

Stochastic Combinatorial Truckload Procurement Auctions

Shervin AhmadBeygi¹ Amy M. Cohn¹ Richard L. Chen¹
Damian R. Beil² Amitabh Sinha²

¹Department of Industrial and Operations Engineering, University of Michigan

²Stephen M. Ross School of Business, University of Michigan

INFORMS Annual Meeting, November 2007, Seattle

Introduction

- **Combinatorial auctions** can be very beneficial in the context of truckload procurement
- Ideally, they allow bidders to capture the **economies of scale and scope** between items
- In practice, the **exponential number of bids** makes both bidding and winner determination intractable even for a **deterministic case**
- In addition, the problem structure is **uncertain**, mostly to capture **opportunities** in the spot market
- We propose a mathematical programming based approach to implicitly consider all the combinations of loads **without explicit enumeration** for both the **deterministic and stochastic** cases
 - In this talk, we emphasize the stochastic case

Truckload Auctions

- Shippers auction off loads (origin/destination pairs) to trucking companies
- A shipper might have many different loads to offer at the same time
- The price trucking companies offer for a set of loads depends on the distance and the company's cost-per-mile
- It also depends on the other loads the trucking company is moving

Example: Economies of Scope

- When bidding for a load, bidder's cost depends on what other loads they win as well:

- Bidder's cost to carry a load from A to B:

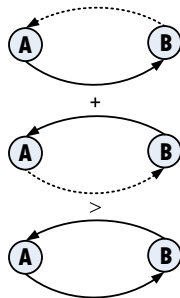
$$\text{loaded} \frac{\text{cost}}{\text{mile}} * (d_{A \rightarrow B}) + \text{empty} \frac{\text{cost}}{\text{mile}} * (d_{B \rightarrow A})$$

- Bidder's cost to carry a load from B to A:

$$\text{loaded} \frac{\text{cost}}{\text{mile}} * (d_{B \rightarrow A}) + \text{empty} \frac{\text{cost}}{\text{mile}} * (d_{A \rightarrow B})$$

- Bidder's cost to carry both loads:

$$\text{loaded} \frac{\text{cost}}{\text{mile}} * (d_{A \rightarrow B} + d_{B \rightarrow A})$$



Basic Combinatorial Auction: Set-Partitioning Formulation

- Bidders construct, price, and submit bids for sets (bundles) of items
- Bidders communicate bundle prices to the auctioneer, who solves a winner determination problem

$$z^*(\mathcal{A}^B) = \min \sum_{k \in \mathcal{K}} \sum_{S \subseteq \mathcal{A}^B} p_S^k x_S^k$$

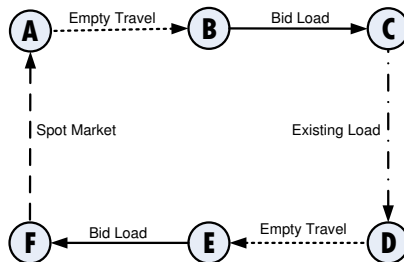
$$\text{subject to} \quad \sum_{k \in \mathcal{K}} \sum_{S \subseteq \mathcal{A}^B: l \in S} x_S^k = 1 \quad \forall l \in \mathcal{A}^B$$

$$x_S^k \in \{0, 1\} \quad \forall k \in \mathcal{K}, S \subseteq \mathcal{A}^B$$

- This is a *set-partitioning* problem with an **exponential number** of variables (bundles): $2^{|\mathcal{A}^B|} - 1$, where $\mathcal{A}^B$ is the set of bid loads
 - For 30 loads we get over a **billion** combinations
 - Practical cases may have tens of thousands of loads

Pricing a Bundle from Carriers' View

- Here is a more complex case:



- Note that the cost of each bundle can be found by solving a min-cost flow problem
- Key Insight:** we can transmit the price function parameters instead of the prices for all the bundles

Literature

- Combinatorial auctions:
 - Pekec and Rothkopf 2003
 - Sandholm et al 2005
 - Crampton et al 2006
- Truckload procurement auctions:
 - Elmaghraby and Keskinocak 2002
 - Caplice and Sheffi 2003
 - Sheffi 2004
 - Ergun et al 2007
- Stochastic multi-commodity flow problem:
 - Birge 1982
 - Aneja and Nair 1982
 - Soroush and Mirchandani 1990
 - Topaloglu and Powell 2004

