Chapter 8. Discrete Point Location

This is an introduction chapter quotation. It is offset three inches to the right.

8.1. Set Covering and Set Partitioning

Set Covering and Set Partitioning Formulation

Every column j corresponds to a feasible alternative service action. Every row i corresponds to a service request.

x_j	1 if alternative j is executed, 0
a_{ij}	1 if alternative j satisfies request i
c_j	cost of alternative j
p_i	cost estimate for servicing request I

The objective is to minimize the overall cost while servicing all the requests. otherwise, which means that the x are binary variables.

If the service request can be served by exactly one service alternative, then the resulting formulation is called the Set Partitioning Problem (SPP) as shown in the formulation 8.1.

$$\begin{aligned} \min \quad & \sum_{j=1}^{N} c_j x_j \\ s.t. \quad & \sum_{j=1}^{N} a_{ij} x_j = 1 \qquad i = 1..M \\ & x_j \in \{0,1\} \end{aligned} \tag{8.1}$$

If the service request can be served by more than one service alternative, then the resulting formulation is called the Set Covering Problem (SCP), as shown in formulation 8.2. Since in most logistics systems, not servicing one particular service request in a feasible service alternative also is a feasible service alternative and the cost of the reduced service alternative is no larger than the cost of the original service request, the Set Partitioning and Set Covering problems can be solved in the same way. The discussion from now on will focus on the Set Partitioning Problem.

$$\begin{aligned} \min \quad & \sum_{j=1}^{N} c_j x_j & & \\ s.t. \quad & \sum_{j=1}^{N} a_{ij} x_j \geq 1 & i = 1..M & \\ & x_j \in \{0,1\} & & \end{aligned} \tag{8.2}$$

Set Partitioning Problem Characteristics

Alternative Selecting Algorithm

Accurate Costs and Feasibility Constraints

Optimal Solution for "Small" Problem Sizes (IP Solver)

Efficient Column Generation and Pricing

Models Complex Problems

Set Partitioning Algorithm

Notation

To facilitate the description and notation of the algorithm the following set notation is introduced. J^+ is the set of service alternatives currently executed. I° is the set of request currently not serviced.

$$\begin{aligned} J_i &= \{ j : a_{ij} = 1 \} \\ I_j &= \{ i : a_{ij} = 1 \} \\ J^+ &= \{ j : x_j = 1 \} \\ I^\circ &= \left\{ i : i \notin \bigcup_{j \in J^+} I_j \right\} \end{aligned} \tag{8.3}$$

Column Generation Heuristic

Algorithm 8.1. Set Partitioning Column Generation

1) Start with a feasible partition J^+

2) Determine row prices by allocating column prices “equitable” such that

$$c_j^+ = \sum_{i=1}^{M} a_{ij} p_i \qquad (8.4)$$

3) Generate and evaluate a column j. If its "reduced price" is negative, add the column to the partitioning master problem.

$$c_j - \sum_{i=1}^{M} a_{ij} p_i \leq 0 \qquad (8.5)$$

4) If enough columns are added, solve the partitioning master problem and go to step 2. If not all columns have been evaluated go to step 3.

5) If all columns have been evaluated and none have been added or the solution or row prices are within the tolerance, stop, else solve the partitioning master problem and go to step 2.

The SPP master problem is a pure integer (binary) programming problem and must be solved with an integer programming solver. Problems of intermediate size can be solved to optimality in a reasonable amount of time. Since only potentially attractive columns are added in step 3, the master problem size is significantly reduced. The number of rejected columns depends on the quality of the row prices which in turn depends on the quality of the current feasible partition. So the algorithm efficiency is greatly enhanced if it is started with a good initial feasible partition. This partition can be generated either with an heuristic algorithm or can be the current configuration for an existing system.

Row Price Allocation Schemes

There are many alternative schemes to determine "equitable" row prices and the selection of such scheme depends on the physical characteristics of the problem. For example, in the case of locating a distribution center that services a number of customers with demand dem_i and distance to the distribution center d_{0i}, a reasonable allocation of the cost can be based on the product of demand and distance to the distribution center, or

$$p_i = \frac{dem_i \cdot d_{0i}}{\sum_{i \in I_j} dem_i \cdot d_{0i}} c_j^+ \tag{8.6}$$

The column generation algorithm is only guaranteed to yield the optimal solution if all feasible columns are evaluated with all possible combinations of row prices satisfying (8.4) and no column can be added to the master problem, i.e., all have a positive reduced cost as computed by (8.5).

Reduction Rules

The size of the master problem can be further reduced by applying a set of reduction rules.

Row Infeasibility

If $J_k = \varnothing$ then the problem is infeasible

Since no service alternative can satisfy service request k the problem has no feasible solution.

Row Feasibility

if $J_r = \{c\}$ then $x_c = 1, c \in J^+$ and eliminate rows $k : k \in I_c$.

Since service request *r* can only be served by service alternative *c*, alternative *c* must be executed and all service request served by it can be eliminated from the problem. The cost of service alternative c must be added to the current objective function value.

Row Dominance

If $J_r \subseteq J_q$ then eliminate row *q*

Since whatever service alternative that services request *r* will also service request *q*, request *q* can be eliminated from the problem. So the "easier" service requests are eliminated from the problem.

Column Dominance

If $I_t = \varnothing$ then $x_t = 0$

If $I_s \supseteq I_t$ *and* $c_s \leq c_t$ then $x_t = 0$

If a particular service alternative does not serve any request, then it can be eliminated from the problem. If a particular service alternative services more requests and costs less than another alternative t, then the alternative t can never be included in an optimal solution and can thus be eliminated from the problem.

The rules can be applied in any sequence and the application should cycle through all the rules until no further reductions can be made.

Reduction Example

A study is being conducted to determine the optimum number and locations of fire towers in a large national forest. The forest consists of 12 tracts that need to be surveyed by rangers located in the fire towers. The objective is to minimize the total cost of establishing the fire towers. Eight locations have been selected as potential sites for the fire towers due to their altitudes and visibility ranges. The coverage matrix is given in Table 8.1 with the last row showing the cost of constructing a fire tower at each of the eight sites.

Table 8.1. Original Cost and Coverage Matrix

i↓ / j→	1	2	3	4	5	6	7	8
1	1	1	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0
3	0	1	1	0	0	0	0	0
4	1	1	0	1	0	0	0	0
5	0	1	1	1	0	1	0	0
6	0	0	1	1	1	1	0	1
7	1	0	1	1	1	1	0	0
8	1	0	0	1	0	0	0	0
9	0	0	0	1	1	1	1	1
10	1	0	0	0	1	0	1	0
11	0	0	0	0	1	1	1	1
12	0	0	0	0	1	1	1	1
c_j	175	225	145	115	105	165	135	195

The objective is to determine the correct locations of fire towers to be established and the minimal cost solution in the most efficient way. The following reductions are possible during the preprocessing phase. In the reduction phase the terms tract and row, and also tower and column will be used interchangeably.

Track 2 requires tower 2 since its row contains only a single 1, so tower 2 must be established. This eliminates all tracks that are served by tower 2, i.e., {1, 2, 3, 4, 5} and it also eliminates column 2. The resulting matrix is as follows.

Table 8.2. Cost and Coverage Matrix after One Reduction

i↓ / j→	1	3	4	5	6	7	8
6	0	1	1	1	1	0	1
7	1	1	1	1	1	0	0
8	1	0	1	0	0	0	0
9	0	0	1	1	1	1	1
10	1	0	0	1	0	1	0
11	0	0	0	1	1	1	1
12	0	0	0	1	1	1	1
c_j	175	145	115	105	165	135	195

Tract 9 is dominated by tract 12, since row 9 has a one wherever row 12 has a one and thus if tract 12 is covered then tract 9 will also be covered. Similarly, track 11 is dominated by tract 12, and track 7 is dominated by tract 8. This eliminates rows 7, 9, and 11. The resulting matrix is as follows.

Table 8.3. Cost and Coverage Matrix after Two Reductions

i↓ / j→	1	3	4	5	6	7	8
6	0	1	1	1	1	0	1
8	1	0	1	0	0	0	0
10	1	0	0	1	0	1	0
12	0	0	0	1	1	1	1
c_j	175	145	115	105	165	135	195

Tower 5 dominates tower 6 since its column has a one wherever column 6 has a one and its cost is lower, so tower 6 can be eliminated. Tower 5 dominates tower 7 and its cost is lower, so tower 7 can be eliminated. Tower 5 dominates tower 8 and its cost is lower, so tower 8 can be eliminated. Tower 4 dominates tower 3 and its cost is lower, so tower 3 can be eliminated. This eliminates columns 3, 6, 7, and 8. The resulting matrix is as follows.

Table 8.4. Cost and Coverage Matrix after Three Reductions

i↓ / j→	1	4	5
6	0	1	1
8	1	1	0
10	1	0	1
12	0	0	1
c_j	175	115	105

Track 12 requires tower 5 since its row contains only a single 1, so tower 5 must be established. This eliminates all tracks that are served by tower 5, i.e., {6, 10, 12} and it also eliminates column 5. The resulting matrix is as follows.

