Logistics Systems Design: Discrete Point Location

7. Continuous Point Location
8. Discrete Point Location
9. Supply Chain Models
10. Facilities Design
11. Computer Aided Layout
12. Layout Models

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Discrete Point Location

❶ Set Covering and Set Partitioning
- Problem and Formulation
- Column Generation
- Master Problem Solution

❷ Generalized n-Mediam

❸ Warehouse Location Problem
- Erlenkotterís Dual Algorithm

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Discrete Capacitated Location

❶ Covering Problem

❷ Generalized Assignment Problem

❸ Warehouse Location Problem

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Set Partitioning Problem Definition

- Every Column j = Feasible Alternative Action
- Every Row i = Service Request
- Minimize Overall Cost while Servicing All Requests

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Set Partitioning Notation

- x_j = 1 if Alternative j is Executed (Binary)
- a_{ij} = 1 if Alternative j Satisfies Request i
- c_j = Cost of Alternative j
- p_i = Cost Estimate for Servicing Request i

Set Partitioning Formulation

$$\min \sum_{j=1}^{N} c_j x_j$$

$$s.t. \sum_{j=1}^{N} a_{ij} x_j = 1 \qquad i = 1..M$$

$$x_j \in \{0,1\}$$

Set Covering Formulation

$$\min \sum_{j=1}^{N} c_j x_j$$

$$s.t. \sum_{j=1}^{N} a_{ij} x_j \geq 1 \qquad i = 1..M$$

$$x_j \in \{0,1\}$$

Set Partitioning Characteristics

- Alternative Selecting Algorithm
- Accurate Costs and Feasibility Constraints
- Optimal Solution for ìSmallî Problem Sizes (IP Solver)
- Efficient Column Generation and Pricing
- Models Complex Problems

Set Definitions

$$J_i = \{ j : a_{ij} = 1 \}$$

$$I_j = \{ i : a_{ij} = 1 \}$$

$$J^+ = \{ j : x_j = 1 \}$$

$$I^\circ = \left\{ i : i \notin \bigcup_{j \in J^+} I_j \right\}$$

Set Partitioning Algorithm

❶ *Start with a feasible partition J^+*

❷ *Determine row prices by allocating column prices ìequitableî such that*

$$c_j^+ = \sum_{i=1}^{M} a_{ij} p_i$$

Example:

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

Set Partitioning Algorithm (2)

❸ *Generate and evaluate new column j, if*

$$c_j - \sum_{i=1}^{M} a_{ij} p_i \le 0$$

add column j to the partitioning problem

❹ *If enough new columns are added, solve the partitioning master problem else go to step 3*

Set Partitioning Algorithm (3)

❺ *If all columns have been evaluated and none added or solution is within tolerance, stop else solve partitioning master problem and go to step 2*

Set Covering Reduction Rules

❶ **Row Infeasibility**

$J_k = \varnothing \quad \Rightarrow \quad infeasible$

❷ **Row Feasibility**

$J_r = \{c\} \quad \Rightarrow \quad x_c = 1, \quad c \in J^+$

$eliminate \;\; rows \;\; k : k \in I_c$

❸ **Row Dominance**

$J_r \subseteq J_q \quad \Rightarrow \quad eliminate \;\; row \;\; q$

Set Covering Reduction Rules (2)

❹ **Column Dominance**

$I_s = \varnothing \Rightarrow \quad x_t = 0$

$I_s \supseteq I_t \;\; and \;\; c_s \le c_t \quad \Rightarrow \quad x_t = 0$

❺ **Cycle Through Rules**

Set Covering, Reduction Data

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

Set Covering, Reduction 1

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

Set Covering, Reduction 2

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

Set Covering, Reduction 3 & 4

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

i↓ / j→	1	4
8	1	1
c_j	175	115

Set Covering, Greedy Heuristic

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

$\text{while } I^\circ \neq \varnothing$ {

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

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

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

} *endwhile*

Set Covering Heuristic

- ìReduced Costî function f(c, k)
  - Many alternatives
  - Inverse is ìMost Bang for Buckî

