specific problem with a given capacity vector, an optimal solution is obtained for a related problem with a suitably adjusted capacity vector. Bitran et al [5] provide such an analysis for CPLP.

We now present some methods that can be used either alone or in conjunction with a branch and bound algorithm.

## 6. Dual Ascent Method

Erlenkotter [16] proposed the dual ascent method for solving the SPLP. This method was also developed independently, around the same time, by Bilde and Krarup [4]. Erlenkotter [16] 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 [26].

We will describe the dual ascent procedure for CPLP in this section. Consider the LP relaxation of CPLP and its dual as given below.

$$\min \Sigma_i \Sigma_j c_{ij} x_{ij} + \Sigma_j f_j y_j \tag{1.10}$$

subject to

$$\Sigma_j x_{ij} = 1 \ \forall\ i$$

$$-\Sigma_i d_i x_{ij} + s_j y_j \geq 0 \ \forall\ j$$

$$-y_j \geq -1 \ \forall\ j$$

$$y_j \geq 0 \ \forall\ j$$

The dual of the above problem is

$$\max \Sigma_i u_i - \Sigma_j t_j \tag{1.11}$$

subject to

$$u_i - d_i v_j \leq c_{ij} \ \forall\ i,j$$

$$s_j v_j - t_j \leq f_j \ \forall\ j$$

$$v_j, t_j \geq 0 \ \forall\ j.$$

In the above problem we can write $v_j$ and $t_j$ as,

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

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

and

$$t_j \geq s_j v_j - f_j \ \forall\ j$$

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

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

With the above expressions for $v_j$ and $t_j$ we can write the dual problem in the condensed form as

$$\max_u \Sigma_i u_i - \Sigma_j [(s_j/d_i)(u - c_{ij})^+ - f_j]^+ \qquad (1.12)$$

$$u_i \ \text{unrestricted} \ \forall\ i$$

If $\Sigma_j[(s_j/d_i)(u_i - c_{ij})^+ - f_j]^+ > 0$, then some $u_i$ can be decreased without decreasing the objective function (1.12). Also, if $u_i < \min_j c_{ij}$ then some $u_i$ can be increased without decreasing the objective function (1.12). These results provide another condensed dual

$$\max_u \Sigma_i i_i \qquad (1.13)$$

$$(s_j/d_i)(u_i - c_{ij})^+ - f_j \leq 0 \ \forall\ j \qquad (1.14)$$

$$u_i \geq \min_j c_{ij} \ \forall\ i \qquad (1.15)$$

The dual ascent method develops a set of feasible $\{u_i^t\}$ that is optimal or near optimal for the condensed dual formulation (1.13) - (1.15). 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 (1.14) 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 (1.13). 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 [26].

## 7. Benders Decomposition

Benders decomposition is described fully in Magnanti and Wong [43]. The method can be viewed as follows. Consider the mixed integer programming problem (P)

$$\text{(P)} \qquad \text{Min}_{x \in S}\ cx$$

$$Ax \geq b$$

$$\text{where} \qquad S = \{x = (x_i, x_2) : x_1 \geq 0,\ x_2 \text{ integer}\}$$

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

$$\text{(SP)} \qquad c^2 x_2 + \text{Min}_{x_1 \geq 0}\ c^1 x_1$$

$$A^1 x_i \geq b - A^2 x_2$$

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 [22] 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 [65, 67, 69] exploits simultaneously the structure of both the primal and dual problems.

The basic idea underlying cross decomposition, as given by Van Roy [67, 69] is to use both the subproblems in one single decomposition procedure. The procedure is as follows, Van Roy [67]. Consider

$$\text{Min}_{x \in S}\ cx \qquad \text{(MIP)}$$

$$Ax \geq b,$$

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

We define the primal subproblem (SP) to be

$$c^2x_2 + \min c^1x_1 \qquad \text{(SP)}$$

$$A^1x_1 \geq b - A^2x_2$$

$$x_1 \geq 0$$

And we define the dual subproblem (SD) to be

$$\min_{x \in S}\ cx + u_2(b_2 - A_2x) \qquad \text{(SD)}$$

$$A_1x \geq b_1.$$

where $u_2$ are the Lagrange multipliers for the constraints of variables $x_2$. We now give the procedure.

* Step 1: Initialize. Select initial values $u_2^o$ for the Lagrangian multipliers and set up the corresponding dual subproblem.

