Using an interior-point method for huge capacitated multiperiod facility location

J. Castro[1] S. Nasini[2] F. Saldanha-da-Gama[3]

[1]Universitat Politècnica de Catalunya, Barcelona

[2]IESE Business School, Barcelona

[3]University of Lisbon

VI International Workshop on Locational Analysis
Barcelona November 25–27 2015

Supported by MTM2012-31440 grant

Outline

1. Motivation
2. Formulation of the multiperiod facility location problem
3. The cutting-plane approach
4. Solving subproblems by the interior-point
5. Computational results

Outline

Motivation

BlockIP: IPM solver for block-angular problems (f_i convex separable))

$$
\begin{array}{ll}
\min & \sum_{i=0}^{k} f_i(x^i) \\
\text{subject to} & \begin{bmatrix} N_1 & & & \\ & \ddots & & \\ & & N_k & \\ L_1 & \dots & L_k & I \end{bmatrix} \begin{bmatrix} x^1 \\ \vdots \\ x^k \\ x^0 \end{bmatrix} = \begin{bmatrix} b^1 \\ \vdots \\ b^k \\ b^0 \end{bmatrix} \\
& 0 \le x^i \le u^i \quad i = 0, \dots, k.
\end{array}
$$

Goal:

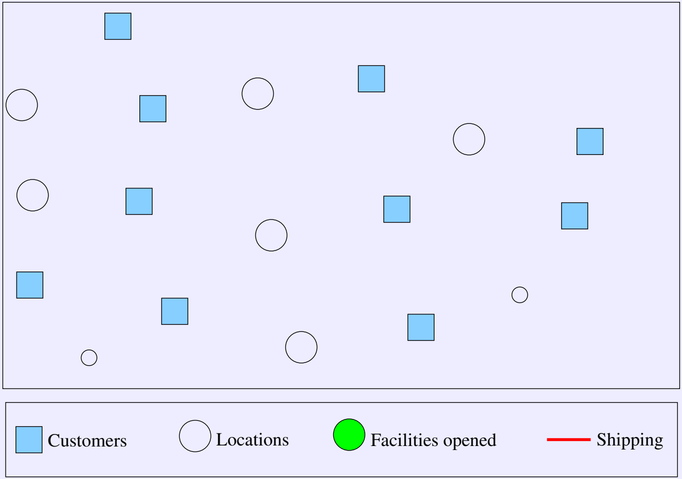
- Using BlockIP for subproblems within cutting-plane approach for MILPs.
- Focus on very large multiperiod facility location problems: e.g., world-wide instances faced by internet-based multinational retailers (hundreds of warehouses, millions of customers).

Outline

1. Motivation
2. Formulation of the multiperiod facility location problem
3. The cutting-plane approach
4. Solving subproblems by the interior-point
5. Computational results

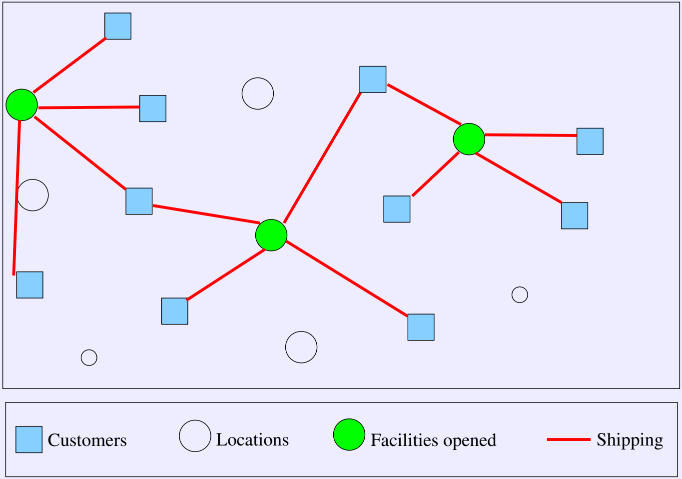
Multiperiod facility location problem: a 2-periods example

