

Optimal resource allocation for multiqueue systems with a shared server pool

Ran Yang · Sandjai Bhulai · Rob van der Mei

Received: 8 July 2010 / Revised: 1 March 2011 / Published online: 30 March 2011
© The Author(s) 2011. This article is published with open access at Springerlink.com

Abstract We study optimal allocation of servers for a system with multiple service facilities and with a shared pool of servers. Each service facility poses a constraint on the maximum expected sojourn time of a job. A central decision maker can dynamically allocate servers to each facility, where adding more servers results in faster processing speeds but against higher utilization costs. The objective is to dynamically allocate the servers over the different facilities such that the sojourn-time constraints are met at minimal costs. This situation occurs frequently in practice, for example, in Grid systems for real-time image processing (iris scans, fingerprints). We model this problem as a Markov decision process and derive structural properties of the relative value function. These properties, which are hard to derive for multidimensional systems, give a full characterization of the optimal policy. We demonstrate the effectiveness of these policies by extensive numerical experiments.

Keywords Constrained Markov decision problems · Monotonicity · Optimal resource allocation · Shared server pool · Queueing theory

Mathematics Subject Classification (2000) Primary: 90C40 · Secondary: 90B22

1 Introduction

In recent years new real-time multimedia services have triggered a tremendous growth in data volumes and computational demand. Typical services include iris-scan and fingerprint systems that make high-resolution scans and require processing

R. Yang · S. Bhulai (✉) · R. van der Mei
VU University, Amsterdam, The Netherlands
e-mail: sbhulai@few.vu.nl

R. Yang · S. Bhulai · R. van der Mei
CWI, Amsterdam, The Netherlands

of the data to identify a person; these services operate in a real-time environment and run under very strict time constraints. To adhere to such constraints, these large-scale services typically use centralized computing clusters to execute on. In these service-based scenarios, a central decision maker then allocates a number of processing resources to different service facilities to process the data. This gives rise to a class of models in which the central decision maker has to allocate the number of resources to ensure that all Quality of Service (QoS) constraints of the different facilities can be met.

In this decision making problem there is a trade-off between the processing time on the one hand and the utilization costs (lease costs, operating costs, etc.) on the other hand. Having too many resources at the server side leads to high costs and also to inefficiency, since only a part of the resources are needed to ensure that the QoS-constraint is satisfied. However, having too few resources leads to a long processing time so that the QoS-constraint of jobs can be violated. Hence, the objective is to find the allocation of the number of resources for the different facilities such that all QoS-constraints are met against minimal costs.

A few papers have been devoted to resource allocation problems closely related to our setting. Perry and Nilson [8] have studied a system in which two types of jobs are served by a single pool of resources. They associate priorities, based on an aging factor that grows proportionally with the waiting time, to these jobs and give an analytical model for computing the expected waiting. This heuristic was first analyzed by Kleinrock [5, 6]. Borst and Seri [3] apply more complex heuristics in a multiskilled queueing system with as performance metric the tail probabilities of the waiting time. They compare the number of jobs in each facility that actually has been served to the number that, nominally, should have been served under a long-run average allocation scheme. The “further behind” the actual number of services, the higher the resulting priority. Bhulai and Koole [2] and Gans and Zhou [4] study a variant with fully cross-trained servers in which only one queue has a QoS-constraint. They use Markov decision processes and Linear Programming to obtain (nearly) optimal control strategies. Stanford and Grassmann [11] simplify the problem by using fixed, static priority policies using matrix-geometric methods. Shumsky [10] divides the state space into regions and uses an approximate analysis for the conditional system performance within each region.

