

Multi-Echelon Inventory Control for Perishable Products

August 13, 2018

Model Setup

- Total T time periods.
- Network with incidence matrix A .
- Single product, life time is m .
- Lead times are fixed (l_{ij}).
- There are yield and storage capacities.
- Sequence of events:
 1. Demand realization
 2. Allocation of inventories to demand
 3. Disposal and expiration
 4. Previous orders arrive
 5. Inventory state update
 6. Place order
- Decisions include inventory attribution, disposal, and ordering.
 - Why not FIFO? Maximizing freshness might require non-FIFO.
 - Why disposal? Disposal makes sense when we have storage capacity.

Single Node Single Source

Assumptions

- Lead time l , same freshness (m remaining) when arriving

Minimizing wastage:

$$J^t(\mathbf{x}^t) = \min_{y \geq 0} \max_{d^t \in U^t} \min_{(\mathbf{s}^t, \mathbf{w}^t) \in V^t(d, y, \mathbf{x})} \mathbf{1}' \mathbf{w}^t + J^{t+1}(f(\mathbf{x}^t, \mathbf{w}^t, \mathbf{s}^t, y))$$

where the feasible region $V^t(d, y, \mathbf{x})$ is

$$\begin{aligned} \sum_{k \in [m]} s_k^t &\geq d^t & \forall d^t \in U^t & \text{(demand satisfaction)} \\ s_k^t + w_k^t &\leq x_k^t & \forall k \in [m] & \text{(inventory allocation)} \\ w_1^t &= x_1^t & & \text{(expiration)} \\ y^t &\leq \bar{y}^t & & \text{(yield capacity)} \\ \sum_{k \in [m]} x_k^t &\leq \bar{x}^t & & \text{(storage capacity)} \end{aligned}$$

Variable	Description
x_k	life- k inventory in the beginning of period, $k = 1, \dots, m + l$
y	order quantity in the beginning of the period
w_k	wastage of life- k inventory after demand realization in t
s_k	demand satisfaction with life- k after demand realization
d	demand
\bar{x}	storage capacity at retailer
\bar{y}	ordering capacity per period

And the linear update rule is

$$f_k(\mathbf{x}, \mathbf{w}, \mathbf{s}, y) := x_{k+1} - w_k - s_k, \quad k = 1, \dots, m + l - 1,$$

$$f_m(\mathbf{x}, \mathbf{w}, \mathbf{s}, y) = y.$$

Last time period:

$$J^T(\mathbf{x}^T) = \min_{y \geq 0} \max_{d^T \in U^T} \min_{(\mathbf{s}^T, \mathbf{w}^T) \in V^T(d, y, \mathbf{x})} \mathbf{1}' \mathbf{w}^T,$$

where $V^T(d, y, \mathbf{x})$ is described by:

$$\begin{aligned} \sum_{k \in [m]} s_k^T &\geq d^T && \forall d^T \in U^T && \text{(demand satisfaction)} \\ s_k^T + w_k^T &\leq x_k^T && \forall k \in [m] && \text{(inventory allocation)} \\ w_1^T &= x_1^T && && \text{(expiration)} \\ y^T &\leq \bar{y}^T && && \text{(yield capacity)} \\ \sum_{k \in [m]} x_k^T &\leq \bar{x}^T && && \text{(storage capacity)} \end{aligned}$$

What value of d^T does it take from U^T ? If the problem is infeasible for some $d^T \in U^T$, then d^T will always take such value (e.g., the largest) to take the cost to infinity. But assuming we have come to a point where feasibility is satisfied, then d^T will actually take the smallest value.

Single Node, Multisource

Bipartite

Variables & Parameters

Variable	Description
x_{jk}^{tl}	life- k in-transit inventory l periods away from j in the beginning of period t
x_{jk}^t	life- k on-hand inventory at j in the beginning of period t
y_{ijk}^t	order quantity of life- k product from i to j by the end of period t
w_{jk}^t	wastage of life- k inventory after demand realization in t
s_{jk}^t	demand satisfaction with life- k after demand realization in t
d_j^t	demand in node j in period t

Formulation

Minimizing wastage:

$$J^t(\mathbf{x}^t) = \min_{(\mathbf{s}^t, \mathbf{w}^t, \mathbf{y}^t) \in V^t} \mathbb{E} [\mathbf{1}' \mathbf{w}^t + J^{t+1}(f(\mathbf{x}^t, \mathbf{w}^t, \mathbf{s}^t, \mathbf{y}^t))]$$

