Journal of the Nigerian Association of Mathematical Physics
***Volume* 22 (November, 2012), pp 273 – 280**

LP-Relaxation of a Single Stage Capacitated Warehouse Location Problem (CWLP): A Study In Formulation Efficiency II

[1]Abdullahi, N. and [2]Sani, B.

**[1]Department of Mathematics and Computer Science,
NDA, Kaduna
[2]Department of Mathematics,
Ahmadu Bello University, Zaria**

Abstract

Formulations proposed in the literature for the Capacitated Warehouse (Facility or Plants) Location Problem (CWLP), are compared. The comparison, with main emphasis on (linear programming) LP-relaxations is based on some theoretical and computational results. The theoretical aspect compares some relations among the subsets of some constraints of the problem sets. On the other hand, the computational aspect compares the relaxations in terms of the quality of the lower bound the formulations produced when solved directly on some small size test problems, having various characteristics.

Keywords: Mixed integer programming, Warehouse (Facility, Plant) Location problem, LP-relaxation.

1.0 Introduction

The Capacitated Warehouse (Facility or Plant) Location Problem (CWLP) is a well known combinatorial optimization problem. It is concerned with determining which warehouses to open from a given set of potential warehouses in different locations and also how to assign customers to these warehouses in-order to satisfy demand. The objective is to minimize total fixed and shipping costs. Constraints are such that each customer's demand must be satisfied and that each warehouse cannot supply more than its capacity if it is opened. The problem class of CWLP we consider in this paper is a one stage single product CWLP. The model to solve this class of problems forms the core or generic model where some special cases of the problem are derived. The problem can be stated as follows; a single product is produced or stocked at some warehouses in order to satisfy customer's demands; the product is then transported from these plants/warehouses to customers directly. On the other hand, the two-stage CWLP is the situation where a single product is produced at some plants (warehouses or facilities) in order to satisfy demands, and then the product is transported from these plants (warehouses, facilities) to some depots before sending it to customers. Other cases which are special classes of this problem are the one stage multi product CWLP, two stage multi product CWLP, etc and the P-median location problems. Further details can be found in [1, 2].

Applications of CWLP exist in the literature which have nothing to do with warehouse location. For example the same mathematical model is appropriate in the areas of production scheduling with set up costs, telecommunication network design and machine replacement and as the basis for some heuristics in vehicle routing, where vehicles have unequal capacities [3, 4]. Methods of solving CWLP can be found in [2, 3 and 4] and the references there in. The rest of the paper is presented as follows; section two is the various problem formulations, section three is the LP-relaxation, section four is the computational results and discussion while section five is the conclusion, followed by the references. The test problems can be found in either [5] or [6].

[1]Corresponding author: ***Abdullahi, N.,*** E-mail: nasirualbani@yahoo.co.uk, Tel. +234 8035587641

2.0 Problem Formulation

A general and well known Mixed Integer Programming formulation of CWLP is as follows [2, 3, 7, and 8]: Suppose there are m possible locations for warehouses to supply a commodity for shipment to n demand points; define a binary variable y_i, that corresponds to the use (open) or disuse (close) of warehouse at location i, according as to whether y_i is 1 or 0; if it is operational y_i is 1 otherwise it is 0. The capacity of the warehouse in this case is s_i and if the location i is used then a fixed cost f_i is incurred. A cost c_{ij} is also incurred for transporting a fraction (or percentage) x_{ij} of the demand d_j from location i, to demand point j. The total volume at the location$i$, is $\sum_j d_j x_{ij}$. The problem to solve is then:

