Mechanism Design Approach in Supply Chain Planning

Corvinus Game Theory Seminar

Péter Egri

MTA SZTAKI, Laboratory on Engineering and Management Intelligence

December 7, 2009 

Outline

Outline

EMI laboratory

- Main research directions
  - Intelligent manufacturing processes and systems
  - Production planning, scheduling and control
  - Management of complexity, changes and disturbances in production
  - Production networks, supply chain information systems
- Applied methods
  - Operations research
  - Constraint programming
  - Discrete event simulation
  - Machine learning
  - Neural networks
- People
  - Head of dept.: Prof. László Monostori, Dr. József Váncza
  - Ca. 20 members
- Fraunhofer Project Center

Outline

Introduction

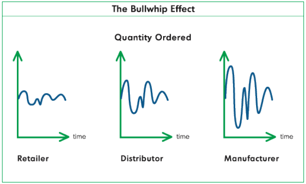
Background

Mechanism design

Application example in supply chains

Situation in manufacturing supply chains

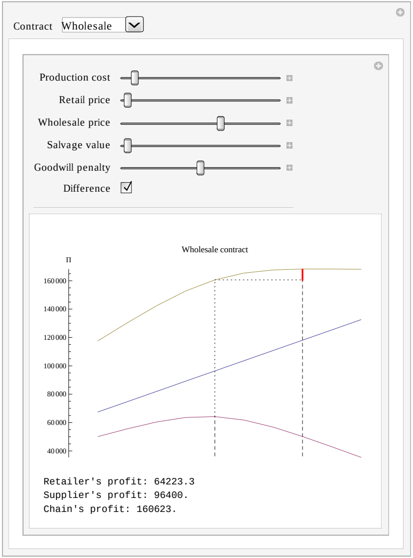
- Bullwhip effect
- Double marginalisation: prisoners' dilemma in supply chains[1]
- Vertical integration
  - Strategic partnerships (e.g., Vendor Managed Inventory, Just-In-Sequence supply, etc.)
  - Electronic Data Interchange (EDI)
  - Collaborative planning, forecasting and replenishment



- Buzzwords...
  - Virtual enterprises
  - Cost and profit sharing
- ...and the reality
  - The dominant position controls the "cooperation"

[1] Jiang, H., 2008, Game Analysis of the Cooperation between Suppliers and Core Enterprises in Supply Chains and Cooperation Mechanism Design.

Example of suboptimal ordering



- Supplier produces to order
- Retailer builds stock based on forecasts
- Non zero-sum "game" with suboptimal order quantity

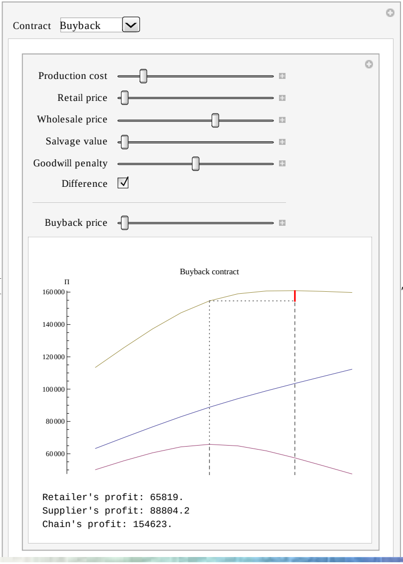
Coordination with contracts

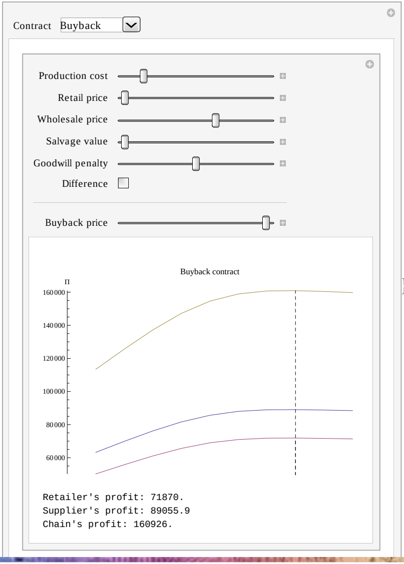
- Model characteristics
  - Parameters of the underlying planning problem?
  - Information structure: what is common knowledge?
    Stackelberg or principal – agent model?
  - Decision structure: Who is the leader?
    Signaling or screening?