In this paper we study and compare the optimal server allocation for the following three related models: (1) each service facility is viewed in isolation having their dedicated servers, (2) a system in which a chosen allocation cannot be changed during a service of a job, and (3) a fully flexible system in which changing the allocation during the service of a job is allowed. The main difference between the existing literature and our work is that we show by studying monotonicity properties of the dynamic programming relative value function that the optimal strategy has appealing structural properties; it is the multidimensional analog of a nondecreasing step function. This structure enables one to find optimal policies relatively easily as compared to solving the dynamic program which suffers from computational tractability. These methods will be illustrated in extensive numerical experiments.

The contributions of the paper are twofold. First, on the methodological side, we provide a full characterization of the optimal policy of a high-dimensional system,

which is numerically intractable. This is quite exceptional, since there is no standard approach to derive monotonicity properties of the relative value function for multidimensional systems other than componentwise and directional monotonicity (see [12, 13] for examples and [7] for an overview of monotonicity results using event-based DP). This is the reason why the literature overview mainly deals with models having only two service facilities [2, 4, 8], or use heuristics [3, 5, 6, 10, 11] (sometimes even without constraints). Secondly, on the application side, we have readily available policies that are easy to implement in systems that are highly relevant in practice. The comparison of the different models provides fundamental insights into the operational flexibility that is needed in the design of these systems.

The paper is organized as follows. In Sect. 2 we formulate the models for the different cases. Next, we derive the structure of the optimal policy in Sect. 3. In Sect. 4 we illustrate these results by numerical experiments. Finally, in Sect. 5 we conclude the paper and discuss topics for further research.

2 Model formulation

Consider N parallel service facilities at which jobs arrive according to a Poisson process with rate λ_i for facility i , $i = 1, \dots, N$. There is a common pool of $A \geq 1$ resources to serve the jobs in the system. When upon arrival of a job at facility i there are no other jobs present, the arriving job is taken into service. However, if there are other jobs present, then the arriving job joins an infinite-capacity queue at facility i and awaits its service in an FCFS manner. When facility i has been allocated a_i resources, the job that is in service has a service duration that is exponentially distributed with parameter $\mu(a_i)$, where $\mu(\cdot)$ is an increasing function. In the ideal case, one would have $\mu(a_i) = \mu a_i$ for some fixed service rate μ . However, in practice, there is communication overhead between multiple resources, and therefore the function μ is typically sublinear. In some cases, resources can cache results so that its effect is that the function μ is superlinear. After a job has received its service, it leaves the system.

Each facility provides a QoS-guarantee on the mean delay to the jobs served at that facility. Although, in practice, the QoS-constraints are usually expressed in terms of tail probabilities, we choose to express the constraints in terms of the mean sojourn time. This choice keeps the already complex model tractable for analysis and serves as a first step toward the analysis with tail probabilities as QoS-constraints. For this purpose, let S_i denote the steady-state sojourn time of an arbitrary job at facility i . Then facility i is constrained by $\mathbb{E}S_i \leq \alpha_i$ for a preset value of α_i . There is a central decision maker that needs to allocate the resources to the different facilities such that the QoS-constraints are met. This gives rise to a problem in which the optimal allocation strategy needs to be determined. However, when facility i uses a_i resources, a cost of $c_i(a_i)$ is incurred by the system with c_i an increasing function in a_i . Therefore, we are simultaneously interested in meeting the N constraints against lowest average costs. The optimal solution provides the value of A for which the optimal allocation strategy meets all the constraints but fails to meet them when the optimal allocation strategy under $A - 1$ resources is used.

We study the optimal number of resources A^* from three different viewpoints. First, we consider the case in which all service facilities operate independently of each other. In this case, the resources are not shared between the different facilities but are dedicated to each facility. Secondly, we study the case in which the resource pool is shared between different facilities. However, we make the assumption that the resource allocation cannot be changed when a job is served; only upon the start of the service of the next job the resource allocation can be changed. This is typically the case in systems where resources need to be reserved in advance. The third case deals with the fully flexible case in which the system can take full advantage of the economies of scale by allowing the resource allocation to change even during the service of a job. Since we can directly observe that going from case 1 to case 3 increases the flexibility, we can expect that $A_1^* \geq A_2^* \geq A_3^*$, with A_i^* the optimal number of resources needed in case i for $i = 1, 2, 3$. However, it is of interest to determine how big the gap between the three cases is and to study how the policy changes from case to case.