$$Z_1 = min \sum_i \sum_j c_{ij} x_{ij} + \sum_i f_i y_i \quad (1)$$

Subject to

$$\sum_i x_{ij} = 1 \quad for\ all\ j \quad (D)$$

$$x_{ij} \le y_i \quad for\ all\ i, j \quad (B)$$

$$\sum_i s_i y_i \ge \sum_j d_j \quad for\ all\ j \quad (T)$$

$$\sum_j d_j x_{ij} \le s_i y_i \quad for\ all\ i \quad (C)$$

$$\sum_i x_{ij} = d_j \quad for\ all\ j \quad (K)$$

$$\sum_j x_{ij} \le s_i y_i \quad for\ all\ i \quad (L)$$

$$\begin{cases} 0 \le x_{ij} \le 1 \\ 0 \le y_i \le 1 \end{cases} \quad for\ all\ i, j \quad (N)$$

$$y_i = 0\ or\ 1 \quad for\ all\ i \quad (I)$$

The constraints (B) and (T) (not usually found in the formulation of CWLP) are called variable upper bound (VUB) and total demand constraints respectively. These two constraints are implied by the constraints (D) (called generalized upper bound, GUB, which says that all demand must be satisfied), (C), (N), and (I). To see that (B) is valid for CWLP observe that $y_i = 0\ or\ 1$ by constraint (I). When $y_i = 1$, (B) follows from $x_{ij} \le 1$ in (N) and when $y_i = 0$, (C) implies that $x_{ij} = 0$ (using $x_{ij} \ge 0\ and\ d_j \ge 0$) therefore (B) holds [8]. Similarly (T) is valid from the constraints (C) and (K).

Because of alternative formulations of the same problem, where a feasible set can be represented by different set of constraints, the various formulations of (1) are obtained by combining some of the constraint sets above. These derived formulations differ both in the quality of the lower bound they produce and the degree of ease of computation, [5].

The following notations were used conventionally except where otherwise stated.
Let:$(S_1, \dots \dots \dots S_k)$, be the sets of equality or inequality constraints.
$F(S_1, \dots \dots \dots S_k)$, the feasible region defined by the constraints$(S_1, \dots \dots \dots S_k)$.
$conv(S_1, \dots \dots \dots S_k)$, be the convex hull of the corresponding region. $Z(.)$, $\bar{Z}(.)$ denote optimal objective values for the integer and lp-relaxation problems respectively corresponding to the problem formulations.

The following are eight various formulations derived for the general CWLP which correspond to some formulations reported in the literature. Later we would show by our computational study that these eight formulations reduced to three equivalent sets of formulations.
In several works [10, 11, 12, 13, 14 and 15] the so called weak formulation of (1) was studied and solved. The formulation is weak because of the absence of constraint sets (D) and (B) [6]. The formulation is as follows;

$$Z_2 = min \sum_i \sum_j c_{ij} x_{ij} + \sum_i f_i y_i \quad (2)$$

$$subject\ to\ (L), (K), (N), and\ (I)$$

Disaggregated version of (2) as reported in [2], is the problem with (L) replaced by
$x_{ij} \le s_i y_i;\ for\ all\ i$ and j. Suppose we denote this constraint by$(\acute{L})$. Hence we have:

$$Z_3 = min \sum_i \sum_j c_{ij} x_{ij} + \sum_i f_i y_i \quad (3)$$

$subject\ to\ (\acute{L}), (K), (N), and\ (I)$

(3) Provides an LP-relaxation that is superior to (2), i.e the latter is tighter than the former and is more suitable to use in some algorithmic schemes such as Branch and Bound and lagrangean relaxation [7, 9].

The so called strong formulation of (1) was studied in [6, 16]. Because of the presence of the constraints sets (B) and (D), this formulation give a tighter lower bound than the previous two formulations when constraint (I) is relaxed. It is given as follows;

$$Z_4 = min \sum_i \sum_j c_{ij}\, x_{ij} \ + \ \sum_i f_i\, y_i \qquad (4)$$

$subject\ to\ (D), (B), (C), (N), and\ (I)$ Relaxed

In certain applications of CWLP, demand for a customer must be supplied from a single warehouse (i.e CWLP with indivisible demand). This formulation, presented in [17], is given as follows:

$$Z_5 = min \sum_i \sum_j c_{ij}\, x_{ij} \ + \ \sum_i f_i\, y_i \qquad (5)$$

$subject\ to\ (D), (B), (C), (N^{\mathbb{Z}})\ and\ (I)$

Where $(N^{\mathbb{Z}})$ has replaced (N), given as $x_{ij} \geq 0\ and\ x_{ij}\ an\ integer$. This constraint ensures that each customer is served from only one warehouse.

