한국경영과학회지
제35권 제1호
2010년 3월

계획생산과 주문생산 시설들로 이루어진 두 단계 공급망에서 재고 할당과 고객주문 수용 통제의 통합적 관리

김 은 갑*

Integrated Inventory Allocation and Customer Order Admission Control in a Two-stage Supply Chain with Make-to-stock and Make-to-order Facilities

Eungab Kim*

■ Abstract ■

This paper considers a firm that operates make-to-stock and make-to-order facilities in successive stages. The make-to-stock facility produces components which are consumed by the external market demand as well as the internal make-to-order operation. The make-to-order facility processes customer orders with the option of acceptance or rejection. In this paper, we address the problem of coordinating how to allocate the capacity of the make-to-stock facility to internal and external demands and how to control incoming customer orders at the make-to-order facility so as to maximize the firm's profit subject to the system costs. To deal with this issue, we formulate the problem as a Markov decision process and characterize the structure of the optimal inventory allocation and customer order control. In a numerical experiment, we compare the performance of the optimal policy to the heuristic with static inventory allocation and admission control under different operating conditions of the system.

Keyword : Two-stage Supply Chain, Inventory Allocation, Admission Control, Coordination, Inventory Control

논문접수일 : 2009년 10월 26일 논문게재확정일 : 2010년 02월 05일
논문수정일(1차 : 2009년 12월 29일, 2차 : 2010년 01월 28일)
* 이화여자대학교 경영대학

1. Introduction

In this paper, we consider a firm that operates two production facilities in successive stages; one is for make-to-stock (MTS) operation and the other is for make-to-order (MTO) operation. The MTS facility produces components which are consumed by the external market demand as well as the internal MTO operation. The MTO facility processes customer orders with the option of acceptance or rejection. In this paper, we address the problem of coordinating how to allocate the component inventory of the MTS facility to internal and external demands and how to control incoming customer orders at the MTO facility so as to maximize the firm's long-run expected profit subject to the system costs.

This work is motivated by a manufacturer that owns two production facilities ; one for semi-conductor chips and the other for printed circuit board (PCB) items. After produced, semi-conductor chips are stored in inventory for the production of PCB items which are requested by the companies that produces electronic products. Depending on inventory available, the manufacturer also sells semi-conductor chips to the PCB manufacturers in the market.

Several issues at the operational level can emerge from the model presented in this paper. If component production times tend to be longer than the delivery time requested by customer orders, components typically have to be stored in advance. In such an environment, the stocking decision for component is no longer a simple matter. Moreover, since there exist external demands on the component, the issue of inventory allocation (whether make-to-stock capacity should be served for external demand or internal demand) is raised. Incorporating an admission control into a make-to-order system is also important due to its applicability to real problems. The primary goal of this paper is (i) to present the formulation of an appropriate model for jointly managing inventory allocation and admission control, (ii) to investigate the structure of the optimal policy, and (iii) to implement the performance comparison of the optimal policy and the heuristic with static inventory allocation and admission control.

There is a vast literature on production control and blocking mechanisms in multi-stage production systems using queueing network modeling (see Dallery and Gershwin [3] for the literature survey). Veatch and Wein [13, 14] studied the problem of production planning in a two-stage production system where unmet demands from finished goods inventory are backordered. Assuming Poisson demand and exponential production times, Veatch and Wein [13] characterized the optimal production decisions as two monotone switching curves. In [14], Veatch and Wein found conditions that certain simple policies are optimal and showed that the base stock policy can never be optimal. Optimal controls were compared with kanban, base stock, and finite buffer control mechanisms.

The model presented in this paper is closely related to the literature dealing with joint admission (inventory rationing) and production control. Benjaafar and ElHafsi [1] considered a single product, multi-components assemble-to-order system with multiple demand classes where components are produced in a make-to-stock fashion. Under the assumption of instantaneous assembly, they showed that a base-stock policy is optimal for controlling production of each component and a component inventory reservation policy with state-dependent rationing levels is optimal for controlling admission of each demand class. ElHafsi [5] extended Benjaafar and ElHafsi [1] to the case that each demand class arrives according to a compound Poisson process. For a single product make-to-stock system with multiple demand classes with lost sales, Ha [6] showed that a base-stock policy is optimal for controlling production and an inventory reservation policy with fixed rationing levels is optimal for controlling admission. Ha [7] and de Vericourt et al. [4] extended these results to the case backordering for two demand classes and N demand classes, respectively. Teunter and Haneveld [12] considered a single product make-to-stock system with two demand classes (critical and non-critical) and studied the rationing policy that the number of items reserved for critical demand depends on the remaining time until the next order arrives. Carr and Duenyas [2] considered a assemble-to-order/make-to-stock system that produces two classes of products and studied the problem of joint admission control and production switching.

All references cited above assumed that (1) the order/demand arrival process follows a Poisson process and the production/assembly times follow the exponential distribution for the mathematical tractability and (2) each production/assembly generates one unit at a time because the setup cost is not included into the models. Despite these restrictions in real applications, the Makov models presented in the literature are known to be useful to gain the strategical insights into the nature of the joint admission and production control problem. In this paper, we adopt the same modeling approach as the above references but our model contributes to the current literature in the following essential aspects. First, we explicitly consider a two-stage production system while the references consider a single stage production system. Although ElHafsi [5] and Benjaafar and ElHafsi [1] considered a two-stage production system (i.e., assembly facility and component production facility), the assembly process is assumed to be instantaneous. Second, we deal with the issue of capacity rationing at the component production level while the references deal with the issue of demand rationing (i.e., which type of demand to accept or reject) based on the finished goods inventory.

