Optimal Control of a Nested-Multiple-Product Assemble-To-Order System

Mohsen ElHafsi
The A. Gary Anderson Graduate School of Management, University of California, Riverside, CA 92521-0203, USA, mohsen.elhafsi@ucr.edu

Herve Camus
Ecole Centrale de Lille
Cité Scientifique – BP 48, 59651 Villeneuve D’Ascq cedex, France, herve.camus@ec-lille.fr

Etienne Craye
Ecole Centrale de Lille
Cité Scientifique – BP 48, 59651 Villeneuve D’Ascq cedex, France, etienne.craye@ec-lille.fr

Abstract: In this paper, we study an assemble-to-order system consisting of \( n \) products assembled from a subset of \( m \) distinct components where the products have a modular nested design. \( i.e., \) product \( i \) has only one additional component more than product \( i-1 \). In particular, we study the optimal production and inventory allocation policies of such systems. Components are produced on independent production facilities one unit at a time, each with a finite production rate and exponentially distributed production times. The components are stocked ahead of demand and therefore incur a holding cost per unit per unit of time. Demand from each product occurs continuously over time according to a Poisson process. The demand for a particular product can be either satisfied (provided all its components are available in stock) or rejected. In the latter case, a product-dependent lost sale cost is incurred. In this situation, a manager is confronted with two decisions: when to produce a component and whether or not to satisfy an incoming product order from on-hand inventory. We show that, for the production of a component, the optimal policy is a base-stock type where the base-stock level depends on all other components’ inventory. We also show that, for inventory allocation, the optimal policy is a multi-level rationing policy where the rationing levels depend on all other components’ inventory. We propose a simple heuristic that we numerically compare against the optimal policy and show that, when carefully designed, it can be very effective.

Keywords: Assemble-to-order, nested-assembly structure, production and inventory control, Markov decision processes, make-to-stock queues
1 Introduction

In many manufacturing systems, modular design is used so that products are made of modules or components where upgraded versions of these products are obtained by only adding few additional modules or components. Examples of such systems include servers for local computer networks and servers for general network structures. In this case, the manufacturer offers a basic variety of servers and upgraded versions are obtained by adding additional circuit boards added to the server’s bays. The system we study can also model a situation where a main product can be sold either separately or combined with another auxiliary product. This could be the case of computers and printers where the demand for computers can induce a demand for printers (or digital cameras). This strategy is usually combined with an assemble-to-order (ATO) production strategy. ATO is a hybrid strategy where components and subassemblies are procured or produced and held in inventory, then assembled into final products as customer demand for specific combinations materializes. Therefore, a manager must decide what components to procure or produce well before final demand materializes. The focus of this paper is on components production and inventory allocation of these components. This strategy has become popular for manufacturing firms that seek to be both responsive and cost-efficient. It is particularly valuable when component supply lead-times are long relative to final assembly or when the supply processes are limited.

Despite their popularity, ATO systems are known to be difficult to analyze and manage. The challenge appears to rise from several factors including correlation of components’ demands, difference in components’ supply lead-times and the fact that order fulfillment is dependent on the availability of several components. As a result, it is difficult to manage components independently. Hence, deciding whether or not to fulfill demand for a particular product type cannot be carried out without considering the inventory level of all components.

In this paper, we consider a system consisting of multiple components assembled into several products using a nested product design. The components are produced one unit at a time on separate production facilities with finite production rates. Components are produced ahead of demand and held in stock, but incur a component-dependent holding cost. In order to satisfy demand for a particular product, all its components must be available. Otherwise, the demand is lost. A demand that is not immediately satisfied from stock incurs a shortage cost that can vary from product to product. A system manager must determine when to produce a particular component and, whenever an order for a particular product is placed, whether or not to satisfy it from on-hand inventory.

We use a Markov decision process (MDP) framework to formulate the problem and determine the structure of the optimal policy. We show that the optimal production policy for each component is a state-dependent base-stock policy, where the state of the system is specified by the vector of component inventory levels. We also show that the optimal inventory allocation policy is a multi-level rationing
policy. An order for a particular product is satisfied only if the current inventory level of each component is above a certain rationing level. The rationing levels for components with respect of the products are state-dependent as well. We also propose simple heuristics that we, numerically, compare their performance against the optimal policy.

The rest of this paper is organized as follows. In section 2, we offer a brief review of related literature. In section 3, we present our model. In section 4, we characterize the optimal policy. In section 5, we present the heuristic policies. In section 6, we present numerical results and comparisons of the heuristics against the optimal policy. In section 7, we offer a summary and few concluding comments.

2 Literature Review
The literature on ATO systems is quite extensive. The reader is referred to the comprehensive literature review of Song and Zipkin (2003). It can be classified into two categories: periodic and continuous review systems. For periodic review systems, most of the literature focuses on performance evaluation of heuristic policies; see for example Hausman et al. (1998), De Kok and Visschers (1999), Cheng et al. (2002), Zhang (1997), Agrawal and Cohen (2001), Frank et al. (2004) and references therein. The aforementioned literature viewed the system as a set of infinite server queues with correlated arrivals. Song (1998) develops an efficient algorithm for computing order fill rates for a system with Poisson demand and deterministic lead-times. Song and Yao (2002) extend this analysis to systems with uncertain component lead-times and study the impact of lead-time variability on the system performance. Gallien and Wein (2001) evaluate a policy consisting of a fixed and common base-stock level for all components and component-dependent order lead-times, such that a replenishment order for each component is synchronized with all the other components. In the above literature, component inventory has been typically assumed to be managed according to a fixed base-stock policy and the focus has been mainly on performance evaluation rather than to determine the optimal production policy. Recently, Benjaafar and Elhafsi (2006) studied a single-product ATO system with multiple demand classes. They showed that the optimal production policy is a state dependent base-stock policy and that the inventory policy is a state-dependent rationing policy.

Our paper fits in the stream of literature on ATO systems with continuous review and load-dependent supply lead-times. We consider an ATO system with similar characteristics to the ones treated in Benjaafar and Elhafsi (2006), Song et al. (1999) and Dayanik et al. (2003). Similar to these papers, we assume that demand forms a Poisson process, assembly is instantaneous, and components are produced on separate single-server production facilities with exponentially distributed production times.
3 Model Formulation

We consider a system consisting of $n$ products. Without loss of generality, we assume that Product $i$ ($i = 1, \ldots, n$) is assembled from components $i, i+1, \ldots, n$, where $n$ is the total number of components available for assembly of the products. For example, in a system with three components, Product 1 is assembled from components 1, 2 and 3, Product 2 is assembled from components 2 and 3 and Product 3 is assembled from Component 3. The components are produced one unit at a time on separate production facilities. Finished components are placed in inventory and hence incur a holding cost that is component dependent. Components production times are independent and exponentially distributed with mean $1/\mu_k$, $k = 1, \ldots, n$. Hence, each facility can be viewed as a single server make-to-stock queue with finite service rate $\mu_k$. Product $i$'s demand takes place continuously over time according to an independent Poisson process with rate $\lambda_i$ and can be satisfied only if components $i, i+1, \ldots, n$ are all available. Otherwise, the demand is considered lost. A lost sale cost $c_i$ per unit of product $i$ is incurred when demand for product $i$ cannot be fulfilled. Without loss of generality, we assume $c_1 \geq c_2 \geq \cdots \geq c_n$. Here, we assume that the marginal benefit from adding a component to the end product increases (which is the case for most applications) and therefore the opportunity cost of not being able to satisfy the demand increases with the number of components of the final Product. Figure 1 shows a graphical depiction of the system under consideration.
Because of the difference in lost sale costs, it may not always be optimal to satisfy the demand of a product even if all its components are available. In fact, it might be more desirable to reject a demand for product 3, for instance, in order to reserve the available inventory for future demand of Product 1 (which has a higher shortage cost and uses component 3 as well for its assembly). As a result, each time an order is placed for a product, the system manager must decide whether or not to satisfy it from on-hand inventory, if any is available. If the manager decides to satisfy an incoming order from available inventory, the product is immediately assembled and shipped to the customer. We assume that there are no costs associated with interrupting production. This is a common assumption in the treatment of production-inventory systems in the literature; see for example (Ha 1997a, 1997b).