Table 8.5. Final Cost and Coverage Matrix after Four Reductions

i↓ / j→	1	4
8	1	1
c_j	175	115

Tower 4 dominates tower 1 and its cost is lower, so tower 1 can be eliminated. This leaves only tower 4 covering tract 8 and the problem has been solved to optimality by reduction only. The optimal solution is $x_2 = x_4 = x_5 = 1$ and all other x_j equal to zero. The optimal solution value is 225 + 115 + 105 = 445.

Algorithm 8.2 Greedy Set Partitioning Heuristic

$I^\circ = I,\ J^+ = \varnothing$

$while\, I^\circ \neq \varnothing$ {

$f(c_v, k_v) = \min_j \left\{ f(c_j, k_j) \middle| j \notin J^+ \right\}$

$J^+ \leftarrow J^+ \cup \{v\}$

$I^\circ \leftarrow I^\circ \setminus \left(I^\circ \cap I_v \right)$

} *endwhile*

There exist many alternatives for the "reduced cost" function *f(c, k)*. A particular alternative is the inverse of the "most bang for the buck" function, where the cost of the service alternative is divided by the number service requests it satisfies and that are not currently serviced, or

$$f(c_j, k_j) = \frac{c_j}{k_j} = \frac{c_j}{\left| I^\circ \cap I_j \right|} \tag{8.7}$$

Linear Relaxation and Cutting Planes

$$\begin{aligned} & \min \sum_{j=1}^{N} c_j x_j \\ & s.t. \sum_{j=1}^{N} a_{ij} x_j \geq 1 \qquad i = 1..M \\ & \qquad 0 \leq x_j \leq 1 \end{aligned} \tag{8.8}$$

The Covering Problem (CP) is a set covering problem for which all cover cost are equal to one. The CP is then the un-weighted or counting variant of the SCP.

For the Covering Problem usually the following single cut added to the linear relaxation suffices to find the optimal integer solution

$$\sum_{j=1}^{N} x_j \geq \left\lceil z_{LR}^* \right\rceil \tag{8.9}$$

Branch-And-Bound

Branch-and-Bound is the most widely used methodology to solve integer programming problems.

Procedures

- Branching Variable Selection
- Lower Bound Computation
- Primal Heuristics (Incumbent Solution)

8.2. Generalized n-Median Problem

Generalized n-Median Formulation

$$\begin{aligned}
\min \quad & z = \sum_{j=1}^{N} \left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right) \\
s.t. \quad & \sum_{j=1}^{N} a_{ij} x_{ij} = 1 \\
& -x_{ij} + y_j \geq 0 \\
& \sum_{j=1}^{N} y_j \leq n \\
& y_j \in \{0,1\} \\
& x_{ij} \geq 0
\end{aligned} \tag{8.10}$$

Lagrangean Relaxation

If we relax the assignment constraints using Lagrangean multipliers,

$$\sum_{j=1}^{N} a_{ij} x_{ij} = 1 \qquad [u_i] \tag{8.11}$$

the resulting objective function is

$$\min \quad z_{LAR} = \sum_{j=1}^{N}\left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right) + \sum_{i=1}^{M} u_i \left(1 - \sum_{j=1}^{N} a_{ij} x_{ij} \right) \tag{8.12}$$

and the Lagrangean relaxation is then

$$\begin{aligned} \min \quad & z_{LAR}(U) = \sum_{j=1}^{N}\left(f_j y_j + \sum_{i=1}^{M} \left(c_{ij} - a_{ij} u_i\right) x_{ij} \right) + \sum_{i=1}^{M} u_i \\ s.t. \quad & -x_{ij} + y_j \geq 0 \\ & \sum_{j=1}^{N} y_j \leq n \\ & y_j \in \{0,1\}, x_{ij} \geq 0 \end{aligned} \tag{8.13}$$

Observe that the objective function of the Lagrangean relaxation is a lower bound to the objective function of the original problem, or

$$z_{LAR}(U) \leq z^* \tag{8.14}$$

We can condense this formulation by introducing a site's relative cost factor

$$\rho_j(U) = f_j + \sum_{i=1}^{M} \min\left\{0, c_{ij} - a_{ij} u_i\right\} \tag{8.15}$$

$$\begin{aligned} \min \quad & z_{LAR}(U) = \sum_{i=1}^{M} u_i + \min \sum_{j=1}^{N} \rho_j(U) y_j \\ s.t. \quad & \sum_{i=1}^{N} y_j \leq n \\ & y_j \in \{0,1\} \end{aligned} \tag{8.16}$$

Lagrangean Subproblem

1. Compute and Sort $\rho_j(U)$ in Increasing Order
2. Discard All Nonnegative $\rho_j(U)$
3. If No $\rho_j(U)$ Remaining, Pick Single Smallest Positive $\rho_j(U)$
4. Else Pick, up to n, Most Negative $\rho_j(U)$

Algorithm 8.3. Greedy Heuristic for the n-Median Lagrangean Relaxation

$J^{+} = \varnothing, k = 0$

$\rho_{r(1)} = \min_{j} \{\rho_j (U^t)\}$

$if \;\; \rho_{r(1)} > 0, \;\; J^{+} \leftarrow \{r(1)\}, \; z_{LAR} = \sum_{i=1}^{M} u_i + \rho_{r(1)}, \; stop$

$else \;\; while \;\; \rho_{r(k)} < 0 \;\; and \;\; \left|J^{+}\right| < n \;\; \{$

$\quad J^{+} \leftarrow J^{+} \cup \{r(k)\}$

$\quad k \leftarrow k+1$

$\quad \rho_{r(k)} = \min_{j \in J \backslash J^{+}} \{\rho_j (U)\}$

$\quad \} \;\; endwhile$

$z_{LAR} = \sum_{i=1}^{M} u_i + \sum_{j \in J^{+}} \rho_j$

Dual Master Problem

Choice of the Lagrangean multipliers.

$$u_i = \frac{1}{N} \sum_{j=1}^{N} c_{ij} \tag{8.17}$$

$$u_i = \max_{j} \{c_{ij}\} \tag{8.18}$$

Lagrangean relaxation dual master problem

$$\max_{U} \; z_{LAR}(U) \tag{8.19}$$

This problem can be solved by subgradient optimization of by heuristic adjustment of the Lagrangean multipliers.

Algorithm 8.4. n-Median Dual Adjustment Heuristic

$J^{+} = \varnothing, k = 0$

for $i = 1$ *to* M

$$u_i^t = \max_j \{c_{ij}\}$$

$\rho_{r(k)} = \min_j \{\rho_j(U^t)$

if $\rho_{r(1)} > 0,\ J^{+} \leftarrow \{r(1)\},\ z = f_{r(1)} + \sum_{i=1}^{M} c_{ir(1)},$ *stop*

else while $\rho_{r(k)} < 0$ *and* $\left|J^{+}\right| < n$ {

$$J^{+} \leftarrow J^{+} \cup \{r(k)\}$$

$$u_i^{k+1} \leftarrow \min\left\{u_i^k, c_{ir(k)}\right\}$$

$$k \leftarrow k+1$$

$$\rho_{r(k)} = \min_{j \in J \setminus J^{+}} \{\rho_j(U)\}$$

} *endwhile*

$$z = \sum_{j \in J^{+}} f_j + \sum_{i=1}^{M} \min_{j \in J^{+}} \{c_{ij}\}$$

8.3. Warehouse Location Problem

Warehouse Location Problem Definition

Determine the location of the warehouses and their associated customer zones

Satisfying the given deterministic customer demands

Minimizing a sum of concave costs (fixed site and constant marginal transportation)

Warehouse Location Formulation

$$
\begin{aligned}
\min \quad & z = \sum_{j=1}^{N}\left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right) \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 \\
& -x_{ij} + y_j \ge 0 \\
& y_j \in \{0,1\} \\
& x_{ij} \ge 0
\end{aligned}
\tag{8.20}
$$

Erlenkotter DUALOC procedure

Compact dual formulation

Dual Ascent, then dual adjustment

Primal heuristic

Used as bound in branch-and-bound

Magnitude faster than contemporary algorithms

Linear relaxation

$$
\begin{aligned}
\min \quad & z_{LR} = \sum_{j=1}^{N}\left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right) \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && [u_i] \\
& -x_{ij} + y_j \ge 0 && [w_{ij}] \\
& y_j \ge 0,\ \ x_{ij} \ge 0
\end{aligned}
\tag{8.21}
$$

Dual formulation

$$
\begin{aligned}
\max \quad & z_D = \sum_{i=1}^{M} u_i \\
s.t. \quad & \sum_{i=1}^{M} w_{ij} \le f_j & [y_j] \\
& u_i - w_{ij} \le c_{ij} & [x_{ij}] \\
& w_{ij} \ge 0 \\
& u_i \quad unrestricted
\end{aligned}
\tag{8.22}
$$