The rest of the paper is organized as follows. Section 2 presents model assumptions and problem formulation. In Section 3, we determined the structure of the optimal policy. In Section 4, we implement a numerical study. The last section states conclusions.

2. Model Definition and Problem Formulation

Customer orders arrive at the MTO facility according to a Poisson process with rate λ. The firm can accept or reject each arriving customer order and a penalty cost of c_r is incurred whenever it is rejected. Each MTO operation requires one unit of component and takes an exponentially distributed amount of time with mean μ_o^{-1}. If it is completed, a revenue of R_o is generated. The MTS facility produces components one unit at a time according to an exponential distribution with mean μ_s^{-1}. If the firm sells one unit of component at the market, a revenue of R_s is generated. A service delay cost is assessed at rate h_1 for each outstanding customer order whereas a holding cost is incurred at rate h_2 for each component in inventory.

The set of decision epochs in our model corresponds to the epochs of customer order arrival, MTO operation completion, and component production. At each epoch of a component production, the firm should determine whether to stock or sell it. Furthermore, at each epoch of customer order arrival, the firm must decide whether to accept or to reject it. The goal of this paper is to find an inventory allocation and admission control which maximizes the expected discounted profit over an infinite horizon. [Figure 1] graphically illustrates a schematic model of the problem described above.

The state of the system can be described by the vector (n_1, n_2) where n_1 is the number of customer orders in queue and n_2 is the inventory level of component that will be used in the MTO operation. We denote the state space by Γ.

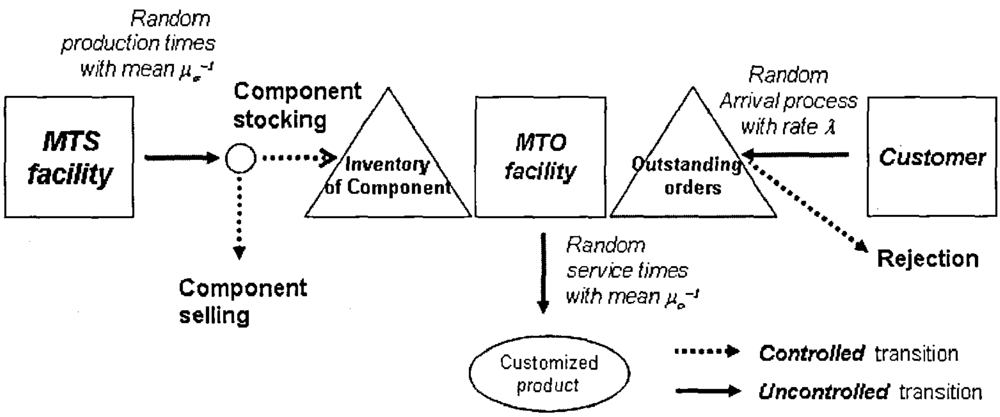
We can formulate the optimal control problem as a Markov Decision Process. Let $v(n_1, n_2)$ be the optimal expected discounted profit over an infinite horizon when the initial state is given by (n_1, n_2). Let $\gamma=\mu_o+\lambda+\mu_s$ and β be a continuous interest rate. Then, the expected length of time per state transition becomes γ^{-1} and the discount factor during γ^{-1} is given by $\gamma/(\beta+\gamma)$. The following state transitions are relevant to (n_1, n_2).

- Customer order arrival with a probability of λ/γ : If it is accepted, the size of outstanding customer orders increases by one. Otherwise, a cost of c_r is incurred.
- Service completion on a customer order with a probability of μ_o/γ : The size of outstanding customer orders and the inventory level decrease by one, respectively.
- Component production epoch with a probability of μ_s/γ : If it is decided to sell, a revenue of R_s is generated. Otherwise, the inventory level increases by one.

Therefore, we can write the optimality equation for $v(n_1, n_2)$ as follows :

$$
\begin{aligned}
v(n_1, n_2) = \frac{1}{\beta+\gamma}[&-(h_1 n_1 + h_2 n_2) \\
&+\mu_o [(v(n_1-1, n_2-1)+R_o)1(n_1>0, n_2>0) \\
&+v(n_1, n_2)1(n_1 n_2=0)] \\
&+\lambda \max[v(n_1+1, n_2), v(n_1, n_2)-c_r] \\
&+\mu_s \max[v(n_1, n_2+1), v(n_1, n_2)+R_s]
\end{aligned} \quad (1)
$$

where the indicator function $1(a)=1$ if a is true,



[Figure 1] A two-stage supply chain with inventory allocation and admission control otherwise, zero. In (1), $-(h_1n_1+h_2n_2)$ is the costs of service delay and inventory holding, the term multiplied by μ_o represents the transitions and revenue generated by the MTO operation, the term multiplied by λ represents the transitions and rejection cost generated by customer order arrival and the term multiplied by μ_s represents the transitions and revenue associated with inventory allocation. <Table 1> summarizes the key notations.