We describe the state of the system at time \( t \) by the vector \( \mathbf{X}(t) = (X_1(t), \ldots, X_n(t)) \), where \( X_i(t), i = 1, \ldots, n \), is a non-negative integer denoting the on-hand inventory for component \( i \) at time \( t \). Because both order inter-arrival times and production times are exponentially distributed, the system is memoryless and decision epochs can be restricted to only times when the state changes (i.e., the production completion of a component or the fulfillment of a demand). As such the problem can be
formulated as a Markov Decision Process (MDP) where we restrict our attention only to the class of Markovian policies for which actions taken at a particular decision epoch depend only on the current state of the system.

Let \( \pi \) be the policy that specifies, for each state \( x = (x_1, \cdots, x_n) \) and let \( h(x) = \sum_{k=1}^{n} h_k(x_k) \), where \( h_k \) is an increasing convex function, denote the inventory holding cost rate when the state is \( x \) and let \( N_i(t) \) denote the number of orders from product \( i \) that have not been fulfilled from on-hand inventory up to time \( t \). Then, the expected discounted cost over an infinite planning horizon \( v^\pi(x) \) obtained under a policy \( \pi \) and a starting state \( x = (x_1, \cdots, x_n) \), can be written as:

\[
v^\pi(x) = E_x \left[ \sum_{k=1}^{n} \int_0^\infty e^{-\alpha t} h_k(X_k(t)) dt + \sum_{i=1}^{n} \int_0^\infty e^{-\alpha t} c_i dN_i(t) \right].
\]

where \( \alpha > 0 \) is the discount rate. To simplify the analysis, we use Lippman’s (1975) uniformization technique to transform the continuous time decision process into a discrete time one. Hence, we define, in each state under any action, the uniform transition rate \( \gamma = \sum_{i=1}^{n} \lambda_i + \sum_{k=1}^{n} \mu_k \) so that the transition times between events is a sequence of i.i.d. exponentially distributed random variables, each with mean \( 1/\gamma \) which leads to a Markov chain defined by \( \{X_s: s \geq 0\} \) where \( X_s \equiv X(t_s) = (X_1(t_s), \cdots, X_n(t_s)) \) is the state resulting from the \( s^{th} \) transition.

We define the operators \( T^i \) and \( T_k \), \( i, k = 1, \cdots, n \), for any real valued function \( v \) over the state space, as follows:

\[
T^i v(x) = \begin{cases} 
v(x) + c_i & \text{if } \prod_{k=1}^{n} x_k = 0, \\
\min \{v(x - e_i), v(x) + c_i\} & \text{otherwise},
\end{cases}
\]

and

\[
T_k v(x) = \min \{v(x + e_k), v(x)\},
\]

where \( e_i = \sum_{k=1}^{n} e_k \) and \( e_k \) is the \( k^{th} \) unit vector of dimension \( n \). Without loss of generality, we rescale time so that \( \alpha + \gamma = 1 \). Then, the optimality equation (see Ross 1970) is given by the following expression:

\[
v^* (x) = h(x) + \sum_{i=1}^{n} \lambda_i T^i v^* (x) + \sum_{k=1}^{n} \mu_k T_k v^* (x)
\]

The operator \( T^i \) is associated with decisions about whether or not to satisfy an order for product \( i \) while the operator \( T_k \) is associated with decisions about whether or not to produce component \( k \). Note that whenever there is on-hand inventory for each component, it is optimal to satisfy demand from product \( i \) if
Similarly, it is optimal to produce component $k$, when the system is in state $x$, if

$$v(x - E_i) \leq v(x) + c_i.$$  

4 Characterization of the Optimal Policy

In this section, we characterize the structure of the optimal policy. In order to do so, we will show that the optimal value function $v^*(x)$, for all states $x$, satisfies certain properties.

**Definition 1:** Let $\mathcal{V}$ be the set of functions defined on $\mathbb{Z}^n$ such that if $v \in \mathcal{V}$, then for $k = 1, \ldots, n$

A1: $v(x + 2e_k) - v(x + e_k) \geq v(x + e_k) - v(x)$;

A2: $v(x + e_k - v(x)) \geq v(x + e_k + e_l) - v(x + e_l)$, for $l \neq k$;

A3: $v(x + e_k + E_i) - v(x + E_i) \geq v(x + e_k) - v(x)$;

A4: $v(x + E_i) + c_i \geq v(x)$.

Property A1 implies that $v(x)$ is convex in $x_k$. Property A2 implies that $v(x)$ is submodular in the direction $(x_k, x_l)$. Property A3 implies that $v(x)$ is supermodular in the direction $(x_k, E_i)$ (see Topkis, 1978 for definition of submodularity and supermodularity). Property A4 implies that it is always optimal to satisfy the demand of Product 1.

**Definition 2:** For $v \in \mathcal{V}$ we define the following quantities.

$$s_k(\tilde{x}(k)) = \min \left\{ x_k \geq 0 \mid v(x + e_k) - v(x) \geq 0 \right\},$$

$$r_{k,i}(\tilde{x}(k)) = \min \left\{ x_k \geq 0, k = i, \ldots, n \mid v(x) + c_i \geq v(x - E_i) \right\},$$

where $\tilde{x}(k)$ is a $n - 1$ dimensional vector defined as $\tilde{x}(k) = (x_1, \ldots, x_{k-1}, x_{k+1}, \ldots, x_n)$.

Property A1 combined with the definition of $s_k(\tilde{x}(k))$ implies

$$\begin{cases} 
  v(x + e_k) - v(x) \geq 0 & \text{if and only if } x_k \geq s_k(\tilde{x}(k)), \\
  v(x + e_k) - v(x) \leq 0 & \text{otherwise.}
\end{cases}$$

and Property A3 combined with the definition of $r_{k,i}(\tilde{x}(k))$ implies

$$\begin{cases} 
  v(x) + c_i \geq v(x - E_i) & \text{if and only if } x_k \geq r_{k,i}(\tilde{x}(k)), \\
  v(x) + c_i < v(x - E_i) & \text{otherwise.}
\end{cases}$$

Using Property A4 and the definition of $r_{k,i}(\tilde{x}(k))$, $k = 1, \ldots, n$, it follows that $r^*_{k,i}(\tilde{x}(k)) = 1$ for $k = 1, \ldots, n$. In other words, the demand for Product 1 is always satisfied as long as there is at least one
unit of each of components $k$, $k=1, \ldots, n$. Moreover, since Component 1 is used in Product 1 only, its inventory is never rationed. Note that there are $n(n-1)/2$ rationing levels in total.

Property A2 implies that $s_k(\tilde{x}(k) + e_j) \geq s_k(\tilde{x}(k))$ for $j \neq k$. i.e., $s_k(\tilde{x}(k))$ is non-decreasing in $x_j$ for $j \neq k$; Property A3 implies that $s_k(\tilde{x}(k) + E_i) \leq s_k(\tilde{x}(k))$ for $i \geq k$. i.e., $s_k(\tilde{x}(k))$ is non-increasing in $E_i$ for $i \geq k$; Property A3 implies that $r_{j,i}(\tilde{x}(k))$ is non-increasing in $x_k$ for $k \geq i$, $i=1, \ldots, n$. Property A4 implies that $r_{i,i}(\tilde{x}(k)) = 1$, $k = 1, \ldots, n$.

**Lemma 1:** If $v \in \mathbb{R}$ then $Tv \in \mathbb{R}$, where $Tv(x) = h(x) + \sum_{i=1}^{n} \lambda_i T^i v(x) + \sum_{i=1}^{n} \mu_1 T^i v(x)$.

All the proofs are included in the Appendix.

We are now ready to state our main result for this section.

