Journal of the Korean
OR/MS Society
Vol. 11, No. 1, Apr. 1986

– Research Paper –

Computational Study on the Simple Plant Location Problem: Variations of the Benders Decomposition Method

Yangyul Kim*

Abstract

We investigate various methods of the Benders decoposition algorithm in its application to the simple plant location problem. We developed six variants. The master problem may be relaxed as an LP problem up to an appropriate point in time, or need not be solved to the optimality before a cut is added. Furthermore, since the subproblem is highly degenerated, we can generate more than one cuts at a time.

The efficiency of the methods are examined using a sample problem. The result showed that the adding two-cut method was superior to the standard method. The LP relaxation and the non-optimization of the master program greatly improved the efficiency. Applying the LP relaxation method, we were able to reduce the computing time by two thirds of the time required by the standard method.

1. Introduction

The problem under consideration is the simple or uncapacitated plant location problem. The problem deals with the supply of a single commodity from a set of supply sources (plants or warehouses) to a set of customers (demand centers) with a known demand for the commodity. Plants are assumed to have unlimited capacity so that any plant can satisfy all demands. The elaborated survey paper by Krarup and Pruzan (1983) provides the state-of-the-art of this type of problems.

* Economic Analysis Group KAIST

** This paper is a revised version of the manuscript, written while I was at the Graduate School of Business, University of Chicago. I am indebted to Linus Schrage, Kipp Martin for many valuable discussions and comments. I also wish to express my gratitude to a referee for his kind comments. I alone am responsible for any remaining errors or ambiguities.

Suppose there are n demand centers. The demand of each demand center must be fully met from one of the warehouses which will be built on m prospective sites. The number of warehouses to be built should be determined to optimize an objective. Here, our objective is to determine how many and where the warehouses should be opened in order to minimize total costs of the fixed setup costs and transportation costs. We also like to know which demand centers each warehouse does supply. We assume that the transportation cost from warehouse i to demand center j, c_{ij}, depends on the distance from i to j. The transportation cost is a linear function of commodity flows and may include production and administrative costs. A formulation of this type of problems is,

$$z^* = \text{Min } z \qquad (0)$$

subject to,

$$z = \Sigma_i f_i y_i + \Sigma_i \Sigma_j c_{ij} x_{ij}$$

$$\Sigma_i y_i \geq MNO \qquad (1)$$

$$\Sigma_i y_i \leq MXO \qquad (2)$$

$$\Sigma_i x_{ij} = 1, \; j = 1, \dots n \qquad (3)$$

$$x_{ij} \leq y_i, \; j = 1, \dots n$$
$$i = 1, \dots m \qquad (4)$$

$$x_{ij} \geq 0, \; y_i = 0 \text{ or } 1, \qquad (5)$$

where f_i is the fixed setup cost required to open a warehouse at site i, MNO and MXO are the pre-determined minimum and maiumum number of warehouses to be opened, and $y_i = 1$ if a warehouse is opened at site i, $x_{ij} = 1$ if the demand of demand center j is supplied from warehouse i.

Since for y_i's fixed the constraints have the property of total unimodularity, the LP relaxation on x_{ij} leads to an interger solution.[see Garfinkel and Nemhauser(1972)]. The constraints (3) are of the type known as generalized upper bound (GUB), and the constraints (4) known as variable upper bound(VUB). Schrage(1975,1978) developed an efficient method to solve the problems with VUB constraints. In the above, we adopted the tight formulation. In general, the tight formulation provides an optimal solution with less fractional variables when the problem is relaxed to an LP. Furthermore, when the Benders decomposition method is applied to solve the problems, the tight fromulation generates better Benders' cuts in the sense that they have a greater right-hand-side constant in the minimization problem.

This type of problems has been studied by Kuehn and Hamburger(1963), Stollsteimer(1963), Balinski(1965), Lasdon(1970), Hansen et al.(1981) and many others. The warehouse capacity is not fixed beforehand, but it will be determined afterward. The warehouses must be built big enough to accomodate the demand by an optimal solution. In the above problem, the transportation cost, c_{ij}, is the cost required to meet the total demand of demand center i from warehouse j.

