An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957 eISSN: 2146-5703
Vol.10, No.1, pp.26-36 (2020)
https://doi.org/10.11121/ijocta.01.2020.00801
RESEARCH ARTICLE

Control of M/Cox-2/s make-to-stock systems

Özgün Yücel, Önder Bulut*
Department of Industrial Engineering, Yaşar University, Turkey
ozgun.yucel@yasar.edu.tr, onder.bulut@yasar.edu.tr

ARTICLE INFO

Article history:
Received: 21 March 2019
Accepted: 14 September 2019
Available Online: 24 September 2019

Keywords:
Make-to-stock
Dynamic programming
Production control
Multi-server systems
Phase-type production times

AMS Classification 2010:
90B30, 90B05, 90C39, 90C40

ABSTRACT

This study considers a make-to-stock production system with multiple identical parallel servers, fixed production start-up costs and lost sales. Processing times are assumed to be two-phase Coxian random variables that allows us to model the systems having rework or remanufacturing operations. First, the dynamic programming formulation is developed and the structure of the optimal production policy is characterized. Due to the highly dynamic nature of the optimal policy, as a second contribution we propose an easy-to-apply production policy. The proposed policy makes use of the dynamic state information and controlled by only two parameters. We test the performance of the proposed policy at several instances and reveal that it is near optimal. We also assess the value of dynamic state information in general by comparing the proposed policy with the well-known static inventory position based policy.

1. Introduction

In a make-to-stock production system, there is always a tradeoff between excess inventory, shortages and production costs. Production control is the main tool handling this tradeoff and providing cost effective operation. In general, in a make-to-stock environment, optimal production control requires starting production at the right time and producing with the optimum number of channels (servers, lines, or machines) to provide sufficient amount of products.

Production policy strategies use the information of inventory status to trigger the production when the inventory status drops below certain threshold levels. Here, inventory status refers a function of the state variables that keep track of the required system information such as inventory level, number of outstanding production orders and their ages. The form of the optimal inventory status function would change from system to system but it is still unknown even for most of the basic make-to-stock production settings. Therefore, most of the studies in the literature, which consider only a single server, assumes that inventory status equals inventory level. There are limited number of studies on multi-server production-inventory systems but they only provide partial characterization of the optimal policy without any discussion on the performances of the static, which should take inventory status as inventory position, or alternative dynamic policies.

In real life production-inventory systems, due to the nature of the environment and its technology, production times might have zero, moderate or high variance. Furthermore, such systems might have rework/inspection or remanufacturing operations. In order to deal with such real life systems, we assume phase-type, in specific two-phase Coxian production times. A busy server (worker or machine) might be either at the first phase (main operation) or at the second phase (inspection/rework) at any given time. A two-phase Coxian random variable has independent exponential phases and there is a certain visiting probability from phase-one to phase-two. Hence, we can create different systems at the boundaries of the visiting probability: when it is set to zero, processing time distribution becomes exponential (which is a typical assumption in the literature), when it is set to one, we can mimic the two-phase general Erlang processing times. Different values of this probability and production rates of phases correspond to systems with different rework characteristics and processing time moments. The representation of a production channel feeding the inventory after a two-phase Coxian processing time is shown in Figure 1. Coxian production times assumption would also help us to assess the value of dynamic state information, i.e. current status of production.

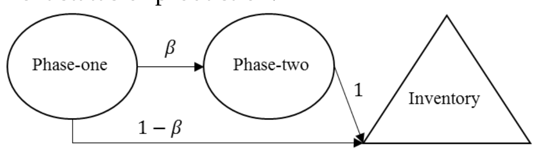

Figure 1. Representation of a Cox-2 production server

*Corresponding author

We charge fixed production (start-up) cost for activating servers, holding cost for each unit of inventory and lost sale cost for each unsatisfied demand. The studies that consider fixed costs in the literature are assuming only a single server. To the best of our knowledge, our study is the first considering multiple parallel production servers and fixed start-up cost at the same time in make-to-stock control environment. There is no study in the literature characterizing the optimal production policy for multi-server systems. For single server backordering systems, it is known that the optimal production policy is a two-critical-number policy. In this study, we aim to characterize the optimal production policy for lost sales multi-server systems with fixed production cost and propose easy to apply alternatives.

We provide the literature review in Section 2. Dynamic programming formulation of the problem is given in Section 3. In Section 4, we numerically characterize the optimal production policy. In Section 5, we propose an alternative production policy and evaluate its performance. Section 6 concludes the paper and provides future research directions.

2. Literature review

In this chapter, we review the production and inventory control literature in the make-to-stock environment. This problem is first attacked by considering the systems having single production channel and single customer/demand class. Analyses are mostly based on queueing theory techniques. Interestingly, the early studies consider the fixed startup or shut-down costs. More recent studies extend the literature by considering multiple production channels without fixed costs. Another common feature of the recent studies is the Markovian structure that enables them to develop Markov Decision Process (MDP) formulation for the control of make-to-stock systems.

Gavish and Graves [1] is the first to study the production-inventory problem assuming single channel, fixed and deterministic production times, independent Exponential inter-demand-arrival times, and backorders. They modeled the problem as an $M/D/1$ make-to-stock queue in the infinite horizon under the time-average cost criterion. This first study is actually the extension of Heyman [2] and Sobel [3] to the make-to-stock production environment. In [2] and [3], $M/G/1$ and $G/G/1$ queueing systems are studied, respectively, operating with server start-up and shutdown costs, and unit service and queue-time costs. For both of the settings, it is shown that the optimal policy is a *two critical number policy* denoted by (S, s) and (M, m) in [2] and [3], respectively. If the queue length is less than or equal to m (or s), service is not provided until queue length increases to M (or S). Service is triggered when the queue length is M and continued until it drops to m again. Although the analyses of [2] and [3] are specific for the queueing environment, we believe that their setting covers the production control for make-to-order systems. The optimal policy structure, which is a *two critical number policy*, is preserved in the make-to-stock production environment setting of [1]. However, the control parameters of the policy are defined on the inventory level: start production when the inventory level hits to the lower control level and continue until it hits to the upper control level. For different settings where two critical number policy is still optimal, see [4] and [5]. Researchers apply different techniques for the analysis of the *two critical number policy*. For example, Lee and Srinivasan [6] considers $M/G/1$ make-to-stock queue with backordering and propose a renewal analysis in order to calculate expected cost. For compound Poisson demand extension of this study see [7].

Recent studies mostly apply MDP techniques for the settings having Markovian structure. This stream of literature usually assumes no fixed production/setup cost. In addition, production is triggered by a single server except Bulut and Fadıloğlu [8]. Ha [9] is the first that uses MDP techniques in problem modeling. [9] addresses $M/M/1$ make-to-stock queue with multiple demand classes and lost sales, and shows that base-stock is optimal production control policy. For backordering case, see [10]. [8] extends the setting by assuming multiple parallel exponential servers and optimal policy is defined as state-dependent base-stock. Ha [11] proves that work storage level is optimal production policy for $M/E_k/1$ make-to-stock queue. Gayon et al. [12] differs from [11] with the backordering assumption. However, in our study, preserving the Markovian structure, we consider multiple parallel production servers allowing reworks and fixed start-up costs at the same time. Interested readers are also directed to the study [13] that considers the control of hybrid make-to-stock/make-to-order systems.