**Theorem 1:** There exists an optimal stationary policy that can be specified as follows. The optimal production policy for components $k$, $k=1, \ldots, n$, is a base-stock policy with a base-stock level $s_k^*(\tilde{x}(k))$ that depends on the inventory level of all other components, such that it is optimal to produce component $k$ if $x_k < s_k^*(\tilde{x}(k))$ and not to produce it otherwise. The optimal inventory allocation policy for component $k$, is a multi-level rationing policy with a vector of rationing levels $r_{j,k}^*(\tilde{x}(k)) = (r_{j,k}^*(\tilde{x}(k)), \ldots, r_{n,k}^*(\tilde{x}(k)))$, such that it is optimal to satisfy Product i's demand, if $x_k \geq r_{j,k}^*(\tilde{x}(k))$ for all $k, k \geq i$, and to reject it otherwise. In addition, the optimal policy exhibits the following properties:

**P1:** $s_k(\tilde{x}(k) + e_j) \geq s_k(\tilde{x}(k))$ for $j \neq k$;

**P2:** $s_k(\tilde{x}(k) + e_j) - 1 \leq s_k(\tilde{x}(k)) \leq s_k(\tilde{x}(k) + e_j)$ for all $j \neq k$;

**P3:** $r_{j,k}^*(\tilde{x}(k) + e_j) \leq r_{j,k}^*(\tilde{x}(k))$ for $j \geq k \geq i$ and $r_{j,k}^*(\tilde{x}(k) + e_j) \geq r_{k,j}^*(\tilde{x}(k))$ for $k \geq j > i$, $i=1, \ldots, n$;

**P4:** $r_{k,k+1}^*(\tilde{x}(k)) \geq r_{k,k+1}^*(\tilde{x}(k))$ for $k \geq i$, $i=1, \ldots, n$;

**P5:** $r_{i,i}^*(\tilde{x}(k)) = 1$, $k = 1, \ldots, n$;

The results of Theorem 1 show that the base-stock level for a component does not decrease with an increase in the inventory level of other components. In other words, increasing the inventory of a component can only increase the desirability to produce more of all the other components, since eventually one unit from each component will be needed for final assembly. Also, a unit increase in the inventory level of component $j$ leads to at most a unit increase in the optimal base-stock level of component $k$ ($k \neq j$) implying that the base stock levels are linearly bounded. For inventory allocation, the results of Theorem 1 show that the products should be treated differently, with each product assigned a vector of inventory rationing level below which the demand from this product would be rejected in favor
of reserving inventory for products with higher lost sales costs. Theorem 1 also shows that it is always optimal to satisfy Product 1’s orders if on-hand inventory is available. Finally, similar to the base-stock levels, the rationing levels are state-dependant and must be adjusted based on the inventory level of all components.

To illustrate the optimal policy, we consider a simple case of a system with two components with the following parameters: $\mu_1=\mu_2=1.0$, $\lambda_1=\lambda_2=0.70$, $c_1=150$, $c_2=80$, $h_1=1.5$, $h_2=1.0$. We used a discount factor $\alpha=0.0001$. Figure 2(a) shows the optimal production policy of components 1 and 2 and Figure 2(b) shows the inventory allocation of Component 2 (recall from the above that component 1’s inventory is never rationed since it is only used by product 1). Here, we note that although the structure of the optimal production policy is similar to that of an ATO system with a single product and multiple demand classes (Benjaafar and Elhafsi (2006)), the inventory allocation policy is completely different in our case.

Figure 2(a) – The structure of the optimal production policy

Figure 2(b) – The inventory allocation policy of Component 2
It is straightforward to extend the analysis to the case where the optimization criterion is the average cost rate instead of the expected discounted cost. Given a policy $\pi$, the average cost rate is given by:

$$v^*(x) = \lim_{T \to \infty} \sup \left\{ \frac{1}{T} \sum_{k=1}^{n} \int_{0}^{T} h_k(X_k(t)) dt + \sum_{k=1}^{n} \int_{0}^{T} c_d N_j(t) \right\},$$

(5)

where the optimal policy $\pi^*$ satisfies $v^*(x) = \inf_{\pi} v^*(x)$ for all states $x$. Using similar arguments as in the discounted cost criterion, the optimality equation of $v^*(x)$ is given by:

$$v^*(x) + g^* = h(x) + \sum_{k=1}^{n} \lambda_k T_k v^*(x) + \sum_{k=1}^{n} \mu_k T_k v^*(x),$$

(6)

where $g^*$ represents the optimal average cost rate. Noting that optimality equation (6) is exactly the same as optimality equation (4) to a scalar ($g^*$), it follows that the optimal policy under the average cost rate criterion is exactly the same as that under the discounted cost criterion. The reader can easily check that the average cost rate exists by showing that the conditions given in Weber and Stidham (1987) are verified in this case.

The advantage of using the average cost criterion instead of the discounted cost criterion resides in the fact that the average cost rate is independent of the initial state and hence more amenable for comparisons and numerical studies.

5 Heuristic Policies

In this section, we propose two heuristics whose performance we compare to that of the optimal policy in section 6. The first heuristic consists of controlling the production of components independently of each other via a vector $s = (s_1, \ldots, s_n)$ of fixed base-stock levels and controlling acceptance/rejection of orders via a vector $r = (r_{i,1}, \ldots, r_{i,j}, \ldots, r_{n,j})$ of fixed rationing levels for each product $i, i = 1, \ldots, n$. We refer to this heuristic as the Multi-Product Independent Base-stock with Rationing or MPIBR for short. Compared to the optimal policy, the MPIBR ignores state dependencies and uses constant thresholds for the production base-stock levels. Similarly it ignores the state dependencies of the inventory rationing levels and use constant thresholds for controlling inventory allocation. The average cost per period of the MPIBR policy can be obtained via the following dynamic programming equation:

$$v^*(x) + g^*_{\text{MPIBR}} = h(x) + \sum_{i=1}^{n} \lambda_i T_{i,\text{MPIBR}} v^*(x) + \sum_{k=1}^{n} \mu_k T_k^{\text{MPIBR}} v^*(x),$$

(7)

where $g^*_{\text{MPIBR}}$ represents the average cost per period of the MPIBR heuristic, and

$$T_k^{\text{MPIBR}} v(x) = \begin{cases} v(x + e_k) & \text{if } x_k < s_k \\ v(x) & \text{otherwise,} \end{cases}$$

and
with the inequality $x \geq r^i$ taken component-wise.

The advantage of the MPIBR policy is that it is much simpler to implement than the optimal policy since it does not require the storage of the state-dependent base-stock levels. This is important when the number of components is large where it becomes difficult to compute the optimal policy because of the exponential growth in the size of the state space. However, the MPIBR policy is clearly sub-optimal and lacks the coordination in managing the production of different components carried out under the optimal policy.

When the number of components is small, it is straightforward to evaluate the cost under the MPIBR heuristic (using the value iteration algorithm for instance) for a given combination of the control parameters (i.e., base-stock and rationing levels). When the number of components is large, exact computation of the cost becomes difficult due to the exponential growth in the size of the state space. However, for a given vector of base-stock levels, and a matrix of rationing levels, it is not difficult to evaluate the MPIBR using simulation, even when the state space is large. Determining optimal values for the control parameters can be carried out via an exhaustive search over a sufficiently large range of feasible values. However, because of the combinatorial nature of the problem, this is possible only when the number of components is small. For large problems, a heuristic search may be needed. An example of a simple heuristic approach to selecting initial values for the base-stock and rationing levels for the MPIBR policy is obtained by decomposing the original problem with multiple components into multiple single component problems. We refer to this heuristic as the MPIBR-simplified (MPIBR-S) heuristic. Therefore, for each component, the optimal fixed base-stock and rationing levels are determined by solving the following single dimension dynamic programming problem:

$$T^{i,\text{MPIBR}} v(x) = \begin{cases} v(x - E_i) & \text{if } x \geq r^i \\ v(x) + c_i & \text{otherwise,} \end{cases}$$ 

(9)

where $g^*_{k,\text{MPIBR-S}}$ represent the average cost per period of the MPIBR-S heuristic applied to component $k$, for $k = 1, \ldots, n$, and

$$T^{\text{MPIBR-S}} k v(x) = \begin{cases} v(x + 1) & \text{if } x < s_k \\ v(x) & \text{otherwise,} \end{cases}$$

(11)

and

$$T^{\text{i,MPiBR-S}} v(x) = \begin{cases} v(x - 1) & \text{if } x \geq 1 \\ v(x) + \tilde{c}_k & \text{otherwise.} \end{cases}$$

(12)
\[ \tilde{\lambda}_k = \sum_{i=1}^k \lambda_i \]
represents the total demand for component \( k \) and \( \tilde{c}_k = \sum_{i=1}^k c_i / \tilde{\lambda}_k \) represents a weighted average lost sale penalty assigned to component \( k \). Note that in the case of the MPIBR-S heuristic, the system is equivalent to an \( M / M / 1 \) queuing system where \( s_k^* \) is the optimal base stock level for component \( k \) when considered in isolation. Hence, \( s_k^* \) can be obtained using the following expression (section 4.3 of Buzacott and Shanthikumar, 1993):

\[
\begin{align*}
    s_k^* &= \arg \min_z \left\{ h_k \left( \frac{1 - \tilde{\rho}_k \left( z + \tilde{\rho}_k \sum z \right) - \tilde{\rho}_k \left( 1 + \tilde{\rho}_k \sum z \right)} {1 - \tilde{\rho}_k \left( 1 + \tilde{\rho}_k \sum z \right)} \right) + \tilde{c}_k \tilde{\lambda}_k \tilde{\rho}_k \left( 1 - \tilde{\rho}_k \right) \right\},
\end{align*}
\]

(13)

where \( \tilde{\rho}_k = \tilde{\lambda}_k / \mu_k \). Once all the base-stock levels are obtained in this fashion, they are used in the dynamic programming equation of the MPIBR heuristic to determine the total cost of the MPIBR-S heuristic. Note that once the \( s_k^* \) parameters are determined using the MPIBR-S heuristic, the \( r_{k,i} \) parameters would be determined in the same way that they are determined under the MPIBR heuristic. In the next section, we evaluate the effectiveness of these two heuristics using numerical results.