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

Set Covering Linear Relaxation

- Larger Feasible Region
- Solvable with Linear Programming

$$\min \sum_{j=1}^{N} c_j x_j$$

$$s.t. \sum_{j=1}^{N} a_{ij} x_j \geq 1 \qquad i = 1..M$$

$$0 \leq x_j \leq 1$$

Cutting Plane

- Additional Valid Constraint, Violated by Current (Optimal) Solution to Relaxation
- Sequence of Cutting Planes Drives Optimal to Feasibility

Covering Problem Cutting Plane

- Covering Problem All $c_j = 1$ (Unweighted or Counting Problem)
- Usually a Single Cut Suffices

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

Branch-And-Bound

- Most Widely Used Solution Method for Discrete Problems
- Procedures
  - Branching Variable Selection
  - Lower Bound Computation
  - Primal Heuristics (Incumbent Solution)

Set Partitioning References

- Cullen F., Jarvis J. and Ratliff H., (1981). "Set Partitioning Based Heuristics for Interactive Routing." Networks, Vol. 11, No. 2, pp. 125-143.
- Balas E. and Padberg M., (1976). "Set Partitioning: A Survey." SIAM Review, Vol. 18, No. 4, pp. 710-760.

Generalized n-Median Problem

$$\min \quad z = \sum_{j=1}^{N} \left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right)$$

$$s.t. \quad \sum_{j=1}^{N} a_{ij} x_{ij} = 1$$

$$-x_{ij} + y_j \geq 0$$

$$\sum_{j=1}^{N} y_j \leq n$$

$$y_j \in \{0,1\}, x_{ij} \geq 0$$

Lagrangean Relaxation

- Relax Assignment Constraints

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

- Penalty Objective Function

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

Lagrangean Relaxation (2)

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

$$s.t. \quad -x_{ij} + y_j \geq 0$$

$$\sum_{j=1}^{N} y_j \leq n$$

$$y_j \in \{0,1\}, x_{ij} \geq 0$$

Lagrangean Relaxation Properties

- Lower Bound

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

- Siteís Relative Cost Factor

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

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Condensed Lagrangean Relaxation

$$\min \quad z_{LAR}(U) = \sum_{i=1}^{M} u_i + \min \sum_{j=1}^{N} \rho_j(U) y_j$$

$$s.t. \quad \sum_{i=1}^{N} y_j \leq n$$

$$y_j \in \{0,1\}$$

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Lagrangean Relaxation Solution

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 $\rho_j(U)$

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

n-Median Lagrangean Relaxation Heuristic

$$J^+ = \emptyset, k = 0$$

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

$$if \quad \rho_{r(1)} > 0,$$

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

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

n-Median Lagrangean Relaxation Heuristic (2)

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

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

$$k \leftarrow k+1$$

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

$$\}\ \ endwhile$$

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

Lagrangean Multipliers

- **Heuristic Values**

$$u_i = \frac{1}{N} \sum_{j=1}^{N} c_{ij}, \qquad u_i = \max_j \{c_{ij}\}$$

- **Lagrangean Master Dual Problem**

$$\max_U \ z_{LAR}(U)$$

  - Subgradient Optimization
  - Dual Adjustment Heuristic

n-Median Dual Adjustment Heuristic

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

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

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

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

$$if\ \ \rho_{r(1)} > 0,$$

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

n-Median Dual Adjustment Heuristic

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

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

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

$$k \leftarrow k+1$$

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

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

Warehouse Location Problem

- 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 Problem

$$\min \quad z = \sum_{j=1}^{N} \left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right)$$

$$\begin{aligned} s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && [u_i] \\ & -x_{ij} + y_j \geq 0 && [w_{ij}] \\ & y_j \in \{0,1\}, x_{ij} \geq 0 \end{aligned}$$

WLP Linear Relaxation

$$\min \quad z_{LR} = \sum_{j=1}^{N} \left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right)$$

$$\begin{aligned} s.t. \quad & \sum_{j=1}^{N} x_{ij} = 1 && [u_i] \\ & -x_{ij} + y_j \geq 0 && [w_{ij}] \\ & y_j \geq 0, \; x_{ij} \geq 0 \end{aligned}$$