〈Table 1〉 Summary of key notations in the model

Notation	Description
μ_o^{-1}	Mean time of MTO operation
μ_s^{-1}	Mean time of component production
λ	Arrival rate of customer order
γ	State transition rate, sum of μ_o, λ, μ_s
β	Continuous interest rate
h_1	Holding cost rate of an outstanding customer order
h_2	Holding cost rate of a component in inventory
R_o	MTO Revenue
R_s	Component sales revenue
c_r	Rejection penalty of customer order
n_1	Size of outstanding orders
n_2	Inventory level of component
(n_1, n_2)	System state
Γ	State space
T	Value iteration operator
$v(n_1, n_2)$	Optimal expected discounted profit given initial state (n_1, n_2)
$A_1(n_2)$	Optimal inventory allocation curve
$A_2(n_1)$	Optimal admission control curve

3. Structure of the Optimal Policy

In this section, we characterize the structure of the optimal inventory allocation and admission control. We will follow Porteus ([9]). The key of this approach is to identify a set of structured value functions and to show that it is preserved under the value iteration operator. Then the corresponding structures of the optimal profit function and the optimal policy can be established. For any real valued function f on Γ, define $D_1f(n_1, n_2)=f(n_1+1, n_2)-f(n_1, n_2)$, $D_2f(n_1, n_2)=f(n_1, n_2+1)-f(n_1, n_2)$, and $D_{11}f(n_1, n_2)=f(n_1+1, n_2+1)-f(n_1, n_2)$. D_1 represents the marginal value of having one more outstanding customer order, D_2 is the additional value of holding one more component in inventory, and D_{11} is the value of holding one more outstanding customer order and one more component in inventory. Operators D_1, D_2, and D_{11} are convenient to define the supermodularity and concavity of f and substantially shorten the analysis. Let V be the set of all functions defined on Γ such that if $f\in V$, then

$$D_2f(n_1, n_2) \le D_2f(n_1+1, n_2), \quad (2)$$
$$D_2f(n_1, n_2) \ge D_2f(n_1+1, n_2+1), \quad (3)$$
$$D_2f(n_1, n_2) \ge D_2f(n_1, n_2+1), \quad (4)$$
$$D_1f(n_1, n_2) \ge D_1f(n_1+1, n_2+1), \quad (5)$$
$$D_1f(n_1, n_2) \ge D_1f(n_1+1, n_2), \quad (6)$$
$$D_{11}f(n_1, n_2) \le R_o \quad (7)$$

Equation (2) establishes the supermodularity of f and it can be rewritten as

$$D_1f(n_1, n_2) \le D_1f(n_1, n_2+1). \quad (8)$$

From (4) and (6), f is concave in n_1 and n_2. Equation (7) implies that the benefit attained from holding one more customer order in queue and one more component in inventory cannot exceed R_o.

To establish the structural properties of the optimal policy, we first show that Equations (2)~(7) are preserved under value iteration operator T. The proof of this and all subsequent results are included in the Appendix.

Lemma 1 : *If $f \in V$, then $Tf \in V$.*

We now state the main result of this section:

Theorem 1 : *(a) $v \in V$.*

(b) Let

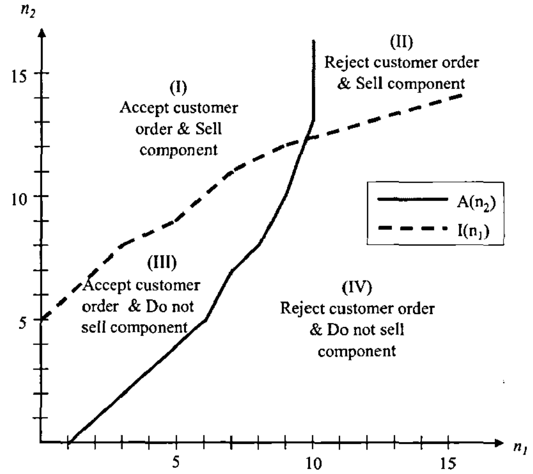
$$A(n_2) := \max\{n_1 : D_1 v(n_1, n_2) \geq -c_r\}, \quad (9)$$

$$I(n_1) := \max\{n_2 : D_2 v(n_1, n_2) \geq R_s\}. \quad (10)$$

The optimal admission control is defined by $A(n_2)$ such that when a customer order arrives in state (n_2, n_2), it is optimal to accept it if $n_1 \leq A(n_2)$, otherwise, to reject it. The optimal inventory allocation is defined by $I(n_1)$ such that if a component is produced in state (n_1, n_2), it is optimal to stock it if $n_2 \leq I(n_1)$, otherwise, to sell it at the market.

(c) *$A(n_2)$ is increasing in n_2 and $I(n_1)$ is increasing in n_1.*

Part (b) of Theorem 1 states that the structure of the optimal inventory allocation and admission control is characterized by two monotone switching curves $A(n_2)$ and $I(n_1)$. The reasoning behind part (c) of Theorem 1 is derived from the fact that the production system should keep a balance between the queue length of customer order and the component inventory to increase its capacity utilization. And this implies that the MTO facility will demand components depending on the ratio of the accepted customer order rate over the MTO service rate, μ_o.