Initial situation



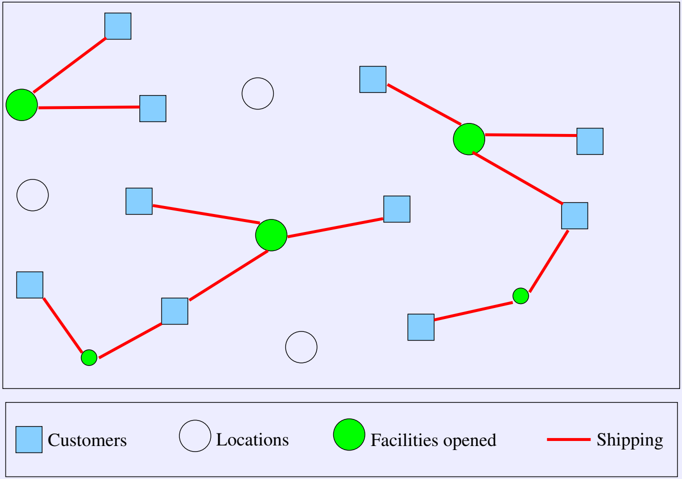
Multiperiod facility location problem: a 2-periods example

1st period: some facilities opened, and shipping



Multiperiod facility location problem: a 2-periods example

2nd period: additional facilities opened, and re-shipping



The multiperiod facility location problem

Data: sets and parameters

- T: Set of time periods in the planning horizon.
- I: Set of candidate locations for facilities, $n = |I|$.
- J: Set of customers, $m = |J|$.

- f_i^t: Cost for operating a facility at location i at period t.
- c_{ij}^t: Unitary transportation cost from facility i to customer j at period t.
- h_j^t: Unitary shortage cost at customer j at period t.

- d_j^t: Demand of customer j at period t.
- q_i: Capacity of a facility located at i.
- p^t: Maximum number of facilities operating at period t.

Variables

- $y_i^t \in \{0,1\}$: if 1 a facility is operating at i during period t; 0 otherwise. Design variables.
- x_{ij}^t: Amount shipped from facility i to customer j at period t.
- z_j^t: Shortage of customer j at period t.

The multi-period facility location problem

Formulation

$$
\begin{aligned}
\min \quad & \sum_{t\in T}\left(\sum_{i\in I} f_i^t y_i^t + \sum_{i\in I}\sum_{j\in J} c_{ij}^t x_{ij} + \sum_{j\in J} h_j^t z_j^t\right), & \\
\text{subject to} \quad & \sum_{i\in I} x_{ij}^t + z_j^t = d_j^t, & t\in T, j\in J, \\
& \sum_{j\in J} x_{ij}^t \le q_i y_i^t, & t\in T, i\in I, \\
& \sum_{i\in I} y_i^t \le p^t, & t\in T, \\
& y_i^t \le y_i^{t+1}, & t\in T\setminus\{|T|\}, i\in I, \\
& y_i^t \in \{0,1\}, & t\in T, i\in I, \\
& x_{ij}^t \ge 0, & t\in T, i\in I, j\in J \\
& z_j^t \ge 0, & t\in T, j\in J.
\end{aligned}
$$

Outline

Equivalent formulation

Projection onto y-space: master problem

$$
\begin{aligned}
\min_{y} \quad & \sum_{t\in T}\sum_{i\in I} f_i^t y_i^t + Q(y), & \\
\text{subject to} \quad & \sum_{i\in I} y_i^t \le p^t, & t\in T, \\
& y_i^t \le y_i^{t+1}, & t\in T\setminus\{|T|\}, i\in I, \\
& y_i^t \in \{0,1\}, & t\in T, i\in I.
\end{aligned}
$$

