Optimal Strategy for Swapping Servers

Bol, J.E.M.

J.E.M. Bol

Optimal Strategy for Swapping Servers

Master thesis

03-07-2022

Thesis supervisor: prof. dr. F.M. Spieksma

Leiden University
Mathematical Institute

Contents

Introduction

A queueing model with service control can be explained as a standard queueing model, where there are two different options for service of which only one can operate at any time. An example is a computer system which can only perform one task at the same time. However, two different types of service are possible for each task, where type 2 provides faster service though uses more energy and therefore is more expensive than type 1 service. Lippman (1975) and Koole (1998) have shown that using the Value Iteration Algorithm can be successful in finding an optimal strategy for such a queueing model. The Value Iteration Algorithm computes the optimal value in each state by recursively improving the value of the previous state. From this value an optimal strategy can be deduced.

The Value Iteration Algorithms needs an initial value function from which the algorithm recursively computes the value in each subsequent time step. In this thesis we will explore which properties we want an initial value function to hold, such that we can find an optimal strategy for using the different types of service.

In the first section of this thesis the model used for this research will be described. Then, we will look at which structural properties are necessary to obtain our desired results and how these properties influence the choice of initial value functions for the Value Iteration Algorithm. Next, the model will be extended with the application of discounting. For the discounting case we will look into how this affects the choice of initial value functions. Finally, we will look at numerical evidence for determining the results of finding an optimal strategy.

Some notations used in this thesis: $x^+ = \max\{0, x\}$, $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$.

1 Description of the model

In this thesis we will look at an $M/M/1$ queueing model with service control. This is a single server model, where the arrivals are all determined by a Poisson process and the service times are exponentially distributed independent of the arrival process. Let the state space X be the number of customers in the system (customers in the queue and customer receiving service) i.e. $X = \mathbb{N}_0$. The customers arrive in the system at rate λ. We have holding costs $c(x)$ for $x \in X$ per time unit. Even though there is only one service station, the service control consists of two possible actions in service: action 1 has a slower service rate (μ_1) and action 2 has a faster service rate (μ_2) such that $\mu_1 < \mu_2$, which are mutually independent. The action space is equal to $A(x) = \{1, 2\}$ for $x \in X$. For all types of service there are holding costs $c(x)$ per unit of time when the state is $x \in X$. However, for action 2 there is an increased cost of $K > 0$ per unit of time. See Figure (1) for this system.

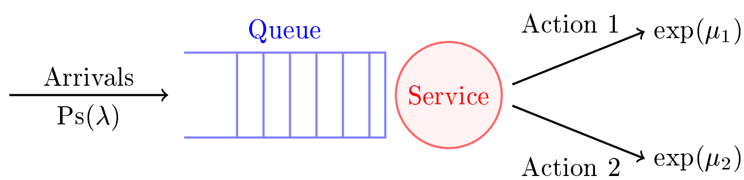

Figure 1: $M/M/1$ queue with service control.

Although this is a continuous Markov Decision Process (MDP), we can analyse this as a uniformised time-discretised MDP, as shown by Serfozo (1979). Even though, this does have an effect on the costs of the system, we are specifically interested in an optimal strategy for choosing which action to apply. Therefore, we will assume that $\lambda + \mu_1 + \mu_2 = 1$. We will consider a stable system, thus, we take $\lambda < \mu_1 < \mu_2$. For $x, y \in X$, the transition probabilities are

$$p_{xy}(a) = \lambda \cdot \mathbb{1}_{y=x+1} + \mathbb{1}_{a=1} \left[\mu_1 \cdot \mathbb{1}_{y=(x-1)^+} + \mu_2 \cdot \mathbb{1}_{y=x} \right] + \mathbb{1}_{a=2} \left[\mu_1 \cdot \mathbb{1}_{y=x} + \mu_2 \cdot \mathbb{1}_{y=(x-1)^+} \right]. \quad (1)$$

The total cost per unit of time, for state $x \in X$, and action $a \in A$, are

$$c(x, a) = c(x) + K \cdot \mathbb{1}_{a=2}. \quad (2)$$

Note that in state $x = 0$ always action 1 will be chosen, as there is no need for extra costs in case of no customers in the system.

The objective function of this system is to minimize the average expected costs. As the slower service speed is cheaper, the question arises as to when one wants to switch to the faster service speed, see Figure 2.

"What is the optimal strategy for choosing between the different possible actions for service?"

Value iteration, a form of dynamic programming, has shown to be a useful tool in finding optimal strategies for this Markov Decision Process by Lippman (1975) and Koole (1998).

Furthermore, the system can be expanded to incorporate discounting such that one can observe how this effects the strategy for switching service speed.

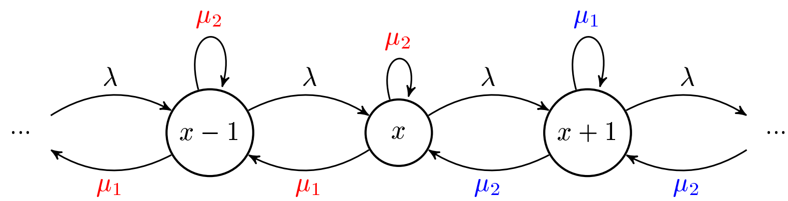

Figure 2: M/M/1 queue with service control (red: action 1, blue: action 2) for threshold at state $x \geq 1$.

2 Determining the optimal strategy for server swapping

For approaching the value of the states in the MDP per unit of time, we will use dynamic programming. The Value Iteration Algorithm allows us to numerically compute values in each state of our state space by determining the optimal action in that state. In each iteration of the Algorithm this value will be updated and improved in a recursive manner, and as such the choice for the optimal action can be updated.

First, we will look at the iterative steps of the Value Iteration Algorithm and how this corresponds to determining which action is optimal for each number of customers in the system. After, we will have a closer look into choices for the initial value function v_0 in the Value Iteration Algorithm. This approach has proven to be successful for our model by Ertiningsih et al (2017).

2.1 Value Iteration

When applying value iteration to an MDP, we first need an initial value function v_0 from which we can compute each next iteration recursively for $x \in X$ taking into account the different possible actions for service. The value iteration for this system is defined as

$$v_{n+1}(x) = \min_{a \in A} \left\{ c(x,a) + \sum p_{xy}(a) \cdot v_n(y) \right\}. \tag{3}$$

Plugging in (1) and (2), for an initial value function $v_0(x)$, $x = 0, 1, 2, ...$, we have

$$\begin{aligned} v_{n+1}(x) &= c(x) + \lambda v_n(x+1) + \min\left\{\mu_1 v_n(x-1)^+ + \mu_2 v_n(x), K + \mu_1 v_n(x) + \mu_2 v_n(x-1)^+\right\} \\ &= c(x) + \lambda v_n(x+1) + \mu_1 v_n(x) + \mu_2 v_n(x-1)^+ \\ &\qquad\qquad + (\mu_2 - \mu_1) \min\left\{ v_n(x) - v_n(x-1)^+, \frac{K}{\mu_2 - \mu_1} \right\}. \end{aligned} \tag{4}$$

2.1.1 Threshold

Perhaps intuitively clear that the optimal strategy has a threshold structure of a certain number of customers in the system for switching from action 1 to action 2. This intuition will be justified in the next section. First, we will introduce a definition of this threshold. We want to switch from service action 1 to action 2 when action 2 is strict ‘cheaper’ than action 1. Therefore, define δ_{n+1} as the threshold for swapping actions in stage $n+1$:

$$\delta_{n+1}^{v_0} = \begin{cases} \min\left\{x | v_n(x) - v_n(x-1)^+ > \frac{K}{\mu_1 - \mu_2}\right\}, & \text{when } \sup_x (v_n(x) - v_n(x-1)^+) > \frac{K}{\mu_2 - \mu_1}, \\ \infty, & \text{otherwise.} \end{cases} \tag{5}$$

This δ_{n+1} exactly describes the threshold for swapping service actions at time step n. Thus, we see that for all $x \geq \delta_{n+1}$, action 2 is optimal. Additionally note, that if in a time step action 2 is never optimal, δ_{n+1} will not exist and is set to ∞.

2.1.2 Convexity and convergence

As seen in Lippman (1975) v_n is found to be a convex function. This is crucial for us, as when $v_n(x+1) - v_n(x) > \frac{K}{\mu_2 - \mu_1}$ we find that for all $y \geq x$ we have $v_n(y+1) - v_n(y) > \frac{K}{\mu_2 - \mu_1}$. This shows us that if in a state x action 2 is optimal, then for all following states $y \geq x$ action 2 is optimal as well. Conversely, likewise when $v_n(x+1) - v_n(x) \leq \frac{K}{\mu_2 - \mu_1}$ for some state $x \in X$ (when action 1 is optimal to choose in this state), this means for all states $y \leq x$ action 1 is the optimal action to choose.

Ertiningsih et al (2017) has shown that $\lim_{n\to\infty}(v_n(x) - v_n(0)) = v^*$, where $v^* = v^*(x)$ is the unique optimal average cost value function (except for a constant) of the system. As v_n convex for all n, convexity also holds for v^*. This is very useful as this also means that $\lim_{n\to\infty} \delta_{n+1} = \delta^*$ is the optimal threshold for switching to service action 2. Indeed, the optimal strategy shows to be finding a threshold for the number of customers in the system.

2.2 Choice of initial value function

Our approach in finding the optimal threshold for switching actions, consists of handy choices of the initial value function of the VI Algorithm.

Lemma 2.1. *Let v_n the value function.*
(a) If

$$v_0^l(x+1) - v_0^l(x) \geq v_1^l(x+1) - v_1^l(x) \tag{6}$$

holds for all $x \in X$, then

$$\lim_{n\to\infty} \delta_{n+1}^{v_0^l} \uparrow \delta^*.$$

(b) If

$$v_0^u(x+1) - v_0^u(x) \leq v_1^u(x+1) - v_1^u(x) \tag{7}$$

holds for all $x \in X$, then

$$\lim_{n\to\infty} \delta_{n+1}^{v_0^u} \downarrow \delta^*.$$

Proof. First, note that if $v_n(x+1) - v_n(x) \geq v_{n+1}(x+1) - v_{n+1}(x)$ holds for an $n \geq 0$ and if in stage n action 2 is optimal for state $x \in X$, then $\frac{K}{\mu_2-\mu_1} > v_n(x+1) - v_n(x)$. So also $\frac{K}{\mu_2-\mu_1} > v_{n+1}(x+1) - v_{n+1}(x)$ and action 2 is optimal in stage $n+1$ for those same states. Hence, $\delta_{n+1} \leq \delta_{n+2}$. Now, by Equations (4) and (6) and by Lemma 2.6 in Ertiningsih et al (2017) we find that v_0^l is a function such that $v_n(x+1) - v_n(x) \geq v_{n+1}(x+1) - v_{n+1}(x)$ holds for all $n \geq 0$, hence, $\lim_{n\to\infty} \delta_{n+1}^{v_0^l} \uparrow \delta^*$.

Conversely, the same arguments hold for a function v_0^u satisfying Equation (7), such that $\lim_{n\to\infty} \delta_{n+1}^{v_0^u} \downarrow \delta^*$. □

Remark. Lemma 2.6 in Ertiningsih et al (2017) calls for the service control operators to be non-decreasing, however, it can be easily checked that this criterion is not necessary for the correctness of Lemma 2.1.

With this approach we can 'squeeze' the optimal threshold and we will know to have found that threshold once $\delta^{v_0^l} = \delta^{v_0^u}$. Therefore, we are interested in two functions for which inequalities in (6) and (7) hold. Furthermore, we presume the optimal threshold to be reached in less iterations when the initial value functions are chosen closer to equality in (6) and (7). We will look at numerical evidence to support this claim and to observe the behaviour of the threshold and the values in each stage of the VI Algorithm.

2.2.1 The zero initial value function

First, we will look at the case where $v_0^l \equiv 0$. In this case it can be easily seen that Equation (7) holds when we have non-decreasing holding costs. To then 'squeeze' the optimal threshold, we will use the initial value function $v_0^u(x)$ of quadratic form, which can be found in the next section. Then, we will use numerical results of different lower initial value functions to be able to see whether convergence occurs faster with an initial value function closer or equal to equality in (7), than the zero initial value function.

For further choices of initial value functions, we find that when substituting Equation (4) in Equations (6) and (7), a common term of $v_0(x+1) - v_0(x)$ can be found on both sides of the inequality sign, leaving a rest of degree one lower than the degree of $c(x+1) - c(x)$. This leads us to the suspicion that a handy choice of the degree of the initial value function is one lower than the degree of the cost function, due to the necessity for the Equations (6) and (7) to hold for all possible states in our state space.

2.2.2 Quadratic initial value function

We are interested in a quadratic v_0 for the VI Algorithm when the holding costs are of linear form. Hence, let $c(x) = c_a x$ and set $v_0(x) = ax^2 + bx + c$, for $x \in X$.

Lemma 2.2. *For*

$$v_0^l(x) = \frac{c_a}{2(\mu_1 - \lambda)}(x^2 + x) \tag{8}$$

and

$$v_0^u(x) = \frac{c_a}{2(\mu_2 - \lambda)}(x^2 + x) \tag{9}$$

equations (6) and (7) respectively hold.

Proof. See Appendix A. □

2.2.3 Cubic initial value function

We are interested in a cubic v_0 for the VI Algorithm when the holding costs are of quadratic form. Hence, let $c(x) = c_a x^2 + c_b x$ and set $v_0(x) = ax^3 + bx^2 + cx + d$, for $x \in X$.

Lemma 2.3. *For*

$$v_0^l(x) = \frac{c_a}{3(\mu_1 - \lambda)}x^3 + \frac{c_b + (\lambda + \mu_1)\frac{c_a}{\mu_1 - \lambda}}{2(\mu_1 - \lambda)}x^2 + \left(\frac{c_b + (\lambda + \mu_1)\frac{c_a}{\mu_1 - \lambda}}{2(\mu_1 - \lambda)} - \frac{c_a}{3(\mu_1 - \lambda)}\right)x \tag{10}$$

and

$$v_0^u(x) = \frac{c_a}{3(\mu_2 - \lambda)}x^3 + \frac{c_b + (\lambda + \mu_2)\frac{c_a}{\mu_2 - \lambda}}{2(\mu_2 - \lambda)}x^2 + \left(\frac{c_b + (\lambda + \mu_2)\frac{c_a}{\mu_2 - \lambda}}{2(\mu_2 - \lambda)} - \frac{c_a}{3(\mu_2 - \lambda)}\right)x \tag{11}$$

equations (6) and (7) respectively hold.