[Figure 2] Graphical representation of the optimal switching curves

[Figure 2] graphically illustrates the structure of the optimal policy for the example with $R_o = 50$, $R_s = 5$, $c_r = 10$, $h_1 = 2$, $h_2 = 1$, $\mu_o = 1$, $\lambda = 0.5$, $\mu_s = 0.5$, and $\beta = 0.0001$. As seen in [Figure 2], $A(n_2)$ and $I(n_1)$ separate the state space into the following four regions : (I) Accept customer order and sell component, (II) Reject customer order and sell component, (III) Accept customer order and do not sell component, and (IV) Reject customer order and do not sell component. For example, in state (5, 5), the optimal inventory allocation is to not sell component and the optimal admission control is to accept an incoming customer order. $A(n_2)$ and $I(n_1)$ in [Figure 2] are found using value iteration (VI) method (see section 6.3.2, Puterman [10]). VI is the most widely used algorithm for solving discounted Markov decision problems. The key of this algorithm is to iteratively solve $v^{k+1}(x_1, x_2) = Tv^k(x_1, x_2), k = 0, 1, \cdots,$ until $|Tv^k - v^k|$ is within some termination criterion where

$$Tv^k(n_1, n_2) = \frac{1}{\beta + \gamma}[-(h_1 n_1 + h_2 n_2)$$
$$+\mu_o[(v^k(n_1 - 1, n_2 - 1) + R_o)1(n_1 > 0, n_2 > 0)$$

$$+v^k(n_1, n_2)1(n_1n_2=0)]$$
$$+\lambda \max[v^k(n_1+1, n_2), v^k(n_1, n_2)-c_r]$$
$$+\mu_s \max[v^k(n_1, n_2+1), v^k(n_1, n_2)+R_s].$$

If VI stops at the l^{th} iteration, $A(n_2)$ and $I(n_1)$ are obtained by setting $v(n_1, n_2) = Tv^l(n_1, n_2)$ and applying (9) and (10).

Next, we consider the case with the average profit criterion. Define the value iteration operator T for the average profit MDP as

$$Tw(n_1, n_2) = \frac{1}{\gamma}[-(h_1n_1+h_2n_2) \quad (11)$$
$$+\mu_o[(w(n_1-1, n_2-1)+R_o)1(n_1>0, n_2>0)$$
$$+w(n_1, n_2)1(n_1n_2=0)]$$
$$+\lambda \max[w(n_1+1, n_2), w(n_1, n_2)-c_r]$$
$$+\mu_s \max[w(n_1, n_2+1), w(n_1, n_2)+R_s]$$

From applying the same arguments as Lemma 1 and Theorem 1, the structure of the optimal policy exploited in the discounted profit MDP is still preserved for the average profit MDP. We also note that the policy of never rejecting customer orders cannot be optimal, since x_1 can go to the infinite, and thus the policy of never selling components cannot be optimal because x_1 is finite. Since (11) has a finite state space and a finite action space, and it is unichain, from Puterman ([10]), there exists an optimal average profit g which satisfies

$$g+w(n_1, n_2) = Tw(v_1, n_2). \quad (12)$$

4. Numerical Study

In this section, we present numerical results that compare the performance of the optimal policy and a policy which controls the decision of admission control and inventory allocation with static two single parameters M_1 and M_2, respectively. The (M_1, M_2) policy accepts each arriving customer order only when $n_1 < M_1$ and produces a component for internal make-to-order operation only when $n_2 < M_2$. The (M_1, M_2) policy is certainly a simple policy to describe and implement in practice. In fact, fixed single parameter controls have been presented and analyzed in many inventory management papers ([8, 11, 14]). For the easiness of the comparison, we compute the optimal average profit rather than the optimal discounted profit. The optimal average profit of (M_1, M_2) policy given M_1 and M_2 is computed using value iteration and the optimal values of M_1 and M_2 are found using a two-dimensional search. More specifically, we first set the possible ranges of M_1 and M_2 values. Second, given some specific values of M_1 and M_2, we iteratively solve $u^{k+1}(n_1, n_2) = Tu^k(n_1, n_2), k=0, 1, \cdots,$ until $|Tu^k - u^k|$ is within some termination criterion for each (n_1, n_2) where

$$Tu^k(n_1, n_2) = \frac{1}{\gamma}[-(h_1n_1+h_2n_2) \quad (13)$$
$$+\mu_o[(u^k(n_1-1, n_2-1)+R_o)1(n_1>0, n_2>0)$$
$$+u^k(n_1, n_2)1(n_1n_2=0)]$$
$$+\lambda[u^k(n_1+1, n_2)1(n_1<M_1)$$
$$+(u^k(n_1, n_2)-c_r)1(n_1=M_1)]$$
$$+\mu_s[u^k(n_1, n_2+1)1(n_2<M_2)$$
$$+(u^k(n_1, n_2)+R_s)1(n_2=M_2)]$$

and $u^0(n_1, n_2)=0$ for each (n_1, n_2). Then, set $u(n_1, n_2) = Tu^k(n_1, n_2)$. For all the combinations of M_1 and M_2 values, we implement VI and find the best of M_1 and M_2 which maximizes $u(n_1, n_2)$. Note that it is conjectured that $u(n_1, n_2)$ is concave in M_1 and M_2 but it cannot be proven.

