Challenging Applications of Stochastic Programming

Anton J. Kleywegt

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, Georgia, USA

Stochastic Programming SPXII Conference
Halifax

14 August 2010

Outline

Two-stage Stochastic Program

- In first stage, decision maker chooses $x \in X$
- Thereafter, in second stage, decision maker observes some random data ξ
- Then decision maker chooses $y(x, \xi) \in Y(x, \xi)$ to maximize $g(x, y, \xi)$
- In first stage, decision maker chooses $x \in X$ to maximize $F(x) := f(x) + \mathbb{E}[\max\{g(x, y, \xi) \; : \; y \in Y(x, \xi)\}] = f(x) + \mathbb{E}[g(x, y(x, \xi), \xi)]$

Stochastic Bilevel Program

- First, in upper level, decision maker 1 chooses $x \in X$
- Thereafter, in lower level, decision maker 2 observes x and some random data ξ
- Then decision maker 2 chooses $y(x, \xi) \in Y(x, \xi)$ to maximize $g(x, y, \xi)$
- In upper level, decision maker 1 wants to maximize $F(x) := \mathbb{E}[f(x, y(x, \xi))]$

Two-stage Stochastic Program

- In first stage, decision maker chooses $x \in X$
- Thereafter, in second stage, decision maker observes some random data ξ
- Then decision maker chooses $y(x, \xi) \in Y(x, \xi)$ to maximize $g(x, y, \xi)$
- In first stage, decision maker chooses $x \in X$ to maximize $F(x) := f(x) + \mathbb{E}[\max\{g(x, y, \xi) : y \in Y(x, \xi)\}] = f(x) + \mathbb{E}[g(x, y(x, \xi), \xi)]$
- Note that two-stage stochastic program is well defined even if set $\arg\max\{g(x, y, \xi) : y \in Y(x, \xi)\}$ of optimal second stage solutions is not a singleton, because optimal second stage objective value $\max\{g(x, y, \xi) : y \in Y(x, \xi)\}$ is well defined

Stochastic Bilevel Program

- First, in upper level, decision maker 1 chooses $x \in X$
- Thereafter, in lower level, decision maker 2 observes x and some random data ξ
- Then decision maker 2 chooses $y(x, \xi) \in Y(x, \xi)$ to maximize $g(x, y, \xi)$
- In upper level, decision maker 1 wants to maximize $F(x) := \mathbb{E}[f(x, y(x, \xi))]$
- Note that objective of decision maker 1 depends on the decision of decision maker 2, and not on the optimal objective value of decision maker 2
- Fundamental problem occurs if set $\arg\max\{g(x, y, \xi) : y \in Y(x, \xi)\}$ of optimal lower level decisions is not a singleton, because in that case upper level objective $F(x)$ is not well defined

Bilevel Program

Bilevel programs are often formulated as follows:

$$\begin{aligned} \max_{x \in X, y(x,\xi) \in Y(x,\xi)} \quad & \{F(x) := \mathbb{E}[f(x, y(x, \xi))]\} \\ \text{subject to} \quad & y(x, \xi) \in \arg\max\{g(x, y, \xi) \,:\, y \in Y(x, \xi)\} \\ & \quad \text{for all } x, \xi \end{aligned}$$

Bilevel Program

Important modeling issues

- Note that the formulation above implies that if $\arg\max\{g(x, y, \xi) : y \in Y(x, \xi)\}$ is not a singleton, then a lower level optimizer $y(x, \xi) \in \arg\max\{g(x, y, \xi) : y \in Y(x, \xi)\}$ that is most favorable to the upper level decision maker will be chosen. In many applications, such an assumption is questionable.

Bilevel Program

Important modeling issues

- Also note that to compute objective $F(x)$, upper level decision maker typically has to compute $y(x, \xi)$, which requires upper level decision maker to know the objective $g(x, y, \xi)$ and the feasible set $Y(x, \xi)$ of the lower level decision maker
- If upper level decision maker does not know the objective or the feasible set of the lower level decision maker, then the upper level decision maker will have errors in the upper level problem formulation
- If the lower level decision maker does not necessarily optimize exactly, then the upper level decision maker will make an error

Outline

Bilevel Programs

Bilevel programs go by many names:

- Leader-follower games
- Stackelberg games
- Principal-agent models

Defender-Attacker Problems

- First, defender chooses a defense strategy $x \in X$
- Thereafter, attacker observes the defense strategy x (this assumption may be questioned) and some random data ξ
- Then attacker chooses a plan of attack $y(x, \xi) \in Y(x, \xi)$ to maximize $g(x, y, \xi)$, for example, maximize probability of success
- Defender wants to maximize $F(x) := \mathbb{E}[f(x, y(x, \xi))]$, for example, minimize expected amount of damage
- Assumption that defender knows attacker's objectives and constraints questionable

Insurance Problems

- First, insurance company chooses a menu of contracts $x \in X$
- Thereafter, customers observe the menu of contracts x and some random data ξ that typically includes private information
- Then each customer chooses a contract $y(x, \xi) \in Y(x, \xi)$ to maximize $g(x, y, \xi)$, for example, maximize expected risk-adjusted benefits
- Insurance company wants to maximize $F(x) := \mathbb{E}[f(x, y(x, \xi))]$, for example, maximize expected profits
- Assumption that insurance company knows customers' objectives, risk assessments, and responses to risk questionable

Outline

Example

- Seller sells 2 substitutable products: product 1 is more expensive and has lower operating cost, and product 2 is cheaper but has higher operating cost
- Seller chooses prices $x = (x_1, x_2) \geq 0$ for products 1 and 2
- Thereafter, buyers observe the prices x and incremental operating cost ξ of product 2, that typically is private information
- Then each buyer chooses amount y_1 of product 1 and amount y_2 of product 2 to buy, to satisfy total demand at minimum total cost
- Total demand is given by $a - b_1 x_1 - b_2 x_2$
- Seller wants to maximize total revenue $x_1 y_1 + x_2 y_2$

Example

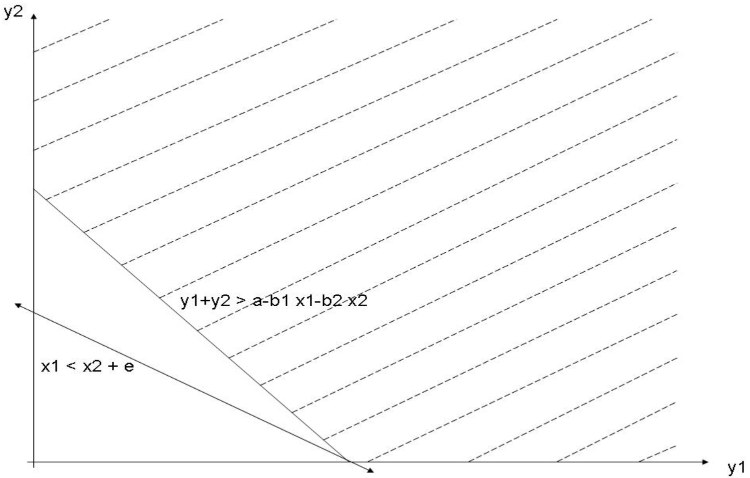
Lower level (buyer's) problem: Choose

$$\begin{array}{rcl} y^*(x,\xi) & \in & \arg\min \quad x_1 y_1 + (x_2 + \xi) y_2 \\ & & \text{subject to} \quad \begin{array}{rcl} y_1 + y_2 & \geq & a - b_1 x_1 - b_2 x_2 \\ y_1, y_2 & \geq & 0 \end{array} \end{array}$$

Upper level (seller's) problem: Choose

$$\begin{array}{rcl} x^* & \in & \arg\max \quad \mathbb{E}[x_1 y_1^*(x,\xi) + x_2 y_2^*(x,\xi)] \\ & & \text{subject to} \quad x_1, x_2 \geq 0 \end{array}$$

Example — Lower Level Problem



Example

Lower level (buyer's) problem:

- If $x_1 < x_2 + \xi$, then $y_1^*(x, \xi) = a - b_1 x_1 - b_2 x_2$, $y_2^*(x, \xi) = 0$
- If $x_1 > x_2 + \xi$, then $y_2^*(x, \xi) = a - b_1 x_1 - b_2 x_2$, $y_1^*(x, \xi) = 0$
- If $x_1 = x_2 + \xi$, then $\arg\max\{g(x, y, \xi) : y \in Y(x, \xi)\}$ is the line segment between $(a - b_1 x_1 - b_2 x_2,\ 0)$ and $(0,\ a - b_1 x_1 - b_2 x_2)$

Example

Upper level (seller's) problem:

- Case 1: $x_1 < x_2 + \xi$: Choose

$$\begin{aligned} x^{1*} \in \arg\max \quad & \mathbb{E}[x_1(a - b_1 x_1 - b_2 x_2)] \\ \text{subject to} \quad & x_1 < x_2 + \xi \\ x_1, x_2 \geq 0 & \end{aligned}$$

- Case 2: $x_1 > x_2 + \xi$: Choose

$$\begin{aligned} x^{2*} \in \arg\max \quad & \mathbb{E}[x_2(a - b_1 x_1 - b_2 x_2)] \\ \text{subject to} \quad & x_1 > x_2 + \xi \\ x_1, x_2 \geq 0 & \end{aligned}$$

- Case 3: $x_1 = x_2 + \xi$: Then upper level problem not well defined

Example

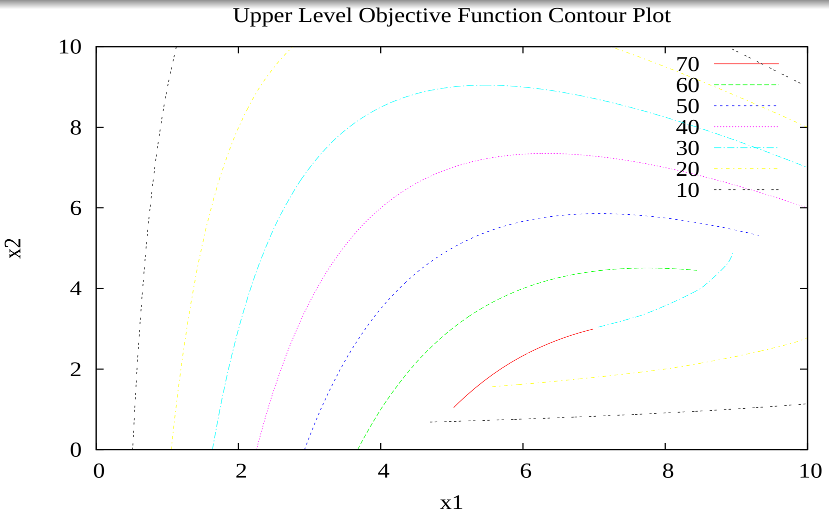
Upper level (seller's) problem:

- Choose

$$
\begin{aligned}
x^* \quad \in \quad \arg\max \qquad & \Big\{ \mathbb{E}[x_1^{1*} y_1^*(x^{1*}, \xi) + x_2^{1*} y_2^*(x^{1*}, \xi)], \\
& \mathbb{E}[x_1^{2*} y_1^*(x^{2*}, \xi) + x_2^{2*} y_2^*(x^{2*}, \xi)] \Big\}
\end{aligned}
$$

- Next plot for $a = 20$, $b_1 = b_2 = 1$, $\xi = 4$

Example


Upper Level Objective Function Contour Plot
70
60
50
40
30
20
10
x2
x1
0
2
4
6
8
10

Example

Important points to note from example:

- Even if lower level problem is very simple (linear program), the upper level problem can be badly behaved (upper level objective function $F(x)$ is discontinuous, and not well defined everywhere)
- Unfortunately, even if the set of bad points (for example discontinuity points) is small, often the optimal solution occurs at such a point
- A small error in the estimate of ξ or its distribution may lead to a large decrease in the objective value
- For example, if the upper level decision maker slightly overestimated ξ, then the actual objective value in the example will turn out to be about 20, as opposed to an optimal value of about 70

Outline

Stochastic MPEC

- First, in upper level, decision maker 1 chooses $x \in X$
- Thereafter, in lower level, multiple decision makers $i = 1, \ldots, n$ observe x and some random data ξ_i
- Each lower level decision maker i has subjective conditional distribution $H_i(\xi_{-i}|\xi_i)$ of $\xi_{-i} = (\xi_1, \ldots, \xi_{i-1}, \xi_{i+1}, \ldots, \xi_n)$ given ξ_i
- Each lower level decision maker i chooses $y_i(x, \xi_i) \in Y_i(x, \xi_i)$ to maximize $\mathbb{E}_{H_i}[g_i(x, y, \xi)|\xi_i]$
- Assume the outcome is an equilibrium $y(x, \xi)$
- In upper level, decision maker 1 wants to maximize $F(x) := \mathbb{E}[f(x, y(x, \xi))]$