Proof. See Appendix B. □

3 Discounting

Adding discounting to the system ensures the value iteration to have lower value in a later time stage. In this section we will look at the effect on this discounting when analysing the optimal threshold for switching service speeds, from the slower service speed (action 1) to the faster yet more expensive service speed (action 2). We will once again use the Value Iteration Algorithm to obtain this desired information, likewise to the previous section where no discounting was applied.

3.1 Value iteration with discounting

Once again, the Value Iteration Algorithm starts with an initial value function $v_0(x)$ for all $x \in X$. However, in this instance, in each iterative step of the VI Algorithm the value of the previous iteration is only worth α as much, for an $\alpha \in (0,1)$. Discounting is applied to events that occur in the future, thus, in the Algorithm we discount each previous iteration. Now, for an initial value function $v_0(x)$, $x = 0, 1, 2, ...$, and $\alpha \in (0,1)$ the value iteration for this system with discounting is defined as

$$v_{n+1}^{\alpha}(x) = \min_{a \in A} \left\{ c(x,a) + \alpha \sum p_{xy}(a) \cdot v_n(y) \right\} \tag{12}$$

Plugging in (1) and (2), this results in

$$\begin{aligned} v_{n+1}^{\alpha}(x) &= c(x) + \alpha \Bigg[\lambda v_n(x+1) \\ &\qquad + \min \left\{ \mu_1 v_n(x-1)^+ + \mu_2 v_n(x), \frac{K}{\alpha} + \mu_1 v_n(x) + \mu_2 v_n(x-1)^+ \right\} \Bigg] \\ &= c(x) + \alpha\lambda v_n(x+1) + \alpha\mu_1 v_n(x) + \alpha\mu_2 v_n(x-1)^+ \\ &\quad + \alpha(\mu_2 - \mu_1) \min \left\{ v_n(x) - v_n(x-1)^+, \frac{K}{\alpha(\mu_2 - \mu_1)} \right\}. \end{aligned} \tag{13}$$

3.1.1 Threshold

In the previous section, we defined the threshold for swapping actions to faster service speed as the lowest number of customers in the system for which action 2 is more optimal than when applying action 1 for service per time stage in the VI Algorithm. Then, the threshold for swapping server speeds with discounting is defined as follows:

$$\delta_{n+1}^{v_0,\alpha} = \begin{cases} \min \left\{ x | v_n(x) - v_n(x-1)^+ > \frac{K}{\alpha(\mu_1 - \mu_2)} \right\}, & \text{when } \sup_x (v_n(x) - v_n(x-1)^+) > \frac{K}{\alpha(\mu_2 - \mu_1)}, \\ \infty, & \text{otherwise.} \end{cases} \tag{14}$$

Note, adding discounting does not influence the convexity or threshold structure of this system. Thus, our approach in determining the optimal threshold remains unchanged.

3.2 Initial value function with discounting

The choice of the initial value function will be done similarly as for the non-discounting model. We will once again try to 'squeeze' the optimal value and the optimal threshold. Lemma 2.1 holds in the exact same way with discounting. By replacing $\frac{K}{\mu_2-\mu_1}$ by $\frac{K}{\alpha(\mu_2-\mu_1)}$ has no influence on the implications in the Lemma. Therefore, we use a similar approach to determining initial value functions, such that we can 'squeeze' the threshold of the number of customers in the system necessary for swapping actions.

Different to the situation without discounting, when plugging in Equation (4) into Equations (6) and (7), we find the term $\alpha(v_0(x-1)-v_0(x))$, such that the term $v_0(x-1)-v_0(x)$ does not get 'cancelled out' as in the previous section. This motivates the idea of using finding value function of the same degree as the cost function has, due to the necessity for the Equations (6) and (7) to hold for all possible states in our state space.

3.2.1 Quadratic initial value function with discounting

We are interested in a quadratic v_0 for the VI Algorithm with discounting (with value α) when the holding costs are of quadratic form. Hence, let $c(x) = c_a x^2 + c_b x$ and set $v_0(x) = ax^2 + bx + c$, for $x \in X$.

Lemma 3.1. *For*

$$v_0^{l,\alpha} = \frac{c_a}{1-\alpha}x^2 + \max\left\{\frac{c_b + 2\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}, \frac{c_b+\alpha(2\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha+\alpha\mu_1}\right\}x \tag{15}$$

and

$$v_0^{u,\alpha} = \frac{c_a}{1-\alpha}x^2 + \min\left\{\frac{c_b + 2\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}, \frac{c_b+\alpha(2\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha+\alpha\mu_2}\right\}x \tag{16}$$

equations (6) and (7) respectively hold.

Proof. See Appendix C. □

3.2.2 Cubic initial value function with discounting

We are interested in a cubic v_0 for the VI Algorithm with discounting (with value α) when the holding costs are of cubic form. Hence, let $c(x) = c_a x^3 + c_b x^2 + c_c x$ and set $v_0(x) = ax^3 + bx^2 + cx + d$, for $x \in X$.

Lemma 3.2. *For*

$$\begin{aligned}
v_0^{l,\alpha}(x) =& \frac{c_a}{1-\alpha}x^3 + \frac{c_b + 3\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}x^2 \\
&+ \max\left\{ \frac{c_c + 3\alpha(\lambda+\mu_1)\frac{c_a}{1-\alpha} + 2\alpha(\lambda-\mu_1)\frac{c_b+3\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha}, \right. \\
&\qquad \frac{c_c + 3\alpha(\lambda+\mu_2)\frac{c_a}{1-\alpha} + 2\alpha(\lambda-\mu_2)\frac{c_b+3\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha}, \\
&\qquad \left. \frac{c_c + \alpha(3\lambda+2\mu_1)\frac{c_a}{1-\alpha} + \alpha(2\lambda-\mu_1)\frac{c_b+3\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha+\alpha\mu_1} \right\} x,
\end{aligned}$$

and

$$\begin{aligned}
v_0^{u,\alpha}(x) =& \frac{c_a}{1-\alpha}x^3 + \frac{c_b + 3\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}x^2 \\
&+ \min\left\{ \frac{c_c + 3\alpha(\lambda+\mu_2)\frac{c_a}{1-\alpha} + 2\alpha(\lambda-\mu_2)\frac{c_b+3\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha}, \right. \\
&\qquad \frac{c_c + 3\alpha(\lambda+\mu_1)\frac{c_a}{1-\alpha} + 2\alpha(\lambda-\mu_1)\frac{c_b+3\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha}, \\
&\qquad \left. \frac{c_c + \alpha(3\lambda+2\mu_2)\frac{c_a}{1-\alpha} + \alpha(2\lambda-\mu_2)\frac{c_b+3\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha+\alpha\mu_2} \right\} x,
\end{aligned}$$

equations (6) and (7) respectively hold.

Proof. See Appendix D. □

Interestingly, this is the first time that that we find both μ_1 and μ_2 in the lower and upper initial value functions.

4 Numerical results

Using MATLAB, we have programmed the Value Iteration Algorithm implementing all the previously discussed initial value functions for various parameters. See Appendix E for this MATLAB code. In this section some examples will be presented to show the effect of the 'squeezing' method of the optimal threshold of swapping service speeds.

4.1 Iterations for optimal threshold for different initial value functions

We want to determine if convergence occurs in less iterations for the found quadratic or cubic initial functions, as given before, than when applying the zero initial function. As, Equation (7) holds for the zero initial function, we will compare this to the upper initial functions of quadratic and cubic degree. The numerical results of the minimum number of iterations necessary for reaching the optimal threshold are shown in Table 1. Indeed, we determine that in each of the 21 different (combinations of) parameters there are less iterations necessary for the optimal threshold to be reached. For some parameters the difference is even substantially large. Especially the difference in the number of iterations between the zero and cubic initial function is significant. The difference ranges from over 2 to 188 times less iterations necessary for the cubic function. For the quadratic initial function we find a range of approximately 1.2 to 3.5 times less iterations to reach the optimal threshold than for the zero initial function.

4.2 Effect of discounting

Example 4.1. *Choose the following parameters:*

$$\lambda = 0.3,\ \mu_1 = 0.32,\ \mu_2 = 0.38,\ K = 20,\ c(x) = 0.1x^2 + x.$$

Various values of α: 0.9, 0.99, 0.999 and 1.

The results are found in Figure 3, the optimal thresholds can be determined numerically:

α	0.9	0.99	0.999	1
δ^*	182	17	8	7

For $\alpha < 1$, the initial functions from Lemma 3.1 are used, and for $\alpha = 1$ the initial functions from Lemma 2.3 are used, as we have a quadratic holding cost function. Although the number of iterations, until the optimal threshold for both the upper and lower initial function is reached, for different values of α varies, we do see that for the initial function from Lemma 3.1 (with $\alpha \in (0, 1)$), less iterations are necessary for lower values of α. Furthermore, we do notice that for higher values of α, the optimal threshold δ^* decreases. This raises the suspicion that this is the case for arbitrary values of α, which we found to be correct, see Theorem 4.1. Although, $(\delta^\alpha)^* \to (\delta^1)^*$, this occurs in a slow manner, as even here $(\delta^{0.999})^* > (\delta^1)^*$.

				v_0^u; (no discounting)			
λ	μ_1	μ_2	K	zero	quad.	zero	cubic
			5	27	21	7	1
0.1	0.3	0.6	10	80	73	8	2
			20	99	89	13	6
			5	63	43	10	1
0.1	0.4	0.5	10	121	97	23	9
			20	234	204	20	4
			5	114	89	15	1
0.1	0.42	0.48	10	201	166	23	4
			20	401	362	32	11
			5	23	12	15	5
0.2	0.25	0.55	10	53	38	11	3
			20	131	112	14	5
			5	42	27	9	1
0.2	0.3	0.5	10	77	56	26	12
			20	162	135	32	19
			5	75	44	54	21
0.2	0.35	0.45	10	168	130	19	1
			20	310	263	40	17
			5	163	48	53	1
0.3	0.31	0.39	10	222	95	41	1
			20	641	460	158	7
			5	222	63	84	1
0.3	0.32	0.38	10	475	276	64	1
			20	623	389	600	115
			5	322	91	175	1
0.3	0.33	0.37	10	560	287	128	1
			20	1449	1101	188	1

Table 1: Number of iterations until threshold is reached for various variables and different initial value functions. The holding costsfor the quadratic and cubic initial functions are $c(x) = x$ and $c(x) = 0.1x^2 + x$ respectively.

Theorem 4.1. *For $\alpha_1, \alpha_2 \in (0, 1]$ with $\alpha_1 < \alpha_2$*

$$\delta_{n+1}^{\alpha_1} \geq \delta_{n+1}^{\alpha_2} \tag{17}$$

holds for all $n \in \mathbb{N}_0$ for the zero initial value function.

Proof. We will prove by induction for all time steps $n \in \mathbb{N}_0$, that $v_n^{\alpha_1}(x) - v_n^{\alpha_1}(x-1)^+ \geq v_n^{\alpha_2}(x) - v_n^{\alpha_2}(x_1)^+$, for all $x \in X$. Consequently, Equation (17) holds.

Note that, when the cost function is an increasing function, we have $c(x) - c(x-1)^+ \geq 0$. As such, with the zero initial value function, $v_n^{0,\alpha}(x) - v_n^{0,\alpha}(x-1)^+ \geq 0$ always holds.

Base step. For all $x \in X$ we find

$$0 = v_0^{\alpha_1}(x) - v_0^{\alpha_1}(x-1)^+ \leq v_0^{\alpha_2}(x) - v_0^{\alpha_2}(x-1)^+ = 0$$

such that $\delta_1^{\alpha_1} \geq \delta_1^{\alpha_2}$, as for $\alpha_1 < \alpha_2$ the inequality $\frac{K}{\alpha_1(\mu_2-\mu_1)} > \frac{K}{\alpha_2(\mu_2-\mu_1)}$ holds for $\alpha \in (0,1]$.

Inductive step. Let $k \in \mathbb{N}$ be arbitrarily given and suppose Equation (17) holds (Induction Hypothesis) for all $0 \leq n \leq k$. Then consider time step $k+1$,

$$\begin{aligned}
v_{k+1}^{\alpha_1}(x) &- v_{k+1}^{\alpha_1}(x-1)^+ - (v_{k+1}^{\alpha_2}(x) - v_{k+1}^{\alpha_2}(x-1)^+) \\
&= c(x) - c(x-1) - (c(x) - c(x-1)) \\
&\quad + \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)^+) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1)^+)\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(x-1)^+ - v_k^{\alpha_1}(x-2)^+) - \alpha_2(v_k^{\alpha_2}(x-1)^+ - v_k^{\alpha_2}(x-2)^+)\big] \\
&\quad + \alpha_1(\mu_2-\mu_1)\Bigg[\min\left\{v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)^+, \frac{K}{\alpha_1(\mu_2-\mu_1)}\right\} \\
&\qquad\qquad - \min\left\{v_k^{\alpha_1}(x-1)^+ - v_k^{\alpha_1}(x-2)^+, \frac{K}{\alpha_1(\mu_2-\mu_1)}\right\}\Bigg] \\
&\quad - \alpha_2(\mu_2-\mu_1)\Bigg[\min\left\{v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1)^+, \frac{K}{\alpha_2(\mu_2-\mu_1)}\right\} \\
&\qquad\qquad - \min\left\{v_k^{\alpha_2}(x-1)^+ - v_k^{\alpha_2}(x-2)^+, \frac{K}{\alpha_2(\mu_2-\mu_1)}\right\}\Bigg]
\end{aligned} \tag{18}$$

First, we will consider the state $x = 0$ followed by state $x = 1$, and finally all states $x \geq 2$.