- Contract types[2]
  - Two-part tariff
  - Sales rebate
  - Quantity discount
  - Buyback/return
  - Quantity flexibility
  - Revenue sharing
  - Options
  - VMI contract[3]

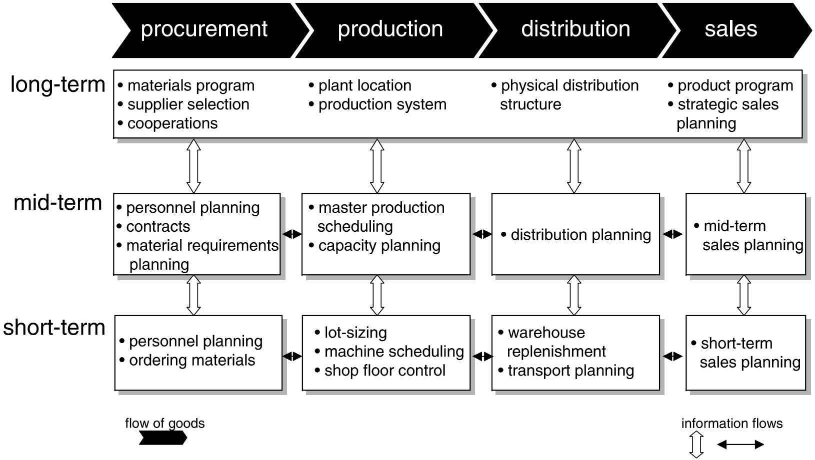
[2] Cachon, G. P., 2003, Supply Chain Coordination with Contracts.
[3] Egri, P., 2008, Coordination in Production Networks.

Example of coordination





More complex supply chain planning problems[4]



[4] Fleischmann, B., Meyr, H., Wagner M., 2002, Advanced Planning.

Mechanism design approach for distributed production

Networked production	Algorithmic mechanism design
enterprises/factories	rational agents
profit max./cost min.	utility maximization
private information	type
global SC optimum	social choice function
contracts	game rules
production planning, etc.	computationally complex problems
electronic data interchange	communication complexity
collaborative planning	distributed mechanisms
(automated) negotiation	iterative mechanisms

Outline

Mechanism design model

n	number of agents (players)
$t_i \in T_i$	type of agent i (private information)
$t \in T = T_1 \times \cdots \times T_n$	type profile
$s_i : T_i \to \Sigma_i$	strategy of agent i
$s(t) \in \Sigma = \Sigma_1 \times \cdots \times \Sigma_n$	strategy profile
$g : \Sigma \to O$	outcome rule
$v_i : O \times T_i \to \mathbb{R}$	valuation function of agent i
$p_i : \Sigma \to \mathbb{R}$	payment for agent i
$u_i = v_i + p_i$	quasi-linear utility of agent i
$\mathcal{M} = (\Sigma, g, p)$	mechanism
$f : T \to O$	social choice function

Notation: $t_{-i} = (t_1, \ldots, t_{i-1}, t_{i+1}, \ldots, t_n) \in T_{-i}$, similarly s_{-i}, Σ_{-i}

sometimes only s instead of $s(t)$

Implementing the social choice function

Definition. s_i is a dominant strategy, if it maximises the utility of agent i, independently of the other's types and strategies:
$\forall t'_{-i}, s' : u_i(s_i(t_i), s'_{-i}(t'_{-i}), t_i) \geq u_i(s'_i(t_i), s'_{-i}(t'_{-i}), t_i)$.

Definition. s is dominant equilibrium, if $\forall i : s_i$ is dominant strategy.

Definition. The $\mathcal{M} = (\Sigma, g, p)$ mechanism implements the f social choice function, if for each t type profile, for each s (dominant) equilibrium $g(s(t)) = f(t)$.

