Multi-Echelon Inventory Optimization: An Overview

LARRY SNYDER
DEPT. OF INDUSTRIAL AND SYSTEMS ENGINEERING
CENTER FOR VALUE CHAIN RESEARCH
LEHIGH UNIVERSITY

EW0 SEMINAR SERIES – NOVEMBER 13, 2008
Outline

- Introduction
- Single-stage models (building blocks)
- Multi-echelon models
  - Network Topology
  - Deterministic Models
  - Stochastic Models
- Decentralized systems
Introduction
Factors Influencing Inventory Decisions

- **Why hold inventory?**
  - Lead times
  - Economies of scale / fixed costs / quantity discounts
  - Service levels
  - Concerns about future availability
  - Sales / promotions

- **Why avoid inventory?**
  - Cost of capital
  - Shelf space
  - Perishability
  - Risk of theft / fire / etc.
Classifying Inventory Models

- Deterministic vs. stochastic
- Single- vs. multi-echelon
- Periodic vs. continuous review
- Discrete vs. continuous demand
- Backorders vs. lost sales
- Global vs. local control
- Centralized vs. decentralized optimization
- Fixed cost vs. no fixed cost
- Lead time vs. no lead time
Costs in Inventory Models

- Holding cost $h$ ($ / item / unit time)
- Stockout penalty $p$ ($ / item / unit time$)
- Fixed cost $k$ ($ / order$)
- Purchase cost $c$ ($ / item$)
  - Often ignored in optimization models
A Brief History of Inventory Theory

- Harris (1913): EOQ model
- ??? (19??): newsvendor model
- Wagner and Whitin (1958): time-varying deterministic demands
- Clark and Scarf (1960): serial stochastic systems
- Roundy (1985): serial deterministic systems w/ fixed costs, power-of-2 policies
- Graves and Willems (2000): guaranteed-service models
Single-Stage Models

(BUILDING BLOCKS)
The EOQ Model

- Continuous, deterministic demand at rate $\lambda$ per year
- Fixed cost $k$ per order
- Holding cost $h$ per item per year
- Stockouts not allowed
The EOQ Model: Optimization

- **Average annual cost:**
  \[ c(Q) = \frac{k\lambda}{Q} + \frac{hQ}{2} \]

- **First-order condition:**
  \[ c'(Q) = -\frac{k\lambda}{Q^2} + \frac{h}{2} = 0 \]

- **Optimal solution:**
  \[ Q^* = \sqrt{\frac{2k\lambda}{h}} \quad c(Q^*) = \sqrt{2k\lambda h} = hQ^* \]
The Newsvendor Model

- Each day, newsvendor buys newspapers from publisher for $0.25 each
- Sells newspapers for $0.75 each
- Unsold papers are sold back to publisher for $0.10
- Daily demand is stochastic, \( \sim N(50, 10^2) \)
- No inventory carryover [perishable inventory]
- No backorder carryover [lost sales]

**How many newspapers to buy?**
- Probably \( >50 \), but how many?
A More General Formulation

- Periodic, stochastic demand
  - pdf $f$, cdf $F$
  - We’ll assume normal distribution ($\phi$, $\Phi$ = standard normal)

- Inventory carryover allowed [non-perishable] or not
  - Either way, “overage” cost = $h$
  - May include salvage value/cost

- Backorders or lost sales
  - Either way, “underage” cost = $p$
  - May include lost profit, loss of goodwill, admin costs

- Decision variable: base-stock level $y$
  - In each period, order up to $y$
Expected Cost Function

\[ c(y) = h \int_{0}^{y} (y - x) f(x)dx + p \int_{y}^{\infty} (x - y) f(x)dx \]