Let $x = 0$ then,

$$\begin{aligned}
v_{k+1}^{\alpha_1}(0) &- v_{k+1}^{\alpha_1}(0) - (v_{k+1}^{\alpha_2}(0) - v_{k+1}^{\alpha_2}(0)) \\
&= c(0) - c(0) - (c(0) - c(0)) \\
&\quad + \lambda\big[\alpha_1(v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(0) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(0) - v_k^{\alpha_2}(0))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(0) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(0) - v_k^{\alpha_2}(0))\big] \\
&\quad + \alpha_1(\mu_2-\mu_1)\Bigg[\min\left\{v_k^{\alpha_1}(0) - v_k^{\alpha_1}(0), \frac{K}{\alpha_1(\mu_2-\mu_1)}\right\} \\
&\qquad\qquad - \min\left\{v_k^{\alpha_1}(0) - v_k^{\alpha_1}(0), \frac{K}{\alpha_1(\mu_2-\mu_1)}\right\}\Bigg] \\
&\quad - \alpha_2(\mu_2-\mu_1)\Bigg[\min\left\{v_k^{\alpha_2}(0) - v_k^{\alpha_2}(0), \frac{K}{\alpha_2(\mu_2-\mu_1)}\right\} \\
&\qquad\qquad - \min\left\{v_k^{\alpha_2}(0) - v_k^{\alpha_2}(0), \frac{K}{\alpha_2(\mu_2-\mu_1)}\right\}\Bigg] \\
&= \lambda\big[\alpha_1(v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0))\big] \\
&\leq 0 \qquad \text{(by Induction Hypothesis)}
\end{aligned}$$

Let $x = 1$ then,

$$
\begin{aligned}
v_{k+1}^{\alpha_1}(1) - v_{k+1}^{\alpha_1}(0) - (v_{k+1}^{\alpha_2}(1) - v_{k+1}^{\alpha_2}(0)) \\
&= c(1) - c(0) - (c(1) - c(1)) \\
&\quad + \lambda\big[\alpha_1(v_k^{\alpha_1}(2) - v_k^{\alpha_1}(1)) - \alpha_2(v_k^{\alpha_2}(2) - v_k^{\alpha_2}(1))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(0) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(0) - v_k^{\alpha_2}(0))\big] \\
&\quad + \alpha_1(\mu_2 - \mu_1)\min\left\{v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0), \frac{K}{\alpha_1(\mu_2 - \mu_1)}\right\} \\
&\quad - \alpha_2(\mu_2 - \mu_1)\min\left\{v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0), \frac{K}{\alpha_2(\mu_2 - \mu_1)}\right\}
\end{aligned}
$$

The remainder of this part of the proof is split over the following three possible cases.

- Case 1: $v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0) \leq \frac{K}{\alpha_1(\mu_2 - \mu_1)}$ and $v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0) \leq \frac{K}{\alpha_2(\mu_2 - \mu_1)}$

$$
\begin{aligned}
v_{k+1}^{\alpha_1}(1) - v_{k+1}^{\alpha_1}(0) - (v_{k+1}^{\alpha_2}(1) - v_{k+1}^{\alpha_2}(0)) \\
&= \lambda\big[\alpha_1(v_k^{\alpha_1}(2) - v_k^{\alpha_1}(1)) - \alpha_2(v_k^{\alpha_2}(2) - v_k^{\alpha_2}(1))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(0) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(0) - v_k^{\alpha_2}(0))\big] \\
&\quad + (\mu_2 - \mu_1)\big[\alpha_1(v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0))\big] \\
&\leq 0 \qquad \text{(by Induction Hypothesis)}
\end{aligned}
$$

- Case 2: $v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0) \leq \frac{K}{\alpha_1(\mu_2 - \mu_1)}$ and $\frac{K}{\alpha_2(\mu_2 - \mu_1)} < v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0)$

$$
\begin{aligned}
v_{k+1}^{\alpha_1}(1) - v_{k+1}^{\alpha_1}(0) - (v_{k+1}^{\alpha_2}(1) - v_{k+1}^{\alpha_2}(0)) \\
&= \lambda\big[\alpha_1(v_k^{\alpha_1}(2) - v_k^{\alpha_1}(1)) - \alpha_2(v_k^{\alpha_2}(2) - v_k^{\alpha_2}(1))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(0) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(0) - v_k^{\alpha_2}(0))\big] \\
&\quad + (\mu_2 - \mu_1)\left[\alpha_1(v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)) - \alpha_2\frac{K}{\alpha_2(\mu_2 - \mu_1)}\right] \\
&\leq \lambda\big[\alpha_1(v_k^{\alpha_1}(2) - v_k^{\alpha_1}(1)) - \alpha_2(v_k^{\alpha_2}(2) - v_k^{\alpha_2}(1))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(0) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(0) - v_k^{\alpha_2}(0))\big] \\
&\quad + (\mu_2 - \mu_1)\left[\alpha_1\frac{K}{\alpha_1(\mu_2 - \mu_1)} - \alpha_2\frac{K}{\alpha_2(\mu_2 - \mu_1)}\right] \\
&\leq 0 \qquad \text{(by Induction Hypothesis)}
\end{aligned}
$$

- Case 3: $\frac{K}{\alpha_1(\mu_2-\mu_1)} < v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)$ and $v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0) \leq \frac{K}{\alpha_2(\mu_2-\mu_1)}$

$$\begin{aligned}
&v_{k+1}^{\alpha_1}(1) - v_{k+1}^{\alpha_1}(0) - (v_{k+1}^{\alpha_2}(1) - v_{k+1}^{\alpha_2}(0)) \\
&\quad = \lambda\big[\alpha_1(v_k^{\alpha_1}(2) - v_k^{\alpha_1}(1)) - \alpha_2(v_k^{\alpha_2}(2) - v_k^{\alpha_2}(1))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(1) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(1) - v_k^{\alpha_2}(0))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(0) - v_k^{\alpha_1}(0)) - \alpha_2(v_k^{\alpha_2}(0) - v_k^{\alpha_2}(0))\big] \\
&\quad + (\mu_2 - \mu_1)\left[\alpha_1\frac{K}{\alpha_1(\mu_2-\mu_1)} - \alpha_2\frac{K}{\alpha_2(\mu_2-\mu_1)}\right] \\
&\quad \leq 0 \qquad \text{(by Induction Hypothesis)}
\end{aligned}$$

Let $x \geq 2$, for Equation (18) to hold we have to consider the following six cases.

- Case 1:

$$v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1),\ v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2) \leq \frac{K}{\alpha_1(\mu_2-\mu_1)},$$

$$v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1),\ v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2) \leq \frac{K}{\alpha_2(\mu_2-\mu_1)}.$$

Then,

$$\begin{aligned}
&v_{k+1}^{\alpha_1}(x) - v_{k+1}^{\alpha_1}(x-1) - (v_{k+1}^{\alpha_2}(x) - v_{k+1}^{\alpha_2}(x-1)) \\
&\quad = c(x) - c(x-1) - (c(x) - c(x-1)) \\
&\quad + \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad + \alpha_1(\mu_2-\mu_1)\big[v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1) - (v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2))\big] \\
&\quad - \alpha_2(\mu_2-\mu_1)\big[v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1) - (v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad = \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad \leq 0 \qquad \text{(by Induction Hypothesis)}
\end{aligned}$$

- Case 2:

$$v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1),\ v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2) \leq \frac{K}{\alpha_1(\mu_2-\mu_1)},$$

$$v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2) \leq \frac{K}{\alpha_2(\mu_2-\mu_1)} < v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1).$$

Then,

$$\begin{aligned}
&v_{k+1}^{\alpha_1}(x) - v_{k+1}^{\alpha_1}(x-1) - (v_{k+1}^{\alpha_2}(x) - v_{k+1}^{\alpha_2}(x-1)) \\
&\quad = c(x) - c(x-1) - (c(x) - c(x-1)) \\
&\quad + \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad + \alpha_1(\mu_2 - \mu_1)\big[v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1) - (v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2))\big] \\
&\quad - \alpha_2(\mu_2 - \mu_1)\left[\frac{K}{\alpha_2(\mu_2 - \mu_1)} - (v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\right] \\
&\quad = \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad + \mu_1\left[\alpha_2\frac{K}{\alpha_2(\mu_2 - \mu_1)} - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\right] \\
&\quad + \mu_2\left[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2\frac{K}{\alpha_2(\mu_2 - \mu_1)}\right] \\
&\quad < \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad + \mu_1\left[\alpha_2\frac{K}{\alpha_2(\mu_2 - \mu_1)} - \alpha_2\frac{K}{\alpha_2(\mu_2 - \mu_1)}\right] \\
&\quad + \mu_2\left[\alpha_1\frac{K}{\alpha_1(\mu_2 - \mu_1)} - \alpha_2\frac{K}{\alpha_2(\mu_2 - \mu_1)}\right] \\
&\quad \leq 0 \qquad \text{(by Induction Hypothesis)}
\end{aligned}$$

- Case 3:

$$v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1),\ v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2) \leq \frac{K}{\alpha_1(\mu_2 - \mu_1)},$$

$$\frac{K}{\alpha_2(\mu_2 - \mu_1)} < v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1),\ v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2).$$

Then,

$$
\begin{aligned}
&v_{k+1}^{\alpha_1}(x) - v_{k+1}^{\alpha_1}(x-1) - (v_{k+1}^{\alpha_2}(x) - v_{k+1}^{\alpha_2}(x-1)) \\
&\quad = c(x) - c(x-1) - (c(x) - c(x-1)) \\
&\quad + \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad + \alpha_1(\mu_2 - \mu_1)\big[v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1) - (v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2))\big] \\
&\quad - \alpha_2(\mu_2 - \mu_1)\left[\frac{K}{\alpha_2(\mu_2 - \mu_1)} - \frac{K}{\alpha_2(\mu_2 - \mu_1)}\right] \\
&\quad = \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad < \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + (\mu_1 + \mu_2)\left[\frac{K}{\mu_2 - \mu_1} - \frac{K}{\mu_2 - \mu_1}\right] \\
&\quad \leq 0 \qquad \text{(by Induction Hypothesis)}
\end{aligned}
$$

- Case 4:

$$
\begin{aligned}
v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2) \leq \frac{K}{\alpha_1(\mu_2 - \mu_1)} < v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1), \\
v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2) \leq \frac{K}{\alpha_2(\mu_2 - \mu_1)} < v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1).
\end{aligned}
$$

Then,

$$
\begin{aligned}
&v_{k+1}^{\alpha_1}(x) - v_{k+1}^{\alpha_1}(x-1) - (v_{k+1}^{\alpha_2}(x) - v_{k+1}^{\alpha_2}(x-1)) \\
&\quad = c(x) - c(x-1) - (c(x) - c(x-1)) \\
&\quad + \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad + \alpha_1(\mu_2 - \mu_1)\left[\frac{K}{\alpha_1(\mu_2 - \mu_1)} - (v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2))\right] \\
&\quad - \alpha_2(\mu_2 - \mu_1)\left[\frac{K}{\alpha_2(\mu_2 - \mu_1)} - (v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\right] \\
&\quad = \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad \leq 0 \qquad \text{(by Induction Hypothesis)}
\end{aligned}
$$

- Case 5:

$$v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2) \le \frac{K}{\alpha_1(\mu_2-\mu_1)} < v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1),$$

$$\frac{K}{\alpha_2(\mu_2-\mu_1)} < v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1),\ v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2).$$

Then,

$$\begin{aligned}
&v_{k+1}^{\alpha_1}(x) - v_{k+1}^{\alpha_1}(x-1) - (v_{k+1}^{\alpha_2}(x) - v_{k+1}^{\alpha_2}(x-1)) \\
&\quad = c(x) - c(x-1) - (c(x) - c(x-1)) \\
&\quad + \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad + \alpha_1(\mu_2-\mu_1)\left[\frac{K}{\alpha_1(\mu_2-\mu_1)} - (v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2))\right] \\
&\quad - \alpha_2(\mu_2-\mu_1)\left[\frac{K}{\alpha_2(\mu_2-\mu_1)} - \frac{K}{\alpha_2(\mu_2-\mu_1)}\right] \\
&\quad < \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_2\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad + (\mu_2-\mu_1)\Big[\alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2)) - \alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2))\Big] \\
&\quad \le \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\big] \\
&\quad + \mu_1\big[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\big] \\
&\quad \le 0 \qquad \text{(Induction Hypothesis)}
\end{aligned}$$

- Case 6:

$$\frac{K}{\alpha_1(\mu_2-\mu_1)} < v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1),\ v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2),$$

$$\frac{K}{\alpha_2(\mu_2-\mu_1)} < v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1),\ v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2).$$

Then,

$$\begin{aligned}
v_{k+1}^{\alpha_1}(x) &- v_{k+1}^{\alpha_1}(x-1) - (v_{k+1}^{\alpha_2}(x) - v_{k+1}^{\alpha_2}(x-1)) \\
&= c(x) - c(x-1) - (c(x) - c(x-1)) \\
&+ \lambda\big[\alpha_1(v_k^{\alpha_1}(x+1) - v_k^{\alpha_1}(x)) - \alpha_2(v_k^{\alpha_2}(x+1) - v_k^{\alpha_2}(x))\big] \\
&+ \mu_1\left[\alpha_1(v_k^{\alpha_1}(x) - v_k^{\alpha_1}(x-1)) - \alpha_2(v_k^{\alpha_2}(x) - v_k^{\alpha_2}(x-1))\right] \\
&+ \mu_2\left[\alpha_1(v_k^{\alpha_1}(x-1) - v_k^{\alpha_1}(x-2)) - \alpha_2(v_k^{\alpha_2}(x-1) - v_k^{\alpha_2}(x-2))\right] \\
&+ \alpha_1(\mu_2-\mu_1)\left[\frac{K}{\alpha_1(\mu_2-\mu_1)} - \frac{K}{\alpha_1(\mu_2-\mu_1)}\right] \\
&- \alpha_2(\mu_2-\mu_1)\left[\frac{K}{\alpha_2(\mu_2-\mu_1)} - \frac{K}{\alpha_2(\mu_2-\mu_1)}\right] \\
&\leq 0 \qquad \text{(Induction Hypothesis)}
\end{aligned}$$

Note that, each inequality holds as $0 < \alpha_1 < \alpha_2 \leq 1$ and $0 < \lambda, \mu_1, \mu_2 < 1$. We can then determine that

$$\delta_{k+2}^{\alpha_1} \geq \delta_{k+2}^{\alpha_2}.$$

Conclusion. By mathematical induction, we can conclude that Equation (17) holds for all $n \in \mathbb{N}_0$. □

Corollary 4.2. *For* $\alpha_1, \alpha_2 \in (0, 1]$ *with* $\alpha_1 < \alpha_2$

$$(\delta^{\alpha_1})^* \geq (\delta^{\alpha_2})^* \tag{19}$$

holds for all $n \in \mathbb{N}_0$ *for all initial value functions.*

Proof. As for the zero initial function $\lim_{n\to\infty} \delta_{n+1}^{v_0} = \delta^*$, the proof follows directly from Theorem 4.1. □

4.3 Results for different initial value functions and different parameters

Example 4.2. *Choose the following parameters:*

$$\lambda = 0.1,\ \mu_1 = 0.4,\ \mu_2 = 0.5,\ K = 5,\ c(x) = x.$$

The results can be seen in the left two graphs in Figure 4. In the first graph we can see the optimal threshold for swapping service speed occurs at 16 customers in the system: $\delta^* = 16$. Furthermore, the lower bound for squeezing the threshold has reached the optimum in it's first iteration, yet the upper bound reaches the optimum threshold in 43 iterations. One can notice that in the lower graph, the optimum value per number of customers in the system seems to lie closer to the lower initial value function.