* Step 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.
* Step 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.
* Step 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 very fast and also gives better bounds than the earlier reported results. Guignard-Spielberg and Kim [27] use the approach of Van Roy for the formulation of CPLP strengthened by (1.6). If we do not obtain the optimal solution after applying the cross decomposition method to the problem, then the optimal solution can be found by using a branch and bound algorithm.

## 9. Reduction Tests

In this section, we give some reduction tests that have been used in the literature to fix plants as either open or closed. 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 we set $y_j = 1$, $j \in J\phi$ if

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

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 we fix plant j open.

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

$$\text{define } \lambda_{ij} = \min_{k \in J\phi - J0 - \{j\}} (c_{ik})$$

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

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

subject to

$$\Sigma_i d_i x_{ij} \le s_j$$

$$0 \le x_{ij} \le 1, \text{ for all i.}$$

If $Z_j \ge f_j$ then $y_j = 1$.

We now give the following two tests for fixing plants closed.

1. We set $y_j = 0$ for $j \in J\phi$ if

$$T(J1) - T(J12 \cup \{j\} \le f_j.$$

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

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

$$\text{define } \lambda_{ij} = u_i - c_{ij}$$

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

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

subject to

$$\Sigma_i d_i x_{ij} \le s_j$$

$$0 \le x_{ij} \le 1, \text{ for all } i.$$

If $Z_j \le f_j$ then $y_j = 1$.

## 10. The CPLP with Single Source Constraints

The single source constraints specify that each customer is served from a single plant. This problem has exactly the same formulation as CPLP except for the integrality constraints on the $x_{ij}$ variables. The single source problem was first formulated by Nagelhout and Thompson [47]. This problem can be formulated as follows.

(CPLPSS)

$$Z = \min \Sigma_i \Sigma_j c_{ij} x_{ij} + \Sigma_j f_j y_j$$

subject to

$$\Sigma_j x_{ij} = 1, \text{ for all } i \qquad (1.16)$$

$$\Sigma_i d_i x_{ij} \le s_j y_j, \text{ for all } j \qquad (1.17)$$

$$x_{ij}, y_j \in \{0,1\}, \text{ for all } i,j \qquad (1.18)$$

Unlike CPLP, this problem has not been studied very extensively in the literature. CPLPSS is also referred to as the fixed-charge assigning users to sources problem, see Neebe and Rao [51].

There are many well known problems that are actually simplifications of CPLPSS. If, for example, $c_{ij} = 0$ for all i.j, then we get the Generalized Bin Packing Problem (GBPP). Lewis and Parker [41] provide an algorithm to solve GBPP. Further, if we set $f_j = 1$ and $s_j = s$ for all j along with $c_{ij} = 0$ in CPLPSS then we get the Bin Packing Problem (BPP). BPP has been studied by Gilmore and Gomory [25], Johnson [30, 31], Yao [71], Karmarkar and Karp [32] among others. When we set all $f_j = 0$ in CPLPSS, we have the problem of assigning sources to uses. This problem has been studied by Sandi [58], DeMaio and Revada [12], Srinivasan and Thompson [62], and Fisk [20]. Yet another variation of CPLPSS is to set all $f_j = 0$ and replace $d_i$ by $d_{ij}$ giving rise to the Generalized Assignment Problem (GAP). GAP has been studied very widely, and some references are Klastorin [36], Ross and Soland [54], and Ross and Zoltners [55].

When we take $d_i = 1$, for all i, and $s_j = k$, for all j, and include a plant $y_0$ with capacity m, we get the Star-Star Concentrator Location Problem (SSCLP). Mirzain [44] provides an approximate algorithm based on a Lagrangian relaxation for SSCLP.

A model for the Capacitated Clustering Problem (CCP) can also be obtained from CPLPSS by setting $f_j = 0$ for all j. When we want p mutually exclusive and collectively exhaustive clusters then the constraint $\Sigma_j y_j = p$ is added to the formulation of CCP. This problem has been formulated and solved by Mulvey and Beck [45].

There are both exact and heuristic methods to solve CPLPSS in the literature. Klincewicz and Luss [37] and Barcelo and Casanovas [3] propose Lagrangian heuristics for the problem. Klincewicz and Luss [37] consider a Lagrangian relaxation where they dualize the capacity constraints while Barcelo and Casanovas [3] consider a relaxation where they dualize the demand constraints. We will describe briefly these two heuristics and the branch and bound method due to Neebe and Rao [51] in the following paragraphs.