Most real-life applications of problem (0)-(5) are too large to be solved economically by existing general mixed integer linear programming codes. Based on its simple and special structure, various solution methods for the problem have been developed, i.e., heuristic methods by Kuehn and Hamburger(1963), Manne(1964), the branch and bound algorithm devised by

Efroymson and Ray(1966), the dual-based Lagrangian relaxation method by Erlenkotter(1978). We may also devise an algorithm exploiting the total unimodularity. This would be an LP relaxation method combined with the VUB algorithm by Scharage (1975, 1978).

In this paper, we will investigate a solution method based on the Benders decomposition algorithm developed by Benders(1962) and Geoffrion(1972). The algorithm has been applied in many areas. Geoffrion and Graves(1974), Polito et al.(1980) and Franca and Luna(1982) are a few examples among others. The Benders decomposition method is especially effective to the problems of which subproblem can be solved easily. Balinski and Wolfe(1963) were the first who tried the Benders decomposition method to solve the simple plant location problem. In this paper, we show that the Benders' algorithm may be improved by modifications in the way of generating the cuts. For the computational works, we use the data in Karg and Thompson(1964).

2. Benders Decomposition

For any feasible vector y^k,i.e., any vector satisfying (1), (2) and (5), we may write the problem as follows,

$$z^*(y^k) = \Sigma_i f_i y^k_i + \text{Min}\, \Sigma_i \Sigma_j c_{ij} x_{ij}$$

subject to,

$$\Sigma_i x_{ij} = 1, \quad j = 1,..,n$$
$$x_{ij} \le y_i^k, \quad j = 1,..,n$$
$$i = 1,....,m$$
$$x_{ij} \ge 0.$$

We minimize the transportation costs for a fixed y. The dual of the above problem is

$$D^*(y^k) = \Sigma_i f_i y_i^k + \text{Max} \;\; \Sigma_j v_j - \Sigma_i \Sigma_j y_i^k w_{ij}$$

subject to,

$$v_j - w_{ij} \le c_{ij}, \quad i = 1,....,m$$
$$j = 1,...,n$$
$$w_{ij} \ge 0, \; v_j\text{:unconstrained.}$$

We know from the duality that

$$z^*(y^k) = D^*(y^k) \ge z^*$$

The last relationship is obvious from the fact that

$$z^* = \text{Min}_{\,y \in Y} \; z^*(y),$$

where Y is a set of y's satisfying (1), (2), and (5). Using the dual problem, we write

$$z^* = \text{Min}_{y \in Y}[\Sigma_i f_i y_i + \text{Max} \;\; \Sigma_j v_j - \Sigma_i \Sigma_j y_i w_{ij}]$$

subject to,

$$v_j - w_{ij} \le c_{ij}, \quad i = 1,...,m$$
$$j = 1,...,n$$

or equivalently,

$$z^* = \text{Min}\, \Sigma_i f_i y_i + G$$

$$\text{subject to,}$$
$$y \in Y$$
$$G \geq \Sigma_j v_j^t - \Sigma_i \Sigma_j y_i w^t_{ij} \text{ for all } t,$$

where $[v^t_j, w^t_{ij}]$ is a set of extreme points of dual constraints. The last version of the problem has as many constraints as the number of extreme points. Instead of considering all the constraints at the same time, the Benders' algorithm solves the problem iteratively by generating constraints only when they are needed.

The iterative procedure is performed as follows. Suppose we know k extreme points. Then, our master problem would be

$$z^*(v^k, w^k) = \text{Min } \Sigma_i f_i y_i + G$$
$$\text{subject to,}$$
$$y \in Y$$
$$G \geq \Sigma_j v_j^t - \Sigma_i \Sigma_j y_i w_i{}^t{}_j, \; t=1,\ldots,k.$$

