Critical level policies in lost sales inventory systems with different demand classes

Aleksander Wieczorek, Ana Bušić, Emmanuel Hyon

▶ **To cite this version:**

INRIA

Critical level policies in lost sales inventory systems with different demand classes

Aleksander Wieczorek   — Ana Bušić   — Emmanuel Hyon

N° 7732

R apport de recherche

Critical level policies in lost sales inventory systems with different demand classes

Aleksander Wieczorek * , Ana Bušić † , Emmanuel Hyon ‡

Abstract: We consider a single-item lost sales inventory model with different classes of customers. Each customer class may have different lost sale penalty costs. We assume that the demands follow a Poisson process and we consider a single replenishment hypoexponential server. We give a Markov decision process associated with this optimal control problem and prove some structural properties of its dynamic programming operator. This allows us to show that the optimal policy is a critical level policy. We then discuss some possible extensions to other replenishment distributions and give some numerical results for the hyperexponential server case.

Key-words: Markov decision process, Critical level policies, Inventory systems.

* INRIA-ENS, 23 avenue d'Italie CS 81321 75214 Paris Cedex 13, France. `aleksanderwieczorek@gmail.com`

† INRIA-ENS, 23 avenue d'Italie CS 81321 75214 Paris Cedex 13, France. `ana.busic@inria.fr`

‡ Université Paris Ouest Nanterre, 200 avenue de la République F-92001 Nanterre, France and LIP6, UPMC, Paris, France `emmanuel.hyon@u-paris10.fr`

Politiques à seuils critiques dans des systèmes de gestion de stock avec pertes et différentes classes de clients

Résumé : On s'intéresse ici à un système de gestion de stock manipulant un seul type d'objet et pour lequel on autorise des pertes de ventes. Ce système manipule différentes classes de clients chacune ayant un coût de perte de vente différent. On suppose que les demandes suivent un processus de Poisson et nous nous intéressons au cas à un seul serveur de réassortiment dont la loi est hypoexponentielle. Nous décrivons le Processus de Décision Markovien associé à un tel problème de contrle optimal et montrons quelques propriétés structurelles de son opérateur de programmation dynamique. Nous en déduisons que la politique optimale est du type politique à seuils critiques. Nous discutons ensuite d'extensions lorsque les distributions de temps de réassortiment sont hyperexponentielles. Nous donnons alors quelques simulations numériques qui tendent à montrer l'existence de propriétés structurelles similaires.

Mots-clés : Processus de Decision Markovien, Politique à seuils critiques, Systèmes de gestion de stock

1 Introduction

We consider an inventory system with single item type and a finite stock capacity. Upon a demand, we can decide to either deliver an item or reject the demand. We assume different demand classes with different rejection penalty costs. The objective is to find a sequence of optimal decisions that minimizes the costs. This problem can be easily modeled as a Markov Decision Process (MDP). We refer the reader to Puterman [13] for general theory on MDPs and to Porteus [12] for inventory systems and MDPs.

Finding a sequence of optimal decisions (an optimal policy) is in general a very difficult problem. When the state space and the set of possible decisions (actions) are finite, one can solve the problem numerically. However, this can be done effectively only for very small instances. The aim is therefore to prove some structural properties of an optimal policy, which then simplifies the computation by reducing significantly the search space of all admissible policies.

We are interested here in the class of critical level policies. A critical level policy can be seen as a set of thresholds (a threshold by class) which determine the optimal action to perform (either rejection or acceptance of a demand). Such policies and their applications have been largely studied in the literature We will focus here only on the results closely related to our problem; for a survey with applications to communication networks see Altman [1] or Koole [8], and for applications to web services see Mazzucco et al. [11].

In the case of an exponential replenishment server, Ha [5] has shown the optimality of a critical level policy. This result was further extended by de Véricourt, Karaesmen and Dallery [2], by authorizing the backordering of unsatisfied demands, where critical level policy are also shown to be optimal. Both previous results assume exponential service times. Dekker et al. [3], and Kranenburg and van Houtum [9] consider general service times and they exhibit exact and heuristic methods to compute the real value of the optimal levels, under the assumption of a critical level policy. Nevertheless, there is no warranty that the critical level policies are optimal under the general service distributions.

In the case of Erlang distributed service times, the optimality of a critical level policy was proven by Ha [6]. However, his model relies on the assumption that the service operator has a constant rate in any state of the model.

We consider here the case of hypoexponential service (see Svoronos [14] for examples of complex services in inventory problems). We prove the optimality of critical level policies. Note that this represents the simplest case of a phase-type service where the service rate depends on the service phase. Also, we consider a two-dimensional state space that is more suitable to possible extensions to other distributions. This new state-space representation allows us to prove that the switching curve is monotone in k (service phase). We expect that our results represent a first step towards general phase-type service. Our numerical experiments suggest that the result may still be preserved in the hyperexponential case.

This paper is organized as follows. In Section 2 we give the formal description of the MDP problem considered in the paper. Section 3 is devoted to the hypoexponential service case. We prove first in Section 3.3 that the dynamic programming operator preserves convexity, which then has two different applications: i) that the optimal policy is of threshold type and ii) that the thresholds are ordered for different classes. The preservation of submodularity, proved in Section 3.5, implies the monotonicity of the switching curve for a fixed demand class. In Section 4 we discuss the possible extensions of the model. Section 5 contains conclusions and some future directions.

2   Problem formulation

2.1   Model description

We consider a single-item inventory model. We denote by S the maximum stock level. This single-item is demanded by J different types of customers (or demand classes). Demands arrive according to a Poisson process of parameter λ and the probability that an arriving demand is of class j is equal to p_j. Let λ_j be the arrival rate of class j. Then $\lambda_j = p_j \lambda$ and $\sum_{j=1}^{J} \lambda_j = \lambda$.

When an item is not delivered, the item is lost and a penalty cost is incurred. We denote by c_j the penalty cost of demand class j and we assume without loss of generality that the costs are ordered such that $c_1 \leq c_2 \leq \ldots \leq c_J$. When a demand arrives, one has to decide whether to satisfy or to refuse it. Since the stock level is finite, rejecting a demand causes a penalty cost but it also saves an item that can be used for future demands (of possibly higher class).

We assume a single replenishment server and the replenishment times can follow different distributions. In this paper we will consider hypoexponential and hyperexponential replenishments.

We detail now how this inventory problem can be expressed as an admission control problem in a queuing system with different classes of customers. We represent the items in replenishment for the inventory system by a queue of capacity S. An arriving demand of class j is an arriving customer of class j. Delivering an item is the equivalent to admitting the customer into the queue. The penalty cost due to a lost demand is equivalent to the cost of rejection. From now on, we will use this queuing model formalism.

2.2   Optimal control problem

The problem of finding the best decision (rejection or acceptance), to perform when customers arrive is an optimal control problem that can be modeled as a (time-homogeneous) MDP. We detail now the elements of this MDP.

The system behaves in continuous time, but we can restrict our attention to the transition epochs. This can be done without loss of generality since we only consider costs that depend on the state of the system. At these transition instants an event occurs and induces a change in the system. When the event is an arrival, then we have to choose the best action to trigger. Although the system behaves continuously, we consider here a discrete time MDP. Indeed, the use of the uniformization standard method makes the problem discrete (see Lippman [10] or Puterman [13]).