Klincewicz and Luss [37] include the capacity constraints in the objective function by the use of Lagrange multipliers thereby obtaining an uncapacitated plant location problem as a subproblem. They solve these subproblems using the dual ascent algorithm of Erlenkotter [16]. They obtain an initial solution by an add heuristic and they also use a final adjustment heuristic to improve the soltuion after the Lagrangian iterations are completed. A brief description of the three phases of their heuristic, the initial add heuristic, the Lagrangian phase, and the final adjustment heuristic is given below.

The initial add heuristic has two stages. In the first stage plants are added one at a time to a set of open plants (this set is empty to start with). The plant to be added next is identified as the one that provides the maximum savings, the savings being a bound on the reduction in the transportation cost minus the fixed cost. An adjustment is made to reflect the capacity restriction. When the total capacity of the open plants exceeds the total demand the second stage is invoked wherein the customers are assigned to plants.

In stage two, for each customer, they compute the cost differential between its best and second best assignment in the set of open plants. Then the customers are ordered in decreasing cost differential. Then for each customer in order, they assign the open plant with minimum assignment cost among those with sufficient remaining capacity. If there is no feasible assignment, then they go back to stage one to open more plants. The procedure stops when all the customers are assigned to plants.

The Lagrangian phase computes a lower bound for the problem. At each iteration of the procedure a simple plant location problem is solved.

The Lagrange multipliers are updated using the subgradient procedure. The upper bound is initialized by the add heuristic value and whenever the simple plant location problem is feasible for CPLPSS the upper bound is updated, if necessary. The procedure stops if (i) the number of iterations has exceeded a prespecified limit, or (ii) the upper bound is less than or equal to 1.001 times the lower bound.

Phase three, the final adjustment heuristic, is invoked only when the Lagrangian phase provides a better upper bound. In this phase, for each customer the cost differential between the best and the second best assignment in the set of open plants is computed. Then the customers are ordered on decreasing cost differential. Then for each customer in order, alternate plant assignments are examined. If a customer can be feasibly moved to another open plant with a lower assignment, then they go back to stage one to open more plants. The procedure stops when all the customers are assigned to plants.

The Lagrangian phase computes a lower bound for the problem. At each iteration of the procedure a simple plant location problem is solved. The Lagrange multipliers are updated using the subgradient procedure. The upper bound is initialized by the add heuristic value and whenever the simple plant location problem is feasible for CPLPSS the upper bound is updated, if necessary. The procedure stops if (i) the number of iterations has exceeded a prespecified limit, or (ii) the upper bound is less than or equal to 1.001 times the lower bound.

Phase three, the final adjustment heuristic, is invoked only when the Lagrangian phase provides a better upper bound. In this phase, for each customer the cost differential between the best and the second best assignment in the set of open plants is computed. Then the

customers are ordered on decreasing cost differential. Then for each customer in order, alternate plant assignments are examined. If a customer can be feasibly moved to another open plant with a lower assignment cost then a reassignment is made. This phase is repeated until no reassignments are possible.

Barcelo and Casanovas [3] propose a Lagrangian relaxation heuristic where they dualize the demand constraints. Their heuristic consists of two stages (i) plant selection, and (ii) assignment. These two stages are described below.

The plant $j_k$ to be selected into the set of open plants in iteration k is found such that $\rho_{j_k}(u^{k+1})$ satisfies

$$\rho_{j_k}(u^{k+1}) = \min_{j \in J-J1}\{\rho_j(u^{k+1})\}$$

where $u^{k+1}$ are the Lagrange multipliers of (1.16) in iteration k+1 and

$$\rho_j(u^{k+1}) = \Sigma_{i \in I}(u^{k+1} - u^k), \ \forall \ j \in J.$$

The Lagrange multipliers $u_i$'s are initialized as

$$u_i^1 = \max_{j \in J} \{c_{ij} + d_i f_j / s_j\}, \ \forall \ i \in I$$

and $u_i^{k+1}$ in iteration k+1 is computed as

$$u_i^{k+1} = u_i^k + \min \{0, c_{ij_k} + d_i f_{j_k} / s_{j_k} - u_i^k\} \ \forall \ i \in I.$$

The plants are added onto the set J1 until the condition $\Sigma_{j \in J1} S_j \geq \Sigma_{i \in I} d_i$ is satisfied. Then either an interchange procedure is used to select an improved list of open plants or the assignment stage is started.