<Table 2> compares the optimal average profits obtained by the optimal policy and (M_1, M_2) policy described for 36 examples. Since the structure of the optimal policy is quite complex, we focused on the development of heuristic policy that is certainly very simple to describe and implement in practice if the appropriate values of the parameters are found. For this reason, the performance comparison is made in terms of the profits obtained by the optimal and heuristic policies rather than the cpu times spent in finding them. In this table, % is defined as the performance difference between the optimal policy and (M_1, M_2) policy to the optimal performance. Test results in <Table 2> suggest that the performance of (M_1, M_2) policy can be sensitive to the example groups. For example, the % difference becomes larger as λ increases. The average performance difference between the optimal policy and (M_1, M_2) policy is 6.5%.

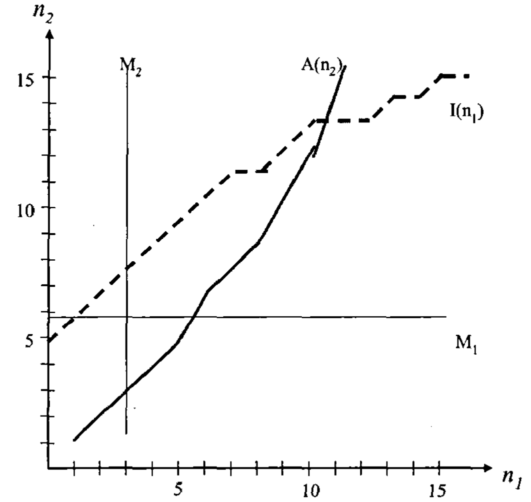
[Figure 3] graphically compares the structure of the optimal policy and (M_1, M_2) policy using Example 13 in <Table 2>. Like the optimal policy, the (M_1, M_2) policy defines four regions that manage inventory allocation and admission control. The advantage of (M_1, M_2) policy is that it is much simpler, particularly when the size of state space is large, to implement than the optimal policy because it does not require the computation of the state dependent $A(n_2)$ and $I(n_1)$. However, the (M_1, M_2) policy is clearly suboptimal as illustrated in [Figure 3] and lacks the coordination in managing inventory allocation and admission control carried out under the optimal policy.

<Table 2> reveals two interesting results. First, the performance difference between the optimal policy and (M_1, M_2) policy can be larger as λ or h_1 increases. Second, it can be smaller as μ_s, R_o, R_s or h_2 increases. These results can be intuitively explained as follows :

- When λ becomes larger, it appears important to effectively handle rejection of customer orders due to the increased traffic intensity, which makes the role of dynamic admission control crucial in reducing excessive rejection costs over the static admission control.
- When μ_s is low (components are produced slowly), the system variability can be amplified. Hence, it is reasonable to think that a dynamic inventory allocation and admission control will be more cost effective, which is consistent with other works in the inventory control literature which show that the dynamic control becomes a more effective way over static rules when lead time is longer.
- If R_o increases, the optimal policy will accept more customer orders and stock more components. Hence, it appears that the cost reduction expected from a dynamic admission control can be limited.
- When h_1 is high, the decision of accepting or rejecting a particular customer order should be done more carefully because holding customer orders in queue becomes more expensive. Hence, we can expect that the dynamic admission control will contribute to reducing the excessive payment of service delay cost.
- As R_s or h_2 becomes larger, the action of sell will be preferable to the action of produce-to-stock, which leads less components to be kept in inventory (thus less customer orders to be accepted). Hence, it can be expected that the role of dynamic inventory allocation and admission control will be reduced.

〈Table 2〉 Performance comparison of the optimal and (M_1, M_2) polices

Ex	R_o	R_s	c_r	h_1	h_2	λ	μ_o	μ_s	g^{Opt}	(M_1, M_2)			%
										g^H	M_1	M_2	
1	100	5	15	2	1	0.45	1	0.2	14.7	14.0	1	20	4.8
2						0.55			13.4	12.7	1	20	5.6
3						0.65			12.1	11.3	1	20	7.0
4						0.75			10.6	9.8	1	20	8.8
5						0.85			9.2	8.3	1	20	11.4
6						0.95			7.7	6.8	1	20	14.7
7	50	10	10	2	1	0.5	1	0.2	5.6	4.9	1	20	13.6
8								0.3	10.2	9.4	2	9	8.4
9								0.4	14.2	13.5	3	7	4.8
10								0.5	17.1	16.7	4	6	2.2
11								0.6	19.4	19.1	5	4	1.7
12								0.7	21.3	21.0	6	4	1.7
13	50	5	5	2	1	0.4	1	0.4	13.4	13.1	3	5	2.2
14	75								22.1	21.9	4	6	1.1
15	100								31.1	30.9	4	7	0.7
16	125								40.2	40.0	5	8	0.5
17	150								49.4	49.2	5	9	0.4
18	175								58.6	58.4	5	9	0.3
19	50	5	5	2	1	0.4	1	0.6	16.7	16.4	5	3	1.9
20		10							17.7	17.4	5	3	1.7
21		15							18.8	18.5	5	3	1.6
22		20							19.9	19.6	4	3	1.5
23		25							21.0	20.7	4	3	1.4
24		30							22.1	21.9	3	2	1.1
25	50	10	5	1	1	0.6	1	0.25	9.4	8.9	1	7	4.9
26				2					9.0	8.3	1	8	8.4
27				3					8.7	7.8	1	8	12.5
28				4					8.4	7.2	1	8	17.3
29				5					8.1	6.6	1	8	23.4
30				6					7.8	6.0	1	9	30.7
31	100	5	5	2	1	0.65	1	0.35	30.7	29.8	2	20	7.8
32					2				29.3	28.5	2	10	7.3
33					3				28.2	27.3	2	7	7.3
34					4				27.0	26.2	2	6	7.1
35					5				26.0	25.2	2	5	6.3
36					6				25.2	24.5	3	5	3.0