The state space of our MDP is $\mathcal{X} = \{1, \ldots, S\} \times \{1, \ldots, N\} \cup \{(0,1)\}$. For any $(x, k) \in \mathcal{X}$, the parameter x denotes the number of customers in the system, while k describes the phase of the customer being served (hypoexponential and hyperexponential have different phases). Actually, in state $(0, 1)$ no customer is served and the indication of the phase is useless but we keep it for readability. Note that the state space does not depend on the classes of customers.

We denote by $\mathcal{A} = \{0, 1\}$ the action set where action 1 is an acceptance and action 0 is a rejection. A *policy* π is a sequence of decision rules that maps the information history (past states and actions) to the action set $\mathcal{A}$. Since in our case the action set and the state space are finite, and since the costs are thus trivially bounded, we restrict our attention to Markov deterministic policies. Indeed, standard results (see Puterman [13]) insure that an optimal

policy belongs to the set of Markov deterministic policies. A Markov deterministic policy is of the form $(a(\cdot), a(\cdot), \ldots)$ where $a(\cdot)$ is a single deterministic decision rule that maps the current state to a decision (hence, in our case $a(\cdot)$ is a function from $\mathcal{X}$ to $\mathcal{A}$). Under a policy π the evolution of the system generates a random sequence of states and actions y_n and a_n.

We denote by $C(y, a)$ with $y \in \mathcal{X}$ and $a \in \mathcal{A}$ the instantaneous cost in state y when action a is triggered. Here $C(y, 0) = c_j$ according to the class of the customer we reject, and $C(y, 1) = 0$ in case of acceptance.

Since we are interested in the average cost criteria, we define the objective, known as the *minimal long-run average costs*, by

$$\bar{v}^* = \min_{\pi} \lim_{n \to \infty} \frac{1}{n} \mathbb{E}_y^{\pi} \left(\sum_{\ell=0}^{n-1} C(y_\ell, a_\ell) \right),$$

where the initial state $y_0 = y$. Our aim is to characterize the optimal policy π^* that reaches $\bar{v}^*$. This means to determine the optimal decision rule, denoted by a^*, with $\pi^* = (a^*, a^*, \ldots)$.

2.2.1 Total cost criteria.

We adopt the framework of Koole [8] and we will work with the total cost criteria to characterize the optimal average policy. We define the *minimal (expected) n-stage total cost* by

$$V_n(y) = \min_{\pi(n)} \mathbb{E}_y^{\pi(n)} \left(\sum_{\ell=0}^{n-1} C(y_\ell, a_\ell) \right), \forall y \in \mathcal{X}, \tag{1}$$

where $\pi(n)$ is a policy over n steps. We assume that the initial state $y_0 = y$ and that, by convention, $V_0(y) = 0$, $\forall y \in \mathcal{X}$. We call V_n the n-stage total cost value function.

From Koole [8] and Puterman [13, Chapter 8], the minimal n-stage total cost value function V_n does not converge when n tends to infinity. Instead, the difference $V_{n+1}(y) - V_n(y)$ converges to the minimal long run average cost. On the other hand, still from Koole [8], the n-stage policy which minimizes V_n tends to the optimal average policy π^* when n tends to infinity.

Finally, since the instantaneous costs are positive, from standard results on MDPs (see Puterman [13]), V_n is well defined for all n and satisfies the dynamic programming equation (Bellman equation):

$$V_{n+1} = TV_n, \tag{2}$$

where T is the dynamic programming operator. More precisely, T operates on the total cost value functions and is defined by:

$$(Tf)(y) = \min_a (\hat{T}f)(y, a) = \min_a \left(C(y, a) + \sum_{y' \in \mathcal{X}} \mathbb{P}\left(y'|(y, a)\right) f(y') \right), \tag{3}$$

with $\mathbb{P}\left(y'|(y, a)\right)$ the probability to move to y' from state $y \in \mathcal{X}$ with action $a \in \mathcal{A}$, and f a function that belongs to the total cost value function set. The optimal decision rule can then be deduced from Eq. (3):

$$a(y) = \min \left(\arg \min_{a \in \mathcal{A}} (\hat{T}f)(y, a) \right). \tag{4}$$

2.3   Critical level policies

In this work, we do not want to numerically compute the optimal policy but rather to characterize some "good" properties that the optimal policy should verify. Indeed, very often the optimal policy admits a simple form. The aim of our work is to prove that the optimal policy for our model is a critical level policy.

Definition 1 (Critical level policy). *A policy is called a critical level policy if for any fixed k and any customer class j it exists a level in x: $t_{k,j}$, such that in state (x, k):*
- for all $0 \leq x < t_{k,j}$ it is optimal to accept any customer of class j,
- for all $x \geq t_{k,j}$ it is optimal to reject any customer of class j.

2.3.1   Structured policy framework.

In order to show that the optimal policy exhibits some special properties we use a standard framework presented in the monograph of Koole [8]. Most of the results presented now follow from this monograph in which detailed proofs can be found. This framework is that of property preservation by the Dynamic Programming operator. It consists of three steps: First, identify two related sets. The one of the value functions that exhibit a special form (say $\mathcal{V}$) and the one one of the decision rules $\mathcal{D}$ that exhibit a special form. The relation between these two sets being such that when the value function belongs to $\mathcal{V}$, then the optimal decision belongs to $\mathcal{D}$. Next, show that the properties of $v \in \mathcal{V}$ are conserved by the operator T. Finally, check that these properties are kept when passing to the limit.

   We present now some special forms for value functions. Let f be a function from $\mathcal{X} \mapsto \mathbb{R}_+$.

Definition 2 (Monotonicity). *We say that a function f is increasing in x (denoted by $Incr(x)$) if, for all $y = (x, k)$, we have $f(x + 1, k) \geq f(x, k)$. Similarly, we say that a function f is decreasing in k (denoted by $Decr(k)$) if, for all $y = (x, k)$, we have $f(x, k + 1) \leq f(x, k)$.*

Definition 3 (Convexity). *We say that a function f is convex in x (denoted by $Convex(x)$) if, for all $y = (x, k)$, we have:*

$$2f(x + 1, k) \leq f(x, k) + f(x + 2, k). \tag{5}$$

Definition 4 (Submodularity). *We say that a function f is submodular in x and k (denoted by $Sub(x, k)$ if, for all $y = (x, k)$, we have:*

$$f(x + 1, k + 1) + f(x, k) \leq f(x + 1, k) + f(x, k + 1). \tag{6}$$

   We give now the relationship between the properties of the value functions and the optimal decisions.

Proposition 5 (Th 8.1 [8]). *Let $a(y)$, with $y = (x, k)$, be the optimal decision rule defined by Eq. (4).*
i) If $f \in Convex(x)$, then $a(y)$ is decreasing in x.
ii) If $f \in Sub(x, k)$, then $a(y)$ is increasing in k.

   Assume a critical level policy and consider a decision for a fixed demand class j.