3. Dynamic programming formulation

We consider a production system including s many identical parallel servers each having two production phases in order to produce a single type of product. Processing times are assumed to be two-phase Coxian random variables where each phase is exponentially distributed with rates μ_1 and μ_2, respectively. Production is started at phase-one, then items are either processed at second phase with a certain probability β, or leave the system without passing second stage with probability $1-\beta$ (Figure 1). Visiting probability β facilitates us to work on more general systems than the ones having exponential processing times, which is a classical assumption in the literature. We model the system as $M/Cox_2/s$ make-to-stock queue with fixed start-up costs and lost sales. In the terminology of production-inventory control literature, the classical Kendall Lee queueing notation is used for the models of make-to-stock systems. However, the meaning of the queuing notation is slightly different in the make-to-stock environment. In our case, M denotes Markovian inter-demand arrival times but the arrived demands do not enter a queue and trigger a production order. Instead, they are either directly satisfied from the inventory or lost, and immediately leave the system. The second entry in the notation, which is "Cox_2" in our case, is for the production time distribution. The inventory is replenished using s many available production channels according to a production policy in anticipation of the future demand arrivals. That is, Coxian-2 is not the "service" time of each demand arrival; it is the replenishment lead-time of any production order triggered according to the policy.

Customer demands arrive according to a stationary Poisson process with rate λ. Lost sale cost c is incurred for each unsatisfied demand. Fixed start-up cost of activating a server is K, inventory holding cost is h and discount rate is denoted by α.

System state is defined with three variables to keep track of the events. Let $x_i(t), i \in \{1,2\}$, be the number of active servers at i^{th} phase and $x_3(t)$ be the inventory level at time t. Then the system state space is

$$SS = \left\{ \begin{matrix} \left(x_1(t), x_2(t), x_3(t)\right) \mid \sum_{i=1}^{2} x_i(t) \le s, \\ x_i(t) \in Z^+ \cup \{0\}, i = 1,2,3 \end{matrix} \right\} \quad (1)$$

Through the Markovian property, decision can be made in either at a phase completion or a demand arrival. For this reason, system state definition $\left(x_1(t), x_2(t), x_3(t)\right)$ is used regardless of time dimension. Since the original problem is a production-inventory control problem in continuous time, we obtain the discrete time equivalent of this problem via uniformization technique ([14]). The uniform transition rate is defined as $\nu = \lambda + s(\mu_1 + \mu_2)$. In our model, production is controlled by the decision variable $u \in \{x_1, \dots, s - x_2\}$, which is the number of busy servers at phase-1 (at the first stage of the production process). Model only controls the number of active servers at stage-one because whenever production is triggered on a server, it starts from stage-one. The production control variable is upper bounded by number of servers that are not at stage-two and lower bounded by number of active servers at phase-one since order cancellation is not allowed. Based on the above definitions, optimal cost-to-go function J is given by

$$\begin{aligned} J(x_1, x_2, x_3) = \frac{1}{\nu + \alpha} \min_{x_1 \le u \le s - x_2} \{ & h x_3 + K(u - x_1) \\ & + u\mu_1 \beta J(u-1, x_2+1, x_3) \\ & + u\mu_1 (1-\beta) min\{J(u-1, x_2, x_3+1), J(u, x_2, x_3+1)\} \\ & + x_2\mu_2 min\{J(u, x_2-1, x_3+1), J(u+1, x_2-1, x_3+1)\} \\ & + (s(\mu_1+\mu_2) - u\mu_1 - x_2\mu_2) J(u, x_2, x_3) + \lambda L(u, x_2, x_3)\} \end{aligned} \quad (2)$$

where L is the lost sales operator expressed by

$$L(x_1, x_2, x_3) = \begin{cases} J(x_1, x_2, x_3 - 1), & x_3 > 0 \\ c + J(x_1, x_2, 0), & x_3 = 0 \end{cases} \quad (3)$$

We aim to identify how many production servers should be active/busy at any given state to minimize the expected discounted system cost. The minimization operation defined with rate $u\mu_1(1-\beta)$ corresponds to the decision at the time of production completion at phase-one: it decides whether to continue production on the server that has just finished processing at the first phase and replenished inventory. The next optimizer, recalled with rate $x_2\mu_2$, is to decide whether to continue production on the server that has just finished processing at the second phase and replenished inventory. One should note that if fixed production cost is zero, these two continuation operators are redundant because the system can reactivate any server with zero cost whenever needed.

The term $(s(\mu_1+\mu_2) - u\mu_1 - x_2\mu_2)J(u, x_2, x_3)$ is necessary for the fictitious self-transitions due to the uniformization. In equation (3), the operator L corresponds to the transitions triggered by demand arrivals: if there is inventory on-hand, it is decreased by one, otherwise lost sales cost is incurred and state remains the same.

4. Characterization of the optimal policy

In this section, we provide a numerical characterization of the optimal production policy under average system cost. Since the system dynamics can be very clearly expressed with discounted cost DPs, we developed our formulation accordingly. However, we conduct numerical studies under average system cost criteria in order to make the performance measure independent of the initial state and the discount factor. We apply the value iteration algorithm to the system defined by equations (2) and (3) with discount rate $\alpha = 0$. Average system cost is calculated as the convergent value of the ratio of the optimal cost-to-go function value and the number of iterations.

Gavish and Graves [4] shows that two-critical-number policy is optimal for backordering $M/G/1$ make-to-stock systems with fixed start-up cost. This policy dictates that production should be triggered when the inventory level drops to the lower control level (I^*) and it should be continued until the inventory level reaches to the upper control level (I^{**}). In Section 4.1, we numerically show that this optimal policy structure is also preserved for lost sales $M/Cox_2/1$ systems. On the other hand, the numerical studies in Section 4.2 illustrates the dynamic nature of the optimal policy for multi-server systems.

4.1. Single server systems

Single server cases are relatively easy to handle because at any state production decision u is either 0 or 1. For the numerical study, we first define a base case as $[\mu_1, \mu_2, \beta, h, \lambda, c] = [3.25, 1.75, 0.15, 3, 6, 3]$. In this subsection, we set $s = 1$ and provide the results while we are changing K or λ. We first assume $K = 0$ and represent the optimal production decisions in Table 1. Rows are for the first two state variables, which are $x_1 =$ the number of active servers at stage-one and $x_2 =$ the number of active servers at stage-two. The columns are for the last state variable, $x_3 =$ the inventory level. The numbers at the intersection of the row and the column axes represent the corresponding optimal decision.