Maximizing freshness:

$$J^t(\mathbf{x}^t) = \max_{(\mathbf{s}^t, \mathbf{w}^t, \mathbf{y}^t) \in V^t} \mathbb{E} \left[\sum_{j \in J} \sum_{k \in [m]} k s_{jk}^t / (m d_j^t) + J^{t+1}(f(\mathbf{x}^t, \mathbf{w}^t, \mathbf{s}^t, \mathbf{y}^t)) \right]$$

where the feasible set for controls in period t is:

$$\begin{aligned} \sum_{k \in [m]} s_{jk}^t &\geq \delta d_j^t & \forall j \in J & \quad (\text{demand satisfaction}) \\ s_{jk}^t + w_{jk}^t &\leq x_{jk}^t & \forall j \in J, k \in [m] & \quad (\text{inventory allocation}) \\ w_{j0}^t &= x_{j0}^t & \forall j \in J & \quad (\text{expiration}) \end{aligned}$$

The linear update rule $f(\cdot)$ is a vector-valued function, dependent on the lead times and orders from upstream nodes:

$$\begin{aligned} x_{jk}^t &= x_{j(k+1)}^{(t-1)1} + x_{j(k+1)}^{t-1} - s_{j(k+1)}^{t-1} - w_{j(k+1)}^{t-1}, \\ x_{jk}^{tl} &= x_{j(k+1)}^{(t-1)(l+1)} + \sum_{i: l_{ij}=l} y_{ijk}^t \end{aligned}$$

Notes

- Decomposable across j if relax yield capacity constraint.
- Reduce the state space and decision space if we are minimizing expiration instead of maximizing freshness: do not have to keep track of the shelf life of inventory arriving from supplier.
- Would maximizing freshness lead to ordering a lot and just serving demands with freshest inventory?

Single Node Uncapacitated Policy

Chen, Pang, and Pan (2014) setup:

- lead time is k
- product life is $l > k$
- FIFO policy
- Ordering cost: c /unit, inventory cost h^+ and h^- , disposal cost θ
- Demand $d_t := D(p) + \epsilon_t$, where
 - $D(p)$ is strictly decreasing
 - ϵ_i is i.i.d., zero mean, with bounded support $[A, B]$
 - Price is bounded $[p, \bar{p}]$
- Decisions: ordering, pricing, disposal.

Simplifications

- Assume no storage capacity, then with low/zero inventory holding cost \rightarrow FIFO.
- No inventory holding cost \rightarrow no disposal.

Meetings and Planning

June 27

1. Include disposal and issuing decisions to keep the model open
2. If supplier capacitated by residual life, then need to keep supplier inventory vector and therefore do inventory update (add a constraint)
3. Would the current case lead to trivial solution that uses demand upperbound
4. Solve the formulation for one node for now, and check against the literature (Chen et al) to see if solution makes sense.
5. Whether expected value or robust problem easier? Explore policy structure or solve numerically?
6. Adaptive/adjustable robust

To address these questions, do the following steps by Friday:

- Formulate the one-node case (with and without capacity)
- Comparison with existing models
- Solution methods, difficulties

June 29 - Project Plan

Model Feature	Model 1 (July 6)	(July 13)	Model 2 (Aug 10)
Network	Single node multi-source	–	Multi-echelon
Objective	$\max \mathbb{E}[\text{freshness} - \text{waste} - \text{demand loss}]$	Sensitivity	–
Capacity	Exogenous capacity	–	Outflow capacitated by inv
Inventory Issuing	FIFO	–	(Active issuing)
Disposal Policy	Expiration only	–	(Active disposal)
Coordination	Centralized decision	–	(Distributed vs. Centralized)
Solution Approach	Heuristic	Implementation	(Bounds and Theory)

Model 1 deviates from the literature mostly on the multi-sourcing aspect. This introduces modeling and computational challenge since we need to introduce an additional dimension when tracking inventory. Model 2 expands to multi-echelon network. Tasks in brackets have lower priority for now.

Model 1: For each period, we use x_{kl} to denote the amount of inventory with k periods left in shelf life, which we have held for l periods. For $l < 0$, the inventory is arriving in $|l|$ periods. Thus when we order from supplier i with lead time L_i , we update the vector $x_{1L_i}, \dots, x_{mL_i}$. For $l > 0$, we are keeping track of how long we have held the inventory in our current location, which allows us to calculate freshness (both for inventory freshness and demand satisfaction freshness).

Model 2: When extending to multi-echelon, we update the supply capacity by the upstream inventory levels, which are subject to state transition.

Comments from the June 29 meeting:

- For model 1, assume single bucket of age and lead time (different each source of course).
- For implementation: 3 tiers including farms, suppliers, and retailers.
- Include a service level constraint, and then penalize the loss demand beyond that.
- System performance comparison (on a simulator)

- Base stock policy, order from closest supplier/farm. [Key]
 - Single-echelon-optimal-policy at last tier (tier n), and base stock everywhere else.
 - Single-echelon-optimal-policy everywhere, empirical demand signal at i from 1 tier down $\mathbb{P}[D_{i+1}]$.
 - Single-echelon-optimal-policy everywhere, demand signal visible $\mathbb{P}[D_n]$, and know $\mathbb{P}[D_i|D_n]$. [blockchain visibility]
 - Multi-echelon centralized decision [blockchain coordination]
- Build the simulator.