Definition 6 (Switching curve). *For every k, we define a level $t(k)$ such that when we are in state (x, k) decision 1 is taken if and only if $x \leq t(k)$ and 0 otherwise. The mapping $k \mapsto t(k)$ is called a switching curve.*

Let us notice that a switching curve separates the state space $\mathcal{X}$ in two regions $\mathcal{X}_0$, in which decision 0 is taken, and $\mathcal{X}_1$, in which decision 1 is taken.

Definition 7 (Monotone switching curve). *We say that a decision rule is of the monotone switching curve type if the mapping $k \mapsto t(k)$ is monotone.*

3   Hypoexponential service

In order to show that the optimal level policy is of critical level type we will prove increasingness (Section 3.2) and convexity (Section 3.3) properties, while submodularity (proved in Section 3.5) implies monotonicity of the switching curve. Furthermore, although the problem was given under its most general formulation in Section 2.2, for the sake of readability and in order to simplify the proofs, in the rest of the paper we will decompose the dynamic programming operator according to the event-based approach presented in Glasserman and Yao [4], and Koole [8]. This means a decomposition of the dynamic programming operator into operators related to a dedicated event and an observation of the system after decision epoch.

3.1   Model description

Let us detail the event-based decomposition. The dynamic programming equation (2) becomes:

$$V_n = T_{unif}\Big(\sum_{i=1}^{J} p_i T_{CA(i)}(V_{n-1}), T_D(V_{n-1})\Big), \tag{7}$$

where $V_0(x, k) \equiv 0$, and T_{unif}, $T_{CA(i)}$ and T_D are different event operators.

If the service (replenishment) time is hypoexponential, then the random value R that describes the service time is defined by $R = \sum_{k=1}^{N} E_k$, where E_k is an exponential random value of parameter μ_k, $1 \leq k \leq N$. Let $\alpha = \max_k \mu_k$. We will use the uniformization rate equal to $\lambda + \alpha$. The uniformization operator, T_{unif}, is defined by:

$$T_{unif}(f(x, k), g(x, k)) = \frac{\lambda}{\lambda + \alpha} f(x, k) + \frac{\alpha}{\lambda + \alpha} g(x, k).$$

Let $\mu_k' = \mu_k/\alpha$, the departure operator, $T_D f$, is defined by:

$$T_D f(x, k) = \mu_k' \begin{cases} f(x, k+1) & \text{if } (k < N) \text{ and } (x > 0), \\ f((x-1)^+, 1) & \text{if } (k = N) \text{ or } (x = 0 \text{ and } k = 0) \end{cases}$$
$$+ (1 - \mu_k') f(x, k).$$

The controlled arrival operator of a customer of class i, $T_{CA(i)}$, is defined by:

$$T_{CA(i)} f(x, k) = \begin{cases} \min\{f(x+1, k), \ f(x, k) + c_i\} & \text{if } x < S, \\ f(x, k) + c_i & \text{if } x = S. \end{cases}$$

3.2 Increasing property

We first notice that the state space admits a linear order (denoted $\preceq$) which is a lexicographic order defined to be increasing with respect to the first dimension and decreasing with respect to the second dimension:

Definition 8 (Linear order $\preceq$). *For every (x,i) and (y,j) in $\mathcal{X}$,*

$$(x,i) \preceq (y,j) \quad \Leftrightarrow \quad (x < y) \ \ or \ \ (x = y, \ i \geq j).$$

Let us define the increasing property of functions with respect to $\preceq$.

Definition 9 (Increasing property). *For any function $f : \mathcal{X} \mapsto \mathbb{R}$ we say that $f \in Incr(\preceq)$ if $(x,i) \preceq (y,j) \Rightarrow f(x,i) \leq f(y,j)$.*

We now show that operators $T_{CA(i)}$ and T_D preserve this increasing property.

Lemma 10. *Let $f : \mathcal{X} \mapsto \mathbb{R}$ such that $f \in Incr(\preceq)$, then $T_{CA(i)}f \in Incr(\preceq)$.*

Proof. Assume that $f \in \mathrm{Incr}(\preceq)$. In order to prove that $T_{CA(i)}f \in \mathrm{Incr}(\preceq)$, it is sufficient to prove that:

$$
\begin{align}
T_{CA(i)}f(x,k) &\leq T_{CA(i)}f(x,k-1), \forall x > 0, \forall k > 1, &(8)\\
T_{CA(i)}f(x,1) &\leq T_{CA(i)}f(x+1,N), \forall x, \forall k. &(9)
\end{align}
$$

Let us consider first (8). For $0 < x < S$ and $k > 1$ we get:

$$
\begin{align}
T_{CA(i)}f(x,k) &= \min\{f(x,k) + c_i, f(x+1,k)\}\\
&\leq \min\{f(x,k-1) + c_i, f(x+1,k-1)\} = T_{CA(i)}f(x,k-1),
\end{align}
$$

For $x = S$ and $k > 1$ we have:

$$T_{CA(i)}f(S,k) = f(S,k) + c_i \leq f(S,k-1) + c_i = T_{CA(i)}f(S,k-1).$$

We now prove (9). For $x < S - 1$ we get:

$$
\begin{align}
T_{CA(i)}f(x,1) &= \min\{f(x,1) + c_i, f(x+1,1)\}\\
&\leq \min\{f(x+1,k) + c_i, f(x+2,k)\} = T_{CA(i)}f(x+1,k).
\end{align}
$$

For $x = S - 1$,

$$T_{CA(i)}f(S-1,1) = \min\{f(S-1,1) + c_i, f(S,1)\} \leq f(S,k) + c_i = T_{CA(i)}f(S,k).$$

$\square$

Lemma 11. *Let $f : \mathcal{X} \mapsto \mathbb{R}$ such that $f \in Incr(\preceq)$, then $T_D f \in Incr(\preceq)$.*

Proof. Assume that f is $\mathrm{Incr}(\preceq)$. In order to prove that $T_D f \in \mathrm{Incr}(\preceq)$, it is sufficient to prove that:

$$
\begin{align}
T_D f(x,k) &\leq T_D f(x,k-1), \forall x > 0, \forall k > 1, &(10)\\
T_D f(x,1) &\leq T_D f(x+1,N), \forall x, \forall k. &(11)
\end{align}
$$

Let us consider first (10). For $x > 0$ and $k = N$,

$$T_D f(x, N) = \mu'_N f(x-1, 1) + (1 - \mu'_N) f(x, N)$$
$$\leq \mu'_{N-1} f(x, N) + (1 - \mu'_{N-1}) f(x, N-1) = T_D f(x, N-1).$$

Indeed, using Definition 9, one has that both $f(x-1, 1)$ and $f(x, N)$ are smaller or equal than $f(x, N)$ and $f(x, N-1)$ and the same holds for any of their convex combinations. Similarly, for $x > 0$ and $1 < k < N$ we have:

$$T_D f(x, k) = \mu'_k f(x, k+1) + (1 - \mu'_k) f(x, k)$$
$$\leq \mu'_{k-1} f(x, k) + (1 - \mu'_{k-1}) f(x, k-1) = T_D f(x, k-1).$$

Let us now consider (11). For $x > 0$:

$$T_D f(x, 1) = \mu'_1 f(x, 2) + (1 - \mu'_1) f(x, 1)$$
$$\leq \mu'_N f(x, 1) + (1 - \mu'_N) f(x+1, N) = T_D f(x+1, N)$$

For $x = 0$: $T_D f(0, 1) = f(0, 1) = T_D f(1, N)$. □

Remark 12. *From Definition 8, it follows that $Incr(\preceq)$ implies $Incr(x)$ and $Decr(k)$.*

3.3 Convexity

In this part we prove that the convexity with respect to the first dimension is preserved by operators $T_{CA(i)}$ and T_D. However, due to the state space form, this requires to introduce an additional property (called augmented convexity) at state $(0, 1)$.