Stochastic MPEC

Review of equilibrium

- $y(x,\xi) = (y_1(x,\xi_i),\ldots,y_n(x,\xi_i))$ is an equilibrium at (x,ξ) if $\mathbb{E}_{H_i}[g_i(x,(y_i(x,\xi_i),y_{-i}(x,\xi_i)),\xi)|\xi_i] \geq \mathbb{E}_{H_i}[g_i(x,(y_i,y_{-i}(x,\xi_i)),\xi)|\xi_i]$ for all $y_i \in Y_i(x,\xi_i)$, for all $i = 1,\ldots,n$
- At a point (x,ξ), an equilibrium may exist or not
- If an equilibrium exists at a point (x,ξ), it may be unique, or not
- If a unique equilibrium does not exist at a point (x,ξ), then the upper level objective $f(x,y(x,\xi))$ is not well defined at (x,ξ)

Mathematical Programs with Equilibrium Constraints

MPECs are often formulated as follows:

$$\max_{x \in X, y_i(x,\xi_i) \in Y_i(x,\xi_i)} \quad \{F(x) := \mathbb{E}[f(x, y(x, \xi))]\}$$

$$\text{subject to} \quad y_i(x, \xi_i) \in \arg\max\{\mathbb{E}_{H_i}[g_i(x, (y_i, y_{-i}(x, \xi_{-i})), \xi)|\xi_i]$$

$$\text{for all } x, \xi, i = 1, \ldots, n$$

Mathematical Programs with Equilibrium Constraints

Important modeling issues

- Note that the formulation above implies that if there is an equilibrium $y(x, \xi)$ but it is not unique, then an equilibrium that is most favorable to the upper level decision maker will be chosen. As before, such an assumption is questionable in many applications.
- In addition, as mentioned, an equilibrium may not exist at (x, ξ), in which case by definition the objective value is $-\infty$

Mathematical Programs with Equilibrium Constraints

Important modeling issues

- Also note that to compute objective $F(x)$, upper level decision maker typically has to compute $y(x, \xi)$, which requires upper level decision maker to know the objective $g_i(x, y, \xi)$, the feasible set $Y_i(x, \xi_i)$, and the subjective conditional distribution $H_i(\xi_{-i}|\xi_i)$ of each lower level decision maker i
- If upper level decision maker does not know the objectives, or the feasible sets, or the subjective distributions of each lower level decision maker, then the upper level decision maker will have errors in the upper level problem formulation

Mathematical Programs with Equilibrium Constraints

Important modeling issues

- If some lower level decision makers do not necessarily optimize exactly, then the upper level decision maker will make an error
- If an equilibrium does not result from the interaction of the lower level decision makers, for example if the equilibrium is unstable, then the upper level decision maker will make an error

Outline

Road Pricing Problems

- First, toll road operator chooses a toll $x \in X$ on each toll link
- Thereafter, each traveler i observes the toll x and some random data ξ_i, often related to travel demand
- Then each traveler i chooses a path $y_i(x, \xi_i)$ to minimize generalized travel cost $\mathbb{E}_{H_i}[g_i(x, y, \xi)|\xi_i]$
- Note that due to congestion effects, the travel cost $g_i(x, y, \xi)$ depends not only on the path $y_i(x, \xi_i)$ chosen by traveler i, but also on the paths $y_{-i}(x, \xi_{-i})$ chosen by the other travelers of traveler
- Toll road operator wants to maximize $F(x) := \mathbb{E}[f(x, y(x, \xi))]$, for example, maximize expected profit

Mechanism Design problems

- MPECs are also called Mechanism Design problems in the economics literature
- In mechanism design, the upper level decision maker often chooses a function of the lower level decisions (or the information revealed by the lower level decision makers)

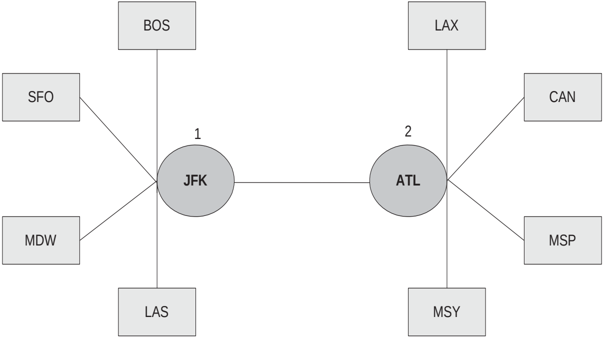
Mechanism Design problems

- The mechanism design literature uses a result called the revelation principle, that states that under some conditions, the upper level decision maker can add constraints, called incentive compatibility constraints, to make it in the best interest of each lower level decision maker to reveal his/her private information truthfully, without loss of optimality
- However, the mechanism design literature almost never reminds the reader that the revelation principle is based on the assumption that the upper level decision maker knows the subjective distribution of each lower level decision maker, and as shown, if the assumption does not hold, then the error may be large