Variable transformations

$$
\begin{aligned}
& w_{ij} \ge u_i - c_{ij} \quad and \quad w_{ij} \ge 0 \\
& w_{ij} = \max\left\{0, u_i - c_{ij}\right\} \\
& \sum_{i=1}^{M} w_{ij} = \sum_{i=1}^{M} \max\left\{0, u_i - c_{ij}\right\} \le f_j \\
& \rho_j(U) = f_j + \sum_{i=1}^{M} \min\left\{0, c_{ij} - u_i\right\} \ge 0
\end{aligned}
\tag{8.23}
$$

Condensed dual formulation

$$
\begin{aligned}
\max \quad & z_D = \sum_{i=1}^{M} u_i \\
s.t. \quad & \rho_j(U) \ge 0 \\
& u_i \quad unrestricted
\end{aligned}
\tag{8.24}
$$

Complementary slackness conditions

$$
y_j \cdot \rho_j(U) = 0 \quad \Rightarrow \quad
\begin{aligned}
& if \quad y_j = 1 \ then \ \rho_j(U) = 0 \\
& if \quad \rho_j(U) \ne 0 \ then \ y_j = 0
\end{aligned}
\tag{8.25}
$$

$$
x_{ij}\left(c_{ij} - u_i + w_{ij}\right) = 0 \quad \Rightarrow \quad
\begin{aligned}
& if \quad x_{ij} = 1 \ then \ u_i \ge c_{ij} \\
& if \quad u_i < c_{ij} \ then \ x_{ij} = 0
\end{aligned}
\tag{8.26}
$$

$$
w_{ij}\left(y_j - x_{ij}\right) = 0 \quad \Rightarrow \quad
\begin{aligned}
& if \quad u_i > c_{ij} \ then \ x_{ij} = y_j \\
& if \quad x_{ij} < y_j \ then \ u_i \le c_{ij}
\end{aligned}
\tag{8.27}
$$

Primal Heuristic

$$J^+ = \left\{ j \middle| \rho_j(U) = 0 \right\} \cap \left\{ j \middle| \exists i \middle| c_{i\alpha(i)} = \min_j \left\{ c_{ij} \middle| \rho_j(U) = 0 \right\} \right\} \cap \left\{ j \middle| \forall i \middle| \exists j \middle| \left(\rho_j(U) = 0\right) \cap \left(u_i > c_{ij}\right) \right\} \tag{8.28}$$

$$\forall i \quad x^+_{i\alpha(i)} = 1 \middle| c_{i\alpha(i)} = \min_j \left\{ c_{ij} \middle| j \in J^+ \right\} \tag{8.29}$$

Dual Ascent Method

Increase each dual variable in turn until one or more dual constraints become binding

$$\begin{aligned} \Delta_1 &= \min_j \left\{ \rho_j(U) \middle| u_i > c_{ij} \right\} \\ \Delta_2 &= \min_j \left\{ c_{ij} - u_i \middle| u_i < c_{ij} \right\} \\ \Delta &= \min\{\Delta_1, \Delta_2\} \end{aligned} \tag{8.30}$$

Algorithm 8.5. Erlenkotter Dual Ascent Algorith m for WLP

$I^a = I, \; u_i \leftarrow \min_j \{c_{ij}\}$

while $I^a \neq \varnothing$

for $i \in I^a$

$$\Delta = \min \left\{ \begin{aligned} &\min_j \left\{ \rho_j(U) \middle| u_i \geq c_{ij} \right\}, \\ &\min_j \left\{ c_{ij} - u_i \middle| u_i < c_{ij} \right\} \end{aligned} \right\}$$

if $\Delta > 0$ *then* $u_i \leftarrow u_i + \Delta,$ *update* $\rho_j(U)$

else $I^a \leftarrow I^a \setminus \{i\}$

Violations of complementary slackness correspond to primal-dual gap. Make set J^+ as small as possible by dropping non-essential sites.

$$z_P - z_D = \sum_{i=1}^{M} \sum_{j \in J^+ \cap j \neq \alpha(i)} w_{ij} \tag{8.31}$$

Table 8.6. Erlenkotter Example 1, Initial Data

	1	2	3	4	5	u_i^1	Δ_1	Δ_2	u_i^2
1	*120*	*210*	*180*	*210*	*170*				
2	*180*	∞	*190*	*190*	*150*				
3	*100*	*150*	*110*	*150*	*110*				
4	∞	*240*	*195*	*180*	*150*				
5	*60*	*55*	*50*	*65*	*70*				
6	∞	*210*	∞	*120*	*195*				
7	*180*	*110*	∞	*160*	*200*				
8	∞	*165*	*195*	*120*	∞				
f_j	*100*	*70*	*60*	*110*	*80*				
ρ_j^1									
1									
2									
3									
4									
5									
6									
7									
8									
ρ_j^2									

Table 8.7. Erlenkotter Example 1, Iteration 1

	1	2	3	4	5	u_i^1	Δ_1	Δ_2	u_i^2
1	*120*	*210*	*180*	*210*	*170*	*120*	*100*	*50*	*170*
2	*180*	∞	*190*	*190*	*150*	*150*	*80*	*30*	*180*
3	*100*	*150*	*110*	*150*	*110*	*100*	*50*	*10*	*110*
4	∞	*240*	*195*	*180*	*150*	*150*	*50*	*30*	*180*
5	*60*	*55*	*50*	*65*	*70*	*50*	*60*	*5*	*55*
6	∞	*210*	∞	*120*	*195*	*120*	*110*	*75*	*195*
7	*180*	*110*	∞	*160*	*200*	*110*	*70*	*50*	*160*
8	∞	*165*	*195*	*120*	∞	*120*	*35*	*45*	*(b)155*
f_j	*100*	*70*	*60*	*110*	*80*	*920*			*1205*
ρ_j^1	*100*	*70*	*60*	*110*	*80*				
1	*50*								
2					*50*				
3	*40*								
4					*20*				
5			*55*						
6				*35*					
7		*20*							
8				*0*					
ρ_j^2	*40*	*20*	*55*	*0*	*20*				

Note that (b) means that the dual variable u_i is blocked after this iteration, i.e. can not be further increased. Blocked variables do not have to the examined in the next iteration.

We can use the current dual variables to obtain a primal solution as explained before. Since only one slack variable equals zero, only one distribution center can be opened, or $y_4 = 1$. This contributes its fixed cost of 110 to the solution value. Since only distribution center is open all customers will be served from it and the non-zero primal transportation variables are then $x_{41} = x_{42} = x_{43} = x_{44} = x_{45} = x_{46} = x_{47} = x_{48} = 1$. The transportation cost is then 210 + 190 + 150 + 180 + 65 + 120 + 160 + 120 = 1195. The corresponding primal feasible cost is then 110 + 1195 = 1305. Only for customers 6 and 8 are the dual variables (gross revenue) larger than or equal to the corresponding transportation costs from the open distribution center, so these are the only two customers we would like to serve.

Table 8.8. Erlenkotter Example 1, Iteration 2

	1	2	3	4	5	u_i^1	Δ_1	Δ_2	u_i^2
1	*120*	*210*	*180*	*210*	*170*	*170*	*20*	*10*	*180*
2	*180*	*∞*	*190*	*190*	*150*	*180*	*10*	*10*	*(b) 190*
3	*100*	*150*	*110*	*150*	*110*	*110*	*0*	*0*	*(b) 110*
4	*∞*	*240*	*195*	*180*	*150*	*180*	*0*	*15*	*(b) 180*
5	*60*	*55*	*50*	*65*	*70*	*55*	*20*	*5*	*60*
6	*∞*	*210*	*∞*	*120*	*195*	*195*	*0*	*15*	*(b) 195*
7	*180*	*110*	*∞*	*160*	*200*	*160*	*0*	*20*	*(b) 160*
8	*∞*	*165*	*195*	*120*	*∞*	*(b) 155*			*(b) 155*
f_j	*100*	*70*	*60*	*110*	*80*	*1205*			*1230*
ρ_j^1	*40*	*20*	*55*	*0*	*20*				
1	*30*				*10*				
2	*20*				*0*				
3									
4									
5		*15*	*50*						
6									
7									
8									
ρ_j^2	*20*	*15*	*50*	*0*	*0*				

Table 8.9. Erlenkotter Example 1, Iteration 3

	1	2	3	4	5	u_i^1	Δ_1	Δ_2	u_i^2
1	*120*	*210*	*180*	*210*	*170*	*180*	*0*	*30*	*(b) 180*
2	*180*	*∞*	*190*	*190*	*150*	*(b) 190*			*(b) 190*
3	*100*	*150*	*110*	*150*	*110*	*(b) 110*			*(b) 110*
4	*∞*	*240*	*195*	*180*	*150*	*(b) 180*			*(b) 180*
5	*60*	*55*	*50*	*65*	*70*	*60*	*15*	*5*	*65*
6	*∞*	*210*	*∞*	*120*	*195*	*(b) 195*			*(b) 195*
7	*180*	*110*	*∞*	*160*	*200*	*(b) 160*			*(b) 160*
8	*∞*	*165*	*195*	*120*	*∞*	*(b) 155*			*(b) 155*
f_j	*100*	*70*	*60*	*110*	*80*	*1230*			*1235*
ρ_j^1	*20*	*15*	*50*	*0*	*0*				
1									
2									
3									
4									
5	*15*	*10*	*45*						
6									
7									
8									
ρ_j^2	*15*	*10*	*45*	*0*	*0*				