Alternative Approach: Implicit Bidding Approach

- **Alternative Approach:** Each carrier (k) submits their underlying cost structure represented by three vectors (d^k, e^k, c^k) , where:
 - d_a^k : cost to carry bid load $a \in \mathcal{A}^B$
 - e_a^k : empty traveling cost on arc $a \in \mathcal{A}^E$
 - c_a^k : cost corresponding to the connection arcs $a \in \mathcal{A}^C$
- We can use the cost structure **directly** in the set-partitioning problem to get a multi-commodity flow formulation
- This approach eliminates the need for carriers to explicitly price all bundles

Deterministic Truckload Procurement Auction Problem

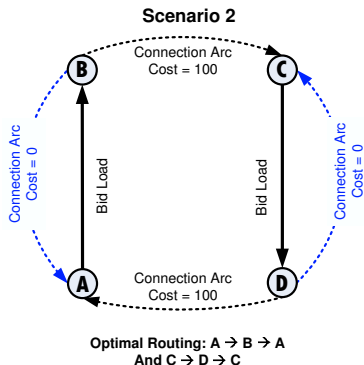
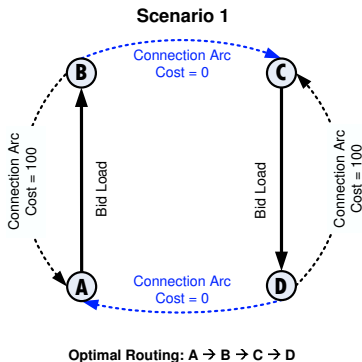
$$\begin{aligned}
 \min \quad & \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^{\mathcal{B}}} d_a^k x_a^k + \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^{\mathcal{E}}} e_a^k y_a^k + \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^{\mathcal{C}}} c_a^k z_a^k \\
 s.t. \quad & \sum_{k \in \mathcal{K}} x_a^k = 1 && \forall a \in \mathcal{A}^{\mathcal{B}} \\
 & \sum_{a \in \mathcal{A}^{\mathcal{B}}: O(a)=i} x_a^k + \sum_{a \in \mathcal{A}^{\mathcal{E}}: O(a)=i} y_a^k + \sum_{a \in \mathcal{A}^{\mathcal{C}}: O(a)=i} z_a^k \\
 = \quad & \sum_{a' \in \mathcal{A}^{\mathcal{B}}: D(a')=i} x_{a'}^k + \sum_{a' \in \mathcal{A}^{\mathcal{E}}: D(a')=i} y_{a'}^k + \sum_{a' \in \mathcal{A}^{\mathcal{C}}: D(a')=i} z_{a'}^k && \forall i \in \mathcal{N}, \forall k \in \mathcal{K} \\
 & z_a^k \leq u_a^k && \forall a \in \mathcal{A}^{\mathcal{C}}, \forall k \in \mathcal{K} \\
 & x_a^k \in \{0, 1\} && y_a^k \in \mathbb{Z}^+ && z_a^k \in \mathbb{Z}^+
 \end{aligned}$$

Discussion

- The proposed formulation is a **multi-commodity flow** problem
- x variables set the **assignment** of the bid loads to the carriers
- y and z variables use the existing loads or empty travel arcs to create a **continuous move**
- If we fix the values of x variables, then the problem **decomposes** into smaller network flows for each carrier
- Therefore, we can **relax the integrality** of y and z variables
- **Next step:** how to consider uncertainty in the cost of connection arcs?
- *The cost of connection arcs is not known at the time of auction!*

Stochastic Truckload Procurement Auction: Example

- The connection costs are not known at the time of auction:



Stochastic Truckload Procurement Auction

- We assume a finite set of **scenarios** with respect to the cost of each connection arc
 - Each scenario $s \in \mathcal{S}$ has probability p^s
 - c_a^{ks} is the cost of connection arc a under scenario s for carrier k
- Carriers may **change** the routing after the information about their connection arc costs are revealed
- In this case we can minimize the **expected cost** over possible scenarios
- This is a **two-stage stochastic programming** problem with recourse