Definition 13 (Augmented convexity). *We say that a function $f : \mathcal{X} \mapsto \mathbb{R}$ is augmented convex in x (denoted by $AConvex(x)$) if for all $k \in \{1, .., N\}$ we have $f(0, 1) + f(2, k) \geq 2f(1, k)$.*

Lemma 14. *Let f be a function such that $f \in Incr(\preceq) \cap Convex(x) \cap AConvex(x)$, then $T_{CA(i)} f \in Incr(\preceq) \cap Convex(x) \cap AConvex(x)$.*

Proof. Assume that f is $Incr(\preceq) \cap Convex(x) \cap AConvex(x)$.
(i) Preservation of $Convex(x)$ shall be proved for $x > 0$. For $x = S - 2$,

$$T_{CA(i)} f(x,k) + T_{CA(i)} f(S,k) = \min \begin{cases} f(x+1,k) + f(S,k) + c_i \\ f(x,k) + c_i + f(S,k) + c_i \geq 2f(x+1,k) + 2c_i \end{cases}$$
$$\geq 2 \min\{f(S,k), f(x+1,k) + c_i\} = 2T_{CA(i)} f(x+1,k).$$

For $0 < x < S - 2$, the sum $T_{CA(i)} f(x, k) + T_{CA(i)} f(x+2, k)$ is equal to

$$\min \begin{cases} f(x+1,k) + f(x+3,k) \\ f(x+1,k) + f(x+2,k) + c_i \\ f(x,k) + c_i + f(x+3,k) + f(x+2,k) - f(x+2,k) \\ f(x,k) + c_i + f(x+2,k) + c_i \end{cases}$$

$$\geq \min \begin{cases} 2f(x+2,k) \\ f(x+1,k) + f(x+2,k) + c_i \\ f(x+1,k) + f(x+2,k) + c_i \\ 2f(x+1,k) + 2c_i \end{cases}$$

$$\geq 2\min\{f(x+2,k), \ f(x+1,k) + c_i\} = 2T_{CA(i)} f(x+1,k).$$

Note that for the third case we used:

$$f(x,k)+c_i+f(x+3,k)+f(x+2,k)-f(x+2,k) \geq 2f(x+1,k)+f(x+3,k)-f(x+2,k)+c_i.$$

(ii) For $x = 0$, we show that augmented convexity is preserved:

$$T_{CA(i)}f(0,1)+T_{CA(i)}f(2,k)=\min \begin{cases} f(1,1)+f(3,k)\geq f(1,k)+f(3,k)\geq 2f(2,k) \\ f(1,1)+f(2,k)+c_i \\ f(0,1)+c_i+f(3,k)+f(2,k)-f(2,k) \\ \quad \geq 2f(1,k)+f(3,k)-f(2,k)+c_i \\ \quad \geq f(1,k)+f(2,k)+c_i \\ f(0,1)+c_i+f(2,k)+c_i \geq 2f(1,k)+2\,c_i \end{cases}$$
$$\geq 2\min\{f(2,k),\ f(1,k)+c_i\} = 2T_{CA(i)}f(1,k),$$

Preservation of Incr$(\preceq)$ follows from Lemma 10. $\square$

Lemma 15. *Let f be a function such that $f \in Incr(\preceq) \cap Convex(x) \cap AConvex(x)$, then $T_D f \in Incr(\preceq) \cap Convex(x) \cap AConvex(x)$.*

Proof. Assume that f is in Incr$(\preceq) \cap$ Convex$(x) \cap$ AConvex(x).
(i) Preservation of Convex(x) shall be proved for $x > 0$. For $x > 0$ and $k < N$:

$$\begin{aligned} &T_D f(x,k) + T_D f(x+2,k) \\ &= \mu'_k f(x,k+1) + (1-\mu'_k)f(x,k) + \mu'_k f(x+2,k+1) + (1-\mu'_k)f(x+2,k) \\ &\geq 2\mu'_k f(x+1,k+1) + 2(1-\mu'_k)f(x+1,k) = 2T_D f(x+1,k) \end{aligned}$$

For $x > 0$ and $k = N$:

$$\begin{aligned} &T_D f(x,N) + T_D f(x+2,N) \\ &= \mu'_N f(x-1,1) + (1-\mu'_N)f(x,N) + \mu'_N f(x+1,1) + (1-\mu'_N)f(x+2,N) \\ &\geq 2\mu'_N f(x,1) + 2(1-\mu'_N)f(x+1,N) = 2T_D f(x+1,N) \end{aligned}$$

(ii) For $x = 0$ augmented convexity is preserved. For $x = 0$ and $k < N$:

$$\begin{aligned} T_D f(0,1)+T_D f(2,k)&=\mu'_1 f(0,1)+(1-\mu'_1)f(0,1)+\mu'_k f(2,k+1) + (1-\mu'_k)f(2,k) \\ &=\mu'_k f(0,1)+(1-\mu'_k)f(0,1)+\mu'_k f(2,k+1)+(1-\mu'_k)f(2,k) \\ &\geq 2\mu'_k f(1,k+1) + 2(1-\mu'_k)f(1,k) = 2T_D f(1,k) \end{aligned}$$

For $x = 0$ and $k = N$:

$$\begin{aligned} T_D f(0,1)+T_D f(2,N)&=\mu'_1 f(0,1)+(1-\mu'_1)f(0,1)+\mu'_N f(1,1)+(1-\mu'_N)f(2,N) \\ &=\mu'_N f(0,1)+\mu'_N f(1,1)+(1-\mu'_N)f(0,1)+(1-\mu'_N)f(2,N) \\ &\geq 2\mu'_N f(0,1) + 2(1-\mu'_N)f(1,N) = 2T_D f(1,N) \end{aligned}$$

Preservation of Incr$(\preceq)$ follows from Lemma 11. $\square$

We aggregate now all the event-based operators to get the preservation of structural properties passing through the whole dynamic programming operator.

Proposition 16. *Let V_n be a n-steps total cost value function that satisfy Equation 7. Then, for all $n \geq 0$, V_n is in $Incr(\preceq) \cap AConvex(x) \cap Convex(x)$.*

Proof. First, let us note that if for all s.t. $i \in \{1, ..., n\}$ we have $f_i \in Incr(\preceq) \cap AConvex(1) \cap Convex(1)$ then we have

$$\sum_{i=1}^{n} p_i f_i \in Incr(\preceq) \cap AConvex(1) \cap Convex(1). \tag{12}$$

This follows from the fact that we have a direct sum of functions each of them satisfying the properties. Furthermore, since the operator T_{unif} is a convex sum of two functions, we get that if $f, g \in Incr(\preceq) \cap AConvex(1) \cap Convex(1)$ then

$$T_{unif}(f, g) \in Incr(\preceq) \cap AConvex(1) \cap Convex(1) \tag{13}$$

Thus, combination of Lemmas 14, 15 with observations (12) and (13) yields

$$V_n \in Incr(\preceq) \cap AConvex(1) \cap Convex(1) \Rightarrow V_{n+1} \in Incr(\preceq) \cap AConvex(1) \cap Convex(1)$$

which, along with the fact that $V_0 \equiv 0$ is in $Incr(\preceq) \cap AConvex(1) \cap Convex(1)$, proves the result. $\square$

3.4 Critical level policy

The next step is to show that the optimal policy, as defined in Section 2.2.1, is a critical level policy (see Definition 1).