2.1 Service facilities with dedicated resources

In this subsection, we assume that the service facilities do not share the resources between each other and thus have their own dedicated resources. This makes service facility i independent of the other facilities and turns the facility into a regular $M/M/1$ queue with arrival rate λ_i and service rate $\mu(A_i)$ when A_i servers have been allocated. In that case, it is well known that the expected sojourn time is given by $1/(\mu(A_i) - \lambda_i)$. Hence,

$$A_1^* = \sum_{i=1}^N A_i^* = \sum_{i=1}^N \lceil \mu^{-1}(\lambda_i + 1/\alpha_i) \rceil,$$

where $\lceil x \rceil$ denotes the smallest integer greater than or equal to x .

2.2 Service facilities with limited resource sharing

In this subsection we focus on the case in which service facilities are allowed to share resources between each other. However, resources become free to be reassigned only at service completion instants. Hence, we make the assumption that the resource allocation for a service facility can only be changed upon the start of the service of a new job. Hence adding/removing resources during a service is not allowed. To study this case, we cast the problem as a Markov decision problem.

Define the state space $\mathcal{X} = \{(x_1, \dots, x_N, a_1, \dots, a_N) \in \mathbb{N}_0^N \times \mathbb{N}_0^N \mid \sum_{i=1}^N a_i \leq A\}$, where $(x, a) \in \mathcal{X}$ means that there are x_i customers at facility i with a_i resources allocated to it for $i = 1, \dots, N$, where $a_i > 0$ also means that a service is ongoing, and $a_i = 0$ means that no job is in service at facility i . When the system is in state $(x, a) \in \mathcal{X}$, the decision maker can choose actions from the action space $\mathcal{A}_{(x,a)} = \{(b_1, \dots, b_n) \in \mathbb{N}_0^N \mid \sum_{i=1}^N (a_i + b_i) \leq A \text{ and } a_i b_i = 0 \text{ for } i = 1, \dots, N\}$, where action $b \in \mathcal{A}_{(x,a)}$ denotes the number of resources that one can allocate. Here, the restriction $a_i b_i = 0$ models the fact that when a service is ongoing (i.e., $a_i > 0$),

the service allocation cannot be changed (i.e., $b_i = 0$). However, after a service completion at facility i , we have that $a_i = 0$ and hence an allocation $b_i > 0$ is allowed. The transition rates when the system is in state $(x, a) \in \mathcal{X}$ and action $b \in \mathcal{A}_{(x,a)}$ is chosen are given by

$$p((x, a), b, (x', b')) = \begin{cases} \lambda_i, & x' = x + e_i, b' = a + b \text{ for } i = 1, \dots, N, \\ \mu(a_i + b_i), & x' = [x - e_i]^+, b' = a + b - a_i e_i - b_i e_i \text{ for } i = 1, \dots, N, \\ 0 & \text{otherwise,} \end{cases}$$

with e_i the zero vector with a one at the i th entry, and $[x]^+$ the componentwise maximum ($\max\{x_1, 0\}, \dots, \max\{x_N, 0\}$). The first line in the expression above models arrivals, the second line models service completions, and the third line prohibits any other state transitions. Note that when a service completion takes place, the resource allocation for that facility is set to zero. Finally, when the system is in state $(x, a) \in \mathcal{X}$ and action $b \in \mathcal{A}_{(x,a)}$ has been chosen, the direct costs $c((x, a), b) = \sum_{i=1}^N c_i(a_i + b_i)$. The quadruple $(\mathcal{X}, \mathcal{A}, p, c)$ completely describes the Markov decision process.