Subproblem $Q(y)$

$$
\begin{aligned}
Q(y) = \quad \min_{x,z} \quad & \sum_{t\in T}\left(\sum_{j\in J}\sum_{i\in I} c_{ij}^t x_{ij}^t + \sum_{j\in J} h_j^t z_j^t\right), & \\
\text{subject to} \quad & \sum_{i\in I} x_{ij}^t + z_j^t = d_j^t, & t\in T, j\in J, \\
& \sum_{j\in J} x_{ij}^t \le q_i y_i^t, & t\in T, i\in I, \\
& x_{ij}^t \ge 0, \quad z_j^t \ge 0 & t\in T, i\in I, j\in J.
\end{aligned}
$$

Subproblems are separable for the $|T|$ time periods

$$Q(y) = \sum_{t \in T} \text{SubLP}(y, t)$$

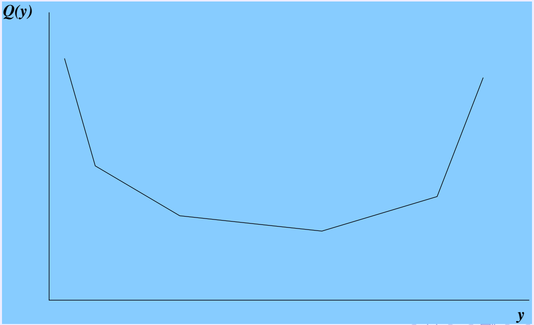
where

$$\begin{aligned}
\text{SubLP}(y,t) = \quad & \min & & \sum_{j \in J} \sum_{i \in I} c_{ij}^t x_{ij}^t + \sum_{j \in J} h_j^t z_j^t, & & \\
& \text{subject to} & & \sum_{i \in I} x_{ij}^t + z_j^t = d_j^t, & & j \in J, \\
& & & \sum_{j \in J} x_{ij}^t \leq q_i y_i^t, & & i \in I, \\
& & & x_{ij}^t \geq 0, & & i \in I, j \in J, \\
& & & z_j^t \geq 0, & & j \in J.
\end{aligned}$$

$Q(y)$ is convex nondifferentiable function

- $Q(y)$ convex:
$$Q(y) \geq Q(y^v) + \partial Q(y^v)(y - y^v)$$
- Thus it can be lower approximated by cutting planes: the solution of $Q(y)$ provides $Q(y^v)$ and its Lagrange multipliers belong to the subdifferential $\partial Q(y^v)$.

Graph of $Q(y)$ for continuous y



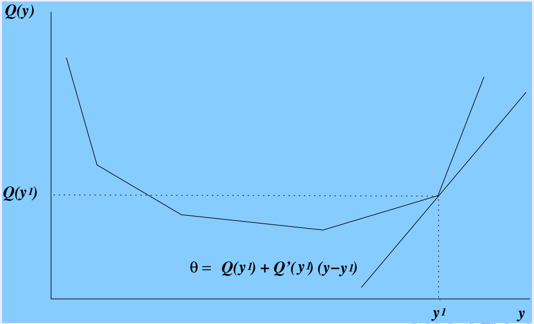
$Q(y)$ is convex nondifferentiable function

- $Q(y)$ convex:

$$Q(y) \geq Q(y^v) + \partial Q(y^v)(y - y^v)$$

- Thus it can be lower approximated by cutting planes: the solution of $Q(y)$ provides $Q(y^v)$ and its Lagrange multipliers belong to the subdifferential $\partial Q(y^v)$.