In a situation where the customers' demands are all equal, say $d_j = d\ for\ all\ j$, this differs from (5) by replacing (C) with $\sum_j x_{ij} \leq s_i y_i\ , \forall i$. Computationally these two similar formulations are the same in terms of the lower bound obtained when solved. We therefore have not listed it here [4].

The formulation in [9] is almost the same with the generalized CWLP shown above and is given as follows;

$$Z_6 = min \sum_i \sum_j c_{ij}\, x_{ij} \ + \ \sum_i f_i\, y_i \qquad (6)$$

$subject\ to\ (D), (C), (T), (B), (N), and\ (I)$

Computationally (6) is among the three best formulations of CWLP in terms of the lower bound it produces.

In [18] another parameter is defined as $g_{ij} = \min\{s_i, d_j\}$ which is augmented within the constraint sets, and the constraint (K) was expressed in terms of (≥) inequality rather than equality as the previous cases. The CWLP using these changes is defined as:

$$Z_7 = min \sum_i \sum_j c_{ij}\, x_{ij} \ + \ \sum_i f_i\, y_i \qquad (7)$$

$subject\ to(\grave{K}), (L)(N), (I)$ and $x_{ij} \leq g_{ij}\ for\ al\ i, j$

where $(\grave{K})$ is (K) with "=" being replaced by " ≥".

A disaggregated form of (7) with the constraint (L) replaced by
$x_{ij} \leq s_i y_i\ for\ all\ i, j$ also yields a tighter lower bound than (7). Denote this by $(\acute{L})$; thus we have

$$Z_8 = min \sum_i \sum_j c_{ij}\, x_{ij} \ + \ \sum_i f_i\, y_i \qquad (8)$$

$subject\ to(\grave{K}), (\acute{L})(N), (I)$ and $x_{ij} \leq g_{ij}\ for\ all\ i, j.$

Another interesting thing is that [19] obtained a tighter lower bound by appending to the relaxed (7), the constraints $x_{ij} \leq g_{ij} y_i\ for\ all\ i, j$. The authors however did not use direct LP solution to solve the resulting model, instead they used Benders decomposition. Their formulations is as follows:

$$Z_9 = min \sum_i \sum_j c_{ij}\, x_{ij} \ + \ \sum_i f_i\, y_i \qquad (9)$$

$subject\ to(\grave{K}), (L)(N), (\grave{I})\ \ x_{ij} \leq g_{ij}\ for\ all\ i, j$ and $x_{ij} \leq g_{ij}y_i\ for\ all\ i, j.$

3.0 LP-Relaxations

Formally, a relaxation of a minimization problem is defined as follows [20]:
Definition: Problem (RZ): $\min\{g(x,y)|x, y \in W\}$ is a relaxation of problem (Z): min $\{f(x,y)|x, y \in V\}$, with the same decision variables, iff

i. $F(RZ)$ contains $F(Z)$ i.e $F(RZ) \supseteq F(Z)$
ii. Over $F(Z)$, the objective function of (RZ) dominates (i.e is better than) that of (Z) i.e $\forall x, y \in V, g(x,y) \leq f(x,y)$, where $V \subseteq W$.