Table 1. Optimal production policy: s=1, K=0

$u^*(x_1, x_2, x_3)$	x_3						
(x_1, x_2)	***0***	***1***	***2***	***3***	***4***	***5***	***6***
[0, 0]	1	1	1	1	0	0	0
[1, 0]	1	1	1	1	1	1	1
[0, 1]	0	0	0	0	0	0	0

The optimal decision at state (x_1, x_2, x_3), denoted by $u^*(x_1, x_2, x_3)$, is the optimal number of busy servers at phase-1 as explained in Section 3. For instance, $u^*(0, 0, 0) = 1$ implies that the server, which is currently idle, should be activated if the inventory level is zero. Since $K = 0$ and continuation decisions are redundant, Table 1 fully characterizes the optimal policy. For single server systems the decision is trivial at states $(1,0,x_3)$ and $(0,1,x_3)$. At such states the server is already busy (there is no idle server to activate) and the decision is automatically x_1. Hence, control is only for the states of $(0,0,x_3)$ type. Table 1, which is an example case for our extensive numerical study, shows that the optimal production policy is of base stock type: it is optimal to produce below the maximum inventory level (base stock level) and not to produce otherwise. For the setting considered in the table, optimal base stock level, BS^*, is 4.

However, when there is fixed start-up cost optimal production policy cannot be described with a single parameter as Gavish and Graves [4] shows for single-server backordering systems. As exemplified in Table 2, our numerical studies depict that *two-critical-number policy* is optimal also for lost sales systems. In the u^*part of the table, it is seen that production is started when inventory level drops to 2. When $K > 0$, in addition to the number of active servers decision, the continuation decisions are also required and provided in c_1^* and c_2^* parts of Table 2. Referring to the DP model of Section 3, we define c_1^* and c_2^* as follows: $c_1^* = 1$ if $min\{J(u-1, x_2, x_3+1), J(u, x_2, x_3+1)\} = J(u, x_2, x_3+1)$, i.e. it is optimal to immediately start new production at stage-1 when the whole production process has completed after stage-1 (without visiting stage-2). Similarly, $c_2^* = 1$ if $min\{J(u, x_2-1, x_3+1), J(u+1, x_2-1, x_3+1)\} = J(u+1, x_2-1, x_3+1)$ corresponding to the event where the production process has completed after stage-2 and it is optimal to immediately start new production on stage-1 of the process. Otherwise, the server that has just finished processing is turned-off and, c_1^* and c_2^* are set to zero. By definition, $c_1^*(x_1, x_2, x_3)$ and $c_2^*(x_1, x_2, x_3)$ are relevant only when $x_1 > 0$ and $x_2 > 0$, respectively. Otherwise continuation decisions are not applicable (NA). As seen from Table 2, $c_1^*(1,0,x_3) = c_2^*(0,1,x_3)$ for all inventory levels x_3. This holds because there is only one available server and the inventory level just after production completion would be the same independent of the last stage visited. If it is optimal to continue production on the server, then the new process is going to start at stage-1 in any case.

For the setting considered in Table 2 the parameters of the *two-critical-number policy* are $(I^*, I^{**}) = (2, 6)$ where I^* is the production trigger level and I^{**} is the maximum inventory level that the system reaches.

Table 2. Optimal production decisions, s=1, K=2

$u^*(x_1, x_2, x_3)$	x_3						
(x_1, x_2)	***0***	***1***	***2***	***3***	***4***	***5***	***6***
[0, 0]	1	1	1	0	0	0	0
[1, 0]	1	1	1	1	1	1	1
[0, 1]	0	0	0	0	0	0	0
$c_1^*(x_1, x_2, x_3)$	x_3						
(x_1, x_2)	***0***	***1***	***2***	***3***	***4***	***5***	***6***
[0, 0]	NA	NA	NA	NA	NA	NA	NA
[1, 0]	1	1	1	1	1	1	0
[0, 1]	NA	NA	NA	NA	NA	NA	NA
$c_2^*(x_1, x_2, x_3)$	x_3						
(x_1, x_2)	***0***	***1***	***2***	***3***	***4***	***5***	***6***
[0, 0]	NA	NA	NA	NA	NA	NA	NA
[1, 0]	NA	NA	NA	NA	NA	NA	NA
[0, 1]	1	1	1	1	1	1	0

After the characterization of the optimal policy we next show in Table 3 how the optimal policy parameters react to changes in traffic intensity and production start-up cost. We change the demand rate while keeping Coxian processing time parameters constant to obtain settings with different traffic intensity (ρ). The effect of Coxian parameters is discussed in Section 5.

Table 3 reveals that the optimal policy parameters are non-decreasing in ρ. At lower ρ values system prefers not to produce at all. That is, the optimal values of the policy parameters are all zero and corresponding average system cost equals λc. On the other hand, it is optimal to produce at some inventory levels beyond certain traffic intensity and thus BS^* and the vector (I^*, I^{**}) are not zero for $K = 0$ and $K > 0$ cases, respectively. When $K > 0$, as the traffic getting heavier the increase in I^{**} is more pronounced than the increase in I^* because the system needs to hold more inventory to meet the increasing demand. For fixed ρ, a similar behavior is observed as the start-up cost K increases: in order to decrease the frequency of production start-up (so the total fixed cost) and to continue with the activated server as much as possible, the gap between the maximum inventory level and the production trigger point, $I^{**} - I^*$, is getting wider.

Table 3. Optimal policy parameters, $s = 1$

		$K = 0$		$K = 1$			$K = 2$			$K = 3$		
λ	**ρ**	**AC**	BS^*	**AC**	I^*	I^{**}	**AC**	I^*	I^{**}	**AC**	I^*	I^{**}
0.50	0.29	1.50	0	1.50	0	0	1.50	0	0	1.50	0	0
0.75	0.43	2.25	0	2.25	0	0	2.25	0	0	2.25	0	0
1.00	0.58	3.00	1	3.00	0	0	3.00	0	0	3.00	0	0
1.50	0.86	3.70	1	4.50	0	1	4.50	0	0	4.50	0	0
2.00	1.15	4.61	1	5.21	0	2	5.56	0	2	5.86	0	3
2.50	1.44	5.66	1	6.06	0	2	6.29	0	3	6.46	0	3
3.00	1.73	6.66	2	6.95	0	3	7.11	0	3	7.19	0	4
3.50	2.01	7.73	2	7.94	1	4	8.00	1	4	8.05	0	5
4.00	2.30	8.90	2	8.99	1	4	9.02	1	5	9.04	1	5

4.2. Multi-server systems

Although the structure of the optimal production policy is known for single server make-to-stock systems, it has not yet been fully characterized for multiple server systems. To the best of our knowledge, the only study addressing the production control of multi-server systems is Bulut and Fadıloğlu [8] and they only provide partial characterization of the policy for the $M/M/s$ case without fixed cost. In this section, we provide numerical analyses to describe the structure of the optimal policy for the $M/Cox_2/s$ make-to-stock systems with fixed cost for the first time in the literature. Single server assumption relatively eliminates the complexity because for such cases the decision is 0-1 for all inventory levels: whether to activate the only available server or not. However, when $s > 1$, the controller should decide how many servers should be active at any system state. Furthermore, this decision would be dependent on the status of the ongoing production, i.e. to the stage/phase information of the active servers.