- Convex \( \Rightarrow \) solve first-order condition (Leibniz’s rule)
- Optimal solution:

\[ y^* = \mu + \sigma \Phi^{-1} \left( \frac{p}{p + h} \right) = \mu + \sigma z_\alpha \]

where \( \alpha = \frac{p}{p + h} \) (the news-vendor ratio)
Interpretation of Optimal Solution

\[ y^* = \mu + \sigma z_\alpha \]

- **No stockouts if demand** \( \leq \mu + \sigma z_\alpha \)
  - Occurs with probability \( \alpha \)
  - \( \alpha = \text{optimal service level} \)

- **If lead time** (\( L \)) > 0:
  \[ y^* = \mu L + \sigma z_\alpha \sqrt{L} \]
Multi-Echelon Models

PART 1:
NETWORK TOPOLOGY
Network Topology

- System is composed of **stages** (nodes, items, sites...)
- Stages are grouped into **echelons**
- Stages can represent:
  - Physical locations
  - Items in BOM
  - Processing activities
Terminology

- Stages to the left are *upstream*
- Those to the right are *downstream*
- Downstream stages face customer demand

- Network topologies, in increasing order of complexity:
Serial System

- Each stage has at most one predecessor and at most one successor
Assembly System

- Each stage has at most one successor
Distribution System

- Each stage has at most one predecessor
Tree System

- No restrictions on neighbors, but no cycles
General System

- No restrictions on cycles
Multi-Echelon Models

PART 2:
DETERMINISTIC SYSTEMS
(WITH FIXED COSTS)
Assumptions

- Each stage functions like an EOQ system:
  - Continuous, deterministic demand (last stage only)
  - Fixed ordering cost
  - No stockouts allowed

- We’ll consider serial systems only
The Optimization Problem

- Need to choose $Q$ at all stages simultaneously

- Properties of optimal solutions:
  - **Zero-inventory ordering (ZIO):** order only when inventory = 0
  - **Stationary:** same $Q$ for every order
    - (but different for different stages)
  - **Nested:** whenever one stage orders, so does its customer

- Instead of optimizing over $Q$, we optimize over $u$ (reorder interval)
  - $u = Q / \lambda$
NLIP Formulation

\[
\min C(u) = \sum_j \left( \frac{k_j}{u_j} + \frac{h_j \lambda u_j}{2} \right)
\]

s.t. \( u_j = \theta_j u_{j+1} \)
\( u_j \geq 0 \)
\( \theta_j \in \{1, 2, 3, \ldots\} \)

- Non-convex mixed-integer NLP
- Optimal solution \( u^* \) is not known
  - In fact, no guarantee an optimal solution exists, except in limit
- Therefore, get lower bound by solving relaxed problem
- And upper bound by rounding relaxed solution to feasible solution
Relaxed Problem

\[
\begin{align*}
\text{min } & \quad C(u) \\
\text{s.t.} & \quad u_j \geq u_{j+1} \\
& \quad u_j \geq 0
\end{align*}
\]

- Convex NLP
- Could solve using NLP solver
- But there’s a better way...
Solving the Relaxed Problem

- Partition the stages:

- In each partition, require every stage to have the same $u_j = u$
  - Find $u$ by solving EOQ—easy!

- If we use the “correct” partition, we solve the relaxed problem
  - Find correct partition by finding upper concave envelope of set of points in 2D—easy!
Power-of-2 Policies

- Let $\hat{u}$ be a fixed base period
  - e.g., 1 week, 3 days, etc.
- **Power-of-2 policy**: each $u_j$ is an integer-power-of-2 multiple of $\hat{u}$
- To get feasible solution, round solution to relaxed problem to *nearest power-of-2 policy*
- Power-of-2 policies are simple to implement and intuitive
  - (Stage 1 orders every 2 weeks, stage 2 orders every week, etc.)
Worst-Case Error Bound

- Let $u^*$ be the (unknown) optimal policy
- Let $u^+$ be the power-of-2 policy

**Theorem (Roundy 1985):** For any $\hat{u}$,

$$\frac{C(u^+)}{C(u^*)} \leq \frac{3}{2\sqrt{2}} \approx 1.06$$

- If we can choose $\hat{u}$, then the bound reduces to 1.02
Multi-Echelon Models