### 6 Numerical Results

To test the performance of the MPIBR and MPIBR-S heuristics against the optimal policy, we carried out a series of numerical experiments for a system with two components and for a wide range of parameter values. We used the relative value iteration algorithm (see Puterman 1994) to solve the dynamic programming equation corresponding to either the optimal or the heuristic policies. We present results for the average cost criterion because they are independent of the starting state and the discount factor. The state space is truncated at \( \{0, n_1^{\text{max}}\} \times \{0, n_2^{\text{max}}\} \) where \( n_k^{\text{max}} \), \( k = 1, 2 \), is a positive integer that is gradually increased until the cost is no longer sensitive to the truncation level. The algorithm is terminated once ten-digit accuracy is obtained.

To evaluate the performance of the above two heuristics as well as properties of the production and allocation policies, we will use a system with two components. For all problem instances we used linear holding costs with \( h_1 \geq 0 \) and \( h_2 \geq 0 \) being the unit holding cost rates for components 1 and 2, respectively. Representative results for two-component systems are shown in Table 1. The parameter values for the 50 examples shown were generated randomly. We observe that the percentage average cost difference between the optimal policy and the MPIBR heuristic is relatively small. For the problems shown in Table 1, the average percentage difference is 0.57\% with a minimum of 0.007\% and a maximum of 4.64\%. The MPIBR-S heuristic does not perform as well as the MPIBR heuristic with an average percentage of 8.11\%, a minimum of 0.007\% and a maximum of 61.36\%. In this case, the MPIBR-S heuristic can be used as a first phase to compute an approximate value of the optimal control parameters of the MPIBR.
heuristic since these can be easily obtained using (13). In a second phase, a local search (rather than an exhaustive search) can be performed to obtain the optimal control parameters of the MPIBR heuristic. In addition, a closer look at Table 1 reveals that the MPIBR-S heuristic performs badly especially in cases where the system is over loaded. This can be observed in cases 10, 13, 18, 22, 24, 27, 29, 34, 47, 48, and 49. Note that in all these cases, the demand rate for one of the components is higher than the corresponding production rate. In general, most systems in practice have enough capacity to satisfy their
Table 1 – Optimal policy versus heuristics (a sample of examples randomly generated with parameter values drawn from uniform distributions as follows \( \mu_k \sim U(1, 10), \lambda_k \sim U(1, 9), h_k \sim U(1,10), c_k \sim U(1,200) \))

<table>
<thead>
<tr>
<th>( \mu_1 )</th>
<th>( \mu_2 )</th>
<th>( \lambda_1 )</th>
<th>( \lambda_2 )</th>
<th>( h_1 )</th>
<th>( h_2 )</th>
<th>( c_1 )</th>
<th>( c_2 )</th>
<th>Average optimal cost</th>
<th>% difference from optimal cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPIBR</td>
<td>MPIBR-S</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>7.79</td>
<td>6.97</td>
<td>8.07</td>
<td>3.18</td>
<td>4.77</td>
<td>2.92</td>
<td>17.15</td>
<td>8.08</td>
<td>2.88</td>
</tr>
<tr>
<td>2</td>
<td>8.66</td>
<td>4.06</td>
<td>4.73</td>
<td>3.06</td>
<td>8.76</td>
<td>178.35</td>
<td>131.67</td>
<td>48.96</td>
<td>0.11</td>
</tr>
<tr>
<td>3</td>
<td>5.39</td>
<td>9.93</td>
<td>3.99</td>
<td>5.25</td>
<td>2.63</td>
<td>5.52</td>
<td>132.43</td>
<td>85.02</td>
<td>4.30</td>
</tr>
<tr>
<td>4</td>
<td>7.06</td>
<td>9.62</td>
<td>2.53</td>
<td>1.89</td>
<td>6.09</td>
<td>9.72</td>
<td>174.17</td>
<td>5.73</td>
<td>2.91</td>
</tr>
<tr>
<td>5</td>
<td>1.24</td>
<td>5.68</td>
<td>2.54</td>
<td>6.73</td>
<td>3.26</td>
<td>9.40</td>
<td>104.80</td>
<td>28.30</td>
<td>14.06</td>
</tr>
<tr>
<td>6</td>
<td>9.06</td>
<td>9.48</td>
<td>3.68</td>
<td>4.50</td>
<td>5.24</td>
<td>2.34</td>
<td>106.97</td>
<td>28.04</td>
<td>1.95</td>
</tr>
<tr>
<td>7</td>
<td>7.53</td>
<td>4.59</td>
<td>3.87</td>
<td>2.88</td>
<td>6.64</td>
<td>193.64</td>
<td>48.99</td>
<td>11.94</td>
<td>0.50</td>
</tr>
<tr>
<td>8</td>
<td>6.76</td>
<td>3.07</td>
<td>6.45</td>
<td>6.33</td>
<td>2.21</td>
<td>1.20</td>
<td>53.18</td>
<td>24.19</td>
<td>14.89</td>
</tr>
<tr>
<td>9</td>
<td>1.62</td>
<td>8.68</td>
<td>2.44</td>
<td>1.26</td>
<td>7.61</td>
<td>5.52</td>
<td>132.43</td>
<td>55.93</td>
<td>6.58</td>
</tr>
</tbody>
</table>

Average optimal cost

<table>
<thead>
<tr>
<th>MPIBR</th>
<th>MPIBR-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>119.28</td>
<td>66.89</td>
</tr>
<tr>
<td>82.14</td>
<td>82.10</td>
</tr>
<tr>
<td>157.84</td>
<td>56.93</td>
</tr>
<tr>
<td>181.50</td>
<td>152.58</td>
</tr>
<tr>
<td>156.19</td>
<td>97.34</td>
</tr>
<tr>
<td>112.60</td>
<td>96.46</td>
</tr>
<tr>
<td>123.78</td>
<td>103.53</td>
</tr>
<tr>
<td>184.84</td>
<td>162.85</td>
</tr>
<tr>
<td>123.78</td>
<td>103.53</td>
</tr>
<tr>
<td>184.84</td>
<td>162.85</td>
</tr>
<tr>
<td>123.78</td>
<td>103.53</td>
</tr>
</tbody>
</table>

Average optimal cost

<table>
<thead>
<tr>
<th>MPIBR</th>
<th>MPIBR-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
</tbody>
</table>

Average optimal cost

<table>
<thead>
<tr>
<th>MPIBR</th>
<th>MPIBR-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
</tbody>
</table>

Average optimal cost

<table>
<thead>
<tr>
<th>MPIBR</th>
<th>MPIBR-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
</tbody>
</table>

Average optimal cost

<table>
<thead>
<tr>
<th>MPIBR</th>
<th>MPIBR-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
</tbody>
</table>

Average optimal cost

<table>
<thead>
<tr>
<th>MPIBR</th>
<th>MPIBR-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
</tbody>
</table>

Average optimal cost

<table>
<thead>
<tr>
<th>MPIBR</th>
<th>MPIBR-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
</tbody>
</table>

Average optimal cost

<table>
<thead>
<tr>
<th>MPIBR</th>
<th>MPIBR-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
</tbody>
</table>

Average optimal cost

<table>
<thead>
<tr>
<th>MPIBR</th>
<th>MPIBR-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
<tr>
<td>188.74</td>
<td>141.19</td>
</tr>
</tbody>
</table>

Average optimal cost
demand. Hence, such large percentage differences will not be observed. This suggests that in practice, simple heuristics with fixed parameters can be effective in managing ATO systems with nested-product structure.