Solution of $Q(y^1)$: cutting plane at y^1



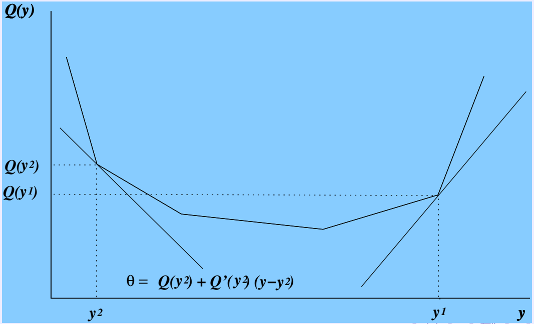
$Q(y)$ is convex nondifferentiable function

- $Q(y)$ convex:

$$Q(y) \geq Q(y^v) + \partial Q(y^v)(y - y^v)$$

- Thus it can be lower approximated by cutting planes: the solution of $Q(y)$ provides $Q(y^v)$ and its Lagrange multipliers belong to the subdifferential $\partial Q(y^v)$.

Solution of $Q(y^2)$: cutting plane at y^2



Solution of master problem

- $Q(y)$ is thus lower approximated by cutting planes.
- Generated cutting planes are iteratively added to the master problem.
- This is in essence Benders decomposition.
- Cuts aggregated for period: efficient enough and simpler master problem.

The master problem, with aggregation of cuts

$$
\begin{array}{rll}
\min & \sum_{t\in T}\sum_{i\in I} f_i^t y_i^t + \theta, & \\
\text{subject to} & \theta \geq \sum_{t\in T}\sum_{j\in J} \lambda_j^{t,v} d_j^t + \sum_{t\in T}\sum_{i\in I} \mu_i^{t,v} q_i y_i^t, & v \in V, \\
& \sum_{i\in I} y_i^t \leq p^t, & t \in T, \\
& y_i^t \leq y_i^{t+1}, & t \in T \setminus \{|T|\}, i \in I \\
& y_i^t \in \{0,1\}, & t \in T, i \in I.
\end{array}
$$

Outline

Block-angular structure of SubLP(t): solved with BlockIP

$$\text{SubLP}(y,t) = \quad \min \quad \sum_{j \in J} \mathbf{c}_j^{t\,\top} \mathbf{x}_j^t$$

$$\text{subject to} \quad \begin{bmatrix} e^\top & & & & \\ & e^\top & & & \\ & & \ddots & & \\ & & & e^\top & \\ L & L & \dots & L & I \end{bmatrix} \begin{bmatrix} \mathbf{x}_1^t \\ \mathbf{x}_2^t \\ \vdots \\ \mathbf{x}_m^t \\ \mathbf{x}_0^t \end{bmatrix} = \begin{bmatrix} d_1^t \\ d_2^t \\ \vdots \\ d_m^t \\ \mathbf{q}^t \end{bmatrix}$$

$$\mathbf{x}_j^t \geq 0, \quad j = 0, 1, \dots, m$$

- $L = [I \mid \mathbf{0}] \in \mathbb{R}^{n \times (n+1)}$.
- $\mathbf{c}_j^t = [c_{1j}^t, \dots, c_{nj}^t, h_j^t]^\top \in \mathbb{R}^{n+1}$: shipping and shortage costs customer j.
- $\mathbf{x}_j^t = [x_{1j}^t, \dots, x_{nj}^t, z_j^t]^\top \in \mathbb{R}^{n+1}$: amount shipped and shortage customer j.
- $\mathbf{q}^t = [q_1 y_1^t, \dots, q_n y_n^t]^\top \in \mathbb{R}^n$: rhs of linking constraints.

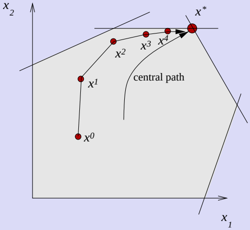
BlockIP is a specialized path-following method

Convex optimization problem

$$(P) \quad \begin{array}{lll} \min & f(x) & \\ \text{s.to} & Ax = b & [\lambda] \\ & x \geq 0 & [s] \end{array}$$

Central path defined by perturbed KKT-μ system

$$\begin{array}{rcl} A^\top \lambda + s - \nabla f(x) & = & 0 \\ Ax & = & b \\ XSe & = & \mu e \quad \mu \in \mathbb{R}^+ \\ s & > & 0 \end{array}$$