Table 8.10. Erlenkotter Example 1, Iteration 4

	1	2	3	4	5	u_i^1	Δ_1	Δ_2	u_i^2
1	*120*	*210*	*180*	*210*	*170*	*(b) 180*			*(b) 180*
2	*180*	*∞*	*190*	*190*	*150*	*(b) 190*			*(b) 190*
3	*100*	*150*	*110*	*150*	*110*	*(b) 110*			*(b) 110*
4	*∞*	*240*	*195*	*180*	*150*	*(b) 180*			*(b) 180*
5	*60*	*55*	*50*	*65*	*70*	*65*	*0*	*5*	*(b) 65*
6	*∞*	*210*	*∞*	*120*	*195*	*(b) 195*			*(b) 195*
7	*180*	*110*	*∞*	*160*	*200*	*(b) 160*			*(b) 160*
8	*∞*	*165*	*195*	*120*	*∞*	*(b) 155*			*(b) 155*
f_j	*100*	*70*	*60*	*110*	*80*	*1235*			*1235*
ρ_j^1	*15*	*10*	*45*	*0*	*0*				
1									
2									
3									
4									
5									
6									
7									
8									
ρ_j^2	*15*	*10*	*45*	*0*	*0*				

The primal solution is now derived based on complementary slackness conditions. Since ρ_4 and ρ_5 are equal to zero distribution centers 4 and 5 are candidates to be opened. Just opening center 4 would not open up a distribution center for customer 1 with cost smaller than or equal to its dual variable. Just opening center 5 would not open up a distribution center for customer 5 with cost smaller than or equal to its dual variable. So both distribution centers 4 and 5 need to be opened or $y_4 = y_5 = 1$, which contributes their fixed costs 110 + 80 = 190 to the primal solution. Then the customers are assigned to the cheapest distribution center, which yields $x_{15} = x_{25} = x_{35} = x_{45} = x_{54} = x_{64} = x_{74} = x_{84} = 1$, which contributes their transportation costs 170 + 150 + 110 + 150 + 65 + 120 + 160 + 120 = 1045 to the primal solution. The primal objective is then 190 + 1045 = 1235, equal to the dual objective. No complementary slackness conditions were violated since the transportation cost to the open distribution center(s) to which each customer is **not** assigned is larger than or equal to the corresponding dual variable.

Dual Adjustment Method

Notation for the sets and costs used in the dual adjustment method

$$
\begin{aligned}
J_i^c &= \left\{ j \in J^+ \middle| u_i \geq c_{ij} \right\} \\
J_i^+ &= \left\{ j \in J^+ \middle| u_i > c_{ij} \right\} \\
I_j^+ &= \left\{ i \middle| J_i^c = \{j\} \right\} \\
c_{i\alpha(i)} &= \min_{j \in J^+} \{c_{ij}\} \\
c_{i\beta(i)} &= \min_{j \in J^+, j \neq \alpha(i)} \{c_{ij}\}
\end{aligned}
\tag{8.32}
$$

Algorithm 8.6. Erlenkotter Dual Adjustment Algorithm for WLP

$for \quad i = 1 \quad to \quad M$

$\quad if \left| J_i^+ \right| > 1$

$\quad\quad if \quad I_{\alpha(i)}^+ \cup I_{\beta(i)}^+ \neq \varnothing$

$\quad\quad\quad \Delta = u_i - \max_j \left\{ c_{ij} \middle| u_i > c_{ij} \right\}$

$\quad\quad\quad for \quad j \in J \middle| u_i > c_{ij}$

$\quad\quad\quad\quad \rho_j(U) \leftarrow \rho_j(U) + \Delta$

$\quad\quad\quad u_i \leftarrow u_i - \Delta$

$\quad\quad\quad I^a \leftarrow I_{\alpha(i)}^+ \cup I_{\beta(i)}^+$

$\quad\quad\quad execute \quad dual \quad ascent$

$\quad\quad\quad I^a \leftarrow I^a \cup \{i\}$

$\quad\quad\quad execute \quad dual \quad ascent$

$\quad\quad\quad I^a \leftarrow I$

$\quad\quad\quad execute \quad dual \quad ascent$

$\quad\quad\quad endif$

$\quad\quad endif$

$\quad endfor$

Table 8.11. Erlenkotter Example 2, Initial Data

	1	2	3	4	5	u_i^1	Δ_1	Δ_2	u_i^2
1	*120*	*210*	*180*	*210*	*170*				
2	*180*	∞	*190*	*190*	*150*				
3	*100*	*150*	*110*	*150*	*110*				
4	∞	*240*	*195*	*180*	*150*				
5	*60*	*55*	*50*	*65*	*70*				
6	∞	*210*	∞	*120*	*195*				
7	*180*	*110*	∞	*160*	*200*				
8	∞	*165*	*195*	*120*	∞				
f_j	*200*	*200*	*200*	*400*	*300*				
ρ_j^1									
1									
2									
3									
4									
5									
6									
7									
8									
ρ_j^2									

Table 8.12. Erlenkotter Example 2, Iteration 1

	1	2	3	4	5	u_i^1	Δ_1	Δ_2	u_i^2
1	120	210	180	210	170	120	200	50	170
2	180	∞	190	190	150	150	300	30	180
3	100	150	110	150	110	100	150	10	110
4	∞	240	195	180	150	150	270	30	180
5	60	55	50	65	70	50	200	5	55
6	∞	210	∞	120	195	120	400	75	195
7	180	110	∞	160	200	110	200	50	160
8	∞	165	195	120	∞	120	325	45	165
f_j	200	200	200	400	300	920			1215
ρ_j^1	200	200	200	400	300				
1	150								
2					270				
3	140								
4					240				
5			195						
6				325					
7		150							
8				280					
ρ_j^2	140	150	195	280	240				

Table 8.13. Erlenkotter Example 2, Iteration 2

	1	2	3	4	5	u_i^2	Δ_1	Δ_2	u_i^3
1	120	210	180	210	170	170	140	10	180
2	180	∞	190	190	150	180	130	10	190
3	100	150	110	150	110	110	120	0	110
4	∞	240	195	180	150	180	220	15	195
5	60	55	50	65	70	55	150	5	60
6	∞	210	∞	120	195	195	205	15	210
7	180	110	∞	160	200	160	145	20	180
8	∞	165	195	120	∞	165	125	30	195
f_j	200	200	200	400	300	1215			1320
ρ_j^2	140	150	195	280	240				
1	130				230				
2	120				220				
3									
4				265	205				
5		145	190						
6				250	190				
7		125		230					
8		95		200					
ρ_j^3	120	95	190	200	190				

Table 8.14. Erlenkotter Example 2, Iteration 3

	1	2	3	4	5	u_i^3	Δ_1	Δ_2	u_i^4
1	120	210	180	210	170	180	120	30	210
2	180	∞	190	190	150	190		0	190
3	100	150	110	150	110	110	90	40	150
4	∞	240	195	180	150	195	120	45	240
5	60	55	50	65	70	60	50	5	65
6	∞	210	∞	120	195	210	75	∞	(b) 285
7	180	110	∞	160	200	180	15	20	(b) 195
8	∞	165	195	120	∞	195	0	∞	(b) 195
f_j	200	200	200	400	300	1320			1530
ρ_j^3	120	95	190	200	190				
1	90		160		160				
2									
3	50		120		120				
4			75	155	75				
5	45	90	70						
6		15		80	0				
7	30	0		65					
8									
ρ_j^4	30	0	70	65	0				

Table 8.15. Erlenkotter Example 2, Iteration 4

	1	2	3	4	5	u_i^4	Δ_1	Δ_2	u_i^5
1	120	210	180	210	170	210	0	0	(b) 210
2	180	∞	190	190	150	190	0	∞	(b) 190
3	100	150	110	150	110	150	0	0	(b) 150
4	∞	240	195	180	150	240	0	∞	(b) 240
5	60	55	50	65	70	65	0	5	(b) 65
6	∞	210	∞	120	195	(b) 285			(b) 285
7	180	110	∞	160	200	(b) 195			(b) 195
8	∞	165	195	120	∞	(b) 195			(b) 195
f_j	200	200	200	400	300	1530			1530
ρ_j^4	30	0	70	65	0				
1									
2									
3									
4									
5									
6									
7									
8									
ρ_j^5	30	0	70	65	0				

The primal solution is now derived based on complementary slackness conditions. Since ρ_2 and ρ_5 are equal to zero distribution centers 2 and 5 are candidates to opened. Just opening 2 would not open up a distribution center for customer 2 with cost smaller than or equal to its dual variable. Just opening 5 would not open up a distribution center for customer 5 with cost smaller than or equal to its dual variable. So both distribution centers 2 and 5 need to be opened or $y_2 = y_5 = 1$, which contributes their fixed costs 200 + 300 = 500 to the primal solution. Then the customers are assigned to the cheapest distribution center, which yields $x_{15} = x_{25} = x_{35} = x_{45} = x_{52} = x_{65} = x_{72} = x_{82} = 1$, which contributes their transportation costs 170 + 150 + 110 + 150 + 55 + 195 + 110 + 165 = 1105 to the primal solution. The primal objective is then 500 + 1105 = 1605, which is not equal to the dual objective function value of 1530. Some complementary slackness conditions have to be violated, i.e. the transportation cost to the open distribution center(s) to which each customer is **not** assigned is strictly smaller than the corresponding dual variable. This is the case for customer 6 with a dual variable of 285 but a second best assignment cost of 210. A dual adjustment phase is required.