The interchange procedure requires the computation of the reduced cost for each plant; if a plant in J1 has negative reduced cost, then that plant will be replaced by a plant from J - J1 such that the total capacity of this new set of plants is greater than or equal to the total demand. If more than one plant in J1 has a negative reduced cost then the plant with the lowest reduced cost is chosen as the one to be replaced.

The Lagrange multipliers $u_i^k$ are updated as given above and this procedure is repeated until (i) $\forall\ j \in J1$, the reduced cost $w_j \geq 0$, or (ii) $J - J1 = \phi$ or (iii) no feasible interchange exists. The reduced cost $w_j$ is computed to be

$$w_j = f_j - s_j p_j \ \forall\ j \in J$$

where

$$p_j = \max_{i \in I} \{(u_i^k - c_{ij})/d_i\} \ \forall\ j \in J.$$

The assignment stage is very similar to the one used by Klincewicz and Luss [37]. Here also a regret heuristic as described previously is used to assign the customers to the plants.

Neebe and Rao [51] propose a branch and bound method for this problem. They formulate CPLPSS as a set partitioning problem (SPP) and then consider the linear programming relaxation of the SPP at each node of the enumeration tree to obtain bounds. The CPLPSS is reformulated as an SPP as follows. First, for each plant j they define an activity to be any assignment of customers to that plant such that the capacity of that plant is not exceeded. Let $m_j$ be the number of activities corresponding to plant j and let $M_j = \{1, \ldots, m_j\}$, and also let the $k^{th}$ activity of plant j be denoted by $a_j^k$, $k \in M_j$.

Then activity $a_j^k = (a_{i1}^k, \ldots, a_{im}^k)$ is represented by a 0-1 vector $i \in I$, where

$$a_{ij}^k = \begin{cases} 1 \text{ if customer } i \text{ is assigned to plant } j \\ 0 \text{ otherwise} \end{cases}$$

and

$$\Sigma_{i \in I} d_i a_{ij}^k \le s_j \tag{1.19}$$

The cost of non-null activity $a_j^k$ equals

$$c_j^k = \Sigma_{i \in I} c_{ij} a_{ij}^k + f_j$$

and the costs of null activities are zero. Finally, they define the decision variables $x_j^k$, $j \in J$, $k \in M_j$, such that

$$x_j^k = \begin{cases} 1 \text{ if activity } a_j^k \text{ is active} \\ 0 \text{ otherwise} \end{cases}$$

Then CPLPSS can be formulated as

$$\min Z = \Sigma_{j \in J} \Sigma_{k \in M_j} c_j^k x_j^k$$

subject to

$$\Sigma_{j \in J} \Sigma_{k \in M_j} a_{ij}^k x_j^k = 1 \ \forall \ i \in I \tag{1.20}$$

$$\Sigma_{k \in M_j} x_j^k \le 1 \ \forall \ j \in J \tag{1.21}$$

$$x_j^k = 0, 1, \ \forall \ j \in J, \ k \in M_j \tag{1.22}$$

This is a set partitioning problem with a side constraint (1.21) specifying that at most one activity from any plant is utilized.

The relaxation of the program (1.20) - (1.22) is obtained when we relax (1.22) to

$$x_j^k \geq 0 \ \forall \ j \in J, \ k \in M_j. \tag{1.23}$$

The formulation (1.20) - (1.22) of CPLPSS will have an enormous number of columns even for a moderate-size problem. This is overcome by the authors by the use of column generating procedures such as the one used by Gilmore and Gomory [25] for the cutting stock problem.

If $\bar{x}$ solves the LP relaxation of (1.20) - (1.22) then the solution of CPLPSS is obtained as

$$x_{ij}^* \ = \ \Sigma_{k \in M_j} a_{ij}^k x_j^k \forall \ i \in I, \ j \in J.$$

If $\bar{x}$ is all integer, than x* is also all integer and therefore optimal for CPLPSS.

If $\bar{x}$ is not all integer, then a branch and bound procedure that fixes the fractional values in $\bar{x}$ to 0 or 1 is used to find the optimal solution for CPLPSS.

Neebe and Rao [51] observe that the LP relaxation (1.20, (1.21) and (1.23) has a high probability of terminating all integer and hence the branch and bound tree to solve CPLPSS is not too large.

Geoffrion and Graves [22] give a solution procedure based on Benders decomposition for a generalized version of CPLPSS which considers multiple commodities and multiple stages in distribution.