With an optimal solution of the master problem, y^{k+1}, we write the subproblem as

$$z^*(y^{k+1}) = \Sigma_i f_i y_i^{k+1} + \text{Min } \Sigma_i \Sigma_j c_{ij} x_{ij}$$
$$\text{subject to,}$$
$$\Sigma_i x_{ij} = 1, \; j=1,\ldots,n \tag{6}$$
$$x_{ij} \leq y_i^{k+1}, \; j=1,\ldots,n, \; i=1,\cdots,m \tag{7}$$
$$x_{ij} \geq 0.$$

After solving the subproblem, using its dual solution we add another constraint to the master problem. The new objective value of the master problem must be greater than the old one. We consistently approach the optimal solution. Furthermore, it is obvious that

$$z^*(v^k, w^k) \leq z^* \leq z^*(y^{k+1}). \tag{8}$$

In other words, an optimal solution to the master problem provides a lower bound, and an optimal solution to the subproblem gives us an upper bound, of the objective value of our problem. Therefore, at iteration k we stop the iterative procedure if for a positive number ϵ,

$$z^k(y^*) - z^*(v^k, w^k) \quad \epsilon \tag{9}$$

where $z^k(y^*) = \min_{t=0,\ldots,k} z^*(y^{t+1})$. Otherwise, we must find a dual solution and add another constraint to the master problem.

The above subproblem is a primal instead of a dual form. The primal problem can be easily solved by inspection. To solve the subproblem, we do not need any LP algorithm. In the following, we discuss how we add the constraints.

2.1 Method A (Standard Method)

Setting the initial values of v_j^o and w_{ij}^t to be zeroes, we solve the master problem. Using an optimal solution, y_i^1, we solve the subproblem and compute the dual solution as follows:

Suppose $a(j)=\text{argmin}_i \; [c_{ij} \mid y_i^1=1]$.
$x_{ij}=1$ if $i=a(j)$, and $x_{ij}=0$ oterwise.
$v_j^1 = c_{a(j),j}$, and
$w^1_{ij}=\text{Max} \; [0, \; c_{a(j),j}-c_{ij}]$.

The primal solution is to assign demand center j to the closest open warehouse. The dual variable v_j is the cost required to satisfy the demand center j's marginal demand, and w_{ij} indicates the opportunity savings to be realized if demand center j is supplied by warehouse i. Once we find a primal optimal solution, a dual solution is easily obtained. Since we have found a dual solution, we can add a cut to the master problem. This process is repeated until the condition (9) is satisfied.

2.2 Method B

The primal solution to the subproblem is usually uniquely determined. But, we have many alternatve dual optimal solutions. Therefore, it is possible to add more than one cuts whenever the condition (9) is not satisfied. We add two cuts whenever we obtain an optimal solution to the master problem. The first cut is constructed by a dual solution which is determined in the same way as the method A, and we generate another cut based on an alternative solution given as follows,

Suppose $s(j)=\text{argmin}_i \; [c_{ij} | y_i^t=1 \text{ and } i \neq a(j)]$. and
$C_j=c_{s(j),j}$ if $s(j)$ exists, else $C_j=c_{a(j),j}$.
Then an alternative dual solution is
$v_j^t= C_j$, and
$w_{ij}^t=\text{Max} \; [0, \; C_j-c_{ij}]$.

In the above, s(j) is the second closest open warehouse to j. Note that if there is no s(j), we may assign any number to C_j. Furthermore, any linear combination of the two solution is also optimal. This model tries to cut the constraint set of the master problem more deeply by adding two constraints at a time. We expect the number of iterations will be reduced. The tradeoff is an increase in computing time needed to solve the master problem. Sometimes we may even see the case that one cut dominates the other in the sense of Magnanti and Wong(1981)'s dominance. The investiagation of relative efficiency of the two methods is an empirical matter.

2.3 Method C

In the above two methods, the master problem must be completely enumerated before a new cut is added. Although the enumeration is a very time-consuming job, this procedure guarantees that the optimal solution to the problem is found in finite iterations. The number of iterations depends on the number of extreme points of dual constraint. In order to reduce the number of pivoting, we may add a new cut whenever we find an integer solution to the master problem. We do not solve the master problem to the optimality. Geoffrion and Graves(1974) also tried this approach. The cut is generated in the same way as the method A. The only difference from A is when a cut is added.