Recalling the base case, we first set $s = 3$ and $K = 0$ and provide the optimal decisions in Table 4(a). Similar to the single-server case, u^* matrix is enough to describe the optimal policy when the production start-up cost is zero. We separate the decision matrix into four layers where each layer corresponds to a particular total number of active servers. In general, if there are s available servers, there would be $(s + 1)$ layers. For the setting presented in Table 4(a), we list our observations on the structure of the optimal policy below:

i. Since all the available servers are busy at the bottom layer, i.e. $x_1 + x_2 = s = 3$, and order cancellation cost is practically infinite, the optimal decision is trivial at all states of the bottom layer: $u^*(x_1, x_2, x_3) = x_1$.

ii. Production decisions are non-increasing in inventory level x_3, because shortage risk is reduced by increasing inventory.

iii. Unlike the classical static policies, which are based on either inventory level or inventory position (e.g. base stock), unit increase in inventory level does not always end up with unit decrease in the optimal number of active servers at phase-one, e.g. $u^*(0,0,1) = 2$ but $u^*(0,0,2) = 0$.

iv. In addition to (*iii*), there is a second level of dynamicity in the structure of the optimal policy; decisions are dependent on the status of ongoing production. One would expect that as the number of completed production stages increases, the total number of active servers decreases or remains the same. This is true for the processing time random variables having increasing failure rate (IFR) such as Erlangian production times. For such settings, as the number of completed stages increases remaining time to replenish the inventory stochastically decreases. However, for the case considered in Table 4(a), Coxian production time random variable has the parameters $(\mu_1, \mu_2, \beta) = (3.25, 1.75, 0.15)$ and more channels are needed if the item being processed visits stage-2. Since stage-1 is much faster than stage-2 and probability of visiting stage-2 is small, expected time to production completion is smaller when the current production is at stage-1 compared to the case where it is at stage-2. In order to make it clearer, let us consider the states $(1,0,1)$ and $(0,1,1)$ of Table 4(a): the inventory level is the same for both of the states but the (only) active server is at stage-1 in the first state and at stage-2 in the second. As seen from the table, $u^*(1,0,1) = u^*(0,1,1) = 2$ and the transitions are to $(2,0,1)$ and $(2,1,1)$ from $(1,0,1)$ and $(0,1,1)$, respectively. Before the decisions, both states have the same number of active servers, which is one, but after the transitions state $(2,1,1)$ has one more active server than $(2,0,1)$.
We increase μ_2 from 1.75 to 7.5 in Table 4(b) and observe completely different production decisions for the states $(1,0,1)$ and $(0,1,1)$: $u^*(1,0,1) = 2$, $u^*(0,1,1) = 0$ an the transitions are to $(2,0,1)$ and $(0,1,1)$ from $(1,0,1)$ and $(0,1,1)$, respectively That is, this time it is optimal to have more active servers when the current production is at phase-1.

v. With its three parameters Coxian production time random variable has the flexibility to obtain increasing and decreasing failure rate (IFR or DFR) settings and, we show in Table 4 (a) and (b) that the structure of the optimal policy changes accordingly.

Table 4. Optimal production decisions s=3, K=0
(a) DFR, (b) IFR

(a) u^*	x_3						(b) u^*	x_3					
(x_1, x_2)	***0***	***1***	***2***	***3***	***4***	***5***	(x_1, x_2)	***0***	***1***	***2***	***3***	***4***	***5***
[0, 0]	3	2	0	0	0	0	**[0, 0]**	3	2	0	0	0	0
[1, 0]	3	2	1	1	1	1	**[1, 0]**	3	2	1	1	1	1
[0, 1]	2	2	0	0	0	0	**[0, 1]**	2	0	0	0	0	0
[2, 0]	3	2	2	2	2	2	**[2, 0]**	3	2	2	2	2	2
[1, 1]	2	2	1	1	1	1	**[1, 1]**	2	1	1	1	1	1
[0, 2]	1	1	0	0	0	0	**[0, 2]**	1	0	0	0	0	0
[3, 0]	3	3	3	3	3	3	**[3, 0]**	3	3	3	3	3	3
[2, 1]	2	2	2	2	2	2	**[2, 1]**	2	2	2	2	2	2
[1, 2]	1	1	1	1	1	1	**[1, 2]**	1	1	1	1	1	1
[0, 3]	0	0	0	0	0	0	**[0, 3]**	0	0	0	0	0	0

Fixed production cost adds more complexity to the structure of the optimal policy. To reveal this, one can compare Table 4(a) and Table 5 where the only difference is the value of the start-up cost K. When fixed cost is larger optimal policy tends to activate less servers at all the states. Specifically, fixed cost prevents activating all the available servers even there is no inventory on hand. On the other hand, the optimal policy balance the holding and shortage trade-off mostly with the continuation decisions c_1^* and c_2^*; production continues with the previously activated servers for some time. However, continuation decisions are also state dependent and are not only determined by the total number of active servers. As opposed to the single-server case shown in Table 2, there exists a, b and x_3 values such that $c_1^*(a, b, x_3) \neq c_2^*(b, a, x_3)$. For instance, $c_1^*(2,0,3) = 0$ but $c_2^*(0,2,3) = 1$. This dynamic behavior of the optimal policy is due to the fact that any active server at stage-1 can replenish the inventory by two different realizations: with probability $(1 - \beta)$ inventory is replenished directly from stage-1, but with probability β stage-2 is visited and then the production is completed. On the other hand, any active server at stage-2 has only one possible realization path to replenish the inventory. Hence, when $s > 1$, continuation decisions are coupled with the number of active servers decision and depending on the values of the Coxian parameters (μ_1, μ_2, β), continuation decisions might be different even for the symmetric states at the same inventory level.