The dual variable u_6 is decreased from 285 to its next lower level of 210. The affected slacks ρ_2, ρ_4, and ρ_5 are each increased by 75. The best and second best distribution center for customer 6 are centers 2 and 5. The customers which can only be served by one of these distribution centers form prime candidates to increase their dual variables, since two dual variables might be increased to take up the slack caused by decreasing the one dual variable. This would yield a net increase in the dual objective function. The customers which can be served ***only*** by center 2 are 5, 7 and 8, i.e., $J_2^+ = \{5,7,8\}$ and the customer which can be served ***only*** by center 5 is 2, i.e., $J_5^+ = \{2\}$. The union of those customers {2, 5, 7, 8} is examined in the first pass of the dual adjustment procedure.

Table 8.16. Erlenkotter Example 2, Iteration 5

	1	2	3	4	5	u_i^5	Δ_1	Δ_2	u_i^6
1	*120*	*210*	*180*	*210*	*170*	*210*			*210*
2	*180*	∞	*190*	*190*	*150*	*190*	*30*	∞	*(b) 220*
3	*100*	*150*	*110*	*150*	*110*	*150*			*150*
4	∞	*240*	*195*	*180*	*150*	*240*			*240*
5	*60*	*55*	*50*	*65*	*70*	*65*	*0*	*5*	*(b) 65*
6	∞	*210*	∞	*120*	*195*	*210*			*210*
7	*180*	*110*	∞	*160*	*200*	*195*	*0*	*5*	*(b) 195*
8	∞	*165*	*195*	*120*	∞	*195*	*40*	∞	*(b) 235*
f_j	*200*	*200*	*200*	*400*	*300*	*1455*			*1525*
ρ_j^5	*30*	*75*	*70*	*140*	*75*				
2	*0*		*40*	*110*	*45*				
5									
7									
8		*35*	*0*	*70*					
ρ_j^6	*0*	*35*	*0*	*70*	*45*				

In the second pass of the dual adjustment procedure, all customers are eligible to have their dual variables changed.

Table 8.17. Erlenkotter Example 2, Iteration 6

	1	2	3	4	5	u_i^6	Δ_1	Δ_2	u_i^7
1	*120*	*210*	*180*	*210*	*170*	*210*	*0*	∞	*(b) 210*
2	*180*	∞	*190*	*190*	*150*	*(b) 220*			*(b) 220*
3	*100*	*150*	*110*	*150*	*110*	*150*	*0*		*(b) 150*
4	∞	*240*	*195*	*180*	*150*	*240*	*0*	∞	*(b) 240*
5	*60*	*55*	*50*	*65*	*70*	*(b) 65*			*(b) 65*
6	∞	*210*	∞	*120*	*195*	*210*	*35*	∞	*(b) 245*
7	*180*	*110*	∞	*160*	*200*	*(b) 195*			*(b) 195*
8	∞	*165*	*195*	*120*	∞	*(b) 235*			*(b) 235*
f_j	*200*	*200*	*200*	*400*	*300*	*1525*			*1560*
ρ_j^6	*0*	*35*	*0*	*70*	*45*				
1									
2									
3									
4									
5									
6		*0*		*35*	*10*				
7									
8									
ρ_j^7	*0*	*0*	*0*	*35*	*10*				

The primal solution is now derived based on complementary slackness conditions. Since ρ_1, ρ_2, and ρ_3 are equal to zero distribution centers 1, 2 and 3 are candidates to opened. In order to serve customer 6 distribution center 2 must be opened, i.e. distribution center 2 is essential. Opening distribution center 2 covers customers 1, 3, 4, 5, 6, 7, and 8 since the transportation cost is lower than or equal to the corresponding dual variable. To serve customer 2, distribution center 1 is opened since it is cheaper with respect to customer 2 than distribution center 3 is. So distribution centers 1 and 2 need to be opened or $y_1 = y_2 = 1$, which contributes their fixed costs 200 + 200 = 400 to the primal solution. Then the customers are assigned to the cheapest distribution center, which yields $x_{11} = x_{21} = x_{31} = x_{42} = x_{52} = x_{62} = x_{72} = x_{82} = 1$, which contributes their transportation costs 120 + 180 + 100 + 240 + 55 + 210 + 110 + 165 = 1180 to the primal solution. The primal objective is then 1180 + 400 = 1580, which is not equal to the dual objective function value of 1560. Some complementary slackness conditions have to be violated, i.e. the transportation cost to the open distribution center(s) to which each customer is **not** assigned is strictly smaller than the corresponding dual variable. This is the case for customer 5 with a dual variable of 65 but a second best assignment cost of 55. A dual adjustment phase is required.

We decrease u_5 from 65 to its next lower value of 60. The affected slacks ρ_1, ρ_2, and ρ_3 are each increased by 5. The cheapest and second cheapest open distribution centers for serving customer 5 are centers 1 and 2. The customers which can be served ***only*** by center 2 are 4, 6 and 8, i.e., $J_2^+ = \{4,6,8\}$ and the customer which can be served ***only*** by center 1 is 2, i.e., $J_1^+ = \{2\}$. The union of those customers {2, 4, 6, 8} is examined in the first pass of the dual adjustment procedure.

Table 8.18. Erlenkotter Example 2, Iteration 7

	1	2	3	4	5	u_i^7	Δ_1	Δ_2		u_i^8
1	120	210	180	210	170	210				210
2	180	∞	190	190	150	220	5	∞	(b)	225
3	100	150	110	150	110	150				150
4	∞	240	195	180	150	240	0	∞	(b)	240
5	60	55	50	65	70	60				60
6	∞	210	∞	120	195	245	5	∞	(b)	250
7	180	110	∞	160	200	195				195
8	∞	165	195	120	∞	235	0		(b)	235
f_j	200	200	200	400	300	1555				1565
ρ_j^7	5	5	5	35	10					
2	0		0	30	5					
4										
6		0		25	0					
8										
ρ_j^8	0	0	0	25	0					

In the second pass of the dual adjustment procedure all customers are eligible to have their dual variables changed.

Table 8.19. Erlenkotter Example 2, Iteration 8

	1	2	3	4	5		u_i^8	Δ_1	Δ_2		u_i^9
1	120	210	180	210	170		210	0	∞	(b)	210
2	180	∞	190	190	150	(b)	225			(b)	225
3	100	150	110	150	110		150	0		(b)	150
4	∞	240	195	180	150	(b)	240			(b)	240
5	60	55	50	65	70		60	0		(b)	60
6	∞	210	∞	120	195	(b)	250			(b)	250
7	180	110	∞	160	200		195	0		(b)	195
8	∞	165	195	120	∞	(b)	235			(b)	235
f_j	200	200	200	400	300		1565				1565
ρ_j^8	0	0	0	25	0						
1											
2											
3											
4											
5											
6											
7											
8											
ρ_j^9	0	0	0	25	0						

The primal solution is now derived based on complementary slackness conditions. Since ρ_1, ρ_2, ρ_3, and ρ_5 are equal to zero distribution centers 1, 2, 3, and 5 are candidates to opened. In order to serve customer 6 either distribution center 2 or 5 must be opened and since distribution center 5 has the cheapest cost it is opened. Opening distribution center 5 covers customers 1, 2, 3, 4, and 6 since the transportation cost is lower than or equal to the corresponding dual variable. Opening distribution center 2 covers the remaining customers 5, 7, and 8. So distribution centers 2 and 5 are opened or $y_2 = y_5 = 1$, which contributes their fixed costs 200 + 300 = 500 to the primal solution. Then the customers are assigned to the cheapest distribution center, which yields $x_{15} = x_{25} = x_{35} = x_{45} = x_{52} = x_{65} = x_{72} = x_{82} = 1$, which contributes their transportation costs 170 + 150 + 110 + 150 + 55 + 195 + 110 + 165 = 1105 to the primal solution. The primal objective is then 500 + 1105 = 1605, which is not equal to the dual objective function value of 1565. Some complementary slackness conditions have to be violated, i.e. the transportation cost to the open distribution center(s) to which each customer is **not** assigned is strictly smaller than the corresponding dual variable. This is the case for customer 6 with a dual variable of 250 but a second best assignment cost of 210. A dual adjustment phase is required.