Direct mechanisms and the revelation principle

- Direct mechanism: agents should declare their types (not necessarily truthfully), i.e., $\Sigma = T$.
- Strategy-proof mechanism: a direct mechanism, where truthfulness ($s(t) = t$) is dominant equilibrium.
- Revelation principle: for any arbitrary (not necessarily direct) mechanism there exist an equivalent strategy-proof mechanism.

Nice-to-have properties

- Efficiency: it maximises the total valuation of the agents:
  $\forall o : \sum_{i=1}^{n} v_i(g(s), t_i) \geq \sum_{i=1}^{n} v_i(o, t_i)$
- Budget balance: the payment is between the agents:
  $\sum_{i=1}^{n} p_i(s) = 0$
- Individual rationality: each agent realises non-negative utility:
  $\forall i : u_i(s, t_i) \geq 0$

Nice-to-have properties

- Efficiency: it maximises the total valuation of the agents:
  $\forall o : \sum_{i=1}^{n} v_i(g(s), t_i) \geq \sum_{i=1}^{n} v_i(o, t_i)$
- Budget balance: the payment is between the agents:
  $\sum_{i=1}^{n} p_i(s) = 0$
- Individual rationality: each agent realises non-negative utility:
  $\forall i : u_i(s, t_i) \geq 0$

These properties are conflicting!

Some mechanisms, extensions and possible applications

1. Vickrey – Clarke – Groves (VCG) mechanism
   - Supply chain procurement auctions[5]
2. Mechanisms with verification
   - Task allocation problem
   - Modelling commitments
3. Stochastic mechanism design problems
   - Demand forecasts, supply uncertainties, manufacturing disturbances
4. Distributed mechanism design
   - Distributed implementation
   - Automated negotiation[6,7]

[5] Lok, R. B., 2007, Auction Mechanism in Supply Chain Optimisation.

[6] Dudek, G., 2009, Collaborative Planning in Supply Chains.

[7] Albrecht, M., 2010, Supply Chain Coordination Mechanisms.

1. Vickrey – Clarke – Groves (VCG) mechanism

Groves' scheme (strategy-proof and efficient mechanism):

- $\Sigma = T$,
- $g(s) \in \operatorname{argmax}_o \sum_{i=1}^{n} v_i(o, s_i)$,
- $p_i(s) = \sum_{j \neq i} v_j(g(s), s_j) + h_i(s_{-i})$, where h_i is arbitrary.

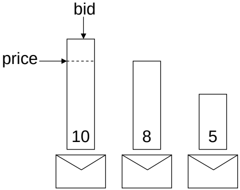
Special case (Clarke): $h_i(s_{-i}) = -\max_o \sum_{j \neq i} v_j(o, s_j)$, i.e.,

$$p_i(s) = \underbrace{\sum_{j \neq i} v_j(g(s), s_j)}_{\text{other's value}} - \underbrace{\max_o \sum_{j \neq i} v_j(o, s_j)}_{\text{other's possible value}} \leq 0,$$

i.e., it is weakly budget balanced: $\sum_{i=1}^{n} p_i \leq 0$.

If furthermore $\forall i, t_i, o : v_i(o, t_i) \geq 0$ then it is also individually rational.

Example: Vickrey auction (second price sealed bid)



- Type is the subjective value of the item: $T_i = \mathbb{R}$
- Bid: $\Sigma_i = \mathbb{R}$
- The highest bid wins: $g(s) \in \mathrm{argmax}_i \, s_i$ $(O = \{1, \ldots, n\})$
- The winner pays the second highest bid
- Winner's utility: $v_{g(s)} = t_{g(s)}$, $p_{g(s)} = -\max_{i \neq g(s)} s_i$
- Other's utility: $v_i = p_i = 0$ $(i \neq g(s))$

Strategy-proof, efficient, weakly budget balanced and individual rational.

2. Mechanisms with verification[8]

Two phases: declaration and execution

Model:

- Strategy: $\Sigma = D \times E$
- Mechanism's decision: $g : D \rightarrow K$
- Execution: $e_i(k, t_i)$
- Output (part of the problem specification): $o(k, e)$
- Utility: $u_i = v_i(o, t_i) + p_i(d, e)$

There are some problems, which can be solved with a mechanism with verification, but not with any traditional mechanism.