PART 3:
STOCHASTIC SYSTEMS
(WITHOUT FIXED COSTS)
Assumptions

- Each stage functions like a newsvendor system:
  - Periodic, stochastic demand (last stage only)
  - No fixed ordering cost
  - Inventory carryover and backorders
- Each stage follows base-stock policy

- Lead time ($L$) = deterministic transit time between stages
- Waiting time ($W$) = stochastic time between when stage places an order and when it receives it
  - Includes $L$ plus delay due to stockouts at supplier
Stochastic- vs. Guaranteed-Service Models

- **Two main modeling approaches**
- **Stochastic-service models:**
  - Each stage meets demands from stock whenever possible ($w = l$)
  - Excess demands are backordered and incur $w > l$
- **Guaranteed-service models:**
  - Each stage sets a committed service time (CST) and guarantees that $w = CST$ for every demand
  - Demand is assumed to be bounded
- **Let $\alpha = \text{service level} (\% \text{ with } w \leq CST)**
  - Stochastic service: $CST = 0$, $\alpha < 1$
  - Guaranteed service: $CST > 0$, $\alpha = 1$
Stochastic-Service Models
Serial Systems: The Clark-Scarf Algorithm

- Objective function:
  \[ c(y) = \sum_j [hE[\text{on-hand inventory}] + pE[\text{backorders}]] \]

- \(E[\text{on-hand}]\) and \(E[\text{backorders}]\) at stage \(j\) depend on \(y\) at \(j\) and upstream

- Clark and Scarf (1960) rewrite \(c(y)\) so that system decomposes by stage
  - \(y_j\) can be determined at each stage in sequence
  - Use decisions from downstream stages but ignore upstream ones
  - At each stage, solve 1-variable convex minimization problem
  - (At last stage, it’s a newsvendor problem)

- Easy computationally but cumbersome to implement
- Good heuristics exist: e.g., Shang and Song (1993)
• **Theorem (Rosling 1989):** Every assembly system can be reduced to an equivalent serial system
  - Solve using Clark-Scarf algorithm

• Based on **inventory balance principle**:

  - If inventory of 2 > inventory of 3, the extra is useless
  - Therefore, attempt to keep $I_2 = I_3$ at all times
Distribution Systems

- Inventory balance principle does not apply
- Allocation rule becomes critical factor

- The one-warehouse, multiple retailer (OWMR) system
  - Famous special case
  - Exact algorithm: Axsäter 1993
  - Heuristics:
    - Sherbrooke 1968 (METRIC): approximate waiting time with its mean
    - Graves 1985: 2-moment approximation of backorder levels
    - Gallego, Özer, and Zipkin 2007: newsvendor approximation
    - Rong, Bulut, and Snyder 2008: decompose into serial systems
Extensions

- Fixed ordering costs
- Stochastic lead times
- Limited capacity
- Imperfect quality

Some are hard, some are not
- Tractability of standard problems is somewhat “fragile”
Guaranteed-Service Models
Guaranteed-Service Models: Overview

- Each stage promises to deliver *every* item within a fixed number of periods
  - Called the **committed service time (CST)**
- Requires assumption that demand is *bounded*
  - e.g., $D \leq \mu + \sigma z_\alpha$
  - Equivalently, ignore excess demand when $D$ exceeds bound
- CST assumption allows us to treat waiting time ($W$) as **deterministic**

Net Lead Time

- Each stage has:
  - Processing time $T$
  - CST $S$
- **Net lead time (NLT)** at stage $i = S_{i+1} + T_i - S_i$
  - “bad” LT
  - “good” LT
Suppose $S_i = S_{i+1} + T_i$
- e.g., inbound CST = 4, proc time = 2, outbound CST = 6
- Don’t need to hold any inventory
- Operate entirely as pull (make-to-order, JIT) system