During a previous iteration, distribution centers 1 and 2 were opened and this is again acceptable since their corresponding slack variables are again equal to zero. Opening centers 1 and 2 contributes their fixed costs 200 + 200 = 400 to the primal solution. Then the customers are assigned to the cheapest distribution center, which yields $x_{11} = x_{21} = x_{31} = x_{42} = x_{52} = x_{62} = x_{72} = x_{82} = 1$, which contributes their transportation costs 120 + 180 + 100 + 240 + 55 + 210 + 110 + 165 = 1180 to the primal solution. The primal objective is then 400 + 1180 = 1580, which is not equal to the dual objective function value of 1565. The gap between the primal objective function value and the dual lower bound is 15 or less than 1 %. Some complementary slackness conditions have to be violated, i.e. the transportation cost to the open distribution center(s) to which each customer is **not** assigned is strictly smaller than the corresponding dual variable. This is the case for customer 7 with a dual variable of 195 but a second best assignment cost of 180. A dual adjustment phase is required.

Alternative Aggregated Formulation

$$\begin{aligned}
\min \quad & z = \sum_{j=1}^{N}\left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right) \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && \forall i \\
& \sum_{i=1}^{M} x_{ij} - M y_j \le 0 && \forall j \\
& y_j \in \{0,1\}, x_{ij} \ge 0
\end{aligned} \tag{8.33}$$

Its linear relaxation

$$\begin{aligned}
\min \quad & z = \sum_{j=1}^{N}\left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right) \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && \forall i \\
& \sum_{i=1}^{M} x_{ij} - M y_j \le 0 && \forall j \\
& y_j \ge 0, x_{ij} \ge 0
\end{aligned} \tag{8.34}$$

Substitution at optimality

$$\sum_{i=1}^{M} x_{ij}^* = M y_j^* \tag{8.35}$$

Condensed linear relaxation

$$\begin{aligned}
\min \quad & z = \sum_{j=1}^{N}\sum_{i=1}^{M}\left(\frac{f_j}{M} + c_{ij} \right) x_{ij} \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && \forall i \\
& x_{ij} \ge 0
\end{aligned} \tag{8.36}$$

Greedy optimal algorithm

Algorithm 8.7. Optimal Greedy Algorithm for Linear Relaxation of the Aggregate WLP

$for \;\; i = 1 \;\; to \;\; M$

$$x_{ik}^{*} = 1 \left| \left(\frac{f_k}{M} + c_{ik} \right) = \min_j \left\{ \frac{f_j}{M} + c_{ij} \right\} \right.$$

$for \;\; j = 1 \;\; to \;\; N$

$$y_j^{*} = \frac{1}{M} \sum_{i=1}^{M} x_{ij}^{*}$$

Comments

IP are identical, but linear relaxations are not!

Aggregated versus disaggregated constraints

“Weak” and “Strong” formulation

Example

Table 8.20. Aggregate WLP Example Data

	1	2	3	4	5
1	*120*	*210*	*180*	*210*	*170*
2	*180*	*∞*	*190*	*190*	*150*
3	*100*	*150*	*110*	*150*	*110*
4	*∞*	*240*	*195*	*180*	*150*
5	*60*	*55*	*50*	*65*	*70*
6	*∞*	*210*	*∞*	*120*	*195*
7	*180*	*110*	*∞*	*160*	*200*
8	*∞*	*165*	*195*	*120*	*∞*
f_j	*100*	*70*	*60*	*110*	*80*
f_j/M	*12.50*	*8.75*	*7.50*	*13.75*	*10.00*
1					
2					
3					
4					
5					
6					
7					
8					
y_j					

Table 8.21. Aggregate WLP Solution

	1	2	3	4	5
1	*132.50*	*218.75*	*187.50*	*223.75*	*180.00*
2	*192.50*	∞	*197.50*	*203.75*	*160.00*
3	*112.50*	*158.75*	*117.50*	*163.75*	*120.00*
4	∞	*248.75*	*202.50*	*193.75*	*160.00*
5	*72.50*	*63.75*	*57.50*	*78.75*	*80.00*
6	∞	*218.75*	∞	*133.75*	*205.00*
7	*192.50*	*118.75*	∞	*173.75*	*210.00*
8	∞	*173.75*	*202.50*	*133.75*	∞
f_j	*100.00*	*70.00*	*60.00*	*110.00*	*80.00*
f_j/M	*12.50*	*8.75*	*7.50*	*13.75*	*10.00*
1	*1*				
2					*1*
3	*1*				
4					*1*
5			*1*		
6				*1*	
7		*1*			
8				*1*	
y_j	*0.250*	*0.125*	*0.125*	*0.250*	*0.250*

Fixed costs = 88.75

Allocation costs = 970

Total cost = 1058.75 (lower bound)

Round-up cost = 1390, (primal heuristic cost)

Optimal cost = 1235

8.4. Capacitated Covering

Formulation

In many cases the existing facilities have a certain resource demand and the new facilities have a resource capacity. Examples are the clustering of customers into truck routes or the location of fire stations where each fire station can serve at most a certain number of neighborhoods. The objective is to minimize the number of new facilities required to serve the existing facilities. The formulation for this problem is given next.

$$
\begin{aligned}
&\text{Min.} && \sum_{i=1}^{M} y_i \\
&\text{s.t.} && \sum_{i=1}^{M} x_{ij} = 1 && \forall j \\
& && \sum_{j=1}^{N} x_{ij} w_j \le cap_i \cdot y_i && \forall i \\
& && x_{ij} \le y_i && \forall i, \forall j \\
& && x_{ij} \in \{0,1\} && \forall i, \forall j \\
& && y_i \in \{0,1\} && \forall i
\end{aligned}
\qquad (8.37)
$$

Circle Covering Heuristic

Savelsbergh and Goetschalckx (1994) have developed an efficient heuristic for this problem. Initially, for each existing facility a covering circle is determined with is center at this existing facility and with a radius so that the sum of the resource requirements of all existing facilities inside the circle does not exceed the capacity. In a second step, the covering circles are added by increasing distance until all existing facilities have been covered. This algorithm is formalized below.

Algorithm 8.8. Circle-Covering (Savelsbergh & Goetschalckx)

$for \;\; i = 1 \;\; to \;\; M \;\; \{$

$\quad rad_i = 0, \;\; C_i = \{i\}, \;\; R_i = r_i$

$\quad while \;\; C_i \subset I \;\; \{$

$\quad\quad k \leftarrow d_{ik} = \min_h \{d_{ih} | h \in I \setminus C_i\}$

$\quad\quad if \;\; R_i + r_k \le s_j$

$\quad\quad then \;\; rad_i \leftarrow d_{ik}, \;\; C_i \leftarrow C_i \cup \{k\},$

$\quad\quad\quad R_i \leftarrow R_i + r_k$

$\quad\quad else \;\; next \;\; i$

$\quad\quad \} \;\; endwhile$

$\quad \} \;\; endfor$

$C = \varnothing$

$while \;\; C \ne I \;\; \{$

$\quad k \leftarrow rad_k = \min_h \{rad_h | h \in I \setminus C\}$

$\quad C \leftarrow C \cup C_k$

$\quad \} \;\; endwhile$

In the first step the circle covers are generated. For each existing facility *i*, the other existing facilities are sorted by increasing distance to facility *i* and then indexed by index k. For each existing facility i the set its current cover to this facility i itself, the radius of the cover to zero, and the total demand in the cover to demand of facility i. While the current cover does not contain all facilities, we check if adding the next closest facility k to the current cover would violate the cover capacity. If so, we stop growing the current cover and go to the next facility i. If not, facility k is added to the current cover and the radius and the total demand in the cover are updated.

In the second step, all covers are ranked by their non-decreasing radius and indexed by index k.

In the third step, all facilities are initially marked as uncovered. While there are uncovered facilities remaining, select the cover with the next smallest radius and centered at an uncovered facility. This cover will be part of the solution. Mark all facilities in this current cover as covered.