[Figure 3] Structural comparison of the optimal policy and the (M_1, M_2) policy

5. Conclusions

In this paper, we address the problem of integrating inventory allocation and admission control for a production system with make-to-stock and make-to-order facilities in successive stages. In particular, we dealt with the issue of inventory allocation at the component production level, which has not been treated in the related literature. Using a Markov decision process framework, we characterized the structure of the optimal inventory allocation and admission control by two monotone switching curves.

To provide a better understanding of the structure of the optimal policy, we compared the performance of the optimal policy to the policy with static inventory allocation and admission control under different operating conditions of the system. Through extensive numerical experiments, we examined how much a dynamic inventory allocation and admission control is effective and under what conditions it is much more favorable over static controls. Test results showed that the static policy works poor when component production time is longer, the inter-arrival time of customer order is shorter, or service delay cost of customer order is larger. This suggests that developing more efficient and computationally inexpensive heuristic policy is demanding.

References

[1] Benjaafar, S. and M. Elhafsi, "Production and inventory control of a single product assemble-to-order system with multiple customer classes," *Management Science,* Vol. 52(2006), pp.1896-1912.

[2] Carr, S. and I. Duenyas, "Optimal Admission Control and Sequencing in a Make-To-Stock/Make-To-Order Production System," *Operations Research,* Vol.48(2000), pp.709-720.

[3] Dallery, Y. and S. Gershwin, "Manufacturing flow line systems : A review of models and analytical result," *Queueing Systems,* Vol.12(1992), pp.3-94.

[4] de Vericourt, F. and Y. Karaesmen, "Stock allocation for a capacitated supply chain," *Management Science,* Vol.48(2002), pp.486-1501.

[5] Elhafsi, M., "Optimal integrated production and inventory control of an assemble-to-order system with multiple non-unitary demand classes," *European Journal of Productional Research,* Forthcoming.

[6] Ha, A.Y., "Inventory rationing in a make-to-stock production system with several demand classes and lost sales," *Management Science,* Vol.43(1997a), pp.1093-1103.

[7] Ha, A.Y., "Stock rationing policy for a make-to-stock production system with two

priority classes and backordering," *Naval Research Logistics*, Vol.44(1997b), pp.457-472.

[8] Muckstadt, J.A. and S.R. Tayur, "A Comparison of Alternative Kanban Control Mechanisms. I. Background and Structural Results," *IIE Transactions*, Vol.27(1995), pp.140-150.

[9] Porteus, E., "Conditions for Characterizing the Structure of Optimal Strategies in Infinite-Horizon Dynamic Programs," *Journal of Optimization Theory and Applications*, Vol.36(1982), pp.419-432.

[10] Puterman, M., *Markov Decision Processes*, John Wiley and Sons, 2005.

[11] Song, J. and D. Yao, "Performance analysis and optimization of assemble-to-order systems with random lead times," *Operations Research*, Vol.50(2002), pp.889-903.

[12] Teunter, R. and W. Haneveld, "Dynamic inventory rationing strategies for inventory systems with two demand classes, Poisson demand and backordering," *European Journal of Productional Research*, Vol.190(2008), pp.156-178.

[13] Veatch, M. and L. Wein, "Monotonic control of queueing networks," *Queueing Systems*, Vol.12(1992), pp.391-408.

[14] Veatch, M. and L. Wein, "Optimal control of a two-station tandem production/inventory system," *Operation Research*, Vol.42 (1994), pp.337-350.

〈Appendix〉

Proof of Lemma 1 : Denote by $(A/R/S/N)$ the optimal action in (n_1, n_2) where A, R, S and N represent *accept, reject, sell,* and *do not sell* actions, respectively. Define

$T_1f(n_1, n_2) = (v(n_1-1, n_2-1)+R_o)1(n_1>0, n_2>0)+v(n_1, n_2)1(n_1n_2=0)$

$T_2f(n_1, n_2) = \max\{f(n_1+1, n_2), f(n_1, n_2)-c_r\}$

$T_3f(n_1, n_2) = \max\{f(n_1, n_2+1, f(n_1, n_2)+R_s\}$

$Tf(n_1, n_2) = \frac{1}{\beta+\gamma}[-(h_1n_1+h_2n_2)+\mu_oT_1f(n_1, n_2)+\lambda T_2f(n_1, n_2)+\mu_sT_3f(n_1, n_2)]$

To prove Lemma 1, we show that (2)~(7) are preserved under T_1, T_2, and T_3.

(i) Let $\triangle^i = D_2T_if(n_1, n_2)-D_2T_if(n_1+1, n_2)$.