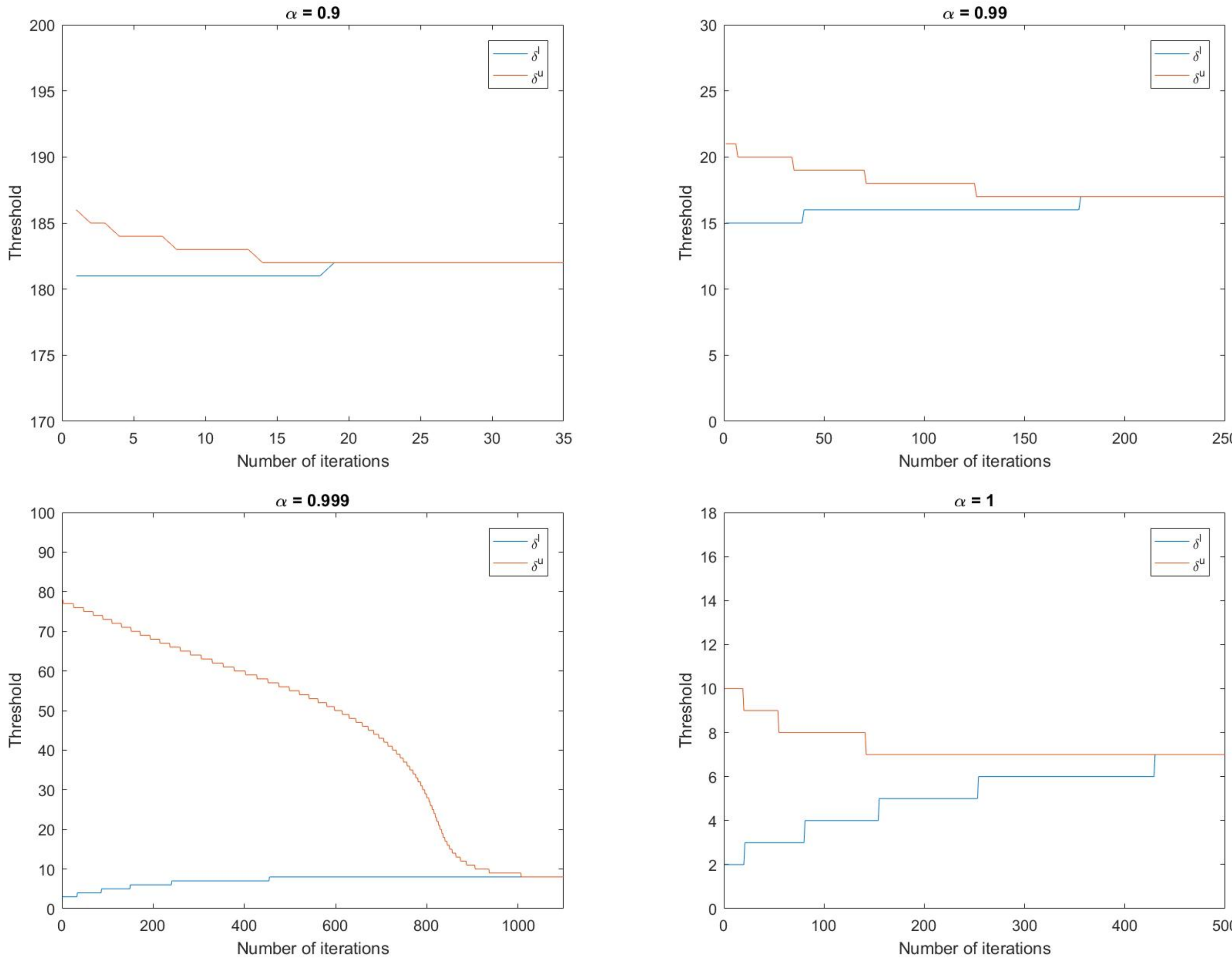

Figure 3: Effect of α on the threshold with quadratic holding costs.

Example 4.3. *Choose the following parameters:*

$$\lambda = 0.3,\ \mu_1 = 0.32,\ \mu_2 = 0.38,\ K = 20,\ c(x) = x.$$

The results can be seen in the right two graphs in Figure 4. In the first graph we can see the optimal threshold for swapping service speeds occur at 14 customers in the system: $\delta^* = 14$. Furthermore, the lower and upper bound for the optimal threshold seem to roughly follow the same pattern, where solely the last step of the lower bound towards reaching the optimal threshold takes over another 200 iterations to reach. In the lower graph, we notice the value per number of customers in the system for both initial value functions lie a lot further apart than in the lower left graph in Figure 4.

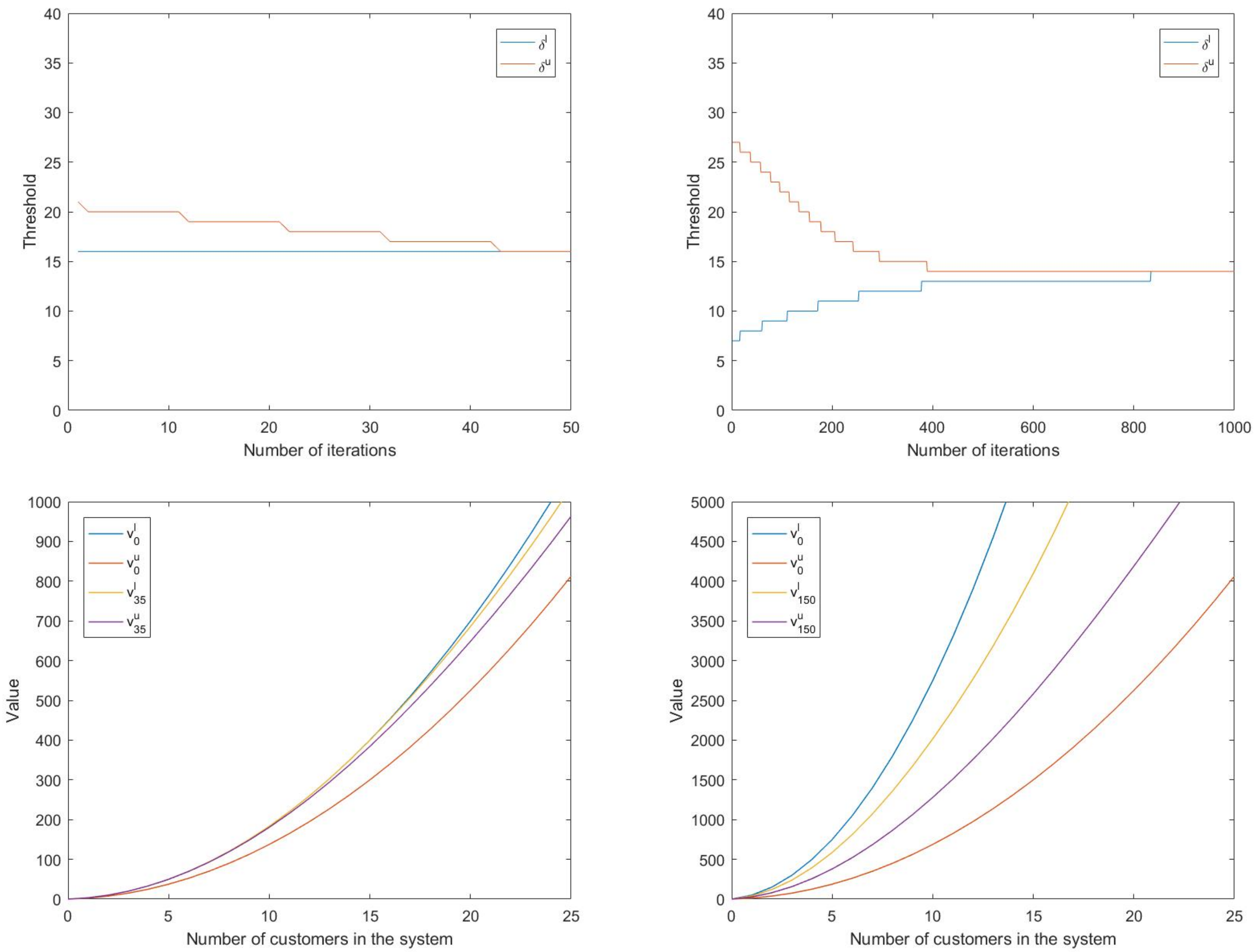

Figure 4: Optimal threshold and value of VI Algorithm for a quadratic initial value function.

Example 4.4. *Choose the following parameters:*

$$\lambda = 0.1,\ \mu_1 = 0.4,\ \mu_2 = 0.5,\ K = 5,\ c(x) = 0.1x^2 + x.$$

The results can be seen in the right two graphs in Figure 5. In the first graph we can see the optimal threshold for swapping service speeds occur at 9 customers in the system: $\delta^* = 9$. The lower bound for squeezing the threshold reaches the optimal threshold in the first iteration, and the upper bound takes 12 iteration, however, only to lower the threshold by one.

Example 4.5. *Choose the following parameters:*

$$\lambda = 0.3,\ \mu_1 = 0.32,\ \mu_2 = 0.38,\ K = 20,\ c(x) = 0.1x^2 + x.$$

The results can be seen in the right two graphs in Figure 5. In the first graph we can see the optimal threshold for swapping service speeds occur at 7 customers in the system: $\delta^* = 7$.

Example 4.6. *Choose the following parameters:*

$$\lambda = 0.3,\ \mu_1 = 0.32,\ \mu_2 = 0.38,\ K = 20,\ \alpha = 0.9,\ c(x) = 0.1x^2 + x.$$

The results can be seen in the left two graphs in Figure 6. In the first graph we can see the optimal threshold for swapping service speeds occur at 182 customers in the system: $\delta^* = 182$.

Example 4.7. *Choose the following parameters:*

$$\lambda = 0.3,\ \mu_1 = 0.32,\ \mu_2 = 0.38,\ K = 20,\ \alpha = 0.99,\ c(x) = 0.1x^2 + x.$$

The results can be seen in the right two graphs in Figure 6. In the first graph we can see the optimal threshold for swapping service speeds occur at 17 customers in the system: $\delta^* = 17$. This is the first time we see that the value of the upper function can take negative value and decrease, even though we see that after 50 iterations, the value is no longer negative for any number of customers in the system.

Example 4.8. *Choose the following parameters:*

$$\lambda = 0.3,\ \mu_1 = 0.32,\ \mu_2 = 0.38,\ K = 20,\ \alpha = 0.9,\ c(x) = 0.01x^3 + 0.1x^2 + x.$$

The results can be seen in the left two graphs in Figure 7. In the first graph we can see the optimal threshold for swapping service speeds occur at 33 customers in the system: $\delta^* = 33$. Remarkably, both the lower and upper bounds for the threshold are reached in the first iteration.

Example 4.9. *Choose the following parameters:*

$$\lambda = 0.3,\ \mu_1 = 0.32,\ \mu_2 = 0.38,\ K = 20,\ \alpha = 0.99,\ c(x) = 0.01x^2 + 0.1x^2 + x.$$

The results can be seen in the right two graphs in Figure 7. In the first graph we can see the optimal threshold for swapping service speeds occur at 9 customers in the system: $\delta^* = 9$. Once again, we determine the optimal threshold to be lower for $\alpha = 0.99$ than for $\alpha = 0.9$.