For the method A and B, the master problem has been optimized before we get into the subproblem. The master problem should be completely enumerated to find out an optimal solution y^* which gives a minimum to $z^*(v^k, w^k)$. This guarantees the lower bound of our problem, $z^*(v^k, w^k)$, is non-decreasing. Thus we approach the optimal solution consistently. The method C is expected to be helpful for us to reduce the number of pivoting operations for each iteration. But, it costs consistency. Since the master problem is not solved to the optimality, the relationship (8) does not hold any more. The non-decreasing property of $z^*(v^k, w^k)$ is lost. The lower bound given by $z^*(v^k, w^k)$ has no significant meaning, but the upper bound is the cost of the current solution. Although we do not solve the master problem to the optimality, we search for an integer solution which gives the smaller objective function value than the current best upper bound, $z^k(y^*)$, which was defined previously. This allow us to keep the lower bound to be below the upper bound. Furthermore, we solve the master problem until we find an integer solution for which the objective value is smaller than $z^k(y^*) - \epsilon$. When we cannot find such a solution, we stop the iterative procedure.

Note that in spite of the nonoptimality of y^k, $z^*(y^k)$ is always greater than z^*. Since the master problem is not solved to the optimality, we suppose probabilistically that the resulting cut will not be as efficient as the cut by the standard method.

2.4 Method D

Instead of adding one cut, we may add two cuts whenever we find an integer solution to the master problem. The method of generating cuts is exactly same as the method B. We use the same stopping rule applied for the method C. The difference of this method from the method C is compared to the difference of the mathod B from the method A.

2.5 Method E (LP Relaxation)

The standard method solves the master problem as an IP problem. Solving an IP problem takes much more time than solving an equivalent LP problem. As an effort to reduce the computing time, our next approach is an LP relaxation of the master problem. We solve the master problem without integer constraints until certain number of cuts are accumulated, and then we take the integrality into consideration. The optimal solution to our problem is likely to be found only after a number of cuts are added. We avoid of wasting time in the early stage of generating cuts.

First, we solve the master problem using an LP algorithm. Even though the resulting solution is not integral, we are easily able to find an optimal dual solution to the corresponding subproblem. The primal variables x_{ij} must satisfy (6) and (7). Note that the primal subproblem can be separated into n independent problems. So, for any j, its demand is first supplied by the closest open plant. If (6) is not satisfied, we find the next closest open plant and assign its supply to the demand center. We repeat this procedure until (6) is

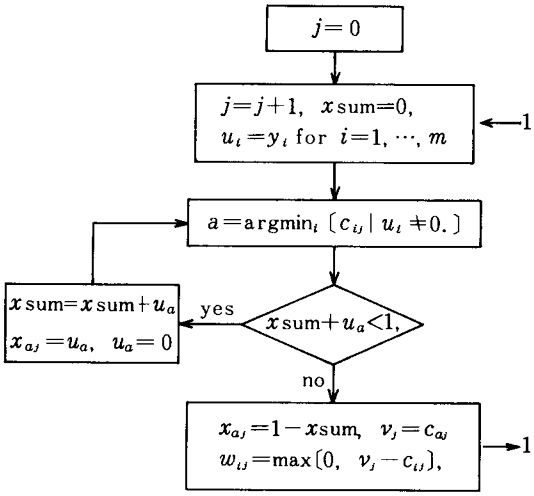

Figure 1. Flow Diagram for the Subproblem solution

satisfied. The dual value for (6) v_j is, then, the distance to the warehouse which supplied marginally. The dual price for (7) w_{ij} is determined by the v_j less c_{ij}. If this value is negative, w_{ij} is set to be zero. Figure 1 shows the procedure.

It is obvious that, for both cases of the standard and the LP relaxation, the lower bound increases monotonically, and never be beyond the objective value of the subproblem. But, it is not clear which method approaches the optimal point more rapidly. Because the two methods generate a different cut for every iteration. It is likely that the LP relaxation approach generally generates more cuts than the standard method before we reach an optimum. Therefore, if we convert to the IP problem too late, we might have wasted time to generate unnecessary cuts. On the other hand, if we add the integrality constraint too early, it will reduce the effectiveness of the LP relaxation method. What is the optimal conversion strategy? The question cannot be answered immediately. The optimal strategy seems to be different case by case.