Stochastic Programming Formulation

$$\min \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^{\mathcal{B}}} d_a^k x_a^k + Q^P(x)$$

$$\sum_{k \in \mathcal{K}} x_a^k = 1 \quad \forall a \in \mathcal{A}^{\mathcal{B}}$$

$$x_a^k \in \{0, 1\} \quad \forall a \in \mathcal{A}^{\mathcal{B}}, \quad \forall k \in \mathcal{K}$$

- The **first-stage** decision, x assigns loads to carriers
- $Q^P(x)$, is the objective function of the **primal recourse problem** corresponding to the first stage decision x
- Second-stage decisions here are y and z

Decomposition of the Recourse Problem

- The recourse problem can be **decomposed** by scenario:

$$\mathcal{Q}^P(x) = \sum_{s \in \mathcal{S}} p^s \mathcal{Q}_s^P(x)$$

- Furthermore, the set of scenarios ($\mathcal{S}$) can also be reduced to **carrier-specific** scenario sets ($\mathcal{S}^k$):

$$\mathcal{Q}^P(x) = \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}^k} p^s \mathcal{Q}_{ks}^P(x)$$

- As we decompose the scenarios by carriers, the total number of combinations we need to consider drops dramatically from $|\mathcal{S}| = |\mathcal{S}^1 \times \mathcal{S}^2 \times \dots \times \mathcal{S}^k|$ to $\sum_{k \in \mathcal{K}} |\mathcal{S}^k|$

Carrier Specific Scenario Sets: Example

- Consider a case where there are two carriers and each have three possible outcomes: L, M, H
- The set of all possible scenarios is:
 - $\mathcal{S} = \{(L, L), (L, M), (L, H), (M, L), \dots, (H, H)\}$
 - $|\mathcal{S}| = 9$
- However, carrier-specific scenario sets are:
 - $\mathcal{S}^1 = \{L, M, H\}$
 - $\mathcal{S}^2 = \{L, M, H\}$
 - Therefore: $\sum_{k \in \mathcal{K}} |\mathcal{S}^k| = 6$

The Structure of the Recourse Problem

The decomposed recourse problem corresponding to carrier k under scenario s can be formulated as:

$$Q_{ks}^P(x) = \min \sum_{a \in \mathcal{A}^E} e_a^k y_a^k + \sum_{a \in \mathcal{A}^C} c_a^{ks} z_a^k$$

$$\begin{aligned} & \sum_{a \in \mathcal{A}^E : O(a)=i} y_a^k - \sum_{a' \in \mathcal{A}^E : D(a')=i} y_{a'}^k + \sum_{a \in \mathcal{A}^C : O(a)=i} z_a^k - \sum_{a' \in \mathcal{A}^C : D(a')=i} z_{a'}^k \\ &= \sum_{a' \in \mathcal{A}^B : D(a')=i} x_{a'}^k - \sum_{a \in \mathcal{A}^B : O(a)=i} x_a^k \quad \forall i \in \mathcal{N} \\ & z_a^k \leq u_a^k \quad \forall a \in \mathcal{A}^C \end{aligned}$$

$$\begin{array}{ll} y_a^k \in \mathbb{Z}^+ & \forall a \in \mathcal{A}^E \\ z_a^k \in \mathbb{Z}^+ & \forall a \in \mathcal{A}^C \end{array} \Rightarrow \begin{array}{ll} y_a^k \geq 0 & \forall a \in \mathcal{A}^E \\ z_a^k \geq 0 & \forall a \in \mathcal{A}^C \end{array}$$

Discussion

- The proposed formulation for the recourse problem is a **min-cost flow** problem, therefore:
 - The recourse problems are **very easy** to solve
 - As the first stage decisions are set, the LP-relaxation of the recourse problem has **integer solutions**
 - Therefore, there is **no duality gap**
- **Next step:** How can we solve the stochastic programming formulation?
- We can use an **algorithmic approach** based on Bender's decomposition, known as the L-shaped algorithm