Aug 1 - Value of Visibility.

1. Issuing policy: we find that in the farm-supplier-retailer chain, if farm issues ages m and $m-1$ (50/50), vs. is better than issuing age $m-1$ and $m-2$ (50/50). This is a very specific made-up example (not sure what decision mechanism will lead to these numbers), but this shows the importance of (downstream lead time aware) issuing policy.
2. Value of centralization must be strictly better than the value of visibility (with decentralized operations). So if we can quantify the value of centralization, we can upper bound the value of decentralized and visible operations. But the centralized case should be easy to implement for at least the base stock policy for a chain – just assume the whole chain as a single echelon. To achieve this centralization, the information to be shared is just the lead times at every echelon, and the total inventory at each node. This looks trivial. Am I missing something?
3. To extend the intuition from chain to network, look at Cachon and Fisher (2000) since they considered a hub and spoke system.
4. Literature review: Cachon on supply chain contract. Other papers on (two-echelon) supply chain information sharing.

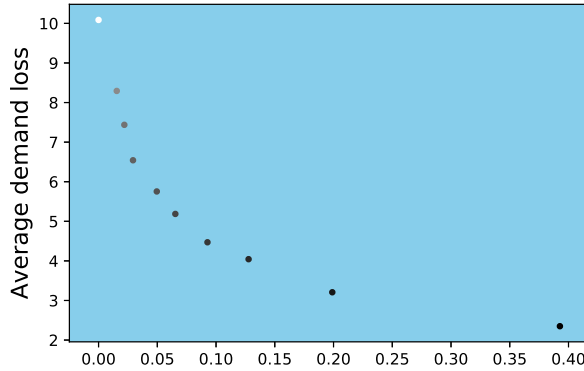


Figure 1: Decentralized base stock policies.

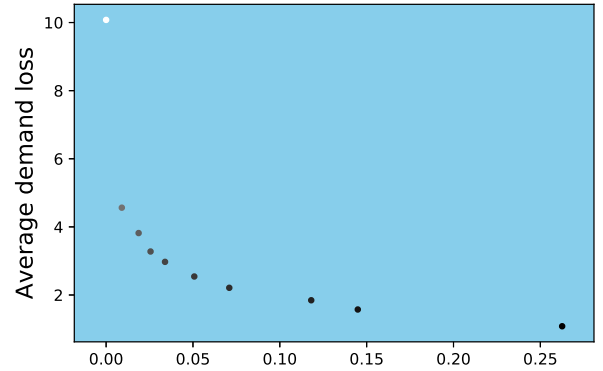


Figure 2: Centralized base stock policies.

Aug 6 - Blockchain governance.

Goal for the next three weeks: literature review of supply chain contract, blockchain, and blockchain application in the contracts.

At least two ways that blockchain contributes. Both ways are basically having the blockchain act as a game designer.

1. Blockchain can limit the space of actions (my thought).
2. Blockchain can change the types of contract from bilateral to multi-lateral (Ramesh).

Aug 7 - Supply chain contracts day 1: newsvendor model.

- Negotiation and blockchain: supply chain contract literature does not focus on the negotiation process. But in practice, this is a costly multi-period problem. Can blockchain provide anything here?

Aug 8 - Incentive for data sharing.

- With blockchain, incentive is not about whether truthfulness of data, but about the revelation of data.
- Valuation of data: Munther's paper and Shapley value. For data duplication problems, check the dimensionality of the data.
- Privacy: zero-knowledge proof.

Aug 9 - Vishal Gaur talk: the case of blockchain in supply chain management.

- Three flows: physical, financial, and document. Physical flow not as well tracked as financial flow. The flows are usually disconnected. Do audits afterwards, but is post factor and expensive.
- Difference between scm and bitcoin: (1) anonymity not good for sc transactions. (2) Majority based distributed consensus not necessary for sc. Byzantine. (3) Both financial and non-financial data should be posted for sc. (4) Smart contracts, IoT, and distributed apps become far more useful in sc. (5) digitization of physical assets such inv and capacities. (6) Blockchain not designed for fast storage and retrieval, not good for certain type of actions (slow is related to the majority based distributed consensus).

I feel that the key is data privacy in sharing onto a distributed ledger. Because like we thought about at the lunch meeting with Ashish and Pavithra, the key incentive question in these applications is not truthfully reporting data, but revealing certain data at all. Because untruthful reporting is now a documented fraud in blockchain. So the incentive problem changed. Furthermore, I think that if we focus on the soft (economic) incentive of sharing data at all, we have to work with specific problem instances, because this thing is highly sensitive to problem setup and blockchain implementation (?). Now think about