In this study, we suggest and implement three conversion strategies: (a) when the gap between lower bound and upper bound is less than 2%, (b) when the number of cuts generated is greater than the number of warehouse sites, (c) when the consecutive two solutions to the master problem are the same. If at least one of the conditions is statisfied, conversion to the IP problem takes place. When we adopt LP relaxation method, we may sometimes encounter the condition (a) is never met. There exists such a case that the lower bound from the LP relaxed master problem does not approach the optimal solution of the problem (0)–(5). Without the condition (c), sometimes we might have generated redundant constraints if an optimal solution to the LP problem were obtained before the condition (a) and (b) are satisfied. Once we have reached the LP optimum point, we can no longer increase the lower bound from the master problem in the LP stage. This implies that we cannot add a different cut after the LP optimum. Thus, when the consecutive two solutions to the master problem are the same, we have already reached the LP optimal solution. At this point, we must convert to the integer problem by adding integrality contraints. The conversion rule (b) is quite arbitrary. We assume that a problem with the more candidate sites requires the more cuts before it reaches an optimal solution.

After conversion to the IP problem, we apply the method A. Thus, the only difference from the method A is that the method E has a warm-up stage to accumulate a certain number of cuts.

2.6 Method F

We have seen that the dual of the subproblem has a number of alternative solutions. Even though the dual price v_j is given by the next closest distance among the open warehouses, the objective value remains unchanged. Since the Benders cuts are generated on the basis of the dual prices, a different set of dual prices generates a different cut. If it is possible to generate a deeper cut than the one by the method A, the expected number of cuts at the optimal solution will be decreased. The following definition of dual prices generally provides us with a deeper cut.

$$c_{aj} = \min_i \ [c_{ij} \mid y_i = 1]$$
$$I_j = \{i \mid c_{ij} \leq c_{aj},\ i \neq a\}$$
$$v_j = \begin{cases} c_{aj}, & \text{if } I_j \neq \emptyset \\ \min_i \ [c_{ij} \mid i \neq a], & \text{if } I_j = \emptyset \end{cases}$$
$$w_{ij} = \max\ [0,\ v_j - c_{ij}]$$

This set of dual prices always generates a Bender's cut with a greater right hand side constant term than a cut generated in a standard fashion. We can easily show that the cut by the above dual prices dominates the standard cut. Therefore, when we implement the LP relaxation method, we will get a better cut if we adopt the above alternative definition of dual prices to generate the Benders' cut.

Our last experiment will be a combination of the LP relaxation and the improved cut genertion method. In the following, we compare the methods. The relative performance will be measured by the cpu time.

3. Computational Results

We use the data by Karg and Thompson(1964) to implement and compare the methods. We exhibit the data on Exhibit A. The transportation costs are symmetric and assumed to be proportional to the distance among the cities. We have 5 demand centers. The warehouses can be opened on any of these cities, and the fixed cost of opening a warehouse is the same for all the cities. We test four cases: (2,4,0), (3,3,0), (2,4,100), and (3,3,100), where the first and second

Exhbit A. Transportation Cost (c_{ij})

i \ j	1	2	3	4	5
1	0				
2	30	0			
3	26	24	0		
4	50	40	24	0	
5	40	50	26	30	0

index indicate the minimum and maximum number of warehouses to be opened and the last index is the cost required for opening a warehouse.

We are easily able to program the methods using LINDO developed by Schrage of the University of Chicago. To implement the method C, we add a constraint

$$\Sigma_i f_i y_i + G \leq z^k(y^*) - \varepsilon$$

to the master problem. The constraint guarantees that the lower bound never be beyond the upper bound. We stop solving the problem when the master problem has no feasible solution. We used DEC−20 interactive computer. In the following, the computing time refers DEC−20 cpu time. We display the results on Exhibit B.