In Figures 3, we compare the average cost of the optimal policy to the average cost of the FCFS policy as well as to that of the MPIBR policy for three values of the total per unit shortage cost, $c_1+c_2$, while varying the ratio $c_1/c_2$. Figure 3 shows that the FCFS policy gets even worst as the values of $c_1+c_2$ and $c_1/c_2$ increase. On the other hand, Figure 3 shows that the percentage difference of MPIBR policy exhibits no pattern and stays relatively low as $c_1+c_2$ and $c_1/c_2$ vary. In particular, we note that the maximum percentage difference for all cases combined is only 2.05%. This is consistent with the results of Table 1 and supports the argument that simple heuristics can be effective for systems with lost sales.

The results of this section highlight the importance of inventory rationing. Regardless of whether the rationing is done optimally or via a heuristic, it is important to make inventory allocation decisions that take into account the cost differences between different products. In fact, inventory allocation among products seems to be more critical than production coordination among components.

![Figure 3](image-url)  
**Figure 3** – The effect of differences in the shortage cost on the performance of the FCFS policy  
($\mu_1=0.4$, $\mu_2=1.0$, $\lambda_1=0.3$, $\lambda_2=0.6$, $h_1=h_2=1.0$)
7 Conclusion

In this paper, we considered the optimal production control and inventory allocation of an ATO system consisting of \( n \) products assembled from \( n \) components according to a nested-assembly architecture. We formulated the problem as a Markov decision process and characterized the structure of the optimal control policy. We showed that components’ optimal production policy is of the base-stock type with the base-stock level for each component non-decreasing in the inventory level of other components (state-dependent base-stock levels). We showed that components’ optimal inventory allocation is a multi-level rationing policy with the rationing level for each component, with respect to a particular product, depending on the inventory level of all other components (state-dependent rationing levels). As dynamic programming algorithms suffer from the curse of dimensionality (and therefore obtaining the optimal control parameters can be computing intensive, especially when the number of components becomes large), we proposed a simple heuristic which manages components independently as opposed to the optimal policy which coordinates the production of components. Our numerical results suggest that such a policy can be effective if carefully designed in terms of the choice of the optimal control parameters. We also proposed a second heuristic (MPIBR-S) that did not perform as good as the MPIBR heuristic. But nevertheless, this heuristic can be used to obtain initial values of the control parameters of the MPIBR heuristic which can be refined (using a local search algorithm for instance) in a second step to reduce the computational effort.

References


Appendix

Proof of Lemma 1: We divide the proof into three parts. In Part 1, we prove that if $v \in \mathcal{V}$ then $T_k v \in \mathcal{V}$ for all $k$. In part 2, we prove that if $v \in \mathcal{V}$ then $T^i v \in \mathcal{V}$ for all $i$. In part 3, we use the results of parts 1 and 2 to complete the proof and show that if $v \in \mathcal{V}$ then $T v \in \mathcal{V}$. First, we introduce the following difference operators for functions $v$ defined on $\mathbb{Z}^+$, where $\mathbb{Z}^+$ is the set of non-negative integers:
\[
\Delta_k v(x) = v(x + e_j) - v(x), \quad \text{for } k = 1, \ldots, n,
\]
\[
\Delta_{k_1 + k_2 + \cdots + k_p} v(x) = v(x + e_{k_1} + e_{k_2} + \cdots + e_{k_p}) - v(x),
\]
and combinations of such operators, including
\[
\Delta_{j,k} v(x) = \Delta_j \Delta_k v(x) = \Delta_j v(x + e_j) - \Delta_k v(x).
\]
where \( x = (x_1, \ldots, x_n) \) and the variables \( x_k \) are state variables associated with an assembly system per the description of Section 3. Note that the order in which the differences are taken does not matter \(-i.e., \) \( \Delta_{j,k} v(x) = \Delta_{k,j} v(x) \).

First, we note that Property A1 is a consequence of properties A3 and A2. Indeed, using A3 and A2, we have
\[
\Delta_{i,k} v(x) = \Delta_i v(x + e_i) - \Delta_k v(x) \geq \Delta_i v(x + e_i + e_k) - \Delta_k v(x)
\]
\[
\Delta_k v(x + e_i + e_k) - \Delta_k v(x) \geq \Delta_k v(x + e_i + e_k + e_j) - \Delta_k v(x)
\]
\[\vdots\]
\[
\Delta_k v(x + e_i + \cdots + e_k) - \Delta_k v(x) \geq \Delta_k v(x + e_i + \cdots + e_k + e_j) - \Delta_k v(x) \geq 0.
\]

Also, to simplify the notation, we define the set of indexes \( S_j = \{i, \ldots, n\} \).

**Operators** \( T_k \)

In order to show that \( T_{k} v \in \mathcal{V} \), we need to show that \( T_{k} v \) satisfies properties A2-A4 since Property A1 is directly obtained as a consequence of properties A2 and A3. First, we note that the following holds:
\[
T_{k} v(x) = \min \{v(x + e_k), v(x)\} = v(x) + \min \{\Delta_k v(x), 0\},
\]
\[
\Delta_{i,j} T_{k} v(x) = \Delta_{i,j} v(x) + \min \{\Delta_i v(x + e_j), 0\} - \min \{\Delta_i v(x), 0\},
\]
and
\[
\Delta_{i,j,k} T_{k} v(x) = \Delta_{i,j,k} v(x) + \min \{\Delta_i v(x + e_j + e_k), 0\} - \min \{\Delta_i v(x + e_j), 0\} - \min \{\Delta_i v(x), 0\},
\]
for all values of \( i, j, \) and \( k \).

**Property A2:** In order to show that functions \( T_{k} v \) satisfy A2, we must show that \( \Delta_{i,j} T_{k} v(x) \leq 0 \), for \( i \neq j \).

We distinguish three cases: \( j=k, i=k, \) and \( j \neq k \).

**Case** \( j \neq k \): By A2, we have \( \Delta_k v(x) \geq \Delta_k v(x + e_j) \geq \Delta_k v(x + e_j + e_i) \) and \( \Delta_k v(x) \geq \Delta_k v(x + e_j) \geq \Delta_k v(x + e_j + e_i) \), leading to six possible sub-cases:

1. \( \Delta_k v(x) \geq \Delta_k v(x + e_j) \geq 0 \) and \( \Delta_k v(x) \geq \Delta_k v(x + e_j + e_i) \geq 0 \) \( \Rightarrow \)
\[
\Delta_{i,j} T_{k} v(x) = \Delta_{i,j} v(x) \leq 0. \quad \text{(By A2)}
\]
2. \( \Delta_k v(x) \geq \Delta_k v(x + e_j) \geq 0 \) and \( \Delta_k v(x) \geq \Delta_k v(x + e_j + e_i) \geq 0 \) \( \Rightarrow \)
\[
\Delta_{i,j} T_{k} v(x) = \Delta_{i,j} v(x + e_j) + \Delta_{i,j} v(x + e_j + e_i) \leq 0. \quad \text{(By A2)}
\]
3. \( \Delta_k v(x) \geq \Delta_k v(x + e_j) \geq 0 \) and \( \Delta_k v(x) \geq \Delta_k v(x + e_j) \geq 0 \) \( \Rightarrow \)
\[
\Delta_{i,j} T_{k} v(x) = \Delta_{i,j} v(x + e_j) + \Delta_{i,j} v(x + e_j + e_i) \leq 0. \quad \text{(By A2)}
\]
4. \( 0 \geq \Delta_k v(x) \geq \Delta_k v(x + e_j) \) and \( 0 \geq \Delta_k v(x) \geq \Delta_k v(x + e_j) \) \( \Rightarrow \)
\[
\Delta_{i,j} T_{k} v(x) = \Delta_{i,j} v(x + e_j + e_i) \leq 0. \quad \text{(By A2)}
\]
5. \( \Delta_k v(x) \geq 0 \geq \Delta_k v(x + e_j) \) and \( \Delta_k v(x) \geq \Delta_k v(x + e_j + e_i) \geq 0 \) \( \Rightarrow \)
\[
\Delta_{i,j} T_{k} v(x) = \Delta_{i,j} v(x) + \Delta_{i,j} v(x + e_j + e_i) - \Delta_{i,j} v(x + e_j) - \Delta_{i,j} v(x + e_j + e_i) \leq 0. \quad \text{(By A2)}
\]
The other two cases are treated similarly. It follows that the functions $T_k v$ satisfy property A2.

**Property A3:** To show that the functions $T_k v$ satisfy A3, we must show that $\Delta_{k,j} T_k v(x) \geq 0$ for all $j$.