Table 5. Optimal production decisions, s=3 and K=2

u^*	x_3						c_1^*	x_3						c_2^*	x_3					
(x_1, x_2)	***0***	***1***	***2***	***3***	***4***	***5***	(x_1, x_2)	***0***	***1***	***2***	***3***	***4***	***5***	(x_1, x_2)	***0***	***1***	***2***	***3***	***4***	***5***
[0, 0]	2	1	0	0	0	0	**[0, 0]**	NA	NA	NA	NA	NA	NA	**[0, 0]**	NA	NA	NA	NA	NA	NA
[1, 0]	2	1	1	1	1	1	**[1, 0]**	1	1	1	1	1	0	**[1, 0]**	NA	NA	NA	NA	NA	NA
[0, 1]	1	0	0	0	0	0	**[0, 1]**	NA	NA	NA	NA	NA	NA	**[0, 1]**	1	1	1	1	1	0
[2, 0]	2	2	2	2	2	2	**[2, 0]**	1	1	1	0	0	0	**[2, 0]**	NA	NA	NA	NA	NA	NA
[1, 1]	1	1	1	1	1	1	**[1, 1]**	1	1	1	1	0	0	**[1, 1]**	1	1	1	0	0	0
[0, 2]	0	0	0	0	0	0	**[0, 2]**	NA	NA	NA	NA	NA	NA	**[0, 2]**	1	1	1	1	0	0
[3, 0]	3	3	3	3	3	3	**[3, 0]**	1	1	0	0	0	0	**[3, 0]**	NA	NA	NA	NA	NA	NA
[2, 1]	2	2	2	2	2	2	**[2, 1]**	1	1	0	0	0	0	**[2, 1]**	1	1	0	0	0	0
[1, 2]	1	1	1	1	1	1	**[1, 2]**	1	1	0	0	0	0	**[1, 2]**	1	1	0	0	0	0
[0, 3]	0	0	0	0	0	0	**[0, 3]**	NA	NA	NA	NA	NA	NA	**[0, 3]**	1	1	0	0	0	0

As the discussion on tables 4(a), 4(b) and 5 exhibits, for the multi-server systems, optimal policy is highly dynamic/state-dependent and cannot be fully described with two static parameters such as inventory level or inventory position. The highly dynamic structure of the optimal policy would reduce its value for practitioners. In practice, controllers are mostly after easy-to-apply approximate policies. We therefore propose an alternative production policy that is controlled by two parameters and can quickly adapt itself to IFR and DFR cases. Next two sections are devoted to the introduction and performance evaluation of our policy.

5. An alternative policy structure

As we have discussed in Section 4, the values of the Coxian parameters directly affect the structure of the optimal policy. In order to first guarantee that our policy structure responds to the changes in input parameters (μ_1, μ_2, β), we define $E(1,0,x_3)$ and $E(0,1,x_3)$ as the expected remaining production times if the current production is on stage-1 and stage-2, respectively. Since the stages are memoryless, $E(1,0,x_3) = \frac{1}{\mu_1} + \beta \frac{1}{\mu_2}$ and $E(0,1,x_3) = \frac{1}{\mu_2}$. We aim to identify IFR and DFR cases by comparing these expected times to production completion. $E(0,1,x_3) < E(1,0,x_3)$ or equivalently $r = \frac{E(0,1,x_3)}{E(1,0,x_3)} < 1$ implies that expected remaining production time decreases when stage-2 is visited. For such settings, since the probability of demand arrivals before inventory replenishment decreases, our policy should demotivate activating new servers when stage-2 is visited. Otherwise, we are in a DFR case and the policy should motivate (or at least should not demotivate) activating new servers when stage-2 is visited.

Second, for the sake of applicability we aim to propose a policy structure that can be controlled by only two parameters. We stick to the notation used in Section 4: I^* and I^{**} are the production trigger and the maximum levels, respectively. This two-critical-number policy is optimal for single-server settings and the parameters of the policy are defined in terms of inventory level. So as to better capture the dynamic nature of the optimal policy of an $M/Cox_2/s$ make-to-stock system, we define I^* and I^{**} in terms of a function of the system state vector referred as *inventory status* (*IS*). At any state (x_1, x_2, x_3),

$$IS(x_1, x_2, x_3) = \sum_{i=1}^{3} a_i x_i \tag{4}$$

where a_1, a_2 and a_3 are the weights of the number of active servers at stage-1, the number of active servers at stage-2 and inventory level, respectively. The above definition of inventory status allows us to trace a policy space including the classical inventory level ($IS = IL$ when $(a_1, a_2, a_3) = (0,0,1)$) and inventory position ($IS = IP$ when $(a_1, a_2, a_3) = (1,1,1)$) based policies.

Based on the above discussion, we propose the below policy structure that computes $u(x_1, x_2, x_3)$ = the number of active/busy servers at stage-1, $c_1(x_1, x_2, x_3)$ = continuation decision for the server that has just finished stage-1 and replenished inventory, and $c_2(x_1, x_2, x_3)$ = continuation decision for the server that has just finished stage-2 and replenished inventory.

$$u(x_1, x_2, x_3) = \begin{cases} \lfloor min\{(I^* + 1) - IS(x_1, x_2, x_3) + x_1, (s - x_2)\}\rceil, & IS \le I^* \\ x_1, & IS > I^* \end{cases} \tag{5}$$

$$c_1(x_1, x_2, x_3) = \begin{cases} 1, & IS(x_1 - 1, x_2, x_3 + 1) < I^{**} \\ 0, & IS(x_1 - 1, x_2, x_3 + 1) \ge I^{**} \end{cases} \tag{6}$$

$$c_2(x_1, x_2, x_3) = \begin{cases} 1, & IS(x_1, x_2 - 1, x_3 + 1) < I^{**} \\ 0, & IS(x_1, x_2 - 1, x_3 + 1) \ge I^{**} \end{cases} \tag{7}$$

For the states whose inventory status is at or below the production trigger level I^*, the proposed policy tries to raise IS to $(I^* + 1)$. This can only be achieved with $(I^* + 1) - IS(x_1, x_2, x_3)$ many new active servers at stage-1 additional to x_1. However, as discussed in the dynamic programming formulation of Chapter 3, $u(x_1, x_2, x_3)$ is bounded above by $(s - x_2)$. In Equation (5), $\lfloor\cdot\rceil$ is to return the nearest integer for the calculated value as the number of active servers at stage-1. For the other states, $IS(x_1, x_2, x_3) > I^*$ and we do no nothing: $u(x_1, x_2, x_3)$ returns the current number of busy servers at stage-1.

Continuation decisions of the policy are defined by (6) and (7), which are only applicable when $x_1 > 0$ and $x_2 > 0$, respectively. The policy keeps the previously activated servers busy until target level I^{**} is reached. In (6) and (7), decisions are given just after production completion (in c_i, x_i is decreased by 1, i = 1,2) and inventory replenishment (x_3 is increased by 1).