WLP Dual Formulation

$$\max \quad z_D = \sum_{i=1}^{M} u_i$$

$$\begin{aligned} s.t. \quad & \sum_{i=1}^{M} w_{ij} \leq f_j && [y_j] \\ & u_i - w_{ij} \leq c_{ij} && [x_{ij}] \\ & w_{ij} \geq 0 \\ & u_i \;\; unrestricted \end{aligned}$$

Variable Transformations

$$w_{ij} \ge u_i - c_{ij} \;\; \text{and} \;\; w_{ij} \ge 0$$

$$w_{ij} = \max\{0, u_i - c_{ij}\}$$

$$\sum_{i=1}^{M} w_{ij} = \sum_{i=1}^{M} \max\{0, u_i - c_{ij}\} \le f_j$$

$$\rho_j(U) = f_j + \sum_{i=1}^{M} \min\{0, c_{ij} - u_i\} \ge 0$$

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Condensed Dual Formulation

$$\max \;\; z_D = \sum_{i=1}^{M} u_i$$

$$s.t. \;\; \rho_j(U) \ge 0$$

$$u_i \;\; \text{unrestricted}$$

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Complementary Slackness

$$y_j \cdot \rho_j(U) = 0 \quad \Rightarrow \quad \text{if } y_j = 1 \text{ then } \rho_j(U) = 0$$

$$\text{if } \rho_j(U) \ne 0 \text{ then } y_j = 0$$

$$x_{ij}\left(c_{ij} - u_i + w_{ij}\right) = 0 \quad \Rightarrow \quad \text{if } x_{ij} = 1 \text{ then } u_i \ge c_{ij}$$

$$\text{if } u_i < c_{ij} \text{ then } x_{ij} = 0$$

$$w_{ij}\left(y_j - x_{ij}\right) = 0 \quad \Rightarrow \quad \text{if } u_i > c_{ij} \text{ then } x_{ij} = y_j$$

$$\text{if } x_{ij} < y_j \text{ then } u_i \le c_{ij}$$

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Primal Heuristic

$$J^+ = \left\{ j \,\middle|\, \rho_j(U) = 0 \right\} \cap$$

$$\left\{ j \,\middle|\, \exists i \,\middle|\, c_{i\alpha(i)} = \min_j \left\{ c_{ij} \,\middle|\, \rho_j(U) = 0 \right\} \right\} \cap$$

$$\left\{ j \,\middle|\, \forall i \,\middle|\, \exists j \,\middle|\, \left(\rho_j(U) = 0\right) \cap \left(u_i > c_{ij}\right) \right\}$$

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

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Dual Ascent Maximum Increase

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

$$\Delta_1 = \min_j \left\{ \rho_j(U) \middle| u_i > c_{ij} \right\}$$

$$\Delta_2 = \min_j \left\{ c_{ij} - u_i \middle| u_i < c_{ij} \right\}$$

$$\Delta = \min\{\Delta_1, \Delta_2\}$$

Dual Ascent Algorithm

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

$$\textit{while}\ \ I^a \neq \emptyset$$

$$\textit{for}\ \ i \in I^a$$

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

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

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

Violations and Primal-Dual Gap

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

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

Erlenkotter DUALOC Algorithm

- Compact dual formulation
- Dual ascent, then dual adjustment
- Primal heuristic
- Used as bound in branch-and-bound
- Magnitude faster than contemporary algorithms

Case 1, 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									

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 1, Dual Ascent, 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				

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 1, Dual Ascent, 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				

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 1, Dual Ascent, 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				

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 1, Dual Ascent, 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				

Case 1, Solution

- Fixed costs = 190
- Allocations costs = 1045
- Total cost = 1235
- Lower bound = 1235
- Optimal (primal feasible) no complementary slackness violations