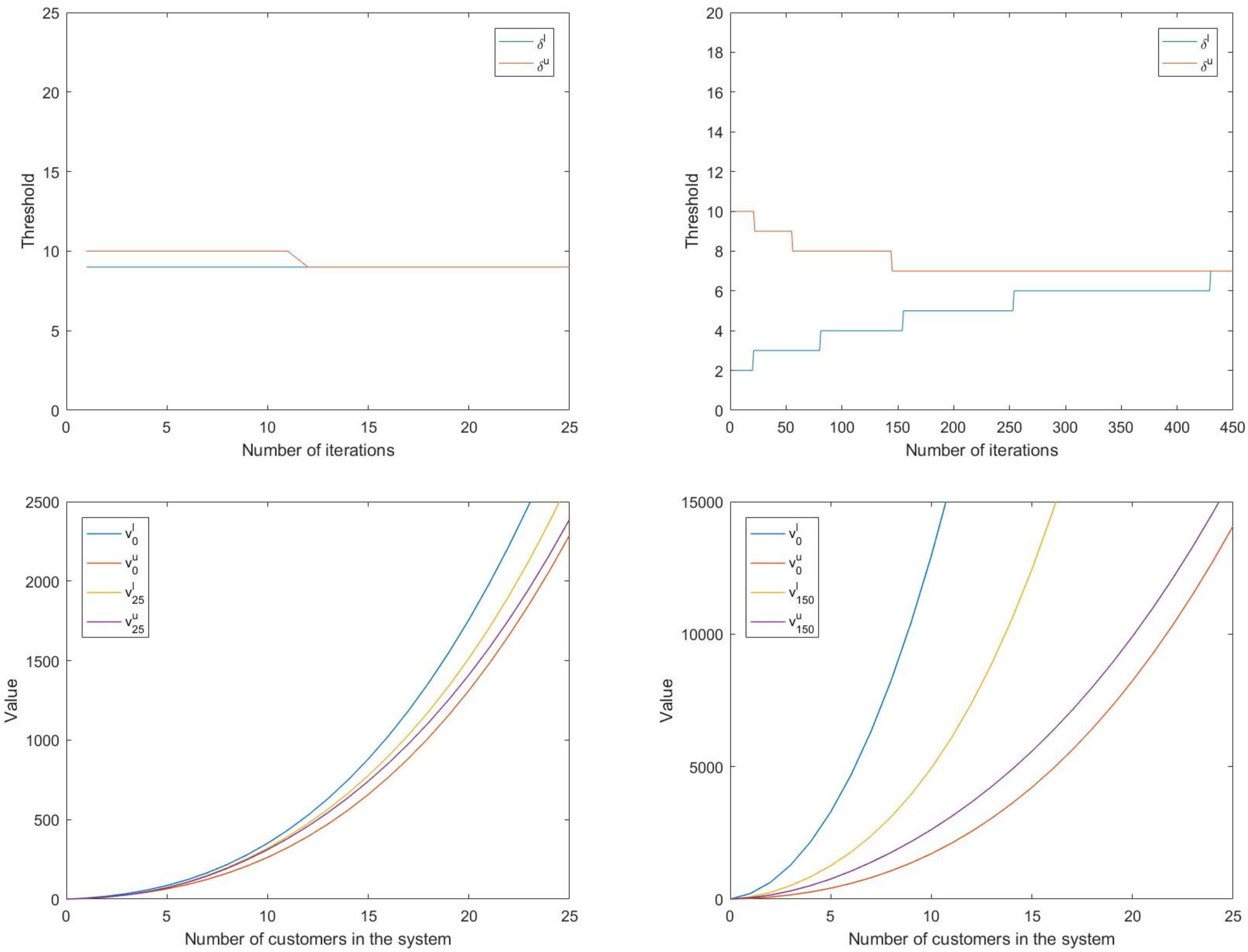

Figure 5: Optimal threshold and value of VI Algorithm for a cubic initial value function.

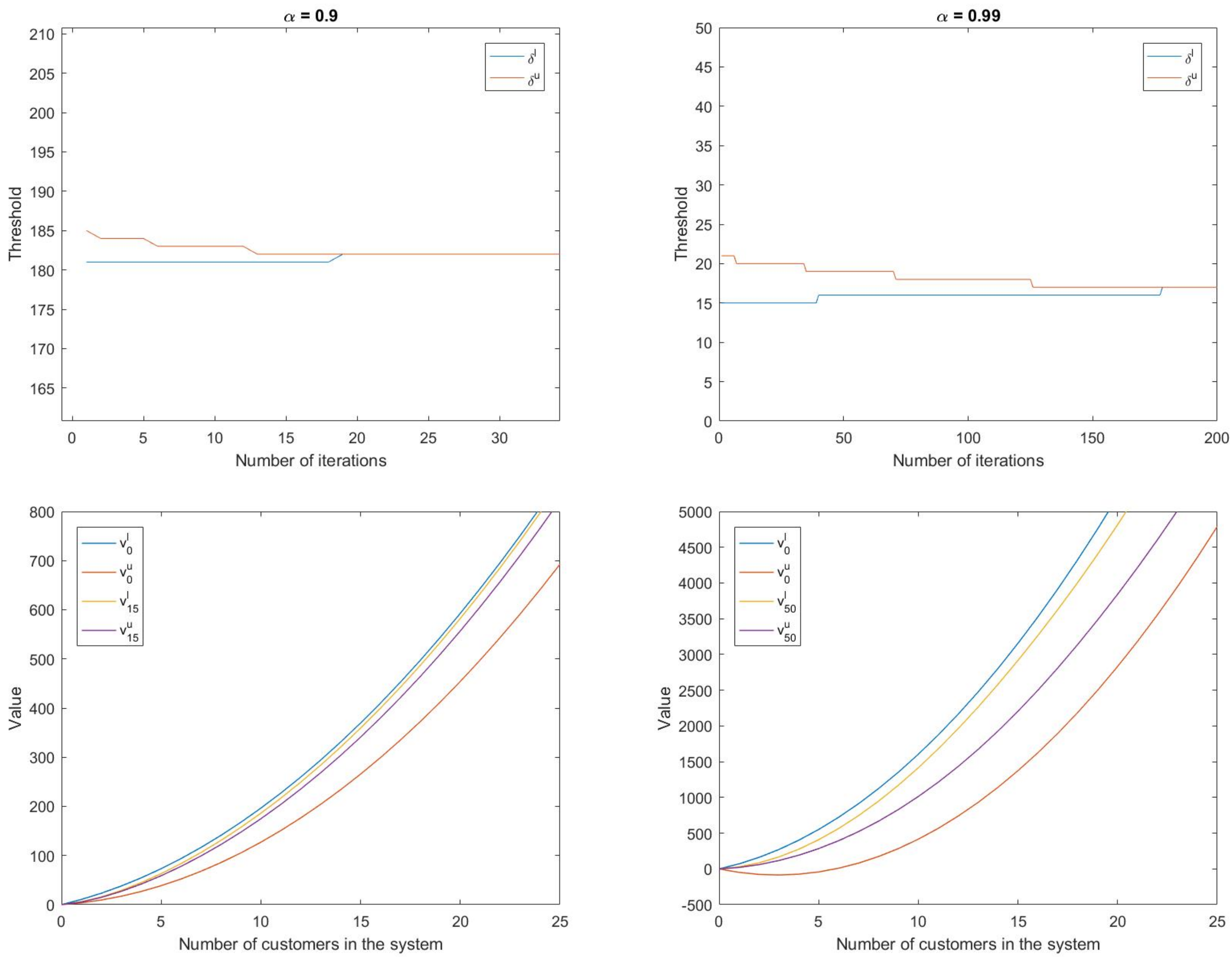

Figure 6: Optimal threshold and value of VI Algorithm for a quadratic initial value function with discounting.

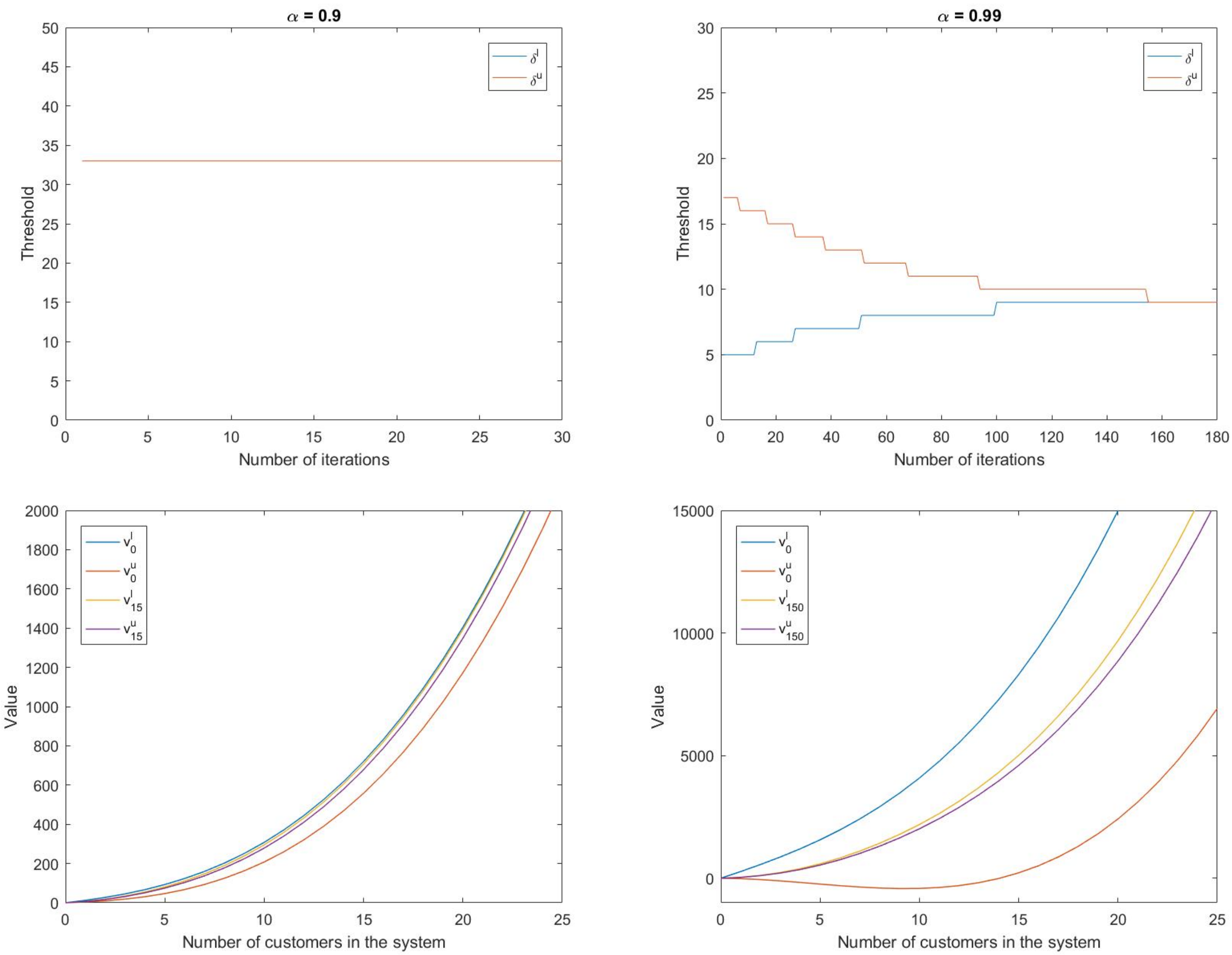

Figure 7: Optimal threshold and value of VI Algorithm for a cubic initial value function with discounting.

4.4 Effect of value of K

We have looked into the behaviour of the value of the extra costs K for the faster service.

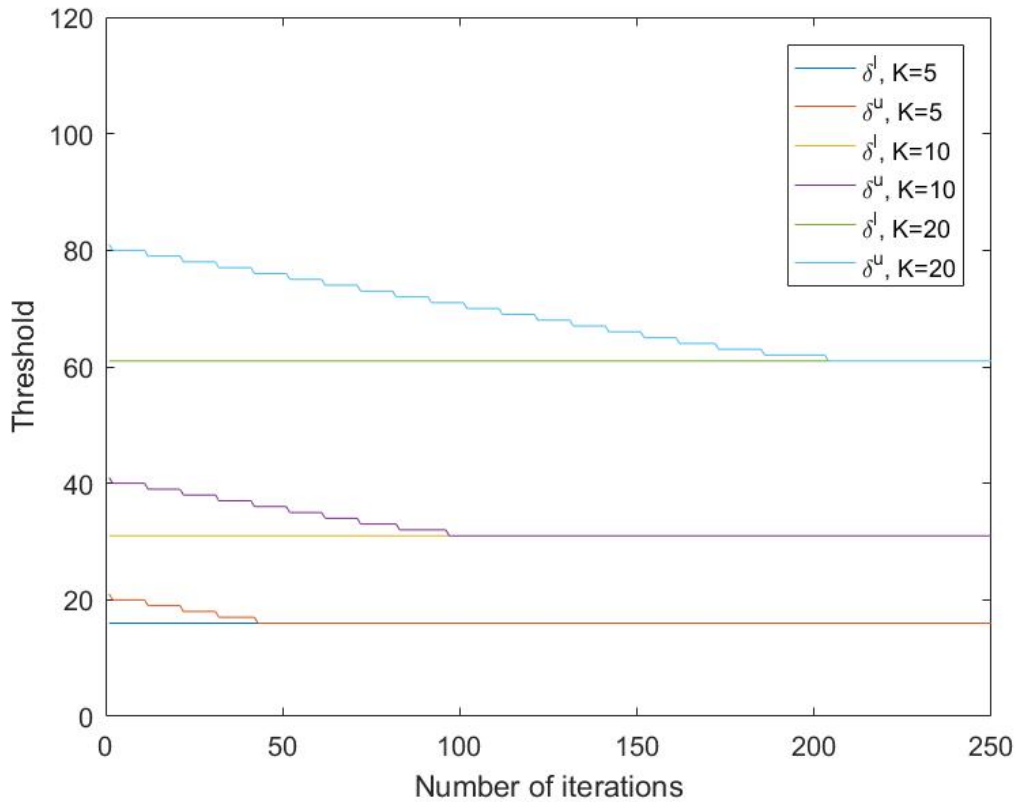

Figure 8: Optimal threshold of the VI Algorithm for a quadratic initial function and various values for K.

Example 4.10. *Choose the following parameters*

$$\lambda = 0.1,\ \mu_1 = 0.4,\ \mu_2 = 0.5,\ \alpha = 1,\ c(x) = x.$$

Various values for K: 5, 10 and 20.

The results can be seen in Figure 8. Logically we immediately see that for a higher value of K, both the thresholds for the upper and lower initial value functions are higher, and as such the optimal threshold too. Remarkably, each set of thresholds for the same value of K show a similar pattern. The lower initial value function reaches the optimal threshold in the first iteration for each value of K. For these parameters, the lower initial value function, found in Lemma (2.2), shows to be close to the optimal value, as can also be seen in the bottom left graph of Figure 4.

5 Conclusion and Discussion

As seen in the result section, we can conclude that our approach in constructing initial value functions for the Value Iteration Algorithm is successful in determining an optimal strategy for faster service in an $M/M/1$ queue with two possible service speeds. This takes into account the models with and without discounting.

In general, we see that the lower initial function reaches the optimal threshold in less iterations than the upper initial value does, and therefore seems to be close to the optimal average cost value function. Furthermore, from the first iteration of the VI Algorithm, it is possible to get an upper and lower bound for the threshold, and as such provides an interval that bounds the optimal strategy.

Also, we have chosen a polynomial in the form $\sum_{i=0}^{n} a_i x^i$ as initial value function. It might be interesting to look at functions of a different form, and if the lower and upper function converge faster.