The above defined policy structure has three weight and two control parameters: (a_1, a_2, a_3) and (I^*, I^{**}). First we develop the following approach to find the setting specific values of the weights (a_1, a_2, a_3): We structure our policy based on the relative values of $a_i{}'$s. Thereby, the degrees of freedom of finding the values of $a_i{}'$s is decreased to two. Without loss of generality, we set the value of an active server at stage-1 to 1, i.e.,

$$a_1 = 1 \tag{8}$$

Then, the weight of an on-hand inventory relative to the weight of an outstanding order at stage-1 is calculated as:

$$a_3 = \frac{\frac{1}{\mu_1} + \beta\frac{1}{\mu_2}}{\frac{1}{\mu_1}} = \frac{\beta\mu_1 + \mu_2}{\mu_2} \tag{9}$$

where $\frac{1}{\mu_1} + \beta\frac{1}{\mu_2}$ and $\frac{1}{\mu_1}$ are the expected time to complete the whole production and stage-1, respectively. That is, if an item at stage-1, which is going to spend (on the average) $\frac{1}{\mu_1}$ time units to complete the stage, has the weight a_1, then the relative weight of an item in the inventory, which has on the average spent $\frac{1}{\mu_1} + \beta\frac{1}{\mu_2}$ time units in the production facility, is $\frac{\frac{1}{\mu_1} + \beta\frac{1}{\mu_2}}{\frac{1}{\mu_1}}$ times a_1. The weight of an item at stage-2, a_2, on the other hand, is set to different values for IFR and DFR cases. As discussed in Section 4, depending on the values of Coxian parameters (μ_1, μ_2, β) more active servers might be needed if the item being processed visits stage-2. Our policy structure gains this flexibility with a_2. We let

$$a_2 = \begin{cases} \frac{a_1 + a_3}{2}, & if\ r = \frac{E(0,1,x_3)}{E(1,0,x_3)} < 1 \\ 0, & if\ r \ge 1 \end{cases} \tag{10}$$

The ratio $r = \frac{E(0,1,x_3)}{E(1,0,x_3)}$ is less than 1 if the expected time to production completion decreases when stage-2 is visited. For such IFR cases, we set the weight of an outstanding order at stage-2 to the average of the weights of the items that are at stage-1 and in the inventory. In this way, for the IFR cases we obtain a weight structure satisfying $a_1 < a_2 < a_3$.

On the other hand, if the case is DFR, the weight is set to 0 in order to motivate the system to activate more servers whenever the slower stage (stage-2) is visited. In this case, $a_2 < a_1 < a_3$.

One can prefer to select the "best" values of (a_1, a_2, a_3) using an optimization routine applied over the DP formulation. However, the next section shows that the performance of the proposed structure under our *intelligent guesses* (8), (9) and (10) is very close to the optimal's. That is, without undertaking the computation cost of any optimization algorithm, we obtain a very good approximation to the "ideal" weights by exploring the structure of the optimal policy (in Section 4) and selecting the values accordingly.

On the other hand, it is hard to develop a similar intuition for the control levels (I^*, I^{**}). We therefore use our DP formulation (2) and (3) as the optimization routine: For given values of (I^*, I^{**}) such that $I^* < I^{**}$, DP is fed with the decision set of the proposed policy, (5), (6) and (7), and the value iteration algorithm is run to calculate the average system cost. We then search for the optimal values of the parameters in the integer domain. As the results presented in Section 6 show that optimizing (I^*, I^{**}) in the integer space results in a well-performing near-optimal policy.

It should be noted that a_1 can also be set to any arbitrary positive value. In such cases, the values of a_2, a_3 and thus *IS* would also be changed relative to a_1. Hence, the optimal values of (I^*, I^{**}) would be also altered/shifted in order to find the same cost minimizer $u(x_1, x_2, x_3)$ values. That is, larger values of a_i's would result in larger values of (I^*, I^{**}) so as to find the same u value.

Although we obtain the results in a reasonable amount of time one can further fasten the routine if the search first visits the space around $(I^{**} - I^*) = EOQ$. The approximate value of the batching decision of the classical inventory systems would here help us to capture the effect of the fixed cost on the length of the non-production period. One should note that our make-to-stock production-inventory environment is different than the classical inventory settings in terms of capacity (there are only *s* many servers) and one-at-a-time replenishment as the active servers complete production. A classical inventory system having stochastic lead-times that is controlled by lot-for-lot policy can be modeled using our approach only if *s* tends to infinity, which requires to guarantee an uncapacitated system.

6. Performance evaluation of the proposed policy

In this section we present the numerical study assessing the performance of the policy structure described by (5), (6) and (7). We test the performance of the structure with the (a_1, a_2, a_3) values given in (8), (9) and (10), which defines the specific policy that we propose. We also evaluate the performance of the inventory position based static policy (IP Policy) in order to quantify the value of dynamic state information. Our policy structure already has the flexibility to cover the IP Policy: in (4), we let $(a_1, a_2, a_3) = (1,1,1)$ to obtain the inventory position as the sum of the on-hand inventory and the number of outstanding production orders (the number of items that are being processed).

The main goal of this section is to reveal the effects of Coxian parameters (μ_1, μ_2, β), the demand rate λ and the fixed cost *K* on the performances of the considered policies. The results of the numerical study are summarized in the tables provided at the end of the section. While changing the above mentioned parameters, without loss of generality we fix the values of the holding cost rate *h* and the unit lost sales cost *c* to 3. Each table includes five different instances with different traffic intensities (ρ) ranging from 0.50 to 1.50. For each instance, average costs of the optimal, proposed and IP policies, and their optimal control levels (I^*, I^{**}) are reported. For the proposed and IP policies, the optimality gap, defined as the percent cost deviation from the optimal, is also provided.

Table 6 shows the results when there are two parallel servers with no start-up costs and Coxian production times have decreasing failure rates (DFR). Our dynamic policy performs very well in the environment of Table 6. The optimality gap of the proposed policy is less than 0.5% at all the instances of the table. Furthermore, IP Policy is also a notable alternative of the optimal policy when the capacity is tight: when there are limited of number of servers or traffic intensity is high. As ρ increases or equivalently as the capacity is getting tighter, more and more servers would be activated independent of the status of the production. That is, all the plausible policies, including the optimal one, utilize all the servers at higher traffic intensity values. This observation is valid not just for Table 6 but for all the tables of the chapter: The proposed policy is near optimal at all the traffic intensities of all the considered cases, and IP Policy is a second alternative for the highly utilized systems.

Table 7 and Table 8 are the 'positive fixed cost' and 'more server' extensions of Table 6, respectively. When the production startup cost K is 0.5, the maximum optimality gap of the Proposed Policy is 2.55% and of the IP Policy is 3.22%. Both maximums are observed at the same instance where $\rho = 0.5$. For the systems with higher server activation cost, the distance between the upper and lower control limits, which is $(I^{**} - I^*)$, should be larger. That is, instead of activating servers at higher inventory levels it is more economical to increase I^{**} in order to both postpone the production cycle and to continue production on the previously activated servers once it is started until reaching I^{**} again.

On the other hand, in Table 8, the number of available servers is higher (and fixed cost is zero) and the maximum deviations from the optimality are 3.21% and 6.89% for the Proposed and IP policies, respectively. However, for the Proposed Policy, the average of the five optimality gaps reported in the table is only 1.54%, which is the highest average among all the tables. Since the Proposed Policy makes use of the dynamic production status information of all the servers, it outperforms the static IP policy as the number of available servers increases.