## 11. Conclusion

In this paper we surveyed both the heuristic and exact solution methods for capacitated plant location problems. As was shown in the introduction a number of decision problems can be obtained as special cases of CPLP and CPLPSS 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", *Mgt. Sc.*, 23, 6, 585-594.

Baker, B.M. (1982) "Linear relaxations of the capacitated warehouse location problem", *Jour. of Opnl. Res. Soc.*, 33, 475-479.

Barcelo J. and Casanovas J. (1984) "A heuristic Lagrangean algorithm for the capacitated plant location problem", *European J. of Opnl. Res.* 15 (1984), 212-226.

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.

Bitran, G.R., Chandru Vijaya, Sempolenski, D.E. and Shapiro, J.F. (1981) "Inverse Optimizations: An Application to the Capacitated Plant Location Problem", *Mgt. Sc.*, 1120-41.

Christofides, N. and Beasley, J.E. (1983) "An algorithm for the capacitated warehouse location problem", *European J. Oper. Res.* 12, 1, 19-28.

Cooper, L. (1963), "Location-Allocation Problems", *Opns. Res.* 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", *Mgt. Sc.* 23 (1977), 789-810.

Cornuejols, G., Nemhauser, G.L. and Wolsey, L.A. (1983) The uncapacitated facility location problem. MSRR 493, Grad. Sch. of Ind. Admin., CMU.

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

Dearing, P.M. and Newruck, F.C. (1979) "A capacitated bottleneck facility location problem", *Mgt. Sc.* 25, 11, 1093-1104.

DeMaio, A.O. and Roveda, C.A. (1971) "An all zero-one algorithm for a certain class of transportation problems", *Opns. Res.* 19, 1406-1418.

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

El-Shaieb, A.M. (1973) "A new algorithm for locating sources among destinations", *Mgt. Sc.* 20, 2, 221-233.

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

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

Eschenback, T.G. and Carlson, R.C. (1975) The capacitated multi-period location - allocation problem. 75-27, System Opt. Lab., Stanford Univ.

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

Fisher, M.L. (1981) "The Lagrangean Relaxation Method for Solving Integer Programming Problems", *Mgt. Sc.* 27, 1, 1-18.

Fisk, J. (1975) "A solution procedure for a special class of transportation problem containing multiple choice constraints", School of Business, State Univ. of New york at Albany, 1975.

Francis, R.L. and Goldstein, J.M. (1974) "Location theory: a selective bibliography", *Operations Research* 22, 400-410.

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

Geoffrion, A.M. (1974) "Lagrangian relaxation and its uses in integer programming", *Math. Prog. Study 2*, 82-114.

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

Gilmore, P.C. and Gomory, R.E. (1961) "A linear programming approach to the cutting stock problem", *Operations Research* 9, 849-859.

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

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

Jacobson, S.K. (1983) "Heuristics for the capacitated plant location model", *European J. of Opns. Res.* 12, 253-261.

Jandy, G. (1967) Approximate algorithm for the fixed charge capacitated site location problem. 67-3, O.R. House, Stanford Univ., April.

Johnson, D.S. (1973) *Near-optimal bin packing algorithms.* Ph.D. Th., Massachusetts Institute of Technology.

Johnson, D.S. (1974) "Fast algorithms for bin packing", *Journal of Computer and Systems Sciences*, 8, 272-314.

Karmarkar, N. and Karp, R.M. (1982) "An efficient approximation scheme for the one-dimensional bin packing problem". *Proc. 23rd Annual Symposium on Foundations of Computer Science*, 312-330.

Kelly, D.L. and Khumawala, B.M. (1982) "Capacitated warehouse location with concave costs", *J. of Opnl. Res. Soc.* 33, 817-826.

Khumawala, B.M. and Kelly, D.L. (1974) "Warehouse location with concave costs", *INFOR 12*, 55-65.

Khumawala, B.M., (1974) "An efficient heuristic procedure for the capacitated warehouse location problem", *Naval Res. Log. Quart.* 21, 4, 609-623.

Klastorin, T.D. (1977) An effective subgradient algorithm for the generalized assignment problem. Univ. of Washington.

Klincewicz, J.G. and Luss, H., (1986) "A Lagrangian relaxation algorithm for capacitated facility location with single-source constraints". *J. of Opnl. Res. Soc.*, Vol. 37.

Kochman, G.A. and McCallum, C.J. Jr. (1981) "Facility location models for planning transatlantic communications network", *Euro. J. of Opnl. Res. 6,* 2, 205-211.