References

Ertiningsih D., Bhulai S., Spieksma F. (2018). A novel use of value iteration for deriving bounds for threshold and switching curve optimal policies. *Naval Research Logistics 65(8)*, 638–659. https://doi.org/10.1002/nav.21824

Koole, G. (1998). Structural results for the control of queueing systems using event-based dynamic programming. *Queueing Systems 30*, 323–339. https://doi.org/10.1023/A:1019177307418

Lippman, S. A. (1975). Applying a new device in the optimization of exponential queueing systems. *Operational Research 23(4)*, 687–710. https://doi.org/10.1287/opre.23.4.687

Serfozo, R. F. (1979). An Equivalence between Continuous and Discrete Time Markov Decision Processes. *Operations Research, 27(3)*, 616–620. http://www.jstor.org/stable/170221

A Quadratic initial value function

Proof of Lemma 2.2. For linear holding costs, we will look into an initial value function of quadratic form. Thus, let $c(x) = c_a x$ and $v_0(x) = ax^2 + bx + c$. We will first try to find conditions for a for the case where $x \geq 1$ and then consider the case where $x = 0$ and find solutions for b and c such that equations (6) and (7) hold for all $x \in X$. Initially, we will only consider the lower initial function and take into account that $\lambda < \mu_1 < \mu_2$. Symmetry arguments will lead to an expression for the upper function.

Consider $x \geq 1$**.**

$$
\begin{aligned}
& v_0^l(x+1) - v_0^l(x) \geq v_1^l(x+1) - v_1^l(x) \\
\iff \quad & 2ax + a + b \geq c_a + \lambda(2ax + 3a + b) + \mu_1(2ax + a + b) + \mu_2(2ax - a + b) \\
& \qquad + (\mu_2 - \mu_1)\left[\min\left\{2ax + a + b, \frac{K}{\mu_2 - \mu_1}\right\} - \min\left\{2ax - a + b, \frac{K}{\mu_2 - \mu_1}\right\}\right] \\
\iff \quad & 2ax + a + b \geq c_a + (\lambda + \mu_1 + \mu_2)(2ax + a + b) + 2(\lambda - \mu_2)a \\
& \qquad + (\mu_2 - \mu_1)\left[\min\left\{2ax + a + b, \frac{K}{\mu_2 - \mu_1}\right\} - \min\left\{2ax - a + b, \frac{K}{\mu_2 - \mu_1}\right\}\right] \\
\iff \quad & 0 \geq c_a + 2(\lambda - \mu_2)a \\
& \qquad + (\mu_2 - \mu_1)\left[\min\left\{2ax + a + b, \frac{K}{\mu_2 - \mu_1}\right\} - \min\left\{2ax - a + b, \frac{K}{\mu_2 - \mu_1}\right\}\right].
\end{aligned}
$$

- **Case 1:** $2ax - a + b, 2ax + a + b \leq \frac{K}{\mu_2 - \mu_1}$

$$
\begin{aligned}
& 0 \geq c_a + 2(\lambda - \mu_2)a + (\mu_2 - \mu_1)(2a) \\
\iff \quad & 2(\mu_1 - \lambda)a \geq c_a \\
\iff \quad & a \geq \frac{c_a}{2(\mu_1 - \lambda)}.
\end{aligned}
$$

- **Case 2:** $\frac{K}{\mu_2 - \mu_1} < 2ax - a + b, 2ax + a + b$

$$
\begin{aligned}
& 0 \geq c_a + 2(\lambda - \mu_2)a \\
\iff \quad & 2(\mu_2 - \lambda)a \geq c_a \\
\iff \quad & a \geq \frac{c_a}{2(\mu_2 - \lambda)}.
\end{aligned}
$$

- **Case 3:** $2ax - a + b \leq \frac{K}{\mu_2 - \mu_1} < 2ax + a + b$

$$
0 \geq c_a + 2(\lambda - \mu_2)a + (\mu_2 - \mu_1)\left[\frac{K}{\mu_2 - \mu_1} - (2ax - a + b)\right].
$$

  This case reduces to **Case 1**, using that $\frac{K}{\mu_2 - \mu_1} < 2ax + a + b$.

As $\mu_2 > \mu_1$, we find that when $a \geq \frac{c_a}{2(\mu_1 - \lambda)}$, Equation (6) holds for all states $x \geq 1$.

Consider $x = 0$ and let $a = \frac{c_a}{2(\mu_1 - \lambda)}$.

$$v_0^l(1) - v_0^l(0) \geq v_1^l(1) - v_1^l(0)$$
$$a + b \geq c_a + \lambda(3a + b) + \mu_1(a + b) + (\mu_2 - \mu_1) \min\left\{a + b, \frac{K}{\mu_2 - \mu_1}\right\}.$$

- **Case 1:** $a + b \leq \frac{K}{\mu_2 - \mu_1}$

$$\begin{aligned}
& a + b \geq c_a + \lambda(3a + b) + \mu_1(a + b) + (\mu_2 - \mu_1)(a + b) \\
\iff \quad & a + b \geq c_a + (\lambda + \mu_1 + \mu_2)(a + b) + 2\lambda a - \mu_1(a + b) \\
\iff \quad & 2(\mu_1 - \lambda)a - \mu_1 a + \mu_1 b \geq c_a \\
\iff \quad & 2(\mu_1 - \lambda)\frac{c_a}{2(\mu_1 - \lambda)} - \mu_1 \frac{c_a}{2(\mu_1 - \lambda)} + \mu_1 b \geq c_a \\
\iff \quad & b \geq \frac{c_a}{2(\mu_1 - \lambda)}.
\end{aligned}$$

- **Case 2:** $\frac{K}{\mu_2 - \mu_1} < a + b$

$$a + b \geq c_a + \lambda(3a + b) + \mu_1(a + b) + (\mu_2 - \mu_1)\frac{K}{\mu_2 - \mu_1}.$$

  This case reduces to **Case 1**, using that $\frac{K}{\mu_2 - \mu_1} < a + b$.

Hence, we find that for $a, b \geq \frac{c_a}{2(\mu_1 - \lambda)}$ and c arbitrary, the Equation (6) holds for all $x \in X$. Therefore, let

$$v_0^l(x) = \frac{c_a}{2(\mu_1 - \lambda)}(x^2 + x).$$

In like manner, we find that Equation (7) holds for all $x \in X$ for $a, b \leq \frac{c_a}{2(\mu_2 - \lambda)}$ and $c = 0$. Thus, let

$$v_0^u(x) = \frac{c_a}{2(\mu_2 - \lambda)}(x^2 + x).$$

□

B Cubic initial value function

Proof of Lemma 2.3. For quadratic holding costs, we will look into an initial value function of cubic form. Thus, let $c(x) = c_a x^2 + c_b x$ and $v_0(x) = ax^3 + bx^2 + cx + d$. We will first try to find conditions for a, b for the case where $x \geq 1$ and then consider the case where $x = 0$ and find solutions for c and d such that equations (6) and (7) hold for all $x \in X$. Our approach in each case is grouping factors of the same degree of x in the equation and then letting the structure of $c(x+1) - c(x)$ lead into finding expressions for a, b, c, d. Initially, we will only consider the lower initial function and take into account that $\lambda < \mu_1 < \mu_2$. Symmetry arguments will lead to an expression for the upper function.

Consider $x \geq 1$**.**

$$
\begin{aligned}
& v_0^l(x+1) - v_0^l(x) \geq v_1^l(x+1) - v_1^l(x) \\
\iff \quad & (3x^2+3x+1)a + (2x+1)b + c \geq (2x+1)c_a + c_b \\
& \quad + \lambda((3x^2+9x+7)a + (2x+3)b + c) \\
& \quad + \mu_1((3x^2+3x+1)a + (2x+1)b + c) \\
& \quad + \mu_2((3x^2-3x+1)a + (2x-1)b + c) \\
& \quad + (\mu_2-\mu_1)\left[\min\left\{(3x^2+3x+1)a + (2x+1)b + c, \frac{K}{\mu_2-\mu_1}\right\}\right. \\
& \quad \left. - \min\left\{(3x^2-3x+1)a + (2x-1)b + c, \frac{K}{\mu_2-\mu_1}\right\}\right] \\
\iff \quad & (3x^2+3x+1)a + (2x+1)b + c \geq (2x+1)c_a + c_b \\
& \quad + (\lambda+\mu_1+\mu_2)((3x^2+3x+1)a + (2x+1)b + c) \\
& \quad + \lambda((6x+6)a + 2b) + \mu_2(-6xa - 2b) \\
& \quad + (\mu_2-\mu_1)\left[\min\left\{(3x^2+3x+1)a + (2x+1)b + c, \frac{K}{\mu_2-\mu_1}\right\}\right. \\
& \quad \left. - \min\left\{(3x^2-3x+1)a + (2x-1)b + c, \frac{K}{\mu_2-\mu_1}\right\}\right] \\
\iff \quad & 0 \geq (2x+1)c_a + c_b \\
& \quad + \lambda((6x+6)a + 2b) + \mu_2(-6xa - 2b) \\
& \quad + (\mu_2-\mu_1)\left[\min\left\{(3x^2+3x+1)a + (2x+1)b + c, \frac{K}{\mu_2-\mu_1}\right\}\right. \\
& \quad \left. - \min\left\{(3x^2-3x+1)a + (2x-1)b + c, \frac{K}{\mu_2-\mu_1}\right\}\right].
\end{aligned}
$$

- **Case 1:** $(3x^2-3x+1)a + (2x-1)b + c, (3x^2+3x+1)a + (2x+1)b + c \leq \frac{K}{\mu_2-\mu_1}$

$$
\begin{aligned}
& 0 \geq (2x+1)c_a + c_b + \lambda((6x+6)a + 2b) + \mu_2(-6xa - 2b) + (\mu_2-\mu_1)(6xa + 2b) \\
\iff \quad & 0 \geq (2x+1)c_a + c_b + (\lambda-\mu_1)(3(2x+1)a + 2b) + 3(\lambda+\mu_1)a \\
\iff \quad & 3(\mu_1-\lambda)(2x+1)a + 2(\mu_1-\lambda)b \geq (2x+1)c_a + c_b + 3(\lambda+\mu_1)a.
\end{aligned}
$$

The last inequality holds for $a \geq \frac{c_a}{3(\mu_1-\lambda)}$, $b \geq \frac{c_b+3(\lambda+\mu_1)a}{2(\mu_1-\lambda)}$ for all states $x \geq 1$.

- **Case 2:** $\frac{K}{\mu_2-\mu_1} < (3x^2-3x+1)a + (2x-1)b + c, (3x^2+3x+1)a + (2x+1)b + c$

$$\begin{aligned} & 0 \geq (2x+1)c_a + c_b + \lambda((6x+6)a + 2b) + \mu_2(-6xa - 2b) \\ \iff \quad & 0 \geq (2x+1)c_a + c_b + (\lambda - \mu_2)(3(2x+1)a + 2b) + 3(\lambda+\mu_2)a - 2\mu_2 b \\ \iff \quad & 3(\mu_2-\lambda)(2x+1)a + 2(\mu_2-\lambda)b \geq (2x+1)c_a + c_b + 3(\lambda+\mu_2)a. \end{aligned}$$

The last inequality holds for $a \geq \frac{c_a}{3(\mu_2-\lambda)}$, $b \geq \frac{c_b+3(\lambda+\mu_2)a}{2(\mu_2-\lambda)}$ for all states $x \geq 1$.

- **Case 3:** $(3x^2-3x+1)a + (2x-1)b + c \leq \frac{K}{\mu_2-\mu_1} < (3x^2+3x+1)a + (2x+1)b + c$

$$0 \geq (2x+1)c_a + c_b + \lambda((6x+6)a+2b) + \mu_2(-6xa-2b) + (\mu_2-\mu_1)\left[\frac{K}{\mu_2-\mu_1} - ((3x^2-3x+1)a + (2x-1)b + c)\right].$$

This case reduces to **Case 1**, using $\frac{K}{\mu_2-\mu_1} < (3x^2+3x+1)a + (2x+1)b + c$. As such, the inequality holds for $a \geq \frac{c_a}{3(\mu_1-\lambda)}$, $b \geq \frac{c_b+3(\lambda+\mu_1)a}{2(\mu_1-\lambda)}$ for all states $x \geq 1$.

To determine if one of the values of b found in the three cases is greater or equal to the other, we will see if the function $f(z) := \frac{c_b+(\lambda+z)\frac{c_a}{z-\lambda}}{2(z-\lambda)}$ for $z \in (\lambda, 1)$ is an increasing or decreasing function in z. As, $z > \lambda > 0$ we find that

$$f'(z) = \frac{c_a(-z^2-2\lambda z+3\lambda^2) - c_b(z-\lambda)^2}{(z-\lambda)^4} < \frac{c_a(-\lambda^2-2\lambda^2+3\lambda^2) - c_b(z-\lambda)^2}{(z-\lambda)^4} = -\frac{c_b}{(z-\lambda)^4} < 0.$$

Therefore, we can conclude that as $1 > \mu_2 > \mu_1 > \lambda > 0$, we obtain that

$$\frac{c_b+3(\lambda+\mu_1)a}{2(\mu_1-\lambda)} > \frac{c_b+3(\lambda+\mu_2)a}{2(\mu_2-\lambda)},$$

resulting in all three cases to hold for $a \geq \frac{c_a}{3(\mu_1-\lambda)}$ and $b \geq \frac{c_b+3(\lambda+\mu_1)a}{2(\mu_1-\lambda)}$ for all states $x \geq 1$.

Consider $x = 0$ and let $a = \frac{c_a}{3(\mu_1-\lambda)}$ and $b = \frac{c_b+3(\lambda+\mu_1)a}{2(\mu_1-\lambda)}$.

$$\begin{aligned} & v_0^l(1) - v_0^l(0) \geq v_1^l(1) - v_1^l(0) \\ \iff \quad & a+b+c \geq c_a + c_b + \lambda(7a+3b+c) + \mu_1(a+b+c) + (\mu_2-\mu_1)\min\left\{a+b+c, \frac{K}{\mu_2-\mu_1}\right\}. \end{aligned}$$

- **Case 1:** $a+b+c \leq \frac{K}{\mu_2-\mu_1}$

$$\begin{aligned} & a+b+c \geq c_a + c_b + \lambda(7a+3b+c) + \mu_1(a+b+c) + (\mu_2-\mu_1)(a+b+c) \\ \iff \quad & a+b+c \geq c_a + c_b + (\lambda+\mu_1+\mu_2)(a+b+c) + \lambda(6a+2b) - \mu_1(a+b+c) \\ \iff \quad & 3(\mu_1-\lambda)a + 2(\mu_1-\lambda)b + \mu_1 c \geq c_a + c_b + 3(\lambda+\mu_1)a - \mu_1 a + \mu_1 b \\ \iff \quad & \mu_1 c \geq \frac{b-a}{\mu_1} \\ \iff \quad & c \geq b - a. \end{aligned}$$

- **Case 2:** $\frac{K}{\mu_2-\mu_1} < a+b+c$

$$a+b+c \geq c_a + c_b + \lambda(7a+3b+c) + \mu_1(a+b+c) + (\mu_2-\mu_1)\frac{K}{\mu_2-\mu_1}.$$

  This case reduces to **Case 1**, using $\frac{K}{\mu_2-\mu_1} < a+b+c$.