Table 9 is for the IFR version of Table 8. The table shows that the optimality gap of the Proposed Policy is below 1% at all the instances. Although the performance of IP Policy is also improved from Table 8 to Table 9, that improvement is not as significant as the improvement attained by the Proposed Policy.

At all the tables from 6 to 9, the traffic intensity (ρ) is increased by decreasing μ_1, the processing rate of stage-1. In order to eliminate any bias that can be due to this method, we reconsider the environment of Table 8 where $s = 5$ and $K = 0$, and increase the traffic intensity by decreasing μ_2 this time. The results are reported in the last table, Table 10. The probability of a WIP item being at phase-2 increases as μ_2 decreases that sharpens the DFR nature of the production times. Due to this fact, the performance of our policy is better than Table 8 in Table 10.

Table 6. Performances of the alternative policies: DFR cases with s=2 and K=0

			Optimal Policy	IP Policy			Proposed Policy		
#	ρ	(μ_1, μ_2, β)	Average Cost	Average Cost	(I^*, I^{**})	Optimality Gap %	Average Cost	(I^*, I^{**})	Optimality Gap %
1	0.5	(15, 0.5, 0.05)	7.05	7.20	(1, 2)	2.03	7.09	(1, 7)	0.48
2	0.75	(6.65, 0.5, 0.05)	8.21	8.49	(1, 2)	3.32	8.21	(2, 3)	0.00
3	1.00	(4.25, 0.5, 0.05)	9.19	9.26	(2, 3)	0.70	9.24	(2, 5)	0.45
4	1.25	(3.15, 0.5, 0.05)	9.98	10.01	(2, 3)	0.27	9.99	(3, 4)	0.09
5	1.50	(2.50, 0.5, 0.05)	10.70	10.72	(2, 3)	0.19	10.71	(3, 4)	0.10

Table 7. Performances of the alternative policies with DFR distribution, s=2 and K=0.5

			Optimal Policy	IP Policy			Proposed Policy		
#	ρ	(μ_1, μ_2, β)	Average Cost	Average Cost	(I^*, I^{**})	Optimality Gap %	Average Cost	(I^*, I^{**})	Optimality Gap %
1	0.5	(15, 0.5, 0.05)	8.82	9.12	(1, 5)	3.22	9.06	(1, 8)	2.55
2	0.75	(6.65, 0.5, 0.05)	9.71	9.79	(1, 6)	0.80	9.72	(2, 6)	0.03
3	1.00	(4.25, 0.5, 0.05)	10.24	10.28	(1, 6)	0.45	10.32	(1, 7)	0.80
4	1.25	(3.15, 0.5, 0.05)	10.77	10.78	(1, 7)	0.16	10.80	(2, 8)	0.30
5	1.50	(2.50, 0.5, 0.05)	11.20	11.21	(2, 7)	0.09	11.20	(2, 8)	0.02

Table 8. Performances of the Alternative Policies with DFR distribution, s=5 and K=0

			Optimal Policy	IP Policy			Proposed Policy		
#	ρ	(μ_1, μ_2, β)	Average Cost	Average Cost	(I^*, I^{**})	Optimality Gap %	Average Cost	(I^*, I^{**})	Optimality Gap %
1	0.5	(2.79, 0.5, 0.05)	7.85	8.43	(3, 4)	6.89	7.96	(3, 4)	1.47
2	0.75	(1.90, 0.5, 0.05)	8.47	8.92	(4, 7)	5.01	8.75	(3, 4)	3.21
3	1.00	(1.35, 0.5, 0.05)	9.29	9.43	(4, 5)	1.47	9.38	(4, 5)	0.99
4	1.25	(1.06, 0.5, 0.05)	9.97	10.09	(5, 8)	1.20	10.11	(5, 6)	1.37
5	1.50	(0.87, 0.5, 0.05)	10.67	10.70	(5, 8)	0.32	10.74	(5, 8)	0.68

Table 9. Performances of the Alternative Policies with IFR distribution, s=5 and K=0

			Optimal Policy	IP Policy			Proposed Policy		
#	ρ	(μ_1, μ_2, β)	Average Cost	Average Cost	(I^*, I^{**})	Optimality Gap %	Average Cost	(I^*, I^{**})	Optimality Gap %
1	0.5	(8.50, 2.65, 0.8)	7.97	8.33	(3, 6)	4.26	8.05	(7, 12)	0.99
2	0.75	(3.90, 2.65, 0.8)	8.46	8.81	(3, 6)	3.93	8.52	(5, 9)	0.75
3	1.00	(1.88, 2.65, 0.8)	9.27	9.40	(4, 7)	1.39	9.33	(6, 8)	0.62
4	1.25	(1.35, 2.65, 0.8)	9.88	9.94	(5, 7)	0.58	9.91	(6, 7)	0.24
5	1.50	(1.05, 2.65, 0.8)	10.52	10.57	(5, 7)	0.48	10.55	(7, 8)	0.26

Table 10. Performances of the Alternative Policies with DFR distribution, s=5, K=0 and μ_2 is changing

			Optimal Policy	IP Policy			Proposed Policy		
#	ρ	(μ_1, μ_2, β)	Average Cost	Average Cost	(I^*, I^{**})	Optimality Gap %	Average Cost	(I^*, I^{**})	Optimality Gap %
1	0.5	(14, 2.30, 0.8)	7.83	8.33	(3, 6)	6.00	7.93	(4, 11)	1.32
2	0.75	(14, 1.44, 0.8)	8.44	8.92	(4, 7)	5.34	8.51	(4, 16)	0.81
3	1.00	(14, 1.05, 0.8)	9.24	9.40	(4, 7)	1.73	9.28	(12, 23)	0.49
4	1.25	(14, 0.82, 0.8)	10.02	10.13	(4, 7)	1.10	10.02	(16, 28)	0.06
5	1.50	(14, 0.68, 0.8)	10.74	10.84	(5, 8)	0.92	10.74	(19, 33)	0.02

As the holding cost rate (h) and unit lost sales cost (c) are both set to 3 in all the above examples, we aim to depict the effect of changes in the (holding cost rate)/(unit lost sales cost) ratio in Figure 2. In order to visit different values of this ratio, without loss of generality we only change holding cost rate: h varies from 1 to 14 while c is kept constant at 3. The other parameters are assumed to be $[s, \mu_1, \mu_2, \beta, K, \lambda] = [2, 4.25, 0.5, 0.05, 0.5, 6]$. In the figure, for each increment of h, average system cost of both the optimal and proposed policies, and (I^*, I^{**}) values of the proposed policy are presented. As seen from the height of the bars representing the average system costs of the policies the performances are so close to each other: average and maximum optimality gaps are calculated as 0.25% and 0.90%, respectively. Furthermore, for both of the policies average system cost is concave in h that converges to a certain value (18) when h is above 12. As h increases and becomes larger relative to c, both policies demotivate production. In our example, when $h > 12$ it is optimal not to produce at all. In this case all the incoming demands are lost and the average system cost converges to $\lambda c = 18$ for both of the policies. In parallel to this observation, it is also seen from the figure that both of the optimal control parameters of the proposed policy, which are defined by (I^*, I^{**}), are non-increasing in h. Equivalently we can say that they would be non-decreasing in c. On the hand, when holding cost rate is getting smaller and smaller (compared to the unit lost sales cost) the average cost converges to zero. As h decreases both I^* and I^{**} increase in order to minimize shortage and start-up costs. At the extreme, when $h = 0$, I^* can be set to any value above a threshold that guarantees no shortage. Similarly, I^{**} can be any value (greater than I^*) such that fixed cost per each server is incurred only finitely many times. For all such control levels the long-run average system cost would be zero.