It clearly follows that the optimal value of RZ is less than or equal to the optimal value of Z i.e, $\overline{RZ}(.) \leq Z(.)$, since RZ has more feasible solutions than Z; the objective function value, $\overline{RZ}(.)$, of RZ is better than (smaller than) that of Z, thus it has a smaller minimum.
The potential usefulness of any relaxation, $\bar{Z}(.)$ of Z, being LP or Lagrangean relaxations is largely determined by how near its optimal value is to that of Z. One of the basic challenges faced in the design of any algorithm for Z is which relaxations to use for lower bounding in the case of minimization. The chosen relaxation must be a suitable compromise between ease of computation on one hand and tightness of the resulting bound on the other. Generally a relaxation which produces a very tight (better) lower bound on the optimal value of Z will be very expensive to compute, whereas an easily optimized relaxation is likely to result in relatively poor bounds, which may affect the convergence of any scheme to solve Z [4, 7, 8, 20]. We concentrate here on the most obvious choice of relaxing Z, i.e the continuous LP relaxations, i.e problem Z with the integrality restriction on constraint (I) dropped i.e $0 \leq y_i \leq 1\ for\ all\ i$. This relaxation exploits the remarkable efficiency of modern LP system based on the presence of GUB constraint (D) which constitutes a large fraction of all constraints.

It was reported in [7] that computational experience indicates that the tightness of the lower bound obtained by Lp-relaxations leaves much to be desired. They observed that on the test problems they used, there is an average gap of more than 7% between the optimal values of $Z(.)\ and\ \bar{Z}(.)$; coincidently we recorded almost the same observation on the test problems we compute, with some few exceptions where there is less than 1% or even 0% gap between $Z(.)\ and\ \bar{Z}(.)$; see Tables 2 through 9.

It was also reported in [20, 21] that if the so called lagrangean dual problem has what they call integrality property (i.e if dropping the integer restriction on the sub-problem does not yield a better result than the usual LP relaxation) then the Lagrangean relaxation solution coincides with the usual LP relaxation. However, [21] emphasized that the integrality property is not defined relative to a given problem class, but relative to a given integer (pure or mix) programming formulation of a problem class. This is an important distinction because a problem class often has more than one formulation as shown in section two of this paper. (Investigation on the integrality property of the formulations considered here is on course and will soon be communicated, since this can only be achieved through Lagrangean relaxation which is not in the scope of this paper). In Theorem 1 of [20], a summary of some of the basic relationships between Z and RZ in the context of lagrangean relaxation were presented and these were also studied in [4]. RZ is the LP relaxation of Z.

4.0 Computational Results

Our computational study compares the relaxations $\bar{Z}(.) = RZ$ to the optimum integer value $Z(.) = Z$. The relative quality of the bound is measured by

$$\xi_{bound} = \frac{Z(.) - \bar{Z}(.)}{Z(.)} * 100\%$$

TABLE 1: CHARACTERISTICS OF TEST PROBLEMS

S/NO	Problem Size	Name	Supply s_i	demand d_j	Fixed charge f_i	Shipping cost c_{ij}
1	3x4	A34	Different for each i, with more than 80% >d_j and f_i	Different for each j, but 100% < both s_i and f_i	Same for all i.	Single digits no zeros all routes allowed.
2	5x4	B54	Same for each i, and generally >d_j	Same for each j, but less than both s_i and f_i	Different for each i, mostly> both s_i and d_j	two digits no zeros all routes allowed
3	5x6	C56	Different for each i, mostly greater than d_j	Different for each j, mostly < s_i	Multiples of hundred, > sum of s_i and d_j	Two digits, within the range of s_i and d_j, all routes allowed.
4	5x8	D58	Same for each i, > d_j	Same for each j, and < s_i	In thousands	In multiples of hundreds some routes not allowed
5	5x8	E58	80% different, > d_j	60% the same, 90% < s_i	Different for each i and in multiples of ten	Two digits with decimals, some routes not allowed
6	10x10	F1010	Same for all i, > d_j	Same for all j and < s_i	Multiples of 100 >> s_i and d_j	In tens with few routes not allowed.
7	10x15	G1015	Same for all i, > d_j	Same for all j all less than s_i	Different for each i, Multiples of hundreds, > s_i and d_j	Two digits with a lot of routes not allowed
8	15x15	H1515	Different for each i, > d_j	Same for all d_j , < s_i	Multiples of thousands Same for all i,	In tens with all routes allowed
9	15x15	I1515	Different for all i.	Different for all j	In thousands Same for all i, >> s_i and d_j	Sparsely represented
10	15x15	J1515	Different for each i, same range with d_j	Mostly different for each j	Multiples of hundreds, >> both s_i and d_j	In tens and a lot of routes not allowed