Hence, we find that for $c \geq \frac{c_b+(\lambda+\mu_1)\frac{c_a}{\mu_1-\lambda}}{2(\mu_1-\lambda)} - \frac{c_a}{3(\mu_1-\lambda)}$ and d arbitrary, Equation (6) holds for all $x \in X$. Therefore, let

$$v_0^l(x) = \frac{c_a}{3(\mu_1-\lambda)}x^3 + \frac{c_b+(\lambda+\mu_1)\frac{c_a}{\mu_1-\lambda}}{2(\mu_1-\lambda)}x^2 + \left(\frac{c_b+(\lambda+\mu_1)\frac{c_a}{\mu_1-\lambda}}{2(\mu_1-\lambda)} - \frac{c_a}{3(\mu_1-\lambda)}\right)x.$$

In like manner, we find that Equation (7) holds for all $x \in X$ for $a \leq \frac{c_a}{3(\mu_2-\lambda)}$, $b \leq \frac{c_b+3(\lambda+\mu_2)a}{2(\mu_2-\lambda)}$, $c \leq \frac{c_b+(\lambda+\mu_2)\frac{c_a}{\mu_2-\lambda}}{2(\mu_2-\lambda)} - \frac{c_a}{3(\mu_2-\lambda)}$ and $d=0$. Thus, let

$$v_0^u(x) = \frac{c_a}{3(\mu_2-\lambda)}x^3 + \frac{c_b+(\lambda+\mu_2)\frac{c_a}{\mu_2-\lambda}}{2(\mu_2-\lambda)}x^2 + \left(\frac{c_b+(\lambda+\mu_2)\frac{c_a}{\mu_2-\lambda}}{2(\mu_2-\lambda)} - \frac{c_a}{3(\mu_2-\lambda)}\right)x.$$

□

C Discounting with a quadratic initial value function

Proof of Lemma 3.1. For a system with discounting with quadratic holding costs, we will look into an initial value function of quadratic form. Thus, let $c(x) = c_a x^2 + c_b x$ and $v_0(x) = ax^2 + bx + c$. We will first try to find conditions for a, b for the case where $x \geq 1$ and then consider the case where $x = 0$ and possibly adjust the solution for b and find an expression for c such that equations (6) and (7) hold for all $x \in X$. Our approach in each case is grouping factors of the same degree of x in the equation and then letting the structure of $c(x+1) - c(x)$ lead into finding expressions for a, b, c. Initially, we will only consider the lower initial function and take into account that $\lambda < \mu_1 < \mu_2$ and that $\alpha \in (0, 1)$. Symmetry arguments will lead to an expression for the upper function.

Consider $x \geq 1$:

$$
\begin{aligned}
& v_0^{l,\alpha}(x+1) - v_0^{l,\alpha}(x) \geq v_1^{l,\alpha}(x+1) - v_1^{l,\alpha}(x) \\
\iff \quad & 2ax + a + b \geq 2c_a x + c_a + c_b + \alpha\lambda(2ax + 3a + b) + \alpha\mu_1(2ax + a + b) + \alpha\mu_2(2ax - a + b) \\
& \qquad + \alpha(\mu_2 - \mu_1)\left[\min\left\{2ax + a + b, \frac{K}{\alpha(\mu_2 - \mu_1)}\right\} - \min\left\{2ax - a + b, \frac{K}{\alpha(\mu_2 - \mu_1)}\right\}\right] \\
\iff \quad & 2ax + a + b \geq 2c_a x + c_a + c_b + \alpha(\lambda + \mu_1 + \mu_2)(2ax + a + b) + 2\alpha(\lambda - \mu_2)a \\
& \qquad + \alpha(\mu_2 - \mu_1)\left[\min\left\{2ax + a + b, \frac{K}{\alpha(\mu_2 - \mu_1)}\right\} - \min\left\{2ax - a + b, \frac{K}{\alpha(\mu_2 - \mu_1)}\right\}\right] \\
\iff \quad & (1-\alpha)(2ax + a + b) \geq 2c_a x + c_a + c_b + 2\alpha(\lambda - \mu_2)a \\
& \qquad + \alpha(\mu_2 - \mu_1)\left[\min\left\{2ax + a + b, \frac{K}{\alpha(\mu_2 - \mu_1)}\right\} - \min\left\{2ax - a + b, \frac{K}{\alpha(\mu_2 - \mu_1)}\right\}\right].
\end{aligned}
$$

- **Case 1:** $2ax - a + b, 2ax + a + b \leq \frac{K}{\alpha(\mu_2 - \mu_1)}$

$$
\begin{aligned}
& (1-\alpha)(2ax + a + b) \geq 2c_a x + c_a + c_b + 2\alpha(\lambda - \mu_2)a + 2\alpha(\mu_2 - \mu_1)a \\
\iff \quad & (1-\alpha)(2x+1)a + (1-\alpha)b \geq (2x+1)c_a + c_b + 2\alpha(\lambda - \mu_1)a.
\end{aligned}
$$

  The last inequality holds for $a \geq \frac{c_a}{1-\alpha}$ and $b \geq \frac{c_b + 2\alpha(\lambda - \mu_1)a}{1-\alpha}$ for all states $x \geq 1$.

- **Case 2:** $\frac{K}{\alpha(\mu_2 - \mu_1)} \leq 2ax - a + b, 2ax + a + b$

$$
\begin{aligned}
& (1-\alpha)(2ax + a + b) \geq 2c_a x + c_a + c_b + 2\alpha(\lambda - \mu_2)a \\
\iff \quad & (1-\alpha)(2x+1)a + (1-\alpha)b \geq (2x+1)c_a + c_b + 2\alpha(\lambda - \mu_2)a.
\end{aligned}
$$

  The last inequality holds for $a \geq \frac{c_a}{1-\alpha}$ and $b \geq \frac{c_b + 2\alpha(\lambda - \mu_2)a}{1-\alpha}$ for all states $x \geq 1$.

- **Case 3:** $2ax - a + b < \frac{K}{\alpha(\mu_2 - \mu_1)} < 2ax + a + b$

$$
(1-\alpha)(2ax + a + b) \geq 2c_a x + c_a + c_b + 2\alpha(\lambda - \mu_2)a + \alpha(\mu_2 - \mu_1)\left[\frac{K}{\alpha(\mu_2 - \mu_1)} - (2ax - a + b)\right].
$$

  This case reduces to **Case 1**, using $\frac{K}{\alpha(\mu_2 - \mu_1)} < 2ax + a + b$. As such, the inequality holds for $a \geq \frac{c_a}{1-\alpha}$ and $b \geq \frac{c_b + 2\alpha(\lambda - \mu_1)a}{1-\alpha}$ for all states $x \geq 1$.

As $\mu_2 > \mu_1 > \lambda$ we find that all three cases hold for $a \geq \frac{c_a}{1-\alpha}$, $b \geq \frac{c_b + 2\alpha(\lambda - \mu_1)a}{1-\alpha}$ for all states $x \geq 1$.

Consider $x = 0$ and let $a = \frac{c_a}{1-\alpha}$:

$$\begin{aligned}
& v_0^{l,\alpha}(1) - v_0^{l,\alpha}(0) \geq v_1^{l,\alpha}(1) - v_1^{l,\alpha}(0) \\
\iff\quad & a + b \geq c_a + c_b + \alpha\lambda(3a+b) + \alpha\mu_1(a+b) \\
& \quad + \alpha(\mu_2 - \mu_1)\min\left\{a+b, \frac{K}{\alpha(\mu_2-\mu_1)}\right\} \\
\iff\quad & a + b \geq c_a + c_b + \alpha(\lambda + \mu_1 + \mu_2)(a+b) + 2\alpha\lambda a - \alpha\mu_2(a+b) \\
& \quad + \alpha(\mu_2 - \mu_1)\min\left\{a+b, \frac{K}{\alpha(\mu_2-\mu_1)}\right\} \\
\iff\quad & (1-\alpha)(a+b) \geq c_a + c_b + 2\alpha\lambda a - \alpha\mu_2(a+b) \\
& \quad + \alpha(\mu_2 - \mu_1)\min\left\{a+b, \frac{K}{\alpha(\mu_2-\mu_1)}\right\}.
\end{aligned}$$

- **Case 1:** $a + b \leq \frac{K}{\alpha(\mu_2-\mu_1)}$

$$\begin{aligned}
& (1-\alpha)(a+b) \geq c_a + c_b + 2\alpha\lambda a - \alpha\mu_2(a+b) + \alpha(\mu_2-\mu_1)(a+b) \\
\iff\quad & (1-\alpha)(a+b) \geq c_a + c_b + 2\alpha\lambda a - \alpha\mu_1(a+b) \\
\iff\quad & (1-\alpha+\alpha\mu_1)b \geq c_b + \alpha(2\lambda - \mu_1)a.
\end{aligned}$$

  The last inequality holds for $b \geq \frac{c_b + \alpha(2\lambda-\mu_1)a}{1-\alpha+\alpha\mu_1}$ for state 0.

- **Case 2:** $\frac{k}{\alpha(\mu_2-\mu_1)} < a + b$

$$(1-\alpha)(a+b) \geq c_a + c_b + 2\alpha\lambda a - \alpha\mu_2(a+b) + \alpha(\mu_2-\mu_1)\frac{K}{\mu_2-\mu_1}.$$

  This case reduces to **Case 1**, using $\frac{K}{\alpha(\mu_2-\mu_1)} < a + b$. As such, the inequality holds for $b \geq \frac{c_b+\alpha(2\lambda-\mu_1)a}{1-\alpha+\alpha\mu_1}$ for state 0.

Now, we have found two different expressions for b. However, neither expression is greater or equal to the other for all $\mu_2 > \mu_1$. Therefore, we will choose the maximum between the two choices depending on the parameter input. This way we can stay as close as possible to equality in Equation (6).

Therefore, let

$$v_0^{l,\alpha} = \frac{c_a}{1-\alpha}x^2 + \max\left\{\frac{c_b + 2\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}, \frac{c_b + \alpha(2\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha+\alpha\mu_1}\right\}x,$$

and

$$v_0^{u,\alpha} = \frac{c_a}{1-\alpha}x^2 + \min\left\{\frac{c_b + 2\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}, \frac{c_b + \alpha(2\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha+\alpha\mu_2}\right\}x,$$

such that equations (6) and (7) hold for all $x \in X$. □

D Discounting with a cubic initial value function

Proof of Lemma 3.2. We are interested in a cubic v_0 for the VI Algorithm with discounting when the holding costs are of cubic form. Thus, let $c(x) = c_a x^3 + c_b x^2 + c_c x$ and $v_0(x) = ax^3 + bx^2 + cx + d$. We will first try to find conditions for a, b for the case where $x \geq 1$ and then consider the case where $x = 0$ and possibly adjust the solution for c and find an expression for d such that equations (6) and (7) hold for all $x \in X$. Our approach in each case is grouping factors of the same degree of x in the equation and then letting the structure of $c(x+1) - c(x)$ lead into finding expressions for a, b, c, d. Initially, we will only consider the lower initial function and take into account that $\lambda < \mu_1 < \mu_2$ and that $\alpha \in (0,1)$. Symmetry arguments will lead to an expression for the upper function.

Consider $x \geq 1$:

$$\begin{aligned}
& v_0^{l,\alpha}(x+1) - v_0^{l,\alpha}(x) \geq v_1^{l,\alpha}(x+1) - v_1^{l,\alpha}(x) \\
\iff\quad & (3x^2+3x+1)a + (2x+1)b + c \geq (3x^2+3x+1)c_a + (2x+1)c_b + c_c \\
& \quad + \alpha\lambda((3x^2+9x+7)a + (2x+3)b + c) \\
& \quad + \alpha\mu_1((3x^2+3x+1)a + (2x+1)b + c) \\
& \quad + \alpha\mu_2((3x^2-3x+1)a + (2x-1)b + c) \\
& \quad + \alpha(\mu_2-\mu_1)\left[\min\left\{(3x^2+3x+1)a + (2x+1)b + c, \frac{K}{\alpha(\mu_2-\mu_1)}\right\}\right. \\
& \quad \left. - \min\left\{(3x^2-3x+1)a + (2x-1)b + c, \frac{K}{\alpha(\mu_2-\mu_1)}\right\}\right] \\
\iff\quad & (3x^2+3x+1)a + (2x+1)b + c \geq (3x^2+3x+1)c_a + (2x+1)c_b + c_c \\
& \quad + \alpha(\lambda+\mu_1+\mu_2)((3x^2+3x+1)a + (2x+1)b + c) \\
& \quad + \alpha\lambda((6x+6)a + 2b) + \alpha\mu_2(-6xa - 2b) \\
& \quad + \alpha(\mu_2-\mu_1)\left[\min\left\{(3x^2+3x+1)a + (2x+1)b + c, \frac{K}{\alpha(\mu_2-\mu_1)}\right\}\right. \\
& \quad \left. - \min\left\{(3x^2-3x+1)a + (2x-1)b + c, \frac{K}{\alpha(\mu_2-\mu_1)}\right\}\right] \\
\iff\quad & (1-\alpha)((3x^2+3x+1)a + (2x+1)b + c) \geq (3x^2+3x+1)c_a + (2x+1)c_b + c_c \\
& \quad + \alpha\lambda((6x+6)a + 2b) + \alpha\mu_2(-6xa - 2b) \\
& \quad + \alpha(\mu_2-\mu_1)\left[\min\left\{(3x^2+3x+1)a + (2x+1)b + c, \frac{K}{\alpha(\mu_2-\mu_1)}\right\}\right. \\
& \quad \left. - \min\left\{(3x^2-3x+1)a + (2x-1)b + c, \frac{K}{\alpha(\mu_2-\mu_1)}\right\}\right].
\end{aligned}$$

- **Case 1:** $(3x^2-3x+1)a+(2x-1)b+c, (3x^2+3x+1)a+(2x+1)b+c \leq \frac{K}{\alpha(\mu_2-\mu_1)}$

$$\begin{aligned}
&(1-\alpha)((3x^2+3x+1)a+(2x+1)b+c) \geq (3x^2+3x+1)c_a+(2x+1)c_b+c_c \\
&\qquad +\alpha\lambda((6x+6)a+2b)+\alpha\mu_2(-6xa-2b) \\
&\qquad +\alpha(\mu_2-\mu_1)(6xa+2b) \\
\iff\quad &(1-\alpha)((3x^2+3x+1)a+(2x+1)b+c) \geq (3x^2+3x+1)c_a+(2x+1)(c_b+3\alpha(\lambda-\mu_1)a) \\
&\qquad +c_c+3\alpha(\lambda+\mu_1)a+2\alpha(\lambda-\mu_1)b.
\end{aligned}$$

The last inequality holds for $a \geq \frac{c_a}{1-\alpha}$, $b \geq \frac{c_b+3\alpha(\lambda-\mu_1)a}{1-\alpha}$ and $c \geq \frac{c_c+3\alpha(\lambda+\mu_1)a+2\alpha(\lambda-\mu_1)b}{1-\alpha}$ for all states $x \geq 1$.