Krarup, J. and Pruzan, P.M. (1983) "The simple plant location problem: survey and synthesis", *Euro. J. of Opnl. Res.* 12, 36-81.

Kuehn, A.A., and Hamburger, M.J. (1963) "A heuristic program for locating warehouses", *Mgt. Sc.* 9, 643-666.

Lewis, R.T. and Parker, R.G. (1982) "On a generalized bin packing problem", *Naval Research Logistics Quarterly 29*, 119-145.

Magnanti, T. and Wong, R. (1984) "Network Design and Transportation Planning", *Trans. Sc. 18*, 1-55.

Magnanti, T.L. and Wong, R.T. (1986) "Decomposition methods for facility location problems, *Discrete Location Theory.*

Mirzaian, A. (1985) "Lagrangian relaxation for the star-star concentrator location problem: Approximation algorithm and bounds", *Networks 15*, 1-20.

Mulvey, J.M. and Beck, M.P. (1984) "Solving capacitated clustering problems". *European Journal of Operational Research 18*, 339-348.

Nagelhout, R.V. (1980) *A study of total cost and bottleneck single source transportation and location problems with applications.* Ph.D. Th., Grad. Sch. of Ind. Admn., CMU.

Nagelhout, R.V. and Thompson, G.L. (1980) "A single source transportation algorithm". *J. of Comput. & Ops. Res 7*, 185-198.

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

Nauss, R.M. (1975) An efficient algorithm for the capacitated facility location problem. Univ. of Missouri.

Nauss, R.M. (1978) "An improved algorithm for the capacitated facility location problem". *J. of Opnl. Res. Soc.*, 1195-1201.

Neebe, A.W. and Rao, M.R. (1983) "An algorithm for the fixed-charge assigning users to sources problem". *J. Opl. Res. Soc. 34*, 1107-1113.

Nemhauser, G.L., Wolsey, L.A. and Fisher, M.L. (1978) "An analysis of approximations for maximizing submodular set functions I", *Math. Prof. 14*, 265-294.

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

Ross, G.T. and Soland, R.M. (1975) "A branch and bound algorithm for the generalized assignment problem". *Math. Programming 8,* 91-105.

Ross, G.T. and Zoltners, A.A. (1979) "Weighted assignment models and their applications", *Management Science 25*, 683-696.

Sa, G. (1969) "Branch and bound and approximate solutions to the capacitated plant location problem". *Oper. Res.* 17, 6, 1005-1016.

Salkin, H.M. (1975) *Integer Programming.* Addison-Wesley Publishing Co., Reading, Mass., 1975.

Sandi, C. (1974) Solution of the machine loading problem with binary variables. Pisa, I.B.M. Scientific Center.

Soland, (1974) "Optimal plant location with concave costs". *Operations Research 22*, 373-385.

Srinivasan, V. and G.L. Thompson (1972a) "An operator theory of parametric programming for the transportation problem-I". *Naval Res. Log. Quart. 19*, 205-225.

Srinivasan, V. and G.L. Thompson (1972b) "An operator theory of parametric programming for the transportation problem-II". *Naval Res. Log. Quart. 19*, 227-252.

Srinivasan, V. and Thompson, G.L. (1973) "An algorithm for assigning sources to uses in a special case of transportation problems". *Opns. Res. 21*, 284-295.

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

Thizy, Van Wassenhove, and Khumawala, B.M. (1983) Comparison of exact and approximate methods of solving the uncapacitated plant location problem. 83-15, Dept. of Civil Eng., Princeton Univ.

Van Roy, T.J. (1981) Cross decomposition for mixed integer programming with applications to facility location. *Opns. Res.*, pp.579-587.

Van Roy, T.J. and Erlenkotter, D. (1982) "Dual - based procedure for dynamic facility location". *Mgt. Sc. 28, 1091-1105.*

Van Roy, T.J. (1983) "Cross decomposition for mixed integer programming". *Math Prog. 25* (1983), 46-63.

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

Van Roy, T.J. (1986) Cross decomposition for mixed integer programming with applications to facility location, CORE, University Catholoque de Louvain.

Wong, T.L. (1983) "Recent research on location and network design problems: an annotated bibliography. Proceedings of the summer school on Combinatorial Optimizatioin, National Inst. for Higher Education, Dublin, July, 1983, pp. 4-15.

Yao, A.C. (1980) "New algorithms for bin packing". *J. ACM 27*, 207-227.