8.5. Generalized Assignment Problem

Formulation

$$\begin{aligned}
\min \quad & z = \sum_{j=1}^{N}\sum_{i=1}^{M} c_{ij} x_{ij} & \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 & \forall i \\
& \sum_{i=1}^{M} r_i x_{ij} \le s_j & \forall j \\
& y_j \in \{0,1\} & \\
& x_{ij} \ge 0 &
\end{aligned} \tag{8.38}$$

8.6. Capacitated Warehouse Location Problem

Formulation

$$
\begin{aligned}
\min \quad & z = \sum_{j=1}^{N} \left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right) & & \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 & \forall i & \\
& \sum_{i=1}^{M} r_i x_{ij} \le s_j & \forall j & \\
& -x_{ij} + y_j \ge 0 & \forall ij & \\
& y_j \in \{0,1\} & & \\
& x_{ij} \ge 0 & &
\end{aligned} \tag{8.39}
$$

Various solution and decomposition approaches have been tried

1. Branch-and-Bound with Linear Relaxation
2. Branch-and-Bound with Lagrangean Relaxation (Dual Decomposition)
3. Primal (Bender's) Decomposition
4. Cross Decomposition

Cross Decomposition Solution Algorithm

Primal transportation subproblem

- Given Y
- Yields primal feasible solution
- Determines V

Dual uncapacitated warehouse location problem

- Given V

- Yields lower bound
- Determines Y

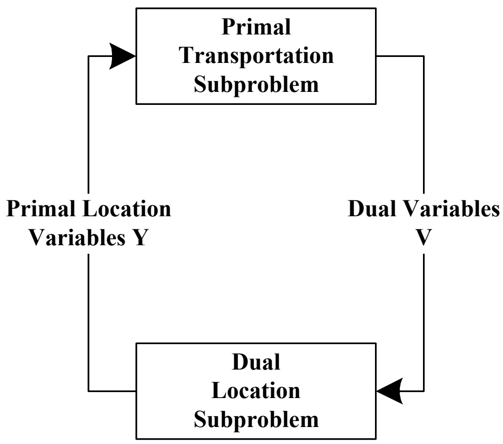

Figure 8.1. Cross Decomposition Algorithm Flow Chart

Primal Transportation Subproblem

$$
\begin{aligned}
\min \quad & z = \sum_{j=1}^{N}\sum_{i=1}^{M} c_{ij} x_{ij} + \sum_{j=1}^{N} f_j y_j \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && [u_i] \\
& \sum_{i=1}^{M} r_i x_{ij} \le s_j y_j && [v_j] \\
& -x_{ij} + y_j \ge 0 && [w_{ij}] \\
& x_{ij} \ge 0
\end{aligned}
\tag{8.40}
$$

Dual Uncapacitated Location Subproblem

$$
\begin{aligned}
\min \quad & z = \sum_{j=1}^{N} \left(f_j - s_j v_j \right) y_j + \sum_{j=1}^{N}\sum_{i=1}^{M} \left(c_{ij} + r_i v_j \right) x_{ij} \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && [u_i] \\
& -x_{ij} + y_j \ge 0 && [w_{ij}] \\
& y_j \in \{0,1\} \\
& x_{ij} \ge 0
\end{aligned}
\tag{8.41}
$$

In order for the dual problem to yield a primal feasible solution, we have to assure that the total demand can be handled by all open facilities.

$$\sum_{j=1}^{N} s_j y_j \geq \sum_{i=1}^{M} r_i \qquad [\lambda] \tag{8.42}$$

$$\begin{aligned}
\min \quad & z = \sum_{j=1}^{N} \left(f_j - s_j v_j\right) y_j + \sum_{j=1}^{N}\sum_{i=1}^{M} \left(c_{ij} + r_i v_j\right) x_{ij} \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && [u_i] \\
& -x_{ij} + y_j \geq 0 && [w_{ij}] \\
& \sum_{j=1}^{N} s_j y_j \geq \sum_{i=1}^{M} r_i && [\lambda] \\
& y_j \in \{0,1\} \\
& x_{ij} \geq 0
\end{aligned} \tag{8.43}$$

Relaxed dual uncapacitated location subproblem.

$$\begin{aligned}
\min \quad & z = \sum_{j=1}^{N} \left(f_j - s_j v_j - \lambda s_j\right) y_j + \sum_{j=1}^{N}\sum_{i=1}^{M} \left(c_{ij} + r_i v_j\right) x_{ij} + \lambda \sum_{i=1}^{M} r_i \\
s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && [u_i] \\
& -x_{ij} + y_j \geq 0 && [w_{ij}] \\
& y_j \in \{0,1\}, \ x_{ij} \geq 0
\end{aligned} \tag{8.44}$$

Exercises

Chemical Arsenal

A military depot is responsible for storing chemical weapons and needs to establish sites for emergency response equipment in case of a chemical accident. The emergency equipment is very expensive and so the depot wants to install as few of these quick response sites as possible. The chemical weapons are stored in bunkers and the emergency site responsible for a particular bunker cannot have a travel time of more than 6 minutes to that bunker. For simplicity, assume that the only possible locations for the emergency sites are next to an existing bunker. Different space constraints at the bunkers generate different costs for establishing the emergency sites. All bunkers must be assigned to an emergency site and no emergency site can be responsible for more than three bunkers. The costs of establishing an emergency site at each of the bunkers are given in Table 8.22. The travel times in minutes between the various locations on the transportation network are given in Figure 8.2. Find the minimum cost configuration for the emergency sites.

Table 8.22. Cost of Emergency Equipment Sites

Site	1	2	3	4	5	6	7	8	9
Cost	105	130	145	125	70	80	110	115	75

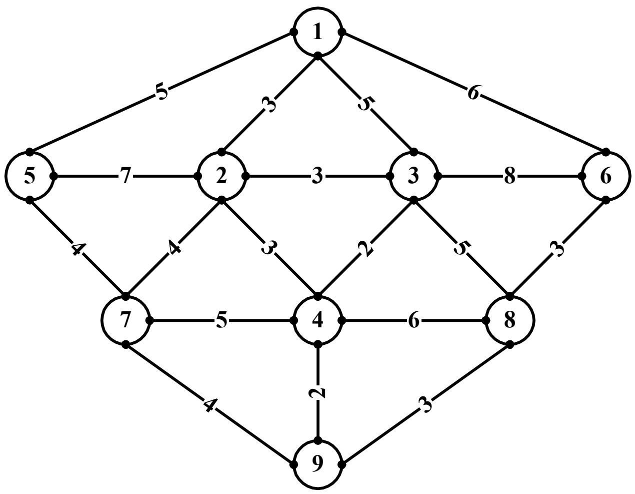

Figure 8.2. Traveling Times between Chemical Storage Bunkers

State Institute of Technology

The State Institute of Technology is in the process of creating a campus master plan. The growing student population and the removal of existing parking areas due to new building construction make it necessary to construct a number of multilevel parking decks. The planning committee has made the promise to the faculty and staff that no academic area will be farther than five minutes walking or 800 feet removed from the nearest parking deck. There exist eight major academic areas on campus and ten possible parking deck locations are being considered. Each of the major academic areas requires a number of parking spaces to accommodate its faculty, staff, and students. The construction costs for each parking deck location are different due to different the different sizes of the decks and due to rock removal and other civil engineering considerations. The following table shows the required number of parking spaces per academic area, the cost of building a parking deck at a location, and the satisfaction matrix, whose elements are equal to a one if a parking deck is within the promised distance to an academic area and zero otherwise. The objective is to build parking decks at the locations that minimize the overall cost and so that the distance constraints are satisfied.

Table 8.23. Parking Data

Academic Area		Parking Deck Location									
	Spaces	1	2	3	4	5	6	7	8	9	10
1	300	1	1	0	1	0	0	1	1	0	0
2	400	1	1	1	1	0	0	1	1	1	0
3	700	1	1	1	0	1	0	1	1	1	0
4	800	1	1	1	1	1	1	1	0	0	0
5	400	0	0	0	0	1	0	0	0	0	0
6	500	0	0	1	1	1	1	1	1	1	1
7	200	0	0	0	1	1	1	1	1	1	1
8	600	0	0	0	0	0	1	1	1	1	1
Cost		380	240	480	450	350	250	850	750	450	175

Solve this problem in the most efficient way with the standard greedy heuristic. Show the computations and the sequential decisions that you made. Summarize your solution and compute the total cost of your solution.

The planning committee decided to allocate the construction cost of a particular parking deck to the various academic areas served by this deck proportional to the number of parking spaces required by the academic areas. Compute and show the allocated cost for each academic area. After the planning phase has been completed, a new parking deck location had become available. The data for this new location are given in the next table. Determine if this new location should be added to the list of ten initial locations and the solution process should be repeated. Justify your answer numerically.

Table 8.24. Additional Parking Deck Location Data

Academic Area		Deck
	Spaces	11
1	300	1
2	400	0
3	700	1
4	800	0
5	400	1
6	500	1
7	200	0
8	600	1
Cost		460

BRAC-99

Reduced tensions in the world have led to a reduced military force for the United States. This reduced military requires a smaller number of support bases and significant costs can be saved if some bases are closed. The Base Realignment and Closing Commission for the fiscal year 1999, (BRAC-99), is in the process of determining which bases to close down. The bases that remain open must be able to provide military support for potential operations in various areas of the world. The world has been divided in ten areas of operations. The following table shows which bases can support operations in which areas of the world and the annual cost in millions of dollars to keep each base open. The objective is to build enough bases so that all areas are covered and to minimize the total cost of the open bases.