- **Case 2:** $\frac{K}{\alpha(\mu_2-\mu_1)} \leq (3x^2-3x+1)a+(2x-1)b+c, (3x^2+3x+1)a+(2x+1)b+c$

$$\begin{aligned}
&(1-\alpha)((3x^2+3x+1)a+(2x+1)b+c) \geq (3x^2+3x+1)c_a+(2x+1)c_b+c_c \\
&\qquad +\alpha\lambda((6x+6)a+2b)+\alpha\mu_2(-6xa-2b) \\
\iff\quad &(1-\alpha)((3x^2+3x+1)a+(2x+1)b+c) \geq (3x^2+3x+1)c_a+(2x+1)(c_b+3\alpha(\lambda-\mu_2)a) \\
&\qquad +c_c+3\alpha(\lambda+\mu_2)a+2\alpha(\lambda-\mu_2)b.
\end{aligned}$$

The last inequality holds for $a \geq \frac{c_a}{1-\alpha}$, $b \geq \frac{c_b+3\alpha(\lambda-\mu_2)a}{1-\alpha}$ and $c \geq \frac{c_c+3\alpha(\lambda+\mu_2)a+2\alpha(\lambda-\mu_2)b}{1-\alpha}$ for all states $x \geq 1$.

- **Case 3:** $(3x^2-3x+1)a+(2x-1)b+c \geq \frac{K}{\alpha(\mu_2-\mu_1)} < (3x^2+3x+1)a+(2x+1)b+c$

$$\begin{aligned}
(1-\alpha)((3x^2+3x+1)a+(2x+1)b+c) \geq\ & (3x^2+3x+1)c_a+(2x+1)c_b+c_c+\alpha\lambda((6x+6)a+2b)+\alpha\mu_2(-6xa-2b) \\
& +\alpha(\mu_2-\mu_1)\left[\frac{K}{\alpha(\mu_2-\mu_1)}-((3x^2-3x+1)a+(2x-1)b+c)\right].
\end{aligned}$$

This case reduces to **Case 1**, using $\frac{K}{\alpha(\mu_2-\mu_1)} < (3x^2+3x+1)a+(2x+1)b+c$. As such, the inequality holds for $a \geq \frac{c_a}{1-\alpha}$, $b \geq \frac{c_b+3\alpha(\lambda-\mu_1)a}{1-\alpha}$ and $c \geq \frac{c_c+3\alpha(\lambda+\mu_1)a+2\alpha(\lambda-\mu_1)b}{1-\alpha}$ for all states $x \geq 1$.

As $\mu_2 > \mu_1 > \lambda$ we find that all three cases hold for $a \geq \frac{c_a}{1-\alpha}$, $b \geq \frac{c_b+3\alpha(\lambda-\mu_1)a}{1-\alpha}$ for all $x \geq 1$. For the different expressions of c we cannot determine one or the other to be greater or equal for all values of $\mu_2 > \mu_1$, therefore we determine that $c \geq \max\left\{\frac{c_c+3\alpha(\lambda+\mu_1)a+2\alpha(\lambda-\mu_1)b}{1-\alpha}, \frac{c_c+3\alpha(\lambda+\mu_2)a+2\alpha(\lambda-\mu_2)b}{1-\alpha}\right\}$ and d arbitrary for all states $x \geq 1$.

Consider $x=0$ and let $a = \frac{c_a}{1-\alpha}$ and $b = \frac{c_b+3\alpha(\lambda-\mu_1)a}{1-\alpha}$:

$$\begin{aligned}
&v_0^{l,\alpha}(1)-v_0^{l,\alpha}(0) \geq v_1^{l,\alpha}(1)-v_1^{l,\alpha}(0) \\
\iff\quad & a+b+c \geq c_a+c_b+c_c+\alpha\lambda(7a+3b+c)+\alpha\mu_1(a+b+c) \\
&\qquad +\alpha(\mu_2-\mu_1)\min\left\{a+b+c, \frac{K}{\alpha(\mu_2-\mu_1)}\right\} \\
\iff\quad & (1-\alpha)(a+b+c) \geq c_a+c_b+c_c+\alpha(6a+2b)-\mu_2(a+b+c) \\
&\qquad +\alpha(\mu_2-\mu_1)\min\left\{a+b+c, \frac{K}{\alpha(\mu_2-\mu_1)}\right\}.
\end{aligned}$$

- **Case 1:** $a+b+c \leq \frac{K}{\alpha(\mu_2-\mu_1)}$

$$\begin{aligned}
& (1-\alpha)(a+b+c) \geq c_a + c_b + c_c + \alpha\lambda(6a+2b) - \alpha\mu_1(a+b+c) \\
\iff & (1-\alpha)(b+c) \geq c_b + c_c + \alpha(6\lambda-\mu_1)a + \alpha(2\lambda-\mu_1)b - \alpha\mu_1 c \\
\iff & 3\alpha(\lambda-\mu_1)a + (1-\alpha)c \geq c_c + \alpha(6\lambda-\mu_1)a + \alpha(2\lambda-\mu_1)b - \alpha\mu_1 c \\
\iff & (1-\alpha+\alpha\mu_1)c \geq c_c + \alpha(3\lambda+2\mu_1)a + \alpha(2\lambda-\mu_1)b
\end{aligned}$$

The last inequality holds for $c \geq \frac{c_c+\alpha(3\lambda+2\mu_1)a+\alpha(2\lambda-\mu_1)b}{1-\alpha+\alpha\mu_1}$ for state 0.

- **Case 2:** $\frac{K}{\alpha(\mu_2-\mu_1)} < a+b+c$

$$(1-\alpha)(a+b+c) \geq c_a + c_b + c_c + \alpha\lambda(6a+2b) - \alpha\mu_2(a+b+c) + \alpha(\mu_2-\mu_1)\frac{K}{\alpha(\mu_2-\mu_1)}.$$

This case reduces to **Case 1**, using $\frac{K}{\alpha(\mu_2-\mu_1)} < a+b+c$. As such, the inequality holds for $c \geq \frac{c_c+\alpha(3\lambda+2\mu_1)a+\alpha(2\lambda-\mu_1)b}{1-\alpha+\alpha\mu_1}$ for state 0.

Now, we have found three expressions for c, however, the maximum of the three is dependent on the parameter choices. Therefore, let

$$\begin{aligned}
v_0^{l,\alpha}(x) = & \frac{c_a}{1-\alpha}x^3 + \frac{c_b + 3\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}x^2 \\
& + \max\left\{ \frac{c_c + 3\alpha(\lambda+\mu_1)\frac{c_a}{1-\alpha} + 2\alpha(\lambda-\mu_1)\frac{c_b+3\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha}, \right. \\
& \quad \frac{c_c + 3\alpha(\lambda+\mu_2)\frac{c_a}{1-\alpha} + 2\alpha(\lambda-\mu_2)\frac{c_b+3\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha}, \\
& \quad \left. \frac{c_c + \alpha(3\lambda+2\mu_1)\frac{c_a}{1-\alpha} + \alpha(2\lambda-\mu_1)\frac{c_b+3\alpha(\lambda-\mu_1)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha+\alpha\mu_1} \right\} x.
\end{aligned}$$

In a like manner, let

$$\begin{aligned}
v_0^{u,\alpha}(x) = & \frac{c_a}{1-\alpha}x^3 + \frac{c_b + 3\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}x^2 \\
& + \min\left\{ \frac{c_c + 3\alpha(\lambda+\mu_1)\frac{c_a}{1-\alpha} + 2\alpha(\lambda-\mu_1)\frac{c_b+3\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha}, \right. \\
& \quad \frac{c_c + 3\alpha(\lambda+\mu_2)\frac{c_a}{1-\alpha} + 2\alpha(\lambda-\mu_2)\frac{c_b+3\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha}, \\
& \quad \left. \frac{c_c + \alpha(3\lambda+2\mu_2)\frac{c_a}{1-\alpha} + \alpha(2\lambda-\mu_2)\frac{c_b+3\alpha(\lambda-\mu_2)\frac{c_a}{1-\alpha}}{1-\alpha}}{1-\alpha+\alpha\mu_2} \right\} x.
\end{aligned}$$

□

E MATLAB code

```
% Input paramaters
lambda = ...; %arrival speed
mu1 = ...; %serving speed slow
mu2 = ...; %serving speed fast
K = ...; %extra cost for faster service
alpha = ...; %discounting factor
ca = ...; %ca*x^3
cb = ...; %cb*x^2
cc = ...; %cc*x
endi = ...; % #iterations
endj = ... + endi + 1; %customers in system + compensation for lambda*v_n(x+1)

%storage arrays for results
A = zeros(endi,endj+1); B = zeros(endi,endj+1); Deltal = zeros(endi,1); Deltau = zeros(endi,1);

% calls the value iteration algorithm and the computation of the threshold
% for swapping server speeds
for i = 0:(endi-1)
    for j = 0:(endj-1)
        A(i+1,j+1) = v(i,j,lambda,mu1,mu2,K,A,alpha,ca,cb,cc,1);
        if (Deltal(i+1,1) == 0)
            Deltal(i+1,1) = lu(i,j,mu1,mu2,K,A,alpha);
        end
        B(i+1,j+1) = v(i,j,lambda,mu1,mu2,K,B,alpha,ca,cb,cc,2);
        if (Deltau(i+1,1) == 0)
            Deltau(i+1,1) = lu(i,j,mu1,mu2,K,B,alpha);
        end
    end
end

% takes the infinity value into account for the threshold of swapping
% server speeds
for i = 1:endi
    if (Deltal(i,1) == 0 && Deltau(i,1) == 0)
        Deltal(i,1) = nan;
        Deltau(i,1) = nan;
    elseif (Deltal(i,1) == 0)
        Deltal(i,1) = nan;
    elseif (Deltau(i,1) == 0)
        Deltau(i,1) = nan;
    else
        break;
    end
end
```

```
% computation of threshold (delta)
function luout = lu(n,x,mu1,mu2,K,A,alpha)
    luout = 0;
    if (n < 1 || x < 1)
        luout = 0;
    else
        if (A(n,x+1) - A(n,x) > (K/(alpha*(mu2-mu1))))
            luout = x;
        end
    end
end

% c*x for the initial value function
function c1out = c1(ca,cb,cc,alpha,lambda,mub,mu)
    c1out = (cc + alpha*(3*lambda + 2*mu)*(ca/(1 - alpha))
            + alpha*(2*lambda - mu)*((cb + 3*alpha*(lambda - mub)*(ca/(1 - alpha)))/(1 - alpha)))
                    /(1 - alpha + alpha*mu);
end

% c*x for the initial value function
function c2out = c2(ca,cb,cc,alpha,lambda,mub,mu)
    c2out = (cc + 3*alpha*(lambda + mu)*(ca/(1 - alpha))
            + 2*alpha*(lambda - mu)*((cb + 3*alpha*(lambda - mub)*(ca/(1 - alpha)))/(1 - alpha)))
                      /(1 - alpha);
end

% initial value function
function vnulout = vnul(x,lambda,mu1,mu2,alpha,ca,cb,cc,lu)
    if lu == 1
        vnulout = (ca/(1 - alpha))*x^3
                  + ((cb + 3*alpha*(lambda - mu1)*(ca/(1 - alpha)))/(1 - alpha))*x^2
                  + max([ c1(ca,cb,cc,alpha,lambda,mu1,mu1) ,
                          c1(ca,cb,cc,alpha,lambda,mu1,mu2) ,
                          c2(ca,cb,cc,alpha,lambda,mu1,mu1) ])*x;
    else
        vnulout = (ca/(1 - alpha))*x^3
                  + ((cb + 3*alpha*(lambda - mu2)*(ca/(1 - alpha)))/(1 - alpha))*x^2
                  + min([ c1(ca,cb,cc,alpha,lambda,mu2,mu1) ,
                          c1(ca,cb,cc,alpha,lambda,mu2,mu2) ,
                          c2(ca,cb,cc,alpha,lambda,mu2,mu2) ])*x;
    end
end
```

```
% holding costs
function cout = c(x,ca,cb,cc)
    cout = ca*x^3 + cb*x^2 + cc*x;
end

% value iteration
function vout = v(n,x,lambda,mu1,mu2,K,A,alpha,ca,cb,cc,lu)
    if n < 1
        vout = vnul(x,lambda,mu1,mu2,alpha,ca,cb,cc,lu);
    else
        if x < 1
            vout = c(x,ca,cb,cc) + alpha*(lambda*A(n,x+2) + mu1*A(n,x+1) + mu2*A(n,x+1));
        else
            vout = c(x,ca,cb,cc) + alpha*(lambda*A(n,x+2) + mu1*A(n,x+1) + mu2*A(n,x)
                  + (mu2-mu1)*min(A(n,x+1)-A(n,x), K/(alpha*(mu2-mu1))));
        end
    end
end
```