Noting that

$$ \Delta_{k,j} T_k v(x) = \Delta_{k,j} v(x) + \min \{ \Delta_k v(x+e_j+E_i), 0 \} - \min \{ \Delta_k v(x+E_i), 0 \} $$

$- \min \{ \Delta_k v(x+e_j), 0 \} + \min \{ \Delta_k v(x), 0 \}$,

Here also we distinguish two cases: $j=k$ and $j \neq k$.

**Case $j=k$:** By A2 and A3, we have $\Delta_k v(x+E_i) \geq \Delta_k v(x) \geq \Delta_k v(x+e_j)$ and $\Delta_k v(x+E_i) \geq \Delta_k v(x+e_j)$, leading to six possible sub-cases:

1. $\Delta_k v(x+e_j) \geq \Delta_k v(x+e_j+E_i) \geq 0$ and $\Delta_k v(x+e_j+E_i) \geq \Delta_k v(x+e_j) \geq 0$ $\Rightarrow$
   $$ \Delta_{j,k} T_k v(x) = \Delta_{j,k} v(x) \geq 0 \text{ (By A3)} $$

2. $\Delta_k v(x+E_i) \geq \Delta_k v(x+e_j) \geq 0 \geq \Delta_k v(x+e_j)$ and $\Delta_k v(x+e_j+E_i) \geq \Delta_k v(x+e_j) \geq 0 \Rightarrow$
   $$ \Delta_{j,k} T_k v(x) = \Delta_{j,k} v(x) - \Delta_k v(x+e_j) \geq 0 \text{ (By A3)} $$

3. $\Delta_k v(x+E_i) \geq \Delta_k v(x+e_j) \geq 0 \geq \Delta_k v(x+e_j)$ and $\Delta_k v(x+e_j+E_i) \geq \Delta_k v(x+e_j) \geq 0 \Rightarrow$
   $$ \Delta_{j,k} T_k v(x) = \Delta_{j,k} v(x) - \Delta_k v(x+e_j) + \Delta_k v(x) - \Delta_{j,k} v(x) \geq 0 \text{ (By A3 and A2)} $$

4. $0 \geq \Delta_k v(x+E_i) \geq \Delta_k v(x) \geq \Delta_k v(x+e_j) \geq 0 \Rightarrow$
   $$ \Delta_{j,k} T_k v(x) = \Delta_{j,k} v(x) - \Delta_k v(x+e_j) \geq 0 \text{ (By A3)} $$

5. $\Delta_k v(x+E_i) \geq \Delta_k v(x) \geq \Delta_k v(x+e_j) \geq 0 \geq \Delta_k v(x+e_j+E_i) \Rightarrow$
   $$ \Delta_{j,k} T_k v(x) = \Delta_{j,k} v(x) - \Delta_k v(x+e_j) + \Delta_k v(x+E_i) \geq 0 \text{ (By A3)} $$

6. $\Delta_k v(x+E_i) \geq \Delta_k v(x) \geq \Delta_k v(x+e_j) \geq 0 \Rightarrow$
   $$ \Delta_{j,k} T_k v(x) = \Delta_{j,k} v(x+e_j) + \Delta_k v(x+E_i) \geq 0 \text{ (By A3)} $$

The case $j=k$ is treated similarly. It follows that the functions $T_k v$ satisfy property A3.

**Property A4:** To show that the functions $T_k v$ satisfy A4, we must show that $\Delta_{k} T_k v(x) \geq -c_1$ for all $k$.

First, we note that

$$ \Delta_{k} T_k v(x) = \min \{ \Delta_k v(x+e_k+E_i), v(x+E_i) \} - \min \{ \Delta_k v(x+e_k), v(x) \} $$

$$ = \min \{ \Delta_k v(x+e_k+E_i), \Delta_k v(x) - \Delta_k v(x) \} + \max \{ \Delta_k v(x), 0 \} \text{ (3)} $$

We then distinguish two possible cases: $\Delta_k v(x) \geq 0$ and $\Delta_k v(x) < 0$. If $\Delta_k v(x) \geq 0$,

$$ \Delta_{k} T_k v(x) = \min \{ \Delta_k v(x+e_k+E_i), \Delta_k v(x), \Delta_k v(x) \} \geq \min \{ -c_1, \Delta_k v(x) - c_1 \} = -c_1 $$

If $\Delta_k v(x) < 0$,

$$ \Delta_{k} T_k v(x) = \min \{ \Delta_k v(x+e_k+E_i), \Delta_k v(x) - \Delta_k v(x) \} \geq \min \{ -c_1, -c_1 - \Delta_k v(x) \} = -c_1 $$

Hence, the functions $T_k v$ satisfy A4. Also, since the functions $T_k v$ satisfy A1-A3, $T_k v \in \mathcal{V}$ for all $k$.

**Operators $T^i$**
In order to show that $iTv \in \mathcal{V}$, we need to show that $iTv$ satisfies properties A2-A4 for $i=1, \ldots, n$.

First note that

$$T'(x) = v(x) + c_i + \begin{cases} 
0 & \text{if } \prod_{k \in S_i} x_k = 0 \\
\min \left\{ -\Delta_{E_i}v(x-E_i) - c_i, 0 \right\} & \text{otherwise.} 
\end{cases}$$

Property A2: We distinguish several cases: $l, j \notin S_i$, $l \in S_i$ and $j \notin S_i$ and $j \in S_i$, and $l, j \in S_i$.

We treat the latter case only. The other cases are less involved and are treated similarly. Now, assuming $l, j \in S_i$, first note that

$$\Delta_{i,j}T'(x) = \Delta_{i,j}v(x) + \begin{cases} 
0 & \text{if } \prod_{k \in S_i, k \neq j} x_k = 0 \\
\min \left\{ -\Delta_{E_i}v(x+e_l + e_j - E_i, E_j) - c_i, 0 \right\} - \min \left\{ -\Delta_{E_i}v(x+e_l - E_i, E_j) - c_i, 0 \right\} \\
- \min \left\{ -\Delta_{E_i}v(x+e_j - E_i, E_j) - c_i, 0 \right\} + \min \left\{ -\Delta_{E_i}v(x-E_i, E_j) - c_i, 0 \right\} & \text{if } \prod_{k \in S_i} x_k \neq 0 \\
\min \left\{ -\Delta_{E_i}v(x+e_l + e_j - E_i, E_j) - c_i, 0 \right\} - \min \left\{ -\Delta_{E_i}v(x+e_l - E_i, E_j) - c_i, 0 \right\} \\
- \min \left\{ -\Delta_{E_i}v(x+e_j - E_i, E_j) - c_i, 0 \right\} + \min \left\{ -\Delta_{E_i}v(x-E_i, E_j) - c_i, 0 \right\} & \text{if } \prod_{k \in S_i} x_k \neq 0, x_j = 0 \\
\min \left\{ -\Delta_{E_i}v(x+e_l + e_j - E_i, E_j) - c_i, 0 \right\} & \text{if } \prod_{k \in S_i, k \neq j} x_k \neq 0, x_j = 0, x_i = 0.
\end{cases}$$

Furthermore, by A3, we have

$$-\Delta_{E_i}v(x-E_i, E_i) - c_i \geq -\Delta_{E_i}v(x+e_j - E_i, E_i) - c_j \geq -\Delta_{E_i}v(x+e_l + e_j - E_i, E_i) - c_j,$$

and

$$-\Delta_{E_i}v(x-E_i, E_i) - c_i \geq -\Delta_{E_i}v(x+e_j - E_i, E_i) - c_j \geq -\Delta_{E_i}v(x+e_l + e_j - E_i, E_i) - c_j.$$

First, we have to distinguish between the cases $i \neq 1$ and $i=1$. For the case $i \neq 1$ there are five sub-cases

Therefore, for $k \in S_i$, we distinguish the cases: $\prod_{k \in S_i} x_k = 0, \prod_{k \in S_i} x_k \neq 0, \prod_{k \in S_i} x_k \neq 0$ and $x_i = 0$, $\prod_{k \in S_i, k \neq j} x_k \neq 0, x_j = 0$ and $x_i = 0$. For brevity, we only illustrate the case $\prod_{k \in S_i} x_k \neq 0$ as it is the most involved when $i \neq 1$. In this case, we distinguish six possible sub-cases:

1. $-\Delta_{E_i}v(x-E_i, E_i) - c_i \geq -\Delta_{E_i}v(x+e_j - E_i, E_i) - c_j \geq 0 \geq 0 \geq -\Delta_{E_i}v(x+e_l + e_j - E_i, E_i) - c_j \geq 0$ and 