Dual Adjustment Sets and Costs

$$J_i^c = \left\{ j \in J^+ \middle| u_i \geq c_{ij} \right\}$$

$$J_i^+ = \left\{ j \in J^+ \middle| u_i > c_{ij} \right\}$$

$$I_j^+ = \left\{ i \middle| J_i^c = \{j\} \right\}$$

$$c_{i\alpha(i)} = \min_{j \in J^+} \{c_{ij}\}$$

$$c_{i\beta(i)} = \min_{j \in J^+, j \neq \alpha(i)} \{c_{ij}\}$$

Dual Adjustment Algorithm

$$
\begin{aligned}
& \textit{for } \ i = 1 \ \textit{to } \ M \\
& \quad \textit{if } \left| J_i^+ \right| > 1 \\
& \qquad \textit{if } \ I_{\alpha(i)}^+ \cup I_{\beta(i)}^+ \neq \varnothing \\
& \qquad\quad \Delta = u_i - \max_j \left\{ c_{ij} \middle| u_i > c_{ij} \right\} \\
& \qquad\quad \textit{for } \ j \in J \middle| u_i > c_{ij} \\
& \qquad\qquad \rho_j(U) \leftarrow \rho_j(U) + \Delta \\
& \qquad\quad u_i \leftarrow u_i - \Delta
\end{aligned}
$$

Dual Adjustment Algorithm (2)

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

execute dual ascent

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

execute dual ascent

$$I^a \leftarrow I$$

execute dual ascent

endif , endif , endfor

Case 2, 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									

Case 2, Dual Ascent, 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				

Case 2, Dual Ascent, 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				

Case 2, Dual Ascent, 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				

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 2, Dual Ascent, 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				

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 2, Ascent Solution

- *Fixed costs = 500*
- *Allocations costs = 1105*
- *Total cost = 1605*
- *Lower bound = 1530*
- *Duality gap = 75 (complementary slackness violation for customer 6)*

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 2, Dual Adjust, 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				

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 2, Dual Adjust, 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				

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 2, Adjustment 1 Solution

- Fixed costs = 400
- Allocations costs = 1180
- Total cost = 1580
- Lower bound = 1560
- Duality gap = 20
  (complementary slackness violations for customer 5 and 7)

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 2, Dual Adjust, 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					

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 2, Dual Adjust, 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						

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Case 2, Adjustment 2 Solution

- Fixed costs = 500
- Allocations costs = 1105
- Total cost = 1605
- Lower bound = 1565
- Duality gap = 40 (complementary slackness violation for customer 6)
- Incumbent = 1580, smallest gap = 15

WLP Alternative Formulation

$$\min \quad z = \sum_{j=1}^{N} \left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right)$$

$$s.t. \quad \sum_{j=1}^{N} x_{ij} = 1 \qquad \forall i$$

$$\sum_{i=1}^{M} x_{ij} - M y_j \le 0 \qquad \forall j$$

$$y_j \in \{0,1\}, x_{ij} \ge 0$$

WLP Alternative Relaxation

$$\min \quad z = \sum_{j=1}^{N} \left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right)$$

$$s.t. \quad \sum_{j=1}^{N} x_{ij} = 1 \qquad \forall i$$

$$\sum_{i=1}^{M} x_{ij} - M y_j \le 0 \qquad \forall j$$

$$y_j \ge 0, x_{ij} \ge 0$$

Linear Relaxation Substitution

- Substitution at optimality

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

Reduced Formulation

$$\min \quad z = \sum_{j=1}^{N} \sum_{i=1}^{M} \left(\frac{f_j}{M} + c_{ij} \right) x_{ij}$$

$$s.t. \quad \sum_{j=1}^{N} x_{ij} = 1 \qquad \forall i$$

$$x_{ij} \geq 0$$

Greedy Algorithm

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

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

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

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

Constraint Disaggregation

- IP are identical, but linear relaxations are not!
- Aggregated versus disaggregated constraints
- ìWeakî and ìStrongî formulation

Case 1, 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					

Case 1, Weak Relaxation Costs

	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

Case 1, Weak Relaxation Solution

- Fixed costs = 88.75
- Allocation costs = 970
- Total cost = 1058.75 (lower bound)
- Round-up cost = 1390 (primal heuristic cost)
- Optimal cost = 1235