- $\triangle^1$: If $n_1n_2>0$, $\triangle^1 \le 0$ by (2). If $n_1=0$ and $n_2>0$, $\triangle^1 = D_2f(0, n_2)-D_2f(0, n_2-1) \le 0$ by (4). If $n_1>0$ and $n_2=0$, $\triangle^1 = D_1f(n_1, 0)-D_1f(n_1-1, 0) \le 0$ by (6). If $n_1=n_2=0$, $\triangle^1 = D_2f(0, 0)-(R_o+f(0, 0)-f(1, 0))$ $\le D_2f(0, 0)-(f(1, 1)-f(1, 0))$(by (7)) ≤ 0 by (2).
- $\triangle^2$: Focusing on the combinations of actions in (n_1, n_2+1), (n_1, n_2), (n_1+1, n_2+1), and (n_1+1, n_2), cases $(R, A, \cdot, \cdot)$ and $(\cdot, \cdot, R, A)$ are excluded by (2), $(\cdot, R, \cdot, A)$ and $(R, \cdot, A, \cdot)$ by (6), and $(\cdot, R, A, \cdot)$ by (5). Cases (A, A, A, A) and (R, R, R, R) follow from (2). For (A, A, R, R), $\triangle^2 = D_2f(n_1+1, n_2)-D_2f(n_1+1, n_2)=0$.

  For (A, A, A, R) and (A, R, R, R), $\triangle^2 \le D_2f(n_1+1, n_2)-D_2f(n_1+1, n_2)=0$.
- $\triangle^3$: Consider the combinations of actions in (n_1, n_2+1), (n_1, n_2), (n_1+1, n_2+1), and (n_1+1, n_2). Cases $(N, S, \cdot, \cdot)$ and $(\cdot, \cdot, N, S)$ are excluded by (4), $(N, \cdot, S, \cdot)$ and $(\cdot, N, \cdot, S)$ by (2), and $(\cdot, S, N, \cdot)$ by (3). Cases (S, S, S, S) and (N, N, N, N) follow from (2). For (S, N, S, N), $\triangle^3 = D_1T_3f(n_1, n_2)-D_1T_3f(n_1, n_2)-D_1T_3f(n_1+1, n_2)=0$. For (S, S, S, N) and (S, N, N, N), $\triangle^3 \le D_1T_3f(n_1, n_2)-D_1T_3f(n_1+1, n_2)=0$.

(ii) Let $\triangle^i = D_2T_if(n_1, n_2)-D_2T_if(n_1+1, n_2+1)$.

- $\triangle^1$: If $n_1n_2>0$, $\triangle^1 \ge 0$ by (3). If $n_1>0$ and $n_2=0$, $\triangle^1 = R_o+f(n_1-1, 0)-f(n_1, 0)-D_2f(n_1, 0) = R_o-D_{11}f(n_1-1, 0) \ge 0$ by (7).

  If $n_1=0$, $\triangle^1 = D_2f(0, n_2)-D_2f(0, n_2)=0$.
- $\triangle^2$: Focusing on the combinations of actions in (n_1, n_2+1), (n_1, n_2), (n_1+1, n_2+2), and (n_1+1, n_2+1), cases $(R, A, \cdot, \cdot)$ and $(\cdot, \cdot, R, A)$ are excluded by (2), $(R, \cdot, \cdot, A)$ by (6), and $(\cdot, R, \cdot, A)$ and $(R, \cdot, A, \cdot)$ by (5). Cases (A, A, A, A) and (R, R, R, R) follow from (3). For (A, A, R, R), $\triangle^2 = D_2f(n_1+1, n_2)-D_2f(n_1+1, n_2+1) \ge 0$ by (4). For (A, R, A, R), $\triangle^2 = D_{11}f(n_1, n_2)-D_{11}f(n_1+1, n_2+1) \ge D_2f(n_1+1, n_2)$ $-D_2f(n_1+2, n_2+1)$ (by (5)) ≥ 0 (by (3)). For (A, A, A, R), $\triangle^2 \ge D_2f(n_1+1, n_2)-D_2f(n_1+2, n_2+1) \ge 0$ by (3).

  For (A, R, R, R), $\triangle^2 \ge D_2f(n_1, n_2)-D_2f(n_1+1, n_2+1) \ge 0$ by (3).
- $\triangle^3$: Consider the combinations of actions in (n_1, n_2+1), (n_1, n_2), (n_1+1, n_2+2), and (n_1+1, n_2+1). Cases $(N, S, \cdot, \cdot)$ and $(\cdot, \cdot, N, S)$ are excluded by (4), and $(S, \cdot, N, \cdot)$ and $(\cdot, S, \cdot, N)$ by (3). Cases (S, S, S, S) and (N, N, N, N) follow from (3). For (S, N, S, N), $\triangle^3 = R_s-R_s=0$. For (S, N, S, S) and

(N, N, S, N), $\Delta^3 \geq f(n_1, n_2+1) - R_s - f(n_1, n_2+1) - [f(n_1+1, n_2+2) - R_s - f(n_1+1, n_2+2)] = 0$

(iii) $D_2 Tf(n_1, n_2+1) \leq D_2 Tf(n_1+1, n_2+1)$ (by (2)) $\leq D_2 Tf(n_1, n_2)$ (by (3)).

(iv) Let $\Delta^i = D_1 T_i f(n_1, n_2) - D_1 T_i f(n_1+1, n_2+1)$.