Outline

Example

Airline -1 operates flights into and out of JFK
Airline 1 operates flights into and out of ATL
Itinerary BOS-JFK-ATL is intraline
Itinerary BOS-JFK-ATL-LAX is interline



Motivation

Major problem in non-alliance setting:

- Customer wants to travel BOS-LAX
- Customer can obtain interline itinerary BOS-JFK-ATL-LAX by purchasing itinerary BOS-JFK-ATL from airline -1 and itinerary ATL-LAX from airline 1
- Note that online reservation systems put together such interline itineraries, with no additional effort required from customer
- Airline -1 sets price p_{-1} for itinerary BOS-JFK-ATL and airline 1 sets price p_1 for itinerary ATL-LAX
- Customer pays price $p_{-1} + p_1$ for interline itinerary BOS-JFK-ATL-LAX

Motivation

Major problem in non-alliance setting (continued):

- Consider pricing game played by airline -1 and airline 1
  - As usual, demand decreases with increase in price
  - Suppose airline -1 sets low price p_{-1} to stimulate demand for itinerary BOS-JFK-ATL
  - Best response of airline 1 is to set high price p_1 for itinerary ATL-LAX to capture most of revenue
  - Result: Total price $p_{-1} + p_1$ in game equilibrium larger than optimal for both airlines combined

Motivation

- Alliances becoming more important in airline and ocean transportation
  - Interline airline itineraries within the U.S. increased from 10% in 1998 to 20% in 2004
  - Interline airline itineraries contributed 46% of revenues of U.S. domestic flights in 2004
- Growth in alliances:
  - Airline industry: Star Alliance, SkyTeam Alliance, Oneworld Alliance
  - Ocean cargo industry: Grand Alliance, New World Alliance, United Alliance
- Revenue management pervasive in airline industry
- However, revenue management in alliance setting not well developed

Alliance Operations

- Each alliance member can sell tickets for itineraries that include flights operated by other alliance members
- Airline that sells ticket called “marketing airline”
- Airline that operates flight called “operating airline”

Types of Alliances

- "Free-sell" or "soft block" alliance
  - Alliance agreement specifies transfer price that marketing airline has to pay operating airline
  - At booking time, operating airline may accept or reject booking request of marketing airline
  - Transfer price may be static or dynamic
  - Examples
    - Static transfer price: When airline -1 sells BOS-JFK-ATL-LAX itinerary, airline -1 pays airline 1 \$200 for each seat sold on flight ATL-LAX. When airline 1 sells BOS-JFK-ATL-LAX itinerary, airline 1 pays airline -1 \$300 for each seat sold on flight BOS-JFK-ATL
    - Static proration: When either airline sells BOS-JFK-ATL-LAX itinerary, airline -1 receives 60% of revenue and airline 1 receives 40% of revenue
    - Dynamic transfer prices

Types of Alliances

- "Resource-exchange" or "hard block" alliance
  - Alliance members exchange resources (seat space on various flights, and possibly money) in advance of sales season
  - For example, airline -1 gives airline 1 30 seats on BOS-JFK, 25 seats on SFO-JFK, 20 seats on MDW-FJK, and 35 seats on LAS-JFK, in exchange for 40 seats on ATL-LAX, 30 seats on ATL-MSP, and $10000
  - After exchange, each airline controls bookings as though they are owner of received resources
  - After exchange, alliance members compete for same demand
  - Used in airline and ocean industries

Types of Alliances

- “Resource-exchange” or “hard block” alliance
  - Design decision: how much of each resource to exchange
  - Existing models do not take into account that after exchange, alliance members will compete