   $-\Delta_{E_i}v(x-E_i, E_i) - c_i \geq -\Delta_{E_i}v(x+e_j - E_i, E_i) - c_j \geq -\Delta_{E_i}v(x+e_l + e_j - E_i, E_i) - c_j \geq 0 \Rightarrow 

   \Delta_{i,j}T'(x) = \Delta_{i,j}v(x) \leq 0.$ (By A2)

2. $-\Delta_{E_i}v(x-E_i, E_i) - c_i \geq -\Delta_{E_i}v(x+e_j - E_i, E_i) - c_j \geq 0 \geq -\Delta_{E_i}v(x+e_l + e_j - E_i, E_i) - c_j$ and 

   $-\Delta_{E_i}v(x-E_i, E_i) - c_i \geq -\Delta_{E_i}v(x+e_j - E_i, E_i) - c_j \geq 0 \geq -\Delta_{E_i}v(x+e_l + e_j - E_i, E_i) - c_j \Rightarrow 

   \Delta_{i,j}T'(x) = \Delta_{i,j}v(x) - \Delta_{E_i}v(x+e_l + e_j - E_i, E_i) - c_i \leq 0.$ (By A2)
3. \(-\Delta_E v(x - E_i) - c_i \geq 0 \geq -\Delta_E v(x + e_j - E_j) - c_j \) and 
\(-\Delta_E v(x - E_i) - c_i \geq -\Delta_E v(x + e_j - E_j) - c_j \geq 0 \geq -\Delta_E v(x + e_i + e_j - E_j) - c_j \Rightarrow \\
\Delta_{ij} T_i^iv(x) = \Delta_{ij} v(x) - \Delta_E v(x + e_j - E_j) + \Delta_E v(x + e_i + e_j - E_j) + \Delta_E v(x + e_i - E_i) + c_i \\
= \Delta_{ij} v(x) - \Delta_E v(x - E_i) \leq 0. \) (By A2 and A3)

4. \(-\Delta_E v(x - E_i) - c_i \geq -\Delta_E v(x + e_j - E_j) - c_j \geq 0 \geq -\Delta_E v(x + e_i + e_j - E_j) - c_j \) and 
\(-\Delta_E v(x - E_i) - c_i \geq 0 \geq -\Delta_E v(x + e_j - E_j) - c_j \geq -\Delta_E v(x + e_i + e_j - E_j) - c_j \Rightarrow \\
\Delta_{ij} T_i^iv(x) = \Delta_{ij} v(x) - \Delta_E v(x + e_j - E_j) + \Delta_E v(x + e_i + e_j - E_j) + \Delta_E v(x + e_i - E_i) + c_i \\
= \Delta_{ij} v(x) - \Delta_E v(x - E_i) \leq 0. \) (By A2 and A3)

5. \(-\Delta_E v(x - E_i) - c_i \geq -\Delta_E v(x + e_j - E_j) - c_j \geq -\Delta_E v(x + e_i + e_j - E_j) - c_j \) and 
\(-\Delta_E v(x - E_i) - c_i \geq 0 \geq -\Delta_E v(x + e_j - E_j) - c_j \geq -\Delta_E v(x + e_i + e_j - E_j) - c_j \Rightarrow \\
\Delta_{ij} T_i^iv(x) = \Delta_{ij} v(x) - \Delta_E v(x + e_j - E_j) + \Delta_E v(x + e_i + e_j - E_j) + \Delta_E v(x + e_i - E_i) + c_i \\
= \Delta_{ij} v(x - E_i) + \Delta_E v(x - E_i) + c_i \leq 0. \) (by A2 and A3)

For the case \(i = 1\), we first note that

\[
\Delta_{ij} T_i^iv(x) = \begin{cases} 
\Delta_{ij} v(x) & \text{if } \prod_{k \neq i, k \neq j} x_k = 0 \\
\Delta_{ij} v(x - E_i) & \text{if } \prod_{k \neq j, k \in S_j} x_k \neq 0 \\
\Delta_{ij} v(x - E_i) - \Delta_{ij} v(x + e_j - E_j) & \text{if } \prod_{k \neq i, k \neq j} x_k \neq 0, x_j = 0 \\
\Delta_{ij} v(x) - \Delta_{ij} v(x + e_i - E_i) & \text{if } \prod_{k \neq i, k \neq j} x_k \neq 0, x_i = 0 \\
\Delta_{ij} v(x) - \Delta_{ij} v(x + e_i + e_j - E_j) - c_i & \text{if } \prod_{k \neq i, k \neq j} x_k \neq 0, x_i = 0, x_j = 0.
\end{cases}
\]

Similarly, using Properties A2 and A3, we show that \(\Delta_{ij} T_i^iv(x) \leq 0\). All the other cases are treated similarly. A consequence of the above is that the functions \(T_i^iv\) satisfy A2.

**Property A3**: To show that the functions \(T_i^iv\) satisfy property A3, we must show that \(\Delta_{ij} T_i^iv(x) \geq 0\) for all \(i\) and \(j\). Noting that
\[\Delta_{j,E_i} T^i v(x) = \Delta_{j,E_i} v(x) + \min \left\{ -\Delta_{E_i} v(x + e_j) - c_i, 0 \right\} - \min \left\{ -\Delta_{E_i} v(x) - c_i, 0 \right\} - \begin{cases} 0 & \text{if } \prod_{k \neq j} x_k = 0 \\ \min \left\{ -\Delta_{E_i} v(x + e_j) - c_i, 0 \right\} & \text{if } \prod_{k \neq j} x_k \neq 0 \\ 0 & \text{if } \prod_{k \neq j} x_k = 0 \\ \min \left\{ -\Delta_{E_i} v(x - E_i) - c_i, 0 \right\} & \text{if } \prod_{k \neq j} x_k \neq 0, \end{cases}\]

Furthermore, by A3 we have

\[-\Delta_{E_i} v(x) - c_i \geq -\Delta_{E_i} v(x + e_j) - c_i \]
and

\[-\Delta_{E_i} v(x - E_i) - c_i \geq -\Delta_{E_i} v(x + e_j - E_i) - c_i.\]

First, we have to distinguish between the cases \(i \neq 1\) and \(i = 1\). For the case \(i \neq 1\) there are three sub-cases. Therefore, we distinguish the cases: \(\prod_{k \neq j} x_k = 0\), \(\prod_{k \neq j} x_k \neq 0\) and \(x_j = 0\), and \(\prod_{k \neq j} x_k \neq 0\). For brevity, we only illustrate the case \(\prod_{k \neq j} x_k \neq 0\) as it is the most involved when \(i \neq 1\). Therefore, there are six possible sub-cases.

1. \(-\Delta_{E_i} v(x) - c_i \geq -\Delta_{E_i} v(x + e_j) - c_i \geq 0\) and \(-\Delta_{E_i} v(x - E_i) - c_i \geq -\Delta_{E_i} v(x + e_j - E_i) - c_i \geq 0 \Rightarrow \Delta_{j,E_i} T^i v(x) = \Delta_{j,E_i} v(x) \geq 0.\) (by A3)
2. \(-\Delta_{E_i} v(x) - c_i \geq 0 \geq -\Delta_{E_i} v(x + e_j) - c_i \) and \(-\Delta_{E_i} v(x - E_i) - c_i \geq -\Delta_{E_i} v(x + e_j - E_i) - c_i \geq 0 \Rightarrow \Delta_{j,E_i} T^i v(x) = \Delta_{j,E_i} v(x) - \Delta_{E_i} v(x + e_j) - c_i = -\Delta_{E_i} v(x - E_i) - c_i \geq 0.\)
3. \(0 \geq -\Delta_{E_i} v(x) - c_i \geq -\Delta_{E_i} v(x + e_j) - c_i \) and \(-\Delta_{E_i} v(x - E_i) - c_i \geq -\Delta_{E_i} v(x + e_j - E_i) - c_i \geq 0 \Rightarrow \Delta_{j,E_i} T^i v(x) = -\Delta_{E_i} v(x) - c_i + \Delta_{E_i} v(x) + c_i = 0.\)
4. \(0 \geq -\Delta_{E_i} v(x - E_i) - c_i \geq -\Delta_{E_i} v(x - E_i) - c_i \) and \(0 \geq -\Delta_{E_i} v(x + e_j - E_i) - c_i \Rightarrow \Delta_{j,E_i} T^i v(x) = \Delta_{j,E_i} v(x - E_i) \geq 0.\) (by A3)
5. \(-\Delta_{E_i} v(x - E_i) - c_i \geq -\Delta_{E_i} v(x) - c_i \geq 0 \geq -\Delta_{E_i} v(x + e_j) - c_i \) and \(0 \geq -\Delta_{E_i} v(x + e_j - E_i) - c_i \Rightarrow \Delta_{j,E_i} T^i v(x) = -\Delta_{E_i} v(x) - c_i + \Delta_{E_i} v(x + e_j - E_i) + c_i \geq 0.\)
6. \(-\Delta_{E_i} v(x - E_i) - c_i \geq 0 \geq -\Delta_{E_i} v(x) - c_i \geq -\Delta_{E_i} v(x + e_j) - c_i \) and \(0 \geq -\Delta_{E_i} v(x + e_j - E_i) - c_i \Rightarrow \Delta_{j,E_i} T^i v(x) = \Delta_{E_i} v(x + e_j - E_i) + c_i \geq 0.\)