Theorem 17. *The optimal policy is a critical level policy.*

Proof. First, any value function V_n satisfying Eq. (7) belongs to $Convex(x)$ this follows from Proposition 16. From Proposition 5 it follows that, for any n, the action minimizing V_n when $V_n \in Convex(x)$ defines a critical level policy. Finally, from the discussion in 2.2.1 it follows that the policy minimizing V_n converges (with a simple convergence) to the optimal policy when n tends to infinity. Since convexity is kept with simple convergence, then the optimal policy is of critical level policy type. $\square$

Example 18. *Consider a model with the following parameters as an example: $J = 3$ (number of customer classes), $N = 5$ (number of phases), $S = 10$ (stock size), $\lambda = 3$, $\mu = [2, 6, 9, 4, 7]$ (service rates in different phases), $p = [0.3, 0.4, 0.3]$ (probabilities of J customer classes), $c = [30, 40, 50]$ (rejection costs).*
  Numerical computation of the optimal policy using value iteration method results in a critical level policy as presented in Figure 1. It can be seen that for every customer class j and for every phase k there exists a unique threshold in x: $t_{k,j}$. These thresholds are represented on the figure as the transition between acceptance for different classes (blue circle – all classes are accepted, green triangle – classes 2 and 3 are accepted, pink square – only class 3 is accepted, red asterisk – rejection of any class). This example and Figure 1 also serve as an illustration for Theorems 19 and 24.

Now we show that when the rejection costs are ordered for consecutive customer classes, the critical levels in x are ordered.
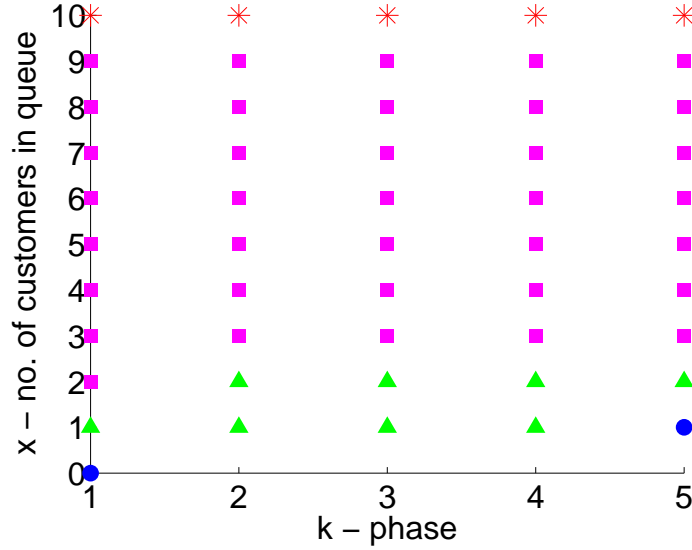
Figure 1: Acceptance points for different customer classes. Blue circle – all classes are accepted, green triangle – classes 2 and 3 are accepted, pink square – only class 3 is accepted, red asterisk – rejection of any class.

Theorem 19. *For any critical level policy (see Definition 1), if the rejection costs are nondecreasing ($c_1 \leq \cdots \leq c_J$), then the levels $t_{k,j}$ are nondecreasing with respect to customer class j, i.e. $t_{k,j} \leq t_{k,j+1}$.*

Proof. First recall that the operator $T_{CA(j)}$ is defined as: $T_{CA(j)}f(x,k) = \min \{f(x+1,k), f(x,k) + c_j\}$ for $x < S$ (for $x = S$ the only allowed action is rejection).

Observe now that by definition $t_{k,j}$ is a number such that: in state (x,k) for all $0 \leq x < t_{k,j}$ it is optimal to accept any customer of class j and to reject it for all $x \geq t_{k,j}$. This, by definition of $T_{CA(j)}$, means that:

$$V_n(x+1,k) - V_n(x,k) < c_j \quad \forall x < t_{k,j} \tag{14}$$
$$V_n(x+1,k) - V_n(x,k) \geq c_j \quad \forall x \geq t_{k,j}$$

Owing to the assumption $c_j \leq c_{j+1}$, it yields $V_n(x+1,k) - V_n(x,k) < c_j \leq c_{j+1}$ which, by definition of $t_{k,j+1}$ holds for all $x < t_{k,j+1}$. But, by Eq. (14), this also holds for all $x < t_{k,j}$. This means that the set of x such that $x < t_{k,j}$ is a subset of the set of x such that $x < t_{k,j+1}$, which, by definition of $t_{k,j}$ implies that $t_{k,j} \leq t_{k,j+1}$. □

3.5　Submodularity

In this part we prove the preservation by operators $T_{CA(i)}$ and T_D of submodularity with respect to the first and second dimension (Sub(x,k)). Similarly to convexity, we need an additional property (boundary-submodularity) and subsequently its preservation along with submodularity shall be proved too.

Definition 20 (Boundary-submodularity). *We say that a function f is in $BSub(x,k)$ if $\forall 0 < x < S$ we have $f(x,1) + f(x,N) \leq f(x-1,1) + f(x+1,N)$.*

Lemma 21. *Let f be a function such that $f \in Sub(x,k) \cap BSub(x,k)$. Then $T_{CA(i)}f \in Sub(x,k) \cap BSub(x,k)$.*

Proof. The proof is postponed in Appendix A.2. □

Lemma 22. *Let f be a function such that $f \in Sub(x,k) \cap BSub(x,k)$. Then $T_D f \in Sub(x,k) \cap BSub(x,k)$.*

Proof. The proof is postponed in Appendix A.3. □

Also here we prove that the submodularity properties are kept by passing through the whole dynamic programming operator.

Proposition 23. *Let V_n be a n-step total cost value function which satisfies Eq. (7). Then for all ($n \geq 0$) V_n is in $Sub(x,k) \cap BSub(x,k)$.*

Proof. Using the same arguments than in the proof of Proposition 16, we obtain that if for all $i \in \{1, \dots, n\}$ we have $f_i \in \mathrm{BSub}(1,2) \cap \mathrm{Sub}(1,2)$ then

$$\sum_{i=1}^{n} p_i f_i \in \mathrm{BSub}(1,2) \cap \mathrm{Sub}(1,2). \tag{15}$$

We also obtain that if $f, g \in \mathrm{BSub}(1,2) \cap \mathrm{Sub}(1,2)$, then

$$T_{unif}(f,g) \in \mathrm{BSub}(1,2) \cap \mathrm{Sub}(1,2) \tag{16}$$

Combination of Lemmas 21, 22 with observations (15) and (16) yields

$$V_n \in \mathrm{BSub}(1,2) \cap \mathrm{Sub}(1,2) \quad \Rightarrow V_{n+1} \in \mathrm{BSub}(1,2) \cap \mathrm{Sub}(1,2)$$

which, along with the fact that $V_0 \equiv 0$ is in $\mathrm{BSub}(1,2) \cap \mathrm{Sub}(1,2)$ proves the result. □

3.6 Monotone switching curve

Henceforth we can now refine the optimal policy using Definition 6.