Discrete Capacitated Location Notation

y_j = binary server status

x_{ij} = customer assignment

f_j = server fixed cost

c_{ij} = assignment cost

r_i = customer service requirement

s_j = server capacity

Capacitated Covering Formulation

$$\min \sum_{j=1}^{N} y_j$$

$$s.t. \sum_{j=1}^{N} x_{ij} \geq 1 \qquad \forall i$$

$$\sum_{i=1}^{M} r_i x_{ij} \leq s_j y_j \qquad \forall j$$

$$y_j \in \{0,1\}, \; x_{ij} \in \{0,1\}$$

Covering Problem Characteristics

- Determine Number and Location of Equal Cost Servers
- Fixed Costs, No Assignment Costs
- Server Status Binary Variables and Assignment Variables

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Capacitated Set Covering Formulation

$$\min \sum_{j=1}^{N} f_j y_j$$

$$s.t. \sum_{j=1}^{N} x_{ij} \geq 1 \qquad \forall i$$

$$\sum_{i=1}^{M} r_i x_{ij} \leq s_j y_j \qquad \forall j$$

$$y_j \in \{0,1\}, \ x_{ij} \in \{0,1\}$$

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Set Covering Problem Characteristics

- Determine Number and Location of Unequal Cost Servers
- Fixed Costs, No Assignment Costs
- Server Status Binary Variables and Assignment Variables

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Circle Covering Heuristic

1. For Each Customer as Center Determine Capacitated Circle Cover of Smallest Diameter
2. While Not All Customers are Covered
   - Select Cover with Smallest Diameter
   - Eliminate All Customers Inside this Cover and Their Associated Covers

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Circle Covering Heuristic: Cover Creation

$$
\begin{aligned}
&\textit{for } i = 1 \textit{ to } M \\
&\quad rad_i = 0,\ C_i = \{i\},\ R_i = r_i \\
&\quad \textit{while } C_i \subset I \\
&\qquad k \leftarrow d_{ik} = \min_h \{d_{ih} | h \in I \setminus C_i\} \\
&\qquad \textit{if } R_i + r_k \le s_j \\
&\qquad \textit{then } rad_i \leftarrow d_{ik},\ C_i \leftarrow C_i \cup \{k\}, \\
&\qquad\quad R_i \leftarrow R_i + r_k \\
&\qquad \textit{else next } i
\end{aligned}
$$

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Circle Covering Heuristic: Cover Selection

$$
\begin{aligned}
&C = \varnothing \\
&\textit{while } C \ne I \\
&\quad k \leftarrow rad_k = \min_h \{rad_h | h \in I \setminus C\} \\
&\quad C \leftarrow C \cup C_k
\end{aligned}
$$

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Generalized Assignment Problem

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

17-Mar-03 © Marc Goetschalckx

Generalized Assignment Characteristics

- Determine Allocation of Customers to Unequal Capacity Servers
- Assignment Costs, No Fixed Costs
- Binary Assignment Variables (Not Automatically Integer)

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Capacitated Warehouse Location Problem

$$\min \quad z = \sum_{j=1}^{N} \left(f_j y_j + \sum_{i=1}^{M} c_{ij} x_{ij} \right)$$

$$s.t. \quad \sum_{j=1}^{N} x_{ij} = 1 \qquad \forall i$$

$$\sum_{i=1}^{M} r_i x_{ij} \le s_j y_j \qquad \forall j$$

$$-x_{ij} + y_j \ge 0 \qquad \forall ij$$

$$y_j \in \{0,1\}, \; x_{ij} \in \{0,1\}$$

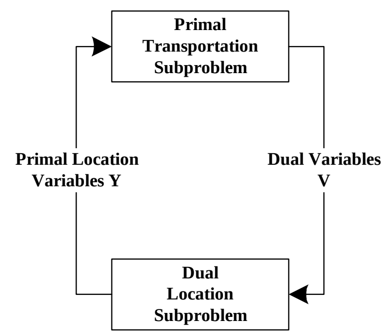
Capacitated Warehouse Location Characteristics