For the case \(i = 1\), we first note that

\[\Delta_{j,E_i} T^i v(x) = \begin{cases} 0 & \text{if } \prod_{k \neq j} x_k = 0 \\ 0 & \text{if } \prod_{k \neq j} x_k \neq 0 \\ -\Delta_{E_i} v(x - E_i) - c_i & \text{if } \prod_{k \neq j} x_k \neq 0 \\ -\Delta_{E_i} v(x + e_j - E_i) - c_i & \text{if } \prod_{k \neq j} x_k \neq 0 \end{cases}\]

We distinguish the following three sub-cases.

1. \(\prod_{k \neq j} x_k = 0 \Rightarrow \Delta_{j,E_i} T^i v(x) = 0.\) (by A3)
2. \(\prod_{k \neq j} x_k \neq 0\) and \(x_j = 0 \Rightarrow \Delta_{j,E_i} T^i v(x) = \Delta_{E_i} v(x + e_j - E_i) + c_i \geq 0.\)
3. \( \prod x_k \neq 0 \Rightarrow \Delta_{j,E_i} T^i v(x) = \Delta_{j,E_i} v(x - E_i) \geq 0. \) (by A3)

All the other cases are treated similarly. A consequence of the above is that the functions \( T^i v \) satisfy A3.

**Property A4:** To show that the functions \( T^i v \) satisfy A4, we must show that \( \Delta_{E_i} T^i v(x) \geq -c_i \) for all \( i \).

First note that

\[
\Delta_{E_i} T^i v(x) = \min \{v(x), v(x + E_i) + c_i\} - v(x) - c_i
\]

if \( \prod x_k = 0 \) we have

\[
\Delta_{E_i} T^i v(x) = \min \{v(x), v(x + E_i) + c_i\} - v(x) - c_i
\]

Hence, if \( \prod x_k = 0 \) we have

\[
\Delta_{E_i} T^i v(x) = \min \{v(x), v(x + E_i) + c_i\} - v(x) - c_i
\]

if \( \prod x_k \neq 0 \) we have

\[
\Delta_{E_i} T^i v(x) = \min \{v(x), v(x + E_i) + c_i\} - v(x) - c_i
\]

Since \( \Delta_{E_i} v(x) \geq \Delta_{E_i} v(x - E_i) \), the following holds: (1) if \( \Delta_{E_i} v(x - E_i) \geq -c_i \) then

\[
\Delta_{E_i} T^i v(x) = \Delta_{E_i} v(x - E_i) \geq -c_i
\]

(2) if \( \Delta_{E_i} v(x - E_i) \leq -c_i \) \( \Delta_{E_i} v(x) \) then \( \Delta_{E_i} T^i v(x) = -c_i \geq -c_i \), and (3) if \( \Delta_{E_i} v(x - E_i) \leq -c_i \) \( \Delta_{E_i} v(x) \) then \( \Delta_{E_i} T^i v(x) = \Delta_{E_i} v(x) \geq -c_i \). Hence, the functions \( T^i v \) satisfy A4.

**Operator T**

Given that \( h(x) \) is separable in the variables \( x_k \) and increasing convex in each \( x_k \), it is straightforward to show that \( h(x) \) satisfies A1-A3. Furthermore, since \( \mathcal{V} \) is closed under multiplication by a scalar and addition, \( Tv \) satisfies properties A1-A3. To show that \( Tv \) also satisfies A4, note that

\[
\Delta_{E_i} Tv(x) = \Delta_{E_i} h(x) + \sum_{i=1}^n \lambda_i \Delta_{E_i} T^i v(x) + \sum_{i=1}^n \mu_i \Delta_{E_i} T_k v(x)
\]

\[
\geq \Delta_{E_i} h(x) - \sum_{i=1}^n \lambda_i c_i - \sum_{i=1}^n \mu_i c_i \quad (\Delta_{E_i} h(x) > 0)
\]

\[
\geq -c_i \left( \sum_{i=1}^n \lambda_i + \sum_{i=1}^n \mu_i \right) = -c_i (1 - \alpha) \geq -c_i.
\]

Hence, \( Tv \in \mathcal{V} \), which completes the proof of lemma 1.

**Proof of Theorem 1:** We need to show that \( v^* \in \mathcal{V} \). Using a value iteration argument, we note that (i) \( v^* = \lim_{n \to \infty} T^n v \) for any \( v \in \mathcal{V} \), where \( T^n \) refers to \( n \) compositions of operator \( T \) and (ii) \( v^* \) is the unique solution of \( v = Tv \) (see Theorem 5.1 of Porteus (1982) and Theorem 6.10.4 of Puterman (1994)). Hence, \( v^* \) satisfies conditions A1-A4 and lemma 1, from which the results of Theorem 1 immediately follow.

Property A1 (i.e., \( \Delta_{E_i} v^*(x) \geq 0 \)) implies that the optimal production policy is of the base-stock type. To show that the optimal inventory allocation policy is of the rationing type, we first note that, for \( k \geq i \) and using A3, we have \( \Delta_{E_i} v^*(x + E_k) \geq \Delta_{E_i} v^*(x + E_i) \geq \Delta_{E_i} v^*(x) \) indicating that \( \Delta_{E_i} v^*(x) \geq 0 \). As a result \( \Delta_{E_i} v^*(x - E_i) = v^*(x) - v^*(x - E_i) \) is non-decreasing in \( x_k \) which means that there exists a unique value
\( r^*_k(\tilde{x}(k)) \), as defined in Definition 2, such that \( v^*(x) - v^*(x - E_i) \geq -c_i \). Hence, the optimal inventory allocation policy is of the rationing type.

**P1** is implied by Property A2.

In order to show **P2**, we have from condition A2 and the definition of \( s_k(\tilde{x}(k)) \),

\[
\Delta_k v \left( (x_1, \ldots, x_{k-1}, s_k(\tilde{x}(k) + e_j), x_{k+1}, \ldots, x_n) \right) \geq \Delta_k v \left( (x_1, \ldots, x_{k-1}, s_k(\tilde{x}(k) + e_j), x_{k+1}, \ldots, x_n) + e_j \right) \geq 0,
\]

and using the definition of \( s_k(\tilde{x}(k)) \), leads to \( s_k(\tilde{x}(k)) \leq s_k(\tilde{x}(k) + e_j) \). From condition A3, we have

\[
\Delta_k v \left( (x_1, \ldots, x_{k-1}, s_k(\tilde{x}(k) + 1, x_{k+1}, \ldots, x_n) + e_j \right) \geq \Delta_k v \left( (x_1, \ldots, x_{k-1}, s_k(\tilde{x}(k)), x_{k+1}, \ldots, x_n) \right) \geq 0,
\]

and using the definition of \( s_k(\tilde{x}(k)) \) leads to \( s_k(\tilde{x}(k) + e_j) \leq s_k(\tilde{x}(k)) + 1 \).

**P3** is implied by the fact that \( \Delta_{k, i} v^*(x) \geq 0 \) for \( k \geq i \) and \( \Delta_{k, i} v^*(x) \leq 0 \) for \( k < i \).

**P4** is implied by the fact that \( \Delta_{k, i} v^*(x - E_i) \) is non-decreasing in \( x_k \) and the ordering \( c_i \geq c_{i+1} \geq \cdots \geq c_n \).

**P5** is due to Property A4 and the definition of \( r^*_k(\tilde{x}(k)) \).

This completes the proof of Theorem 1. ■