Theorem 24. *The optimal policy defines a monotone switching curve.*

Proof. For all $n \geq 0$, a value function V_n that checks Eq. (7) belongs to $\mathrm{Sub}(x,k)$ (this follows from Proposition 23). Thus, by Theorem 17, we know that action is decreasing in x and that there exists a critical level policy. Since the action set consists of two elements and since the switching curve is the boundary between inverse images of these two elements with respect to the Markov deterministic policy, then Theorem 17 yields that for a fixed customer class the boundary (switching curve) is a function $k \mapsto t_{k,j}$. Furthermore, from Prop. 5 it follows that the action minimizing V_n for $V_n \in \mathrm{Sub}(x,k)$ is increasing in k. This, combined with the previous observation, implies that the switching curve is increasing.

Observe now, that (from the discussion in 2.2.1) the policy minimizing V_n converges to the optimal policy when n tends to infinity. Since submodularity is kept by simple convergence, thus the optimal policy is a monotone switching curve.

At last, it has to be noted that submodularity is not well defined for the state $(0,1)$ (indeed when $x = 0$ only one phase is possible). Therefore the same reasoning based on

Proposition 5 cannot be applied to the zero state. However, Theorem 17 ensures that an unique level $t_{1,j}$ for $k = 1$ exists. Submodularity, by application of Proposition 5, implies the fact that the switching curve is increasing for $x > 0$. Finally, the only event not covered by the previous discussion (*i.e.* the existence of the switching point at $(0, 1)$) does not impair the monotonicity of the switching curve, since it is the smallest possible state. $\qquad\square$

4 Model extensions

We discuss the possible extensions of the model in two different directions: hyperexponential service times and including different instantaneous costs.

4.1 Hyperexponential service

We now assume that the service times follow a hyperexponential distribution (an exponential distribution of rate μ_k with a probability γ_k). Let V_n be the n-step value function, the dynamic programming equation is similar: $V_n(x, k) = T_{unif}\left(\sum_{i=1}^{J} p_i T_{CA(i)}(V_{n-1}), T_D(V_{n-1})\right)$, with $V_0(x, k) \equiv 0$. However, since the dynamics of the system changes compared to the hypoexponential case, the definitions of controlled arrivals $T_{CA(i)}$ and departure operators T_D differ.

Let us recall that α is the maximum over all μ_k and that $\mu'_k = \frac{\mu_k}{\alpha}$. We assume, without loss of generality, that $\mu_1 \le \mu_2 \le \ldots \le \mu_N$. The departure operator T_D is now defined by:

$$T_D f(x, k) = \mu'_k \begin{cases} \sum_{i=1}^{N} a_i f(x - 1, i) & \text{if } x > 1 \\ f(0, 1) & \text{if } x \le 1 \end{cases}$$
$$+ (1 - \mu'_k) f(x, k),$$

and the controlled arrivals $T_{CA(i)}$ by:

$$T_{CA(i)} f(x, k) = \begin{cases} f(x, k) + c_i & \text{if } x = S, \\ \min\{f(x + 1, k), \ f(x, k) + c_i\} & \text{if } 0 < x < S, \\ \min\{f(0, 1) + c_i, \ \sum_{j=1}^{N} a_j f(1, j)\} & \text{if } x = 0. \end{cases}$$

Example 25. *Consider a model with the following parameters: $J = 3$ (number of customer classes), $N = 5$ (number of phases), $S = 10$ (stock size), $\lambda = 3$, $\mu = [2, 4, 6, 7, 9]$ (service rates for different phases), $p = [0.3, 0.4, 0.3]$ (probabilities of J customer classes), $c = [0.5, 1, 3]$ (rejection costs). The optimal policy is given in Figure 2. For this example, and many others that we have tested, the optimal policy is still of critical level type. This suggests that even in the hyperexponential service case, the optimal policies are critical level policies. However, since the arrival operator is now more complex at the zero state, the proof technique used for hypoexponential case cannot be used directly, and were not able to prove any results yet about critical level optimality.*

4.2 Adding holding costs

We now discuss the possibility of adding holding costs to Eq. (7). As mentionned in De Véricourt et al. [2], this breaks the queueing model and the inventory system similarities. Thus the two models should be studied separately.
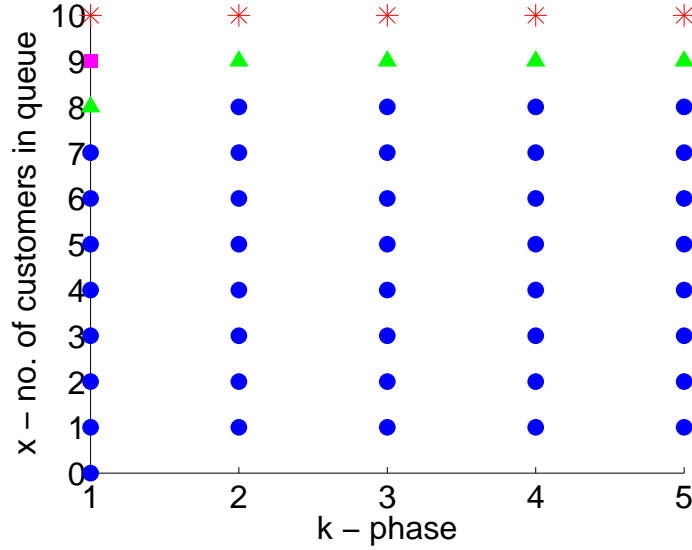
Figure 2: Acceptance points for different customer classes. Blue circle – all classes are accepted, green triangle – classes 2 and 3 are accepted, pink square – only class 3 is accepted, red asterisk – rejection of any class.

The admission control in a queueing system model with holding costs has a wide range of applications (see Kleywegt et al. [7]). In the context of inventory systems, it models the make-to-stock problem (see Ha [5]).

We obtain the similar results as before when we add linear and increasing holding costs in admission control model.

Corollary 26 (Critical level policy with holding costs)**.** *Let T_{cost} be the operator defined by*

$$T_{cost}f(x,k) = \frac{c_h x}{\lambda + \alpha} + f(x,k) ,$$

with c_h the per capita holding cost. The value function is given by $V_n(x,k) = T_{cost}(T_p(V_{n-1}))$ where T_p is the operator given by Eq. (7).

The optimal policy is of critical level policy type. The levels are increasing with respect to the customer class and the optimal policy defines a monotone switching curve.