- Δ^1 : If $n_1 n_2 > 0$, $\Delta^1 \geq 0$ follows by (5). If $n_1 = 0$ and $n_2 > 0$, $\Delta^1 = R_o - D_{11} f(0, n_2-1) \geq 0$ (by (7)). If $n_2 = 0$, $\Delta^1 = D_1 f(n_1, 0) - D_1 f(n_1, 0) = 0$.
- Δ^2 : Consider the combinations of actions in (n_1+1, n_2), (n_1, n_2), (n_1+2, n_2+1), and (n_1+1, n_2+1). Cases $(A, R, \cdot, \cdot)$ and $(\cdot, \cdot, A, R)$ are excluded by (6), $(\cdot, R, \cdot, A)$ and $(R, \cdot, A, \cdot)$ by (5), and $(A, \cdot, \cdot, R)$ by (2). Cases (A, A, A, A) and (R, R, R, R) follow from (5). For (R, A, R, A), $\Delta^2 = -c_r + c_r = 0$. For (A, A, R, A) and (R, A, R, R), $\Delta^2 \geq f(n_1+1, n_2) - c_r - f(n_1+1, n_2) - [f(n_1+2, n_2+1) - c_r - f(n_1+2, n_2+1)] = 0$.
- Δ^3 : Considering the combinations of actions in (n_1+1, n_2), (n_1, n_2), (n_1+2, n_2+1), and (n_1+1, n_2+1), cases $(S, N, \cdot, \cdot)$ and $(\cdot, \cdot, S, N)$ are excluded by (2), $(S, \cdot, N, \cdot)$ and $(\cdot, S, \cdot, N)$ by (3), and $(S, \cdot, \cdot, N)$ by (4). Cases (S, S, S, S) and (N, N, N, N) follow from (5). For (N, N, S, S), $\Delta^3 = D_1 f(n_1, n_2+1) - D_1 f(n_1+1, n_2+1) \geq 0$ (by (6)). For (N, S, N, S), $\Delta^3 = D_{11} f(n_1, n_2) - D_{11} f(n_1+1, n_2+1)) \geq D_2 f(n_1+1, n_2) - D_2 f(n_1+2, n_2+1)$ (by (5)) ≥ 0 by (3). For (N, N, N, S) and (N, S, S, S), $\Delta^3 \geq D_{11} f(n_1, n_2) - D_{11} f(n_1+1, n_2+1)) \geq 0$.

(v) $D_1 Tf(n_1, n_2+1) \geq D_1 Tf(n_1, n_2)$ (by (2)) $\geq D_2 Tf(n_1+1, n_2+1)$ (by (5)).

(vi) Let $\Delta^i = D_{11} T_i f(n_1, n_2)$. For Δ^1, if $n_1 n_2 = 0$, $\Delta^1 = R_o$, otherwise, $\Delta^1 = D_{11} f(n_1-1, n_2-1) \leq R_o$ from (7). For Δ^2, considering the combinations of actions in (n_1+1, n_2+1) and (n_1, n_2), cases (A, A) and (R, R) follows from (7), case (R, A) follows from $\Delta^2 \leq D_{11} f(n_1, n_2) \leq R_o$, and case (A, R) is excluded by (5). For Δ^3, consider the combinations of actions in (n_1+1, n_2+1) and (n_1, n_2). Cases (S, S) and (N, N) follow from (7), case (S, N) follows from $\Delta^2 \leq D_{11} f(n_1, n_2) \leq R_o$, case (N, S) is excluded by (3). Therefore, since $\Delta^i \leq R_o$, $D_{11} Tf(n_1, n_2) = \frac{1}{\beta+\gamma}[-(h_1 n_1 + h_2 n_2) + \mu_o \Delta^1 + \lambda \Delta^2 + \mu_s \Delta^3]$ $\leq R_o$.

Proof of Theorem 1 : (i) From (1) and the definition of T_1, T_2, T_3, and T, we have

$$Tv(n_1, n_2) = \frac{1}{\beta+\gamma}[-(h_1 n_1 + h_2 n_2) + \mu_o T_1 v(n_1, n_2) + \lambda T_2 v(n_1, n_2) + \mu_s T_3 v(n_1, n_2)].$$

Since Lemma 1 holds for any real valued function on Γ, it follows that $v \in V$.

(ii) Suppose $v(n_1+2, n_2) > v(n_1+1, n_2) - c_r$. From the concavity of v in n_1, $v(n_1+1, n_2) > v(n_1, n_2) - c_r$, which implies that if the optimal policy accepts customer order in (n_1+1, n_2), then it does in (n_1, n_2). Suppose $v(n_1, n_2+1) + R_s < v(n_1, n_2+2)$. From the concavity of v in n_2, $v(n_1, n_2) + R_s < v(n_1, n_2+1)$. Hence, if the optimal policy does not sell component in (n_1, n_2+1), then it does not in (n_1, n_2).

(iii) Suppose $v(n_1+1, n_2) > v(n_1, n_2) - c_r$. From the supermodularity of v, $v(n_1+1, n_2+1) > v(n_1, n_2+1) - c_r$. That is, if it is optimal to accept customer order in (n_1, n_2), then it is optimal to do in (n_1, n_2+1). Suppose $v(n_1, n_2) + R_s < v(n_1, n_2+1)$. From the supermodularity of v, $v(n_1+1, n_2) + R_s < v(n_1+1, n_2+1)$. Hence, if it is optimal not to sell component in (n_1, n_2), then it is not optimal to do in (n_1+1, n_2).