Exhibit B. Summary Results[1]

Method \ Case	(2,4,0)[2]	(3,3,0)	(2,4,100)	(3,3,100)
A	7.82(75)	16.03(247)	15.90(227)	15.58(231)
B	6.58(91)	12.00(210)	7.91(108)	9.43(165)
C	4.07(26)	9.47(92)	10.11(82)	9.35(83)
D	3.35(20)	7.98(106)	5.78(69)	7.17(77)
E	6.66(58)	8.18(73)	6.15(52)	7.30(62)
F	5.74(38)	5.01(29)	5.04(36)	4.99(32)

1. Time in cpu seconds. The numbers in parentheses are the number of pivoting operations.
2. The first and second index indicate the minimum and maximum number of warehouses to be opened. The last index is the fixed cost which is applied to all the sites equally.

The method of adding two cuts for each iteration was better than the method of adding one cut. For all cases the nonoptimization methods (C and D) dominate the corresponding standard methods (A and B), respectively. By the nonoptimization method, the number of pivoting operations was greatly reduced. As a result, the computing time was reduced by about 30%. We also observe that the LP relaxation methods (E and F) are generally superior to the nonoptimization method with one cut (C). The relative efficiency of the method D and E is not clear. However, the best result was obtained when we applied the LP relaxation method combined with the improved cut generation method (F). With the method F, the computing time was reduced by

70% of the standard method (A). The number of cuts and the number of pivoting operations were greatly reduced.

We have seen on Exhibit B that it is possible to cut the computing time remarkably when we adopt the LP relaxation method. It solved the small sample problem in a third of the time required by the standard method. We like to see if there exists any difference in applications to bigger problems. We apply the methods, A, E and F, to a problem with 33 demand centers. The data come from Karg and Thompson(1964). We investigate four cases. The results are shown on Exhibit C. The LP relaxation method still dominates the standard method. It solved the problem in one fifth to a half of the time required by the standard method. The improved cut method was especially efficient for the last two cases. When we restrict to open the small number of warehouses out of 33, the effectiveness of the improved cut method is not significant. An explanation for this observation will be found when we look at the procedure to get dual prices.

Exhibit C. Computing Time for 33-city Problem

Method[1] \ Case	(2,3,295)[2]	(5,5,295)	(25,25,295)	(31,31,295)
A	171.33	252.46	372.53	492.46
E	34.96	49.88	102.03	243.72
F	35.48	51.05	19.96	2.91

1. A: standard method, E: LP relaxation with standard cuts, F: LP relaxation with improved cuts.
2. The first and second index indicate the minimum and maximum number of warehouses to be opened. The last index is the fixed cost which is applied to all the sites equally.

In the LP warm–up stage, if the LP optimal solution is fractional, then the resulting dual prices are the same for both cases. According to my experiments, the smaller number of warehouses is to be opened, the LP optimal solution has the more fractional values. Another explanation is given by the set, I_j. The smaller number of warehouses is to be opened, it is more likely that the set I_j is empty. Thus the methods generate the same cut.

Based on the results, we suggest that the LP relaxation method is a good approach to the simple plant location problem. The effectiveness of the improved cut method will be prominent for the case that the master problem has an integer optimal solution.

4. Conclusion

When we apply the Benders decomposition method to solve the simple plant location problem, we observe the subproblem is highly degenerated so that it has many dual solutions. Exploiting this observation, we may add more than one cuts each iteration. Another variation was made to the master problem. We can drop the integrality constraints, or we need not solve the master problem to the optimaltity. the results showed that the two-cut method was superior to the one-cut method, and the LP relaxation and the non-optimization method dominate the standard method. Applying the LP relaxation method with the improved cut generation method, we were able to reduce the computing time by more than half of the time required by the standard method.

The improved cut generation method was especially efficient when we are allowed to open many warehouses. The method introduced in this paper is very naive. Magnanti and Wong(1981) suggest a method to find a pareto-optimal cut. It will be interesting to investigate the relative effectiveness of the pareto-optimal cut and the method in this paper. Although we did not try the non-optimization method combined with the improved cut, we conjecture that it will make no big difference from the method F. It is likely that the larger the problem is, the more effective is the method F. The non-optimization method adds cuts very efficiently in the earlier stage, but it will lose the efficiency as the upper bound approaches the optimal solution. For the implementation of the LP relaxation method, the choice of the conversion strategy is critical.