Model

- 2 sellers, indexed by $i = \pm 1$
- Set $\mathcal{R}$ of resources indexed by r (example, flights operated by either airline)
- Seller i has initial amount $b_{i,r}$ of resource r (example, if airline i operates flight r, then $b_{i,r} =$ number of seats on flight r, otherwise $b_{i,r} = 0$)
- In first stage, seller 1 gives seller -1 the right to use up to x_r units of resource r, with $-b_{-1,r} \leq x_r \leq b_{1,r}$
- After resource exchange, seller i has access to k_i resources, with $b_{i,r} - ix_r$ units of resource r, and can sell m_i products
- Matrix $A_i \in \mathbb{R}^{k_i \times m_i}$ has entries $(A_i)_{rj}$ denoting the amount of resource r required by seller i to provide each unit of product j

Model

- In second stage, each seller i chooses price $y_{i,j}$ of each product j
- Let $y_i := (y_{i,1}, \ldots, y_{i,m_i}) \in \mathbb{R}^{m_i}$
- Linear conditional expected demand:

$$D_{i,j}(y) \;=\; -E_{i,j}^{\mathsf{T}} y_i + B_{-i,j}^{\mathsf{T}} y_{-i} + c_{i,j}$$

  where $E_{i,j} \in \mathbb{R}^{m_i}$, $B_{-i,j} \in \mathbb{R}^{m_{-i}}$
- Let $E_i := (E_{i,1}, \ldots, E_{i,m_i})^{\mathsf{T}} \in \mathbb{R}^{m_i \times m_i}$,
  $B_{-i} := (B_{-i,1}, \ldots, B_{-i,m_i})^{\mathsf{T}} \in \mathbb{R}^{m_i \times m_{-i}}$,
  $c_i := (c_{i,1}, \ldots, c_{i,m_i}) \in \mathbb{R}^{m_i}$
- Let $Q_i := E_i + E_i^{\mathsf{T}} \in \mathbb{R}^{m_i \times m_i}$, assume Q_i positive definite
- Then conditional expected revenue of seller i
  $= \sum_{j=1}^{m_i} y_{i,j} D_{i,j}(y) = -\frac{1}{2} y_i^{\mathsf{T}} Q_i y_i + y_i^{\mathsf{T}} B_{-i} y_{-i} + c_i^{\mathsf{T}} y_i$

Model

In second stage, given first stage resource exchange x, we are interested in Nash equilibrium in which each seller i chooses prices $y_i \in \mathbb{R}^{m_i}$ to solve

$$\begin{array}{rl} \min\limits_{y_i \in \mathbb{R}^{m_i}} & \left\{ \frac{1}{2} y_i^{\mathsf{T}} Q_i y_i - y_i^{\mathsf{T}} B_{-i} y_{-i} - c_i^{\mathsf{T}} y_i \right\} \\ \text{s.t.} & A_i \left(-E_i y_i + B_{-i} y_{-i} + c_i \right) \;\leq\; b_i - ix \\ & l_i \;\leq\; -E_i y_i + B_{-i} y_{-i} + c_i \;\leq\; u_i \end{array}$$

Model

- Matrices B_i, c_i, and E_i, and therefore Q_i, may be random in first stage when resource exchange x is chosen, but become known in second stage before prices y_i are chosen
- For now, assume that for each realization of B_i, c_i, and E_i, equilibrium point exists and is unique
- Let $V_i(x)$, $i = \pm 1$, denote expected optimal objective values of problem above
- First stage problem (alliance design problem) is to choose $x \in [0, u]$ to maximize $V_{-1}(x) + V_1(x)$

Solution Method

Second stage best response problem of seller i

$$\begin{array}{cl} \min\limits_{y_i \in \mathbb{R}^{m_i}} & \left\{ \frac{1}{2} y_i^{\mathsf{T}} Q_i y_i - y_i^{\mathsf{T}} B_{-i} y_{-i} - c_i^{\mathsf{T}} y_i \right\} \\ \text{s.t.} & W_i \left(E_i y_i - B_{-i} y_{-i} \right) \;\geq\; d_i - iMx, \end{array}$$