- Determine Number and Location of Unequal Cost and Unequal Capacity Servers
- Fixed Costs and Assignment Costs
- Server Status Binary Variables and Assignment Variables (Not Automatically Integer)

Cross Decomposition

- Primal Transportation Subproblem
  - Given Y
  - Yields primal feasible solution
  - Determines V
- Dual Uncapacitated Warehouse Location Problem
  - Given V
  - Yields lower bound
  - Determines Y

Cross Decomposition Flow Chart



Primal Transportation Problem

$$\min \quad z = \sum_{j=1}^{N} \sum_{i=1}^{M} c_{ij} x_{ij} + \sum_{j=1}^{N} f_j y_j$$

$$s.t. \quad \sum_{j=1}^{N} x_{ij} = 1 \qquad [u_i]$$

$$\sum_{i=1}^{M} r_i x_{ij} \le s_j y_j \qquad [v_j]$$

$$-x_{ij} + y_j \ge 0 \qquad [w_{ij}]$$

$$x_{ij} \ge 0$$

17-Mar-03 © Marc Goetschalckx

Dual Location Problem

$$\min \quad z = \sum_{j=1}^{N} \left(f_j - s_j v_j \right) y_j + \sum_{j=1}^{N} \sum_{i=1}^{M} \left(c_{ij} + r_i v_j \right) x_{ij}$$

$$s.t. \quad \sum_{j=1}^{N} x_{ij} = 1 \qquad [u_i]$$

$$-x_{ij} + y_j \ge 0 \qquad [w_{ij}]$$

$$y_j \in \{0,1\}, \; x_{ij} \ge 0$$

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Additional Feasibility Constraint

- Sufficient capacity to serve all demands

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

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Dual Location Problem

$$\min \quad z = \sum_{j=1}^{N} \left(f_j - s_j v_j \right) y_j + \sum_{j=1}^{N} \sum_{i=1}^{M} \left(c_{ij} + r_i v_j \right) x_{ij}$$

$$s.t. \quad \sum_{j=1}^{N} x_{ij} = 1 \qquad [u_i]$$

$$-x_{ij} + y_j \ge 0 \qquad [w_{ij}]$$

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

$$y_j \in \{0,1\}, \; x_{ij} \ge 0$$

17-Mar-03 © Marc Goetschalckx

Relaxed Dual Location Problem

$$\min \quad z = \sum_{j=1}^{N} \left(f_j - s_j v_j - \lambda s_j \right) y_j + \sum_{j=1}^{N} \sum_{i=1}^{M} \left(c_{ij} + r_i v_j \right) x_{ij} + \lambda \sum_{i=1}^{M} r_i$$

$$s.t. \quad \sum_{j=1}^{N} x_{ij} = 1 \qquad [u_i]$$

$$-x_{ij} + y_j \geq 0 \qquad [w_{ij}]$$

$$y_j \in \{0,1\}, \; x_{ij} \geq 0$$

17-Mar-03 © Marc Goetschalckx

Various Decomposition Methods

- Branch-and-Bound with Linear Relaxation
- Branch-and-Bound with Lagrangean Relaxation
- Primal (Benderís) Decomposition
- Cross Decomposition

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Decomposition Methods References

- Geoffrion and Graves, (1974). "Multicommodity Distribution System Design by Benders Decomposition." Management Science, Vol. 20, No. 5, pp. 822-844.
- Geoffrion and McBride, (1978). ìLagrangean Relaxation Applied to Capacitated Facility Location Problems.î Operations Research, Vol. 10, No. 1, pp. 40-47.
- Van Roy, (1986). ìA Cross Decomposition Algorithm for Capacitated Facility Location.î Operations Research, Vol. 34, No. 1, pp. 145-163.

17-Mar-03 Logistics Systems Design © Marc Goetschalckx

Decomposition and Mixed Integer Programming References

- Lasdon, (1970). Optimization Theory for Large Systems. MacMillan Publishing Co., New York New York.
- Nemhauser and Wolsey, (1988). Integer and Combinatorial Optimization. John Wiley and Sons, Inc., New York, New York.

17-Mar-03 Logistics Systems Design © Marc Goetschalckx