Table 2: Computational Results for (2)

Instance	$Z(.)$	$\bar{Z}(.)$	ξ_{bound}
A34	3820	3491	9%
B54	412	397	4%
C56	2622	2115	I9%
D58	44170	41970	5%
E58	767.5	764.5	0.4%
F1010	2280	1758	0.23%
G1015	4230	3746	11%
H1515	10528	10380	1.40%
I1515	9209	8417	7%
J1515	11789	10973	7%

Table 3: Computational Results for (3)

Instance	$Z(.)$	$\bar{Z}(.)$	ξ_{bound}
A34	3770	3204	15%
B54	312	303	2.9%
C56	1859	1382	26%
D58	11520	9120	21%
E58	535	460	14%
F1010	1774	715	60%
G1015	2537	1209	52%
H1515	10000	1195	88%
I1515	2131	1739	18%
J1515	470	470	0%

Table 4: Computational Results of (4)

Instance	$Z(.)$	$\bar{Z}(.)$	ξ_{bound}
A34	1007	844	16%
B54	220	201	9%
C56	1759	1687	4%
D58	42362	40162	5%
E58	344	331	4%
F1010	1440	918	36%
G1015	3427	3046	11%
H1515	10264	10264	0%
I1515	9019	8126	10%
J1515	11542	10799	6%

Table 5: Computational Results for (5)

Instance	$Z(.)$	$\bar{Z}(.)$	ξ_{bound}
A34	1007	844	16%
B54	220	201	9%
C56	1759	1687	4%
D58	42362	40162	5%
E58	344	331	4%
F1010	1440	918	36%
G1015	3427	3046	11%
H1515	10264	10264	0%
I1515	9019	8126	10%
J1515	11542	10799	6%

Table 6: Computational Results for (6)

Instance	$Z(.)$	$\bar{Z}(.)$	ξ_{bound}
A34	1007	844	16%
B54	220	201	9%
C56	1759	1687	4%
D58	42362	40162	5%
E58	344	331	4%
F1010	1440	918	36%
G1015	3427	3046	11%
H1515	10264	10264	0%
I1515	9019	8126	10%
J1515	11542	10799	6%

Table 7: Computational Results for (7)

Instance	$Z(.)$	$\bar{Z}(.)$	ξ_{bound}
A34	3820	3491	9%
B54	412	397	4%
C56	2622	2115	I9%
D58	44170	41970	5%
E58	767.5	764.5	0.4%
F1010	2280	1758	0.23%
G1015	4230	3746	11%
H1515	10528	10380	1.40%
I1515	9209	8417	7%
J1515	11789	10973	7%

Table 8: Computational Results for (8)

Instance	$Z(.)$	$\bar{Z}(.)$	ξ_{bound}
A34	3770	3204	15%
B54	312	303	2.9%
C56	1859	1382	26%
D58	11520	9120	21%
E58	535	460	14%
F1010	1774	715	60%
G1015	2537	1209	52%
H1515	10000	1195	88%
I1515	2131	1739	18%
J1515	470	470	0%

Table 9: Computational Results for (9)

Instance	$Z(.)$	$\bar{Z}(.)$	ξ_{bound}
A34	3820	3491	9%
B54	412	397	4%
C56	2622	2115	I9%
D58	44170	41970	5%
E58	767.5	764.5	0.4%
F1010	2280	1758	0.23%
G1015	4230	3746	11%
H1515	10528	10380	1.40%
I1515	9209	8417	7%
J1515	11789	10973	7%