[8] Nisan, N., Ronen, A., 2001, Algorithmic Mechanism Design.

3. Stochastic mechanism design[9]

- Two-stage stochastic optimisation problem
- Two phases: forecasting and realisation
- Both the forecast (probability distribution) and the realised type is private information
- Output can be modified in the second phase
- Realizing and modifying the output is costly for the mechanism

- Extensions: dynamic mechanisms (based on Markov Decision Processes)

[9] Ieong, S., So, A. M.-C., Sundararajan, M., 2007, Stochastic Mechanism Design (Extended Abstract).

4. Distributed mechanism design[10]

The strategy consists of tree parts: $s_i = (e_i, b_i, w_i)$

- Information-revelation strategy: e_i
- Computational strategy: b_i
- Message-passing strategy: w_i

Remark. The information revelation can be incremental, in this case it is called iterative mechanism.

Definition. The $\mathcal{M} = (\Sigma, g, p, s^m)$ is a distributed mechanism, where $s^m : T \to \Sigma$ is the suggested strategy.

[10] Shneidman, J., Parkes, D. C., 2004, Specification Faithfulness in Networks with Rational Nodes.

Distributed mechanism design (cont'd)

Definition. $s : T \to \Sigma$ is *ex post* Nash equilibrium, if no player would like to unilaterally deviate from its strategy, independently from the actual type profile, i.e., $\forall i, t, s'_i : u_i(s(t), t_i) \geq u_i(s'_i(t_i), s_{-i}(t_{-i}), t_i)$.

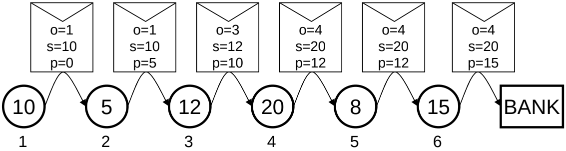
Definition. $\mathcal{M} = (\Sigma, g, p, s^m)$ is a faithful mechanism, if s^m *ex post* Nash equilibrium.

Technics: redundance, catch-and-punish, partitioning, cryptographic protocols,. . .

Remark. Although the decision is made in a distributed manner, a bank is needed for supervising the execution!

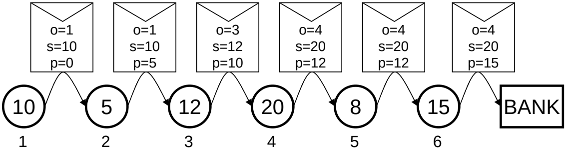
Example: Vickrey auction

Not faithful implementation:

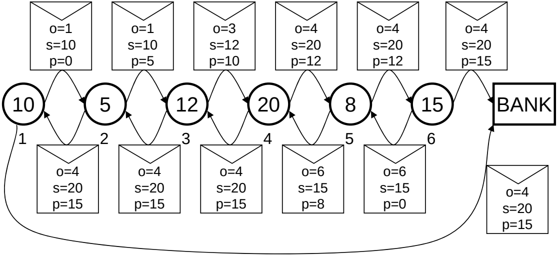


Example: Vickrey auction

Not faithful implementation:

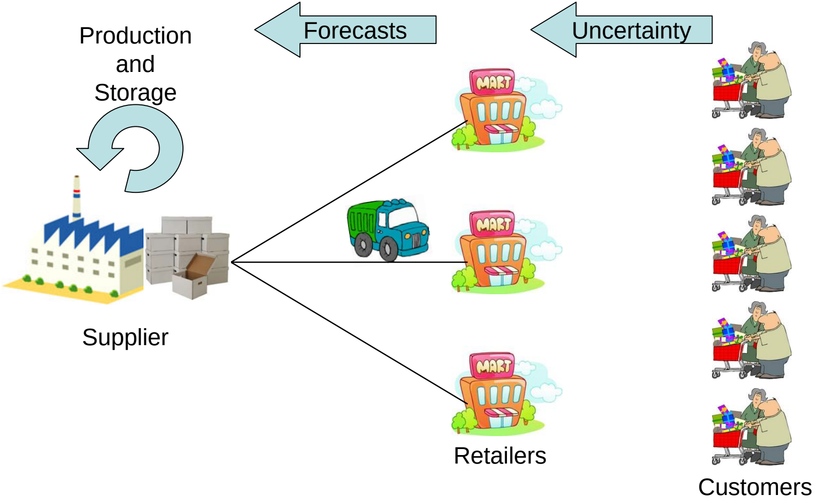


A faithful implementation:



Outline

Procurement with Vendor Managed Inventory



Mechanism design model

Assumptions:

- The supplier is the mechanism, the retailers are the agents
- t_i is the demand forecast
- s_i is the shared forecast (in the 1^{st} phase)
- $g : \Sigma \to O$ determines the production plan of the supplier
- The valuations of the retailers are independent from the supplier's production
- d_i is the realised demand at retailer i, common knowledge in the 2^{nd} phase

Consequences:

- Efficiency and budget balance are not required
- The goal is the truthful forecast sharing (strategy-proofness)
- Similarities with mechanisms with verification and stochastic mechanism design

Specific cases of forecasts

Forecast is given by...

1. ...the **expected value** of the demand
   - Verifying forecast quality is easy, e.g., $|s_i - d_i|$ or $(s_i - d_i)^2$
   - Proposed payment: $p_i(s_i) = \alpha|s_i - d_i| + h_i(d_i)$
     where $\alpha < 0$ is an arbitrary constant, h_i is an arbitrary function
2. ...the **expected value** and the **standard deviation** of the demand
   - How to verify quality of standard deviation based on only one event? (next slide)
3. ...the expected values for **multiple periods**, updated in **rolling horizon** manner[11]

[11] Egri, P., 2008, Coordination in Production Networks.

Forecasts with expected value (m_i) and standard deviation (σ_i)

Theorem

Let $s_i = (\hat{m}_i, \hat{\sigma}_i)$ denote the shared forecasts, then

$$p_i(\hat{m}_i, \hat{\sigma}_i) = \alpha \left(\frac{(\hat{m}_i - d_i)^2}{\hat{\sigma}_i} + \hat{\sigma}_i \right) + h_i(d_i),$$

where $\alpha < 0$ is an arbitrary constant and h_i is an arbitrary function results in a strategy-proof mechanism.

Sketch of proof.

Let $t_i = (m_i, \sigma_i)$ denote the real forecasts for d_i, then

$$\mathbb{E}_{m_i,\sigma_i}[p_i(\hat{m}_i, \hat{\sigma}_i)] = \alpha \left(\frac{(\hat{m}_i)^2 + \sigma_i^2 + m_i^2 - 2\hat{m}_i m_i}{\hat{\sigma}_i} + \hat{\sigma}_i \right) + \mathbb{E}_{m_i,\sigma_i}[h_i(d_i)] \qquad (1)$$

is concave in $\hat{m}_i$, and

$$\frac{\partial \mathbb{E}_{m_i,\sigma_i}[p_i(\hat{m}_i, \hat{\sigma}_i)]}{\partial \hat{m}_i} = \frac{\alpha}{\hat{\sigma}_i}(2\hat{m}_i - 2m_i)$$

thus Eq. (1) has unique maximum in $\hat{m}_i = m_i$. Then Eq. (1) is concave in $\hat{\sigma}_i$, and

$$\frac{\partial \mathbb{E}_{m_i,\sigma_i}[p_i(m_i, \hat{\sigma}_i)]}{\partial \hat{\sigma}_i} = \alpha(1 - \frac{\sigma_i^2}{\hat{\sigma}_i^2}).$$

VMI: conclusions and open questions

- General idea:

$$p_i(s_i) = \underbrace{\alpha(s_i \ominus d_i)}_{\text{compensation for risks}} + \underbrace{h_i(d_i)}_{\text{normal payment}}$$

- Individual rationality?
- Allocation of profits?
- Application of stochastic mechanism design?

Summary

- Distributed planning in supply chains
  - Autonomous enterprises
  - Private information
  - Computationally complex problems
- Mechanism design
  - Classic mechanism design
  - Algorithmic mechanism design with new models
- Supply chain coordination mechanisms
  - Contracts
  - Auctions
  - Negotiation protocols

Thank you for your attention!