Some features of the specialized IPM

- Specialized IPM combines Cholesky factorizations and PCG for solution of Newton directions.
- Ad-hoc preconditioner in PCG is very efficient for facility location problems:
  - Proposition. When $n \to \infty$ the preconditioner becomes exact.
- Inexact cuts easy to obtain with IPMs: compute a suboptimal primal-dual feasible solution, avoiding last expensive PCG interior-point iterations.
- Convergence of cutting-planes guaranteed for inexact cuts [Zakeri, Philpott, Ryan, SIOPT00].
- Efficient implementation of BlockIP: Fully written in C++, about 14000 lines of code.
- BlockIP solves LO, QO, or CO problems.

Outline

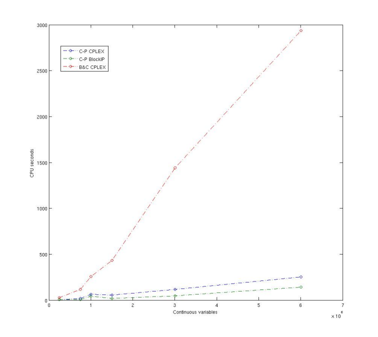
Instances

- Most instances in literature are easy and/or small.
- We developed an instance generator, governed by:
  - n: number of locations.
  - m: number of customers.
  - $|T|$: time periods.
  - $\alpha \in [0,1]$: controls plants capacity (the closer to 1, the larger)
  - $\beta \in [0,1]$: controls number of available plants per period (the closer to 1, the larger)
- We tried several combinations of m ,n, $|T|$, α and β.

Solution of “small” facility location instances

Average numbers for 25 instances with different (α, β)

						Cutting plane CPU		B&C CPU
n	m	$\|T\|$	const.	bin.var.	cont. var.	CPLEX	BlockIP	CPLEX
500	500	1	1001	500	250500	8.1	4.9	27.3
1000	1000	1	2001	1000	1001000	62.2	44.5	257.0
500	500	3	3003	1500	751500	17.0	7.2	118.6
1000	1000	3	6003	3000	3003000	115.7	48.0	1440.1
500	500	6	6006	3000	1503000	56.1	19.1	433.8
1000	1000	6	12006	6000	6006000	253.6	140.1	2936.0



Solution of very large-scale instances with $\varepsilon = 10^{-3}$

- World-wide problems: 100s locations, 100000s of customers.

Optimality tolerance 10^{-3} for the subproblems.

						BlockIP		CPLEX	
n	m	$\|T\|$	const.	bin.var.	cont. var.	gap	CPU	gap	CPU
200	100000	1	100201	200	20100000	0.0010	17.39	0.0002	100.54
200	100000	2	200602	400	40200000	0.0007	32.09	0.0001	297.03
200	100000	3	301003	600	60300000	0.0009	63.55	0.0035	912.05
200	500000	1	500201	200	100500000	0.0010	110.02	0.0000	1146.60
200	500000	2	1000602	400	201000000	0.0010	309.68	‡	
200	500000	3	1501003	600	301500000	0.0010	868.43	†	
200	1000000	1	1000201	200	201000000	0.0009	729.79	†	
200	1000000	2	2000602	400	402000000	0.0010	1109.64	†	
200	1000000	3	3001003	600	603000000	0.0009	3254.21	†	

† CPLEX ran out of memory (required more than 144 Gigabytes of RAM)

‡ CPLEX aborted

Conclusions and future work

- IP solver used for the solution of a large MILP.
- Facility location extensions: stochastic, phase-in/phase-out, etc
- Many additional MILP/MIQP applications with block-angular structure to be tried.

More details in the paper (submitted, available from Optimization Online, June 2015)

BlockIP is available for research purposes from
`www-eio.upc.edu/~jcastro/BlockIP.html`

Thanks for your attention