Unlike Lagrangean relaxations [1, 5, 7, 12, 15, 20, and 21], Lp-relaxations are not instance dependent [1], but depend on the formulation used, as shown in Tables (2 - 9). Table 1 shows the characteristics of the test problems used. The formulations of the optimization models presented in section two were coded in the syntax of the modeling language AMPL [22], and complete problems instance were solved by the system CPLEX 11.2.0.
From Tables 2-9, we have the following;

Remark 1: the following formulations are equal

i. $(3) = (8)$
ii. $(4) = (5) = (6)$
iii. $(2) = (7) = (9)$

Few comments will be appropriate as follows:
From (i), (3) is the disaggregated version of (2) as suggested in [2], and (8) is the disaggregated version of(7), from [18] , and they yield the same bound on Z.

From (ii), the problem studied in [6, 16], [17] and [15] correspond to $(4), (5), and\ (6)$ respectively. They also yield the same bound on Z. The presence of GUB and VUB constraints in these formulations suggest the superiority of the formulations.

From (iii), the problems studied in [10, 12, 13, 14, and 15], [18] and [19] correspond to $(2), (7), and\ (9)$ respectively and the formulations yield the same bound on Z. These bounds, produced by (iii) are weaker than the two other bounds obtained through (i) and (ii). From the foregoing and coupled with the results in Table 10 we have the following remark.

Remark 2: the following relations are valid on the optimal objective values of Z and RZ

i. $(Z_4(.), Z_5(.), Z_6(.)) < (Z_2(.), Z_7(.), Z_9(.))$
ii. $\left(\overline{Z_4}(.), \overline{Z_5}(.), \overline{Z_6}(.)\right) < (\overline{Z_2}(.), \overline{Z_7}(.), \overline{Z_9}(.))$

From A34 to F1010 of the test problem i.e (about 60%)

iii. $(Z_4(.), Z_5(.), Z_6(.)) < (Z_3(.), Z_8(.))$, also ,$(\bar{Z}_4(.), \bar{Z}_5(.), \bar{Z}_6(.))$< $(\bar{Z}_3(.), \bar{Z}_8(.))$

But from G1015 to J1515 of the test problem i.e as the size increase (about 40%)

iv. $(Z_3(.), Z_8(.)) < (Z_4(.), Z_5(.), Z_6(.))$, also, $\bar{Z}_3(.), \bar{Z}_8(.)) < (\bar{Z}_4(.), \bar{Z}_5(.), \bar{Z}_6(.))$

Table 10: Summary of Results

Instance	(2), (7) and (9)		(4), (5), (6)		(3), (8)	
	$Z(.)$	$\bar{Z}(.)$	$Z(.)$	$\bar{Z}(.)$	$Z(.)$	$\bar{Z}(.)$
A34	3820	3491	1007	844	3770	3204
B54	412	397	220	201	312	303
C56	2622	2115	1759	1687	1859	1382
D58	44170	41970	42362	40162	11520	9120
E58	767.5	764.5	344	331	535	460
F1010	2280	1758	1440	918	1774	715
G1015	4230	3746	3427	3046	2537	1209
H1515	10528	10380	10264	10264	10000	1195
I1515	9209	8417	9019	8126	2131	1739
J1515	11789	10973	11542	10799	470	470

5.0 Conclusion

In this paper we have studied Lp relaxation of some formulations of single stage CWLP from two different angles; quality of lower bounds among the corresponding formulations that suits one modeling purpose and which yields a relaxation that can be easily solved among those that yield the same bound. Because of the presence of GUB and VUB constraints, the strong LP-relaxation is relatively easy to solve and provide useful bounds for used in branch and bound algorithms.

The computational results revealed that for the three groups of formulations that yield same bounds, i.e {(2), (7), (9)} {(4), (5), (6)} and {(3), (8)} on the instances having large fixed charges have large integrality gaps than those in the intermediate range.