Solution Method

First order (KKT) necessary and sufficient optimality conditions

$$\begin{aligned}
Q_i y_i - B_{-i} y_{-i} - c_i - E_i^{\mathsf{T}} W_i^{\mathsf{T}} \lambda_i &= 0 \\
-W_i \left(E_i y_i - B_{-i} y_{-i} \right) + d_i - iMx &\leq 0 \\
\lambda_i &\geq 0 \\
\lambda_i^{\mathsf{T}} \left[W_i \left(E_i y_i - B_{-i} y_{-i} \right) - d_i + iMx \right] &= 0
\end{aligned}$$

for $i = \pm 1$

Solution Method

$(y_{-1}^*, y_1^*, \lambda_{-1}^*, \lambda_1^*)$ is a solution of KKT system iff it is an optimal solution of following problem and optimal objective value is zero

$$\begin{array}{rl} \min\limits_{y_{-1},y_1,\lambda_{-1},\lambda_1} & \sum\limits_{i=\pm 1} \lambda_i^{\mathsf{T}} \left[W_i \left(E_i y_i - B_{-i} y_{-i}\right) - d_i + iMx\right] \\ \text{s.t.} & Q_i y_i - B_{-i} y_{-i} - c_i - E_i^{\mathsf{T}} W_i^{\mathsf{T}} \lambda_i \;=\; 0, \quad i = \pm 1 \\ & -W_i \left(E_i y_i - B_{-i} y_{-i}\right) + d_i - iMx \;\leq\; 0, \quad i = \pm 1 \\ & \lambda_i \;\geq\; 0, \quad i = \pm 1 \end{array}$$

It follows from first constraint that

$$\lambda_i^{\mathsf{T}} W_i \;=\; y_i^{\mathsf{T}} Q_i E_i^{-1} - y_{-i}^{\mathsf{T}} B_{-i}^{\mathsf{T}} E_i^{-1} - c_i^{\mathsf{T}} E_i^{-1}$$

Solution Method

After substitution, optimization problem becomes

$$\begin{array}{rl} \min\limits_{y_{-1},y_1,\lambda_{-1},\lambda_1} & y^{\mathsf{T}}\Psi y + \sum\limits_{i=\pm 1} \lambda_i^{\mathsf{T}}\left[-d_i + iMx\right] \\ \text{s.t.} & Q_i y_i - B_{-i} y_{-i} - c_i - E_i^{\mathsf{T}} W_i^{\mathsf{T}} \lambda_i \ = \ 0, \quad i = \pm 1 \\ & -W_i\left(E_i y_i - B_{-i} y_{-i}\right) + d_i - iMx \ \leq \ 0, \quad i = \pm 1 \\ & \lambda_i \ \geq \ 0, \quad i = \pm 1 \end{array}$$

where

$$\begin{aligned} \Psi &:= \begin{bmatrix} Q_{-1} + B_{-1}^{\mathsf{T}} S_1 B_{-1} & \Delta \\ \Delta^{\mathsf{T}} & Q_1 + B_1^{\mathsf{T}} S_{-1} B_1 \end{bmatrix} \\ S_i &:= \left(E_i^{-1} + (E_i^{-1})^{\mathsf{T}}\right)/2 \\ \Delta &:= -(B_1 + B_{-1}^{\mathsf{T}})/2 - \left(Q_{-1} E_{-1}^{-1} B_1 + B_{-1}^{\mathsf{T}} (E_1^{-1})^{\mathsf{T}} Q_1\right)/2 \end{aligned}$$

Solution Method

- Suppose demand $D_{i,j}(y)$ for product j of seller i depends only on the prices $y_{i,j}$ and $y_{-i,j}$ of product j of both sellers
- Then $Q_i = E_i$ and B_{-i} are diagonal and Ψ is positive definite iff $(E_{-1}E_1 - B_{-1}B_1)^2$ is positive definite
- Note that $(E_{-1}E_1 - B_{-1}B_1)^2$ is always positive semidefinite and is positive definite iff $E_{-1}E_1 - B_{-1}B_1$ has no zero diagonal elements
- If demand $D_{i,j}(y)$ depends more strongly on $y_{i,j}$ than on $y_{-i,j}$, then Ψ is positive definite

Solution Method

- If active constraints of quadratic program remain unchanged in a neighborhood of x, then $V_{-1}(x) + V_1(x)$ is continuously differentiable in the neighborhood of x, and derivative is easy to compute
- Problem $\min_{x \in [0,u]} V_{-1}(x) + V_1(x)$ may have multiple local minima
- We used, as a heuristic, trust region method with BFGS construction of second order part of model function
- Use multiple runs with different initial points to check for multiple local minima
- For stochastic problem, use sample average approximation method
- Solution times vary from minutes for deterministic problems to hours for stochastic problems

Local Minima