Table 8.25. Military Base Covering Data

World	Base Locations														
Areas	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	0	1	0	0	1	1	0	0	0	1	0	0	1
2	1	1	1	1	0	0	0	0	1	0	0	1	1	1	0
3	1	1	1	0	1	0	1	0	1	0	0	0	0	0	1
4	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
5	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
6	0	0	1	1	0	1	1	1	1	1	1	1	1	1	0
7	0	0	0	1	1	1	1	0	1	1	1	0	1	0	1
8	0	0	0	0	0	1	1	1	1	1	1	0	1	0	1
9	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
	800	750	650	450	350	250	1050	750	650	450	350	750	550	500	850

Solve this problem in the most efficient way with the standard greedy heuristic. Show the computations and the sequential decisions that you made. Summarize your solution and compute the total cost of your solution.

The planning committee wants to provide insight in how much it costs to support military operations in a certain area of the world. For simplicity, all world areas are assumed to require the same amount of support from the base that supports it. If a world area can be supported by more than one open base, the support function is allocated in equal parts to all the open bases that can support this area and the total cost for this world area is the sum of its costs allocated to the bases that can support it. Compute and show the allocated cost for each world area.

After the planning phase has been completed, the powerful senator and chairperson of the armed services committee wants to establish a new base in his district and uses the argument that it will reduce the overall base cost without reducing support coverage. The data for this new base are given in the next table.

Table 8.26. Additional Base Data

World Areas	Base 16
1	1
2	1
3	1
4	1
5	0
6	0
7	0
8	0
9	1
10	0
	800

Determine if this new base should be added to the list of potential bases to remain open and if the decision process should be repeated. Justify your answer numerically.

Generalize Median Problem

Solve the Generalized Median Problem given in the following table with the Greedy Heuristic for the case of a maximum of two and three service centers. An infinite cost indicates that a particular server cannot service that particular customer. Use the notation used in class and in the book and follow the same tableau structure. Specifically list all Lagrangean multipliers in the columns to the right of the costs, (one column for each action) and list the site dependent costs in the rows below the costs (one row for each stage in the algorithm). List clearly your solution for the case of maximum two service centers and give its costs, and then give your solution for the case of maximum three service centers and give its costs. Which case provides the better solution? Explain and justify your answer. Compare your solutions for the variant of the greedy heuristic where the infinite cost is implemented with a large finite cost (big M method) with the variant of the greedy heuristic where the algorithm is adjusted to skip over the infinite cost combinations. Discuss the differences and similarities in the obtained solutions.

	1	2	3	4	5					
1	*12*	*21*	*18*	*21*	*17*					
2	*18*	*∞*	*19*	*19*	*15*					
3	*10*	*15*	*11*	*15*	*11*					
4	*∞*	*24*	*20*	*18*	*15*					
5	*6*	*6*	*5*	*7*	*7*					
6	*∞*	*21*	*∞*	*12*	*20*					
7	*18*	*11*	*∞*	*16*	*20*					
8	*∞*	*17*	*20*	*12*	*∞*					
f	*100*	*70*	*60*	*110*	*80*					

Dualoc Algorithm by Erlenkotter

Consider the uncapacitated discrete facility location problem and its solution with the Erlenkotter DUALOC algorithm. The transportation cost coefficients are given in the following table.

Table 8.27. Transportation Cost Coefficients

j↓/i→	1	2	3	4	5	u_j
1	120	210	180	210	170	180
2	180	∞	190	205	150	190
3	100	150	110	150	115	110
4	∞	240	195	180	150	195
5	60	55	50	65	70	60
6	∞	210	∞	120	195	210
7	180	110	∞	160	200	180
8	∞	165	195	120	∞	195
f_i	100	300	200	400	350	
s_i	20	195	190	200	240	

The dual variables at the end of iteration two of Erlenkotter's dual ascent method are given to the right of the cost coefficient matrix. The original fixed costs are given on the first line below the cost matrix. The current slack variables are given in the last line below the cost matrix.

Complete the dual ascent phase of the DUALOC algorithm. Clearly indicate major iterations and the corresponding dual variables and slack variables. Use the tableau format developed in class.

At the end of the dual ascent, compute the corresponding primal solution. Verify and show clearly if a dual adjustment phase is required, **but execute only one iteration** if it is required. An iteration consists of the adjustment plus the following ascend steps until all the dual variables are blocked. At the end of this iteration, compute again the primal solution, verify and show clearly if further dual adjustment iterations are required.

Automotive Emission Inspection Stations

The State of Georgia requires an annual emission inspection of all non-commercial automobiles. The CLEANAIR company has agreed to establish enough inspection stations so that the average waiting time is kept below a limit of 5 minutes. It has been determined that 38 possible station locations exist in the service area. Furthermore, the customer demand data has been aggregated into the same 38 locations. Based on the number of cars in the customer zone associated with each location, the contribution of each location to the average waiting time has been computed. It is assumed that the waiting times of the individual customers zones assigned to an inspection station can be added to determine the waiting time of the inspection station. The geographical location of each location is illustrated in Figure 8.3. The coordinates of each location and its associated contribution to the average waiting time is given in Table 8.28.

Determine the minimum number of inspection stations, their locations, and their associated customer zones, i.e. customer assignments, and the radius of the customer zones with the Savelsbergh and Goetschalckx greedy circle covering heuristic. Copy the above Figure 8.3 and draw the solution.

Determine the optimal solution with a mixed integer programming solver such as LINDO or CPLEX. Copy the above Figure 8.3 and draw the solution.

Compare the solution quality and solution times for both procedures. Give a brief assessment of modeling assumptions made for this study.

Table 8.28. Customer Data

#	x	y	r	#	x	y	r
1	1.250	5.687	3.89	21	3.250	2.937	3.84
2	3.750	5.562	0.36	22	2.938	2.812	2.79
3	2.688	5.312	4.22	23	3.125	2.750	1.46
4	1.250	4.937	3.34	24	1.688	2.812	3.99
5	4.250	4.812	4.60	25	5.000	2.750	0.15
6	2.750	4.625	3.86	26	5.813	2.687	3.21
7	6.750	4.312	2.18	27	2.750	2.687	3.77
8	3.875	4.125	1.11	28	2.188	2.625	4.72
9	2.688	4.062	1.32	29	1.215	2.500	1.68
10	2.375	4.000	2.68	30	2.688	2.437	2.90
11	1.938	3.812	3.83	31	3.063	2.437	0.53
12	0.813	3.625	0.88	32	3.313	2.437	0.61
13	6.000	3.562	0.34	33	1.813	2.312	3.18
14	5.188	3.437	1.94	34	2.063	2.312	4.21
15	3.000	3.437	1.86	35	3.063	2.187	4.15
16	2.188	3.375	3.64	36	4.375	2.125	1.01
17	1.813	3.375	1.03	37	2.750	1.000	0.82
18	1.500	3.250	0.96	38	1.125	0.437	0.77
19	2.250	3.000	0.01				
20	4.000	3.000	0.99				

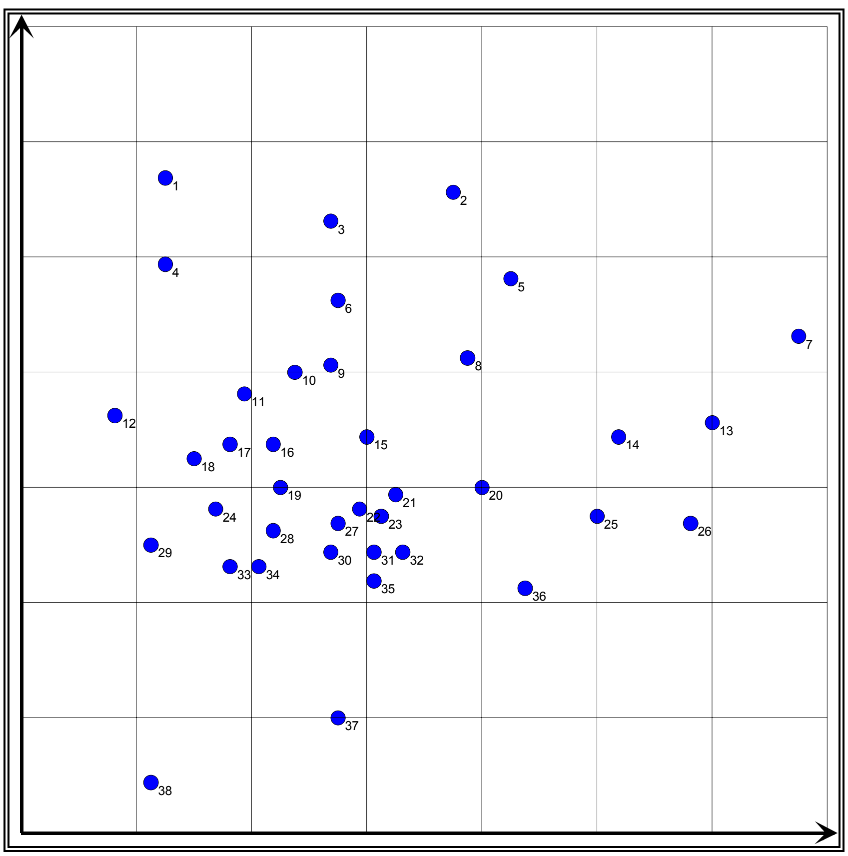

Figure 8.3. Customer Locations

References