Solution Approach - Step 1

Here is the original formulation:

$$\min \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^{\mathcal{B}}} d_a^k x_a^k + \mathcal{Q}^P(x)$$

$$\sum_{k \in \mathcal{K}} x_a^k = 1 \quad \forall a \in \mathcal{A}^{\mathcal{B}}$$

$$x_a^k \in \{0, 1\} \quad \forall a \in \mathcal{A}^{\mathcal{B}}, \quad \forall k \in \mathcal{K}$$

Solution Approach - Step 2

We can decompose the primal recourse problem $Q^P(x)$ by scenario for each carrier:

$$\min \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^{\mathcal{B}}} d_a^k x_a^k + \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}^k} p^s Q_{ks}^P(x)$$

$$\sum_{k \in \mathcal{K}} x_a^k = 1 \quad \forall a \in \mathcal{A}^{\mathcal{B}}$$

$$x_a^k \in \{0, 1\} \quad \forall a \in \mathcal{A}^{\mathcal{B}}, \quad \forall k \in \mathcal{K}$$

Solution Approach - Step 3

We can substitute $Q_{ks}^P(x)$ with $\theta_{ks}(x)$:

$$\min \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^B} d_a^k x_a^k + \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}^k} p^s \theta_{ks}$$

$$\sum_{k \in \mathcal{K}} x_a^k = 1 \quad \forall a \in \mathcal{A}^B$$

$$\theta_{ks} = Q_{ks}^P(x) \quad \forall k \in \mathcal{K}, \forall s \in \mathcal{S}^k$$

$$x_a^k \in \{0, 1\} \quad \forall a \in \mathcal{A}^B, \quad \forall k \in \mathcal{K}$$

Solution Approach - Step 4

We can re-write the formulation in terms of the **dual** recourse problem, $Q_{ks}^D(x)$

$$\min \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^B} d_a^k x_a^k + \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}^k} p^s \theta_{ks}$$

$$\sum_{k \in \mathcal{K}} x_a^k = 1 \quad \forall a \in \mathcal{A}^B$$

$$\theta_{ks} = Q_{ks}^D(x) \quad \forall k \in \mathcal{K}, \forall s \in \mathcal{S}^k$$

$$x_a^k \in \{0, 1\} \quad \forall a \in \mathcal{A}^B, \quad \forall k \in \mathcal{K}$$

Where:

$$Q_{ks}^D(x) = \max \mathbf{B}(x)\pi + \mathbf{u}\alpha$$

$$\text{s.t.} \quad [\pi, \alpha] \in \mathbb{P}$$

$$\mathbb{P} = \left\{ \begin{array}{ll} \pi_{O(a)} - \pi_{D(a)} \leq e_a & \forall a \in \mathcal{A}^E \\ \pi_{O(a)} - \pi_{D(a)} + \alpha_a \leq c_a & \forall a \in \mathcal{A}^C \\ \pi_i \in \mathbb{R} \forall i \in \mathcal{N}, \alpha_a \leq 0 & \forall a \in \mathcal{A}^C \end{array} \right\}$$

Solution Approach - Step 5

We can also write the dual recourse problem $Q_{ks}^D(x)$ over the set of the extreme points ($\mathcal{P}$) of the polyhedron $\mathbb{P}$:

$$\min \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^B} d_a^k x_a^k + \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}^k} p^s \theta_{ks}$$

$$\sum_{k \in \mathcal{K}} x_a^k = 1 \quad \forall a \in \mathcal{A}^B$$

$$\theta_{ks} = Q_{ks}^D(x) \quad \forall k \in \mathcal{K}, \forall s \in \mathcal{S}^k$$

$$x_a^k \in \{0, 1\} \quad \forall a \in \mathcal{A}^B, \quad \forall k \in \mathcal{K}$$

Where:

$$\begin{aligned} Q_{ks}^D(x) = \max \mathbf{B}(x)\pi + \mathbf{u}\alpha \\ \text{s.t.} \quad [\pi, \alpha] \in \mathbb{P} \end{aligned} \quad \Rightarrow \quad \begin{aligned} Q_{ks}^D(x) = \max \mathbf{B}(x)\pi^v + \mathbf{u}\alpha^v \\ \text{s.t.} \quad [\pi^v, \alpha^v] \in \mathcal{P} \end{aligned}$$