Proof. We only give a sketch of proof by noticing that the increasing, convexity, and submodularity properties are kept by the operator. Then the results follow using similar arguments as in the proofs of Theorems 17, 19, and 24. □

5   Conclusions

The main contribution of this paper is the proof that in the case of a hypoexponential service case we still have an optimal policy of critical level type. This extends the previous results from the literature to the simplest case of a phase-type service where the service rate depends on the service phase. We expect that our result is a first step towards more general results for phase-type service. Our numerical experiments suggest that the result is still preserved for

hyperexponential case. However, the particularity of the zero state (i.e. when the queue is empty) becomes more challenging because of the possibility of choice in the arrival operator when the queue is empty. Technically, on the boundary around zero state convexity and submodularity must be replaced with some more elaborate properties. Defining the boundary properties for general phase type service and finding the conditions under which they are preserved by the dynamical programming operator represents our first direction for future extensions of this work. The second direction concerns the extensions of the result for different types of immediate costs. Finally, it would be interesting to consider other types of arrival processes or adding the possibility of decision for service.

5.0.1 Acknowledgements.

References

[1] E. Altman. Applications of Markov Decision Processes in communication networks. In E.A. Feinberg and A. Shwartz, editors, *Handbook of Markov Decision Processes*, chapter 16. Kluwer, 2002.

[2] F. de Véricourt, F. Karaesmen, and Y. Dallery. Optimal stock allocation for a capacitated supply system. *Management Science*, 48:1486–1501, 2002.

[3] R. Dekker, R.M. Hill, M.J. Kleijn, and R.H. Teunter. On the (S-1,S) lost sales inventory model with priority demand classes. *Naval Research Logistics*, 49:593–610, 2002.

[4] P. Glasserman and D. Yao. *Monotone Structure in Discrete-Event Systems*. Wiley, 1994.

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

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

[7] A. J. Kleywegt and J. D. Papastavrou. The dynamic and stochastic knapsack problem. *Operations Research*, 46:17–35, 1998.

[8] G. Koole. Monotonicity in Markov reward and decision chains: Theory and applications. *Foundation and Trends in Stochastic Systems*, 1(1), 2006.

[9] A.A. Kranenburg and G.J. van Houtum. Cost optimization in the (S-1, S) lost sales inventory model with multiple demand classes. *Oper. Res. Lett.*, 35(4):493–502, 2007.

[10] S.A. Lippman. Applying a new device in the optimization of exponential queueing systems. *Operations Research*, 23:687–710, 1975.

[11] M. Mazzucco, I. Mitrani, J. Palmer, M. Fisher, and P. McKee. Web service hosting and revenue maximization. In *Fifth European Conference on Web Services. ECOWS '07.*, pages 45 –54, 2007.

[12] E. L. Porteus. *Foundations of Stochastic Inventory Theory*. Stanford University Press, 2002.

[13] M. Puterman. *Markov Decision Processes Discrete Stochastic Dynamic Programming*. Wiley, 2005.

[14] A. Svoronos and P. Zipkin. Evaluation of one-for-one replenishment policies for multi-echelon inventory systems. *Management Science*, 37(1), 1991.

A Proofs

A.1 Preliminary Lemma

The following observation is used extensively within the proofs:

Lemma 27. *For any real numbers $A \leq B \leq C$ and $0 \leq \theta_1, \theta_2 \leq \gamma$ we have $\theta_1 A + (\gamma - \theta_1)B \leq \theta_2 B + (\gamma - \theta_2)C$.*

A.2 Proof of Lemma 21

Assume that f is $\mathrm{Sub}(x,k) \cap \mathrm{BSub}(x,k)$. Propagation of submodularity (i) will be proved first, followed by the propagation of boundary-submodularity (ii).

(i) Sub is preserved by $T_{CA(i)}$. For $x < S - 1$

$$T_{CA(i)}f(x,k) + T_{CA(i)}f(x+1,k+1)$$

$$= \min \begin{cases} f(x,k) + f(x+1,k+1) + 2c_i \leq f(x,k+1) + f(x+1,k) + 2c_i \\ f(x,k) + f(x+2,k+1) + c_i \\ \quad = f(x,k) + f(x+2,k+1) + f(x+1,k+1) - f(x+1,k+1) + c_i \\ \quad \leq f(x+1,k) + f(x,k+1) + f(x+2,k+1) - f(x+1,k+1) + c_i \\ \quad \leq f(x+2,k) + f(x+1,k+1) + f(x,k+1) - f(x+1,k+1) + c_i \\ \quad \leq f(x,k+1) + f(x+2,k) + c_i \\ f(x+1,k) + f(x+1,k+1) + c_i \\ f(x+1,k) + f(x+2,k+1) \leq f(x+1,k+1) + f(x+2,k) \end{cases}$$

$$= \min\{f(x,k+1) + c_i, \ f(x+1,k+1)\} + \min\{f(x+1,k) + c_i, \ f(x+2,k)\}$$

$$= T_{CA(i)}f(x,k+1) + T_{CA(i)}f(x+1,k),$$

and for $x = S - 1$

$$T_{CA(i)}f(x,k) + T_{CA(i)}f(S,k+1)$$

$$= \min \begin{cases} f(x,k) + f(S,k+1) + 2c_i \leq f(x,k+1) + f(S,k) + 2c_i \\ f(S,k) + f(S,k+1) + c_i \end{cases}$$

$$= f(S,k) + c_i + \min\{f(S,k+1) + c_i, \ f(x,k+1)\}$$

$$= T_{CA(i)}f(S,k) + T_{CA(i)}f(x,k+1).$$

(ii) BSub is preserved by $T_{CA(i)}$. For $x < S - 1$

$$T_{CA(i)}f(x, N) + T_{CA(i)}f(x, 1)$$

$$= \min \begin{cases} f(x, N) + f(x, 1) + 2c_i \le f(x - 1, 1) + f(x + 1, N) + 2c_i \\ f(x, N) + f(x + 1, 1) + c_i = f(x, N) + f(x + 1, 1) + f(x, 1) - f(x, 1) + c_i \\ \quad \le f(x, 1) + f(x + 2, N) + f(x - 1, 1) - f(x, 1) + c_i \\ \quad = f(x + 2, N) + f(x - 1, 1) + c_i \\ f(x + 1, N) + f(x, 1) + c_i \\ f(x + 1, N) + f(x + 1, 1) \le f(x, 1) + f(x + 2, N) \end{cases}$$

$$= \min\{f(x - 1, 1) + c_i, \ f(x, 1)\} + \min\{f(x + 1, N) + c_i, \ f(x + 2, N)\}$$

$$= T_{CA(i)}f(x - 1, 1) + T_{CA(i)}f(x + 1, N),$$

and for $x = S - 1$

$$T_{CA(i)}f(x, N) + T_{CA(i)}f(x, 1) = \min \begin{cases} f(x, N) + f(x, 1) + 2c_i \le f(x - 1, 1) + f(S, N) + 2c_i \\ f(x, N) + f(S, 1) + c_i \\ f(S, N) + f(x, 1) + c_i \\ f(S, N) + f(S, 1) \end{cases}$$

$$\le \min \begin{cases} f(x - 1, 1) + f(S, N) + 2c_i \\ f(x, 1) + f(S, N) + c_i \end{cases}$$

$$= f(S, N) + c_i + \min\{f(x - 1, 1) + c_i, \ f(x, 1)\}$$

$$= T_{CA(i)}f(S, N) + T_{CA(i)}f(x - 1, 1).$$

$\square$

A.3   Proof of Lemma 22

Assume that f is $\mathrm{Incr}(\preceq) \cap \mathrm{Sub}(x, k) \cap \mathrm{BSub}(x, k)$. Propagation of $\mathrm{Incr}(\preceq)$ follows from Lemma 11. Propagation of submodularity (i) will be proved first, followed by the propagation of boundary submodularity (ii). Both properties need to be considered in two cases which makes the proof consist of four parts. All the parts use the same reasoning and make use of Lemma 27 feeding in different inequalities.