References

[1] I. Litvinchev, M. Mata, and S. Rangel, "Calculating the best dual bound for problems with multiple lagrangian relaxation", *Journal of Cumputer and systems sciences International*, vol. **49**, No.6 (2010) pp.915-922.

[2] K. Holmberg, "Transportation and Location problems with staircase costs", Research Report (1989) LiTH-MAT-R-89-12, Linkoping institute of technology, Sweden.

[3] A. Klose and S. Gortz, " A branch and price algorithm for the capacitated facility location problem", *Eur. J. of Oper. Res.* **179**, (2007) 1109-1125.

[4] G. Cornuejols, R. Sridharan and J.M. Thizy, "A Comparisons of heuristics and relaxations for the capacitated plant location problem" *Eur. J. of Oper Res.* **50** (1991) 280-287.

[5] N. Abdullahi and B. Sani, ”Capacitated warehouse location problem: A study in formulation efficiency I “, *Abacus, journal of Mathematical Association of Nigeria*, (2012) to appeared.

[6] G.S. Monique and K. Siwhn, “A strong lagrangean relaxation for capacitated plant location problems”, Department of statistics technical Report no.56, (1983) Wharton School, University of Pennsylvania.

[7] A. Geoffrion and R. Mc Bride,“lagrangean relaxation applied to capacitated facility location problems”. *AIIE Transactions* **10**, (1978) No.1, 40-47.

[8] M. Guignard, “Lagrangean Relaxation”, TOP vol.11, No.2 (2003) pp.151-228.

[9] B. Chen and M. Guignard, “polyhedral analysis and decompositions for capacitated plant location problem”, *Discrete applied Mathematics* **82**, (1998) 79-91.

[10] U. Akinc and B.M. Khumawala, “An efficient Branch and Bound algorithm for the capacitated warehouse location problem”, *Management Science* **23** (1977)585-594.

[11] R.M. Nauss,“An improved algorithm for the capacitated facility location problem”, *J. Oper. Res. Soc* **29** (1978) 1195-1201.

[12] M.B. Baker and P.B. Amandio, “Branch and bound algorithm for a water Authority distribution problem”, *J. Oper. Res. Soc*. **46** (1995) 698-707.

[13] G.K. John and L. Hannan, “A lagrangean relaxation heuristic for capacitated facility location with single source constraints”, *J. Oper. Res. Soc* **37**, (***5***) (1986) 495-500.

[14] M.B. Baker,“A partial dual algorithm for the capacitated ware house location problem”, *Eur J. Oper. Res*. **23** (1986) 48-56.

[15] M.C. Lai, S. Han-Suk, T. Tzu-Liang, and C. Chunkuan,“A hybrid algorithm for capacitated plant location problem”, *Expert systems with applications* **37** (2010) 8599-8605.

[16] T.J. Van Roy,“A cross decomposition algorithm for capacitated facility location problem”, *Oper. Res.* **34,** (**1**) (1986) 145-163.

[17] M.Y. Janny Leung and T.L. Magnanti, “Valid inequalities and facets fo the capacitated plant location problem”, Operations Research report, (1986) OR-149-86,Sloan School of Management MIT.

[18] B.M. Baker, “Linear Relaxations of the Capacitated Warehouse Locations Problems”, *J. Oper. Res. Soc*. Vol. **33** (1982) 475-479.

[19] P.S Davis and T.L Roy, “ A branch and Bound algorithm for the Capacitated Warehouse location problem”, *Naval Res. Logist. Q.* **16** (1969), 331-344.

[20] A.M. Geoffrion, “Lagrangean Relaxation for Integer Programming’, *Mathematical programming study* **2**, (1974) 82-114.

[21] M.L. Fisher, “The lagrangean Relaxation method for solving Integer programming problem” *Management science* vol. **50** No. 12, (2004) 1861-1871.

[22] R. Fourer, D. Gay and B.W. Kernighan, “AMPL: A Modeling language for Mathematical Programming” 2nd Eds (2003), Brooks/Cole.