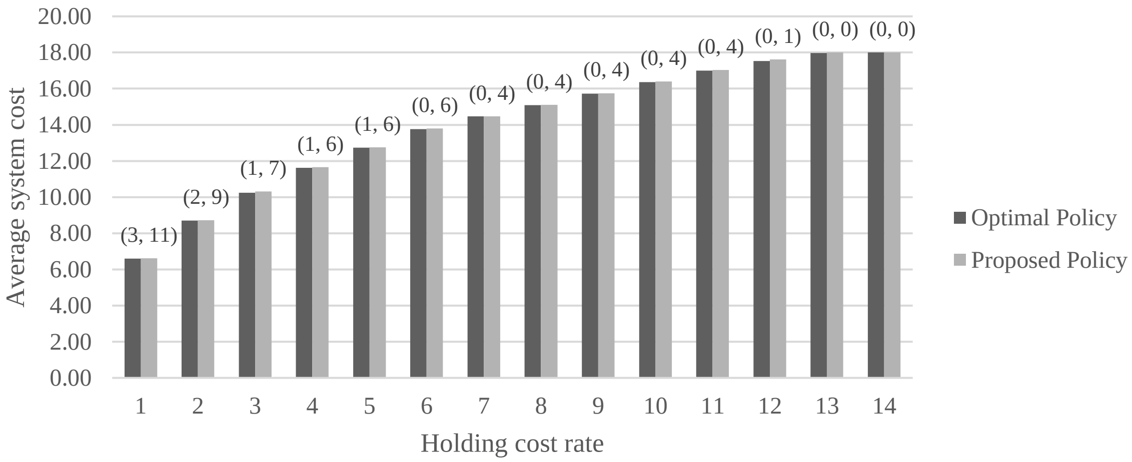

Figure 2. Effects of changes in *h* on the average cost and (I^*, I^{**})

7. Conclusion

This article considers a production-inventory system in a make-to-stock environment with multiple identical production channels (machines, servers or lines), fixed production start-up costs and lost sales. We assume that production times are 2-phase Coxian random variables that allows us to model rework/remanufacturing and repair operations within the production process. Demands are generated according to a stationary Poisson process and unsatisfied demands are immediately lost.

We extend the existing literature by considering phase-type production times and multiple servers with start-up costs in the same model. The system is modeled as an $M/Cox_2/s$ make-to-stock queue and dynamic programming formulation is developed. Thereafter, we first numerically characterize the optimal production policy and reveal that it has a highly dynamic nature. Secondly, we propose a policy structure that aims to capture the dynamic nature of the optimal policy with two control and three weight parameters. Control parameters are to define the maximum inventory and the production start-up levels. The other three parameters are the weights of the number of active servers at stage-1, the number of active servers at stage-2 and the number of items in inventory. This policy structure has the capability to trace a large space of several different policies. Using this structure we specifically propose a policy with fixed weight parameters and test its performance with respect to the optimal. Results reveal that our policy, which is controlled by only two parameters and thus easy-to-apply, is near optimal at all the instances.

Acknowledgments

This work was supported by The Scientific and Technological Research Council of Turkey – TÜBİTAK as a part of the project 213M355.

References

[1] Gavish, B., Graves, S.C., (1980), A one-product production/inventory problem under continuous review policy, *Operations Research*, 28(5), 1228-1236.

[2] Heyman, D. P., (1968). Optimal operating policies for M/G/1 queuing systems, *Operations Research*, 16(2), 362-382.

[3] Sobel, M. J., (1969), Optimal average-cost policy for a queue with start-up and shut-down costs, *Operations Research*, 17(1), 145-162.

[4] Gavish, B., Graves, S.C., (1981), Production/Inventory Systems with a Stochastic Production Rate under a Continuous Review Policy, *Computers & Operations Research*, 8(3), 169-183.

[5] Graves, S. C., Keilson, J., (1981), The compensation method applied to a one-product production/inventory problem. *Mathematics of Operations Research*, 6(2), 246-262.

[6] Lee, H.S., Srinivasan, M.M., (1989), The Continuous Review (s, S) Policy for Production/Inventory Systems with Poisson Demands and Arbitrary Processing Times, *Technical Report*, 87-33.

[7] Lee, H.S., Srinivasan, MM., (1991), Random Review Production/Inventory Systems with Compound Poisson Demands and Arbitrary Processing Times, *Management Science*, 37(7), 813-833.

[8] Bulut, Ö., & Fadiloğlu, M. M., (2011), Production control and stock rationing for a make-to-stock system with parallel production channels, *IIE Transactions*, 43(6), 432-450.

[9] Ha, A. Y., (1997a), Inventory rationing in a make-to-stock production system with several demand classes and lost sales. *Management Science*, 43(8), 1093-1103.

[10] Ha, A.Y., (1997b), Stock rationing policy for a make-to-stock production system with two priority classes and backordering, *Naval Research Logistics*, 43, 458-472.

[11] Ha, A. Y., (2000), Stock rationing in an M/Ek/1 make-to-stock queue, *Management Science*, 46(1) 77-87.

[12] Gayon, J. P., De Vericourt, F., Karaesmen, F., (2009), Stock rationing in an M/Er /1 multi-class make-to-stock queue with backorders, *IIE Transactions*, 41(12), 1096-1109.

[13] Rafiei, H., Rabbani, M., Vafa-Arani, H., & Bodaghi, G. (2017). Production-inventory analysis of single-station parallel machine make-to-stock/make-to-order system with random demands and lead times. *International Journal of Management Science and Engineering Management*, *12*(1), 33-44.

[14] Lippman, S., (1975), Applying a new device in the optimization of exponential queuing systems, *Operations Research*, 23, 687-710.

Özgün Yücel *earned her master's degree from the Industrial Engineering Department of Yaşar University. She has been working as a research assistant and continuing her PhD studies at the same department.*
http://orcid.org/0000-0003-0900-297X

Önder Bulut *is Assistant Professor in the Department of Industrial Engineering at Yaşar University. He received his Ph.D. degree from the Industrial Engineering Department of Bilkent University. His research interests include stochastic modeling, production and inventory systems, simulation, dynamic programming and optimal control.*
http://orcid.org/0000-0003-1476-6333