Solution Approach - Step 6

Therefore:

$$\min \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^B} d_a^k x_a^k + \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}^k} p^s \theta_{ks}$$

$$\sum_{k \in \mathcal{K}} x_a^k = 1 \quad \forall a \in \mathcal{A}^B$$

$$\theta_{ks} = Q_{ks}^D(x) \quad \forall k \in \mathcal{K}, \forall s \in \mathcal{S}^k$$

$$x_a^k \in \{0, 1\} \quad \forall a \in \mathcal{A}^B, \quad \forall k \in \mathcal{K}$$

Where:

$$Q_{ks}^D(x) = \max_{\pi^v, \alpha^v} \mathbf{B}(x) \pi^v + \mathbf{u} \alpha^v \quad \Rightarrow \quad \theta_{ks} \geq \mathbf{B}(x) \pi^v + \mathbf{u} \alpha^v$$

s.t. $[\pi^v, \alpha^v] \in \mathcal{P} \quad \forall v \in \mathcal{P}, \forall k \in \mathcal{K}, \forall s \in \mathcal{S}^k$

Solution Approach - Step 7

Finally we get:

$$\min \sum_{k \in \mathcal{K}} \sum_{a \in \mathcal{A}^B} d_a^k x_a^k + \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}^k} p^s \theta_{ks}$$

$$\sum_{k \in \mathcal{K}} x_a^k = 1 \quad \forall a \in \mathcal{A}^B$$

$$\theta_{ks} \geq \mathbf{B}(x) \pi^v + \mathbf{u} \alpha^v \quad \forall v \in \mathcal{P}, \forall k \in \mathcal{K}, \forall s \in \mathcal{S}^k$$

$$x_a^k \in \{0, 1\} \quad \forall a \in \mathcal{A}^B, \quad \forall k \in \mathcal{K}$$

- Note that this is just a simple substitution
- But it relies on full enumeration of the extreme points of $\mathbb{P}$
- **Next step:** How can we avoid enumeration of all the extreme points?
- We can use cut generation

Solution Algorithm

- **step (0)** initialize the main problem with $\theta_{ks} \geq 0 \quad \forall k \in \mathcal{K}, \forall s \in \mathcal{S}^k$ and solve it to get x .
- **step (1)** using vector x , solve the dual recourse problem $Q_{ks}^D(x) \quad \forall k \in \mathcal{K}, \forall s \in \mathcal{S}^k$.

- **step (2)** use the extreme point (α^v, π^v) found in step 2 to generate a new cut:

$$\theta_{ks} \geq \mathbf{B}(x)\pi^v + \mathbf{u}\alpha^v \quad \forall k \in \mathcal{K}, \forall s \in \mathcal{S}^k$$

and add it to the main problem.

- **step (3)** re-solve the main problem. If the new solution x is the same as the one found in the previous iteration, stop. Otherwise, go to step 1.

Discussion

- At each iteration, we generate a cut for each carrier under each scenario
- This approach leads to stronger cuts
- If a carrier's load assignment does not change from one iteration to another, then we can skip the recourse problems corresponding to that specific carrier
- Each recourse problem is a min-cost flow problem
- Overall, based on preliminary computations, the total number of iterations is relatively small

Summary of Contributions

- Combinatorial truckload procurement auctions are difficult to solve, even in a deterministic setting
- We consider the increased challenge of addressing uncertainty
- We present an idea to use the implicit bidding approach introduced by Beil, Cohn, and Sinha to capture exhaustive bidding
- We implement in a stochastic programming framework
- We exploit the problem structure to model the recourse problem as series of small min-cost flow problems, in order to overcome the difficulties of solving a two-stage integer recourse problems

Future Work

- Implementing the solution approach to evaluate performance of realistic-sized problem instances
- Considering operational side constraints
- Expanding this idea of stochastic auctions to other applications such as energy or telecommunication auctions

Questions