(i) Sub is preserved by T_D. For $x > 0$ and $k < N - 1$ in order to prove propagation of Sub it has to be shown that:

$$T_D f(x, k) + T_D f(x + 1, k + 1)$$
$$= \mu'_k f(x, k + 1) + (1 - \mu'_k) f(x, k) + \mu'_{k+1} f(x + 1, k + 2) + (1 - \mu'_{k+1}) f(x + 1, k + 1)$$
$$\le \mu'_{k+1} f(x, k + 2) + (1 - \mu'_{k+1}) f(x, k + 1) + \mu'_k f(x + 1, k + 1) + (1 - \mu'_k) f(x + 1, k)$$
$$= T_D f(x, k + 1) + T_D f(x + 1, k) \tag{17}$$

Submodularity, given as assumption, at $f(x, k)$ and $f(x, k + 1)$ respectively can be rewritten as:

$$f(x, k) - f(x + 1, k) \le f(x, k + 1) - f(x + 1, k + 1) \tag{18}$$
$$f(x, k + 1) - f(x + 1, k + 1) \le f(x, k + 2) - f(x + 1, k + 2) \tag{19}$$

Now, since the right hand side of inequality (18) is equal to the left hand side of inequality (19), Lemma 27 can be used with $\theta_1 = (1 - \mu'_k)$ and $\theta_2 = (1 - \mu'_{k+1})$ and $\gamma = 1$. Application of this lemma yields:

$$\mu'_k[f(x, k+1) - f(x+1, k+1)] + (1 - \mu'_k)[f(x, k) - f(x+1, k)]$$
$$\leq \mu'_{k+1}[f(x, k+2) - f(x+1, k+2)] + (1 - \mu'_{k+1})[f(x, k+1) - f(x+1, k+1)]$$

which is equivalent to the inequality (17) to be proved (after rearrangement of terms so that every value of f is positive).

For $x > 0$ and $k = N - 1$ in order to prove propagation of Sub it has to be shown that:

$$
\begin{aligned}
&T_D f(x, k) + T_D f(x+1, N) \\
&= \mu'_k f(x, N) + (1 - \mu'_k) f(x, k) + \mu'_N f(x, 1) + (1 - \mu'_N) f(x+1, N) \\
&\leq \mu'_N f(x-1, 1) + (1 - \mu'_N) f(x, N) + \mu'_k f(x+1, N) + (1 - \mu'_k) f(x+1, k) \\
&= T_D f(x, N) + T_D f(x+1, k)
\end{aligned}
\tag{20}
$$

Submodularity, given as assumption, at $f(x, k)$ and boundary-submodularity, also given as assumption, at $f(x, N)$ can be rewritten as:

$$f(x, k) - f(x+1, k) \leq f(x, N) - f(x+1, N) \tag{21}$$
$$f(x, N) - f(x+1, N) \leq f(x-1, 1) - f(x, 1) \tag{22}$$

Now, since the right hand side of inequality (21) is equal to the left hand side of inequality (22), Lemma 27 can be used with $\theta_1 = (1 - \mu'_k)$ and $\theta_2 = (1 - \mu'_N)$ and $\gamma = 1$. Application of this lemma yields:

$$\mu'_k[f(x, N) - f(x+1, N)] + (1 - \mu'_k)[f(x, k) - f(x+1, k)]$$
$$\leq \mu'_N[f(x-1, 1) - f(x, 1)] + (1 - \mu'_N)[f(x, N) - f(x+1, N)]$$

which is equivalent to the inequality (20) to be proved.

(ii) BSub is preserved by T_D. For $x > 1$ in order to prove propagation of BSub it has to be shown that:

$$
\begin{aligned}
&T_D f(x, N) + T_D f(x, 1) \\
&= \mu'_N f(x-1, 1) + (1 - \mu'_N) f(x, N) + \mu'_1 f(x, 2) + (1 - \mu'_1) f(x, 1) \\
&\leq \mu'_1 f(x-1, 2) + (1 - \mu'_1) f(x-1, 1) + \mu'_N f(x, 1) + (1 - \mu'_N) f(x+1, N) \\
&= T_D f(x-1, 1) + T_D f(x+1, N)
\end{aligned}
\tag{23}
$$

Boundary-submodularity, given as assumption, at $f(x, N)$ and submodularity, also given as assumption, at $f(x-1, 1)$ respectively can be rewritten as:

$$f(x, N) - f(x+1, N) \leq f(x-1, 1) - f(x, 1) \tag{24}$$
$$f(x-1, 1) - f(x, 1) \leq f(x-1, 2) - f(x, 2) \tag{25}$$

Now, since the right hand side of inequality (24) is equal to the left hand side of inequality (25), Lemma 27 can be used with $\theta_1 = (1 - \mu'_N)$ and $\theta_2 = (1 - \mu'_1)$ and $\gamma = 1$. Application of this lemma yields:

$$\mu'_N[f(x - 1, 1) - f(x, 1)] + (1 - \mu'_N)[f(x, N) - f(x + 1, N)]$$
$$\leq \mu'_1[f(x - 1, 2) - f(x, 2)] + (1 - \mu'_1)[f(x - 1, 1) - f(x, 1)]$$

which is equivalent to the inequality (23) to be proved.

For $x = 1$ in order to prove propagation of BSub it has to be shown that:

$$T_D f(1, N) + T_D f(1, 1)$$
$$= \mu'_N f(0, 1) + (1 - \mu'_N)f(1, N) + \mu'_1 f(1, 2) + (1 - \mu'_1)f(1, 1)$$
$$\leq \mu'_1 f(0, 1) + (1 - \mu'_1)f(0, 1) + \mu'_N f(1, 1) + (1 - \mu'_N)f(2, N)$$
$$= T_D f(0, 1) + T_D f(2, N) \tag{26}$$

Boundary-submodularity, given as assumption, at $f(1, N)$ and increasing property, also given as assumption, at $f(1, 1)$ ($f(1, 1) \geq f(1, 2)$) respectively can be rewritten as:

$$f(1, N) - f(2, N) \leq f(0, 1) - f(1, 1) \tag{27}$$
$$f(0, 1) - f(1, 1) \leq f(0, 1) - f(1, 2) \tag{28}$$

Now, since the right hand side of inequality (27) is equal to the left hand side of inequality (28), Lemma 27 can be used with $\theta_1 = (1 - \mu'_N)$ and $\theta_2 = (1 - \mu'_1)$ and $\gamma = 1$. Application of this lemma yields:

$$\mu'_N[f(0, 1) - f(1, 1)] + (1 - \mu'_N)[f(1, N) - f(2, N)]$$
$$\leq \mu'_1[f(0, 1) - f(1, 2)] + (1 - \mu'_1)[f(0, 1) - f(1, 1)]$$

which is equivalent to the inequality (26) to be proved. $\qquad\square$