In order to keep the lower bound below the current best upper bound, we added a constraint to the master problem. Similarly, we can add a constraint so that the lower bound increases. One weakness of the method C was the inconsistency of the lower bound. Suppose L is the current best lower bound. If we add a constraint

$$\Sigma_i f_i y_i + G > L$$

to the master problem, the lower bound will go up consistently. And we expect this will improve the performance of the method C and D.

REFERENCES

Balinski, M., "Integer programming: Methods, Uses, Computations", *Management Science*, (March), 1965, PP. 253–313.

Balinski, M.L. and P. Wolfe, "On Benders Decomposition and a Plant Location Problem," Working paper ARO–27, Mathematica, Princeton, N.J. 1963.

Benders, J.F., "Partitioning Procedures for Solving Mixed Variables Programming Problems," *Numerische Mathematik*, Vol. 4, 1962, PP. 238–252.

Efroymson, M.A. and T.L. Ray, "A Branch-Bound Algorithm for Plant Location, *Operations Research*, Vol. 14, 1966, PP. 361–368.

Erlenkotter, D., "A Dual-Based Procedure for Uncapacitated Facility Location," *Operations Research*, Vol. 26, 1978, PP. 992–1009.

Franca, P.M. and H.P.L. Lunã, "Solving Stochastic Transportation Location Problems by Generalized Benders Decomposition," *Transportation Science*, Vol. 16, No. 2, 1982, PP 113–126.

Garfinkel, R.S. and G.L. Nemhauser, *Integer Progamming*, Wiley, New York, 1972.

Geoffrion, A.M., "Generalized Benders Decomposition," *Journal of Optimization Theory and Application*, Vol. 10, No. 4, 1972, PP. 237–260.

Geoffrion, A.M. and G.W. Graves, "Multicommodity Distribution System Design by Benders Decomposition," *Management Science*, Vol. 20, No. 5, 1974, PP. 822–844.

Hansen, P., J.F. Thisse and P. Hanjoul, "Simple Plant Location under Uniform Delivered

Pricing,"*European Journal of Operational Research*, Vol. 6, 1981, PP. 94–103.

Karg, R.L. and G.L. Thompson, "A Heuristic Approach to Solving Travelling Salesman Problems," *Management Science*, (Jan.), 1964, PP. 225–248.

Krarup, J. and P.M. Pruzan, "The Simple Plant Location Problem: Survey and Synthesis," *European Journal of Operational Research*, Vol. 12, No. 1, 1983, PP. 36–81.

Kuehn, A.A. and M.J. Hamburger, "A Heuristic Program for Locating Warehouses," *Management Science*, Vol. 9, 1963, PP. 643–666.

Lasdon, L.S., *Optimization Theory for Large Systems*, Macmillan Publishing Co., New York, 1970.

Magnanti, T. L. and R. T. Wong, "Accelerating Benders Decomposition:Algorithmic Enhancement and Model Selection Criteria", *Operations Research*, 1981, PP. 464–484.

Manne, A.S., "Plant Location under Economies-of-Scale-Decentralization and Computation," *Management Science*, Vol. 11, 1964, PP. 213–235.

Polito, J., B.A. McCarl and T.L. Morin, "Solution of Spatial Equilibrium problems with Benders Decomposition," *Management Science*, Vol. 26, 1980, PP. 593–605.

Schrage, L.E., "Implicit Representation of Variable Upper Bounds in Linear Programming," *Mathematical Programming Study*, Vol. 4, 1975, PP. 118–132.

_________, "Imlicit Representation of Generalized Variable Upper Bounds in Linear Programs," *Mathematical Programming*, Vol. 14, 1978, PP. 11–20.

Stollsteimer, J.F., "A Working Model for Plant Numbers and Locations," *Journal of Farm Economics*, Vol. 45, 1963, PP. 631–651.