Suppose $S_i = 0$
- Promise immediate order fulfillment
- Make-to-stock system
Net Lead Time vs. Inventory

• In general:

\[ y^* = \mu \times NLT + \sigma z_\alpha \sqrt{NLT} \]

• NLT replaces LT in earlier formula

• Choosing inventory levels ⇔ choosing NLTs, i.e., choosing \( S \) at each stage
Objective:
- Find optimal $S$ values (CSTs)
- To minimize expected holding cost
- Subject to end-customer service requirement

Solution methods:
- **Serial systems:** dynamic programming (Graves 1988)
- **Tree systems:** dynamic programming (Graves and Willems 2000)
- **General systems:** piecewise-linear approximation + CPLEX (Magnanti et al., 2006)
Key Insight

• It is usually optimal for only a few stages to hold inventory
  ○ Other stages operate as pull systems

• **Theorem (Graves 1988):** In a serial system, every stage either:
  ○ holds zero inventory (and quotes maximum CST)
  ○ or quotes CST of zero (and holds maximum inventory)
In this solution, inventory is held of finished product and its raw materials.

A Pure Pull System

- Produce to order
- Long CST to customer
- No inventory held in system
A Pure Push System

- Produce to forecast
- Zero CST to customer
- Hold lots of finished goods inventory
A Hybrid Push-Pull System

- Part of system operated produce-to-stock, part produce-to-order
- Moderate lead time to customer
CST vs. Inventory Cost

[Graph showing committed lead time to customer (days) versus inventory cost ($/year).]

- Push System
- Push-Pull System
- Pull System
Optimization Shifts the Tradeoff Curve
Decentralized Systems
Decentralized Systems

- We have assumed the system is centralized
  - Can optimize at all stages globally
  - One stage may incur higher costs to benefit the system as a whole
- What if each stage acts independently to minimize its own cost / maximize its own profit?
Suboptimality

- Optimizing locally results in suboptimality
- Example: upstream stages want to operate make-to-order
  - Results in too much inventory downstream
- Another example:
  - Wholesaler chooses wholesale price
  - Retailer chooses order quantity
  - Optimizing independently, the two parties will always leave money on the table
Supply Chain Contracts / Coordination

- One solution is for the parties to impose a *contracting mechanism*
  - Splits the costs / profits / risks / rewards
  - Still allows each party to act in its own best interest
  - If structured correctly, system achieves optimal cost / profit, even with parties acting selfishly

- There is a large body of literature on contracting
  - Review: Cachon 2003
  - Based on game theory
  - In practice, idea is commonly used
  - Actual OR models rarely implemented
Bullwhip Effect (BWE)

- Demand for diapers:
Irrational Behavior Causes BWE

- Firms over-react to demand signals
  - Order too much when they perceive an upward demand trend
  - Then back off when they accumulate too much inventory
- Firms under-weight the supply line
- Both are irrational behaviors
- Demonstrated by “beer game”
- Sterman 1989
Rational Behavior Causes BWE

- **BWE can be caused by** *rational* *behavior*
  - *i.e.*, by acting in “optimal” ways according to OR inventory models

- **Four causes:**
  - Demand forecast updating
  - Batch ordering
  - Rationing game
  - Price variations

- Lee, Padmanabhan, and Whang 1997
Further Reading

- **Single-stage and multi-echelon stochastic-service models:**
  - Undergrad / MBA textbooks:
    - Chopra and Meindl, 3rd ed., 2006
    - Nahmias, 5th ed., 2004
  - Graduate textbooks:
    - Zipkin, 2000
    - Axsäter, 2nd ed., 2006
    - Porteus, 2002

- **Guaranteed-service models:**
  - Graves and Willems 2003 (book chapter)
Questions?

LARRY.SNYDER@LEHIGH.EDU