Define a decision rule $\pi_{(x,a)}$ as a probability distribution on $\mathcal{A}_{(x,a)}$, i.e., when the system is in state $(x, a) \in \mathcal{X}$, the decision maker chooses action $b \in \mathcal{A}_{(x,a)}$ with probability $\pi_{(x,a)}(b)$. Let the policy π denote the collection of decision rules for all states. Let $u_t^\pi(x, a)$ denote the total expected costs up to time t when the system starts in state (x, a) under policy π . Note that for any stable and work-conserving policy, the Markov chain satisfies the unichain condition, so that the average expected costs $g(\pi) = \lim_{t \rightarrow \infty} u_t^\pi(x, a)/t$ are independent of the initial state (x, a) (see Proposition 8.2.1 of Puterman [9]). The goal is to find a policy π^* that minimizes the long-term average costs under the constraints, that is,

$$\min_{\pi} g(\pi) \quad \text{subject to} \quad \mathbb{E}S_i \leq \alpha_i \quad \text{for } i = 1, \dots, N.$$

Note that due to Little’s Law the number of jobs L_i in facility i can be related to the sojourn time S_i in facility i by $\mathbb{E}L_i = \lambda_i \mathbb{E}S_i$. Using this knowledge, the constrained Markov decision problem can be rewritten as an unconstrained Markov decision problem using Lagrange multipliers (see Sect. 12.6 of Altman [1]). To this end, we uniformize the system (see Sect. 11.5 of Puterman [9]). Therefore, assume that the uniformization constant $\sum_{i=1}^N \lambda_i + N\mu(A) = 1$; we can always get this by scaling. Uniformizing is equivalent to adding dummy transitions (from a state to itself) such that the rate out of each state is equal to 1; then we can consider the rates to be transition probabilities. Now, let $V(x, a)$ be a real-valued function defined on the state space. This function will play the role of the relative value function, i.e., the asymptotic difference in total costs that results from starting the process in state (x, a) instead of some reference state. The long-term average optimal actions are a solution of the optimality equation (in vector notation) $g + V = TV$, where T is the dynamic programming operator acting on V defined as follows:

$$\begin{aligned}
TV(x, a) &= \sum_{i=1}^N \tau_i \frac{x_i}{\lambda_i} + \sum_{i=1}^N c_i(a_i) + \sum_{i=1}^N \lambda_i H(x + e_i, a) \\
&\quad + \sum_{i=1}^N \mu(a_i) H([x - e_i]^+, a - a_i e_i) \\
&\quad + \left(1 - \sum_{i=1}^N \lambda_i - \sum_{i=1}^N \mu(a_i) \right) V(x, a) \\
&= \sum_{i=1}^N \tau_i \frac{x_i}{\lambda_i} + \sum_{i=1}^N c_i(a_i) + \sum_{i=1}^N \lambda_i H(x + e_i, a) \\
&\quad + \sum_{i=1}^N \mu(a_i) H([x - e_i]^+, a - a_i e_i) \\
&\quad + \left(N\mu(A) - \sum_{i=1}^N \mu(a_i) \right) V(x, a), \tag{1}
\end{aligned}$$

where τ_i are Lagrange multipliers, and the function H is given by

$$H(x, a) = \min_{b \in \mathcal{A}(x, a)} \{V(x, a + b)\}.$$

The first term in the dynamic programming operator corresponds to the QoS-constraints of the several facilities. The second term represents the cost of using a resources. The third term is involved with the decision making upon arrival of a job. The fourth term deals with the decision making when a job has finished its service. The final term is the dummy term due to uniformization. Note that the decision making is modeled uniformly through the function H .

Note that when facility i has no holding costs x_i/λ_i , then no resources will be allocated to facility i , since it does not incur any costs from the buildup in jobs. Therefore, when $\tau = e_i$, i.e., $\tau_i = 1$ and $\tau_j = 0$ for $j \neq i$, the optimal strategy will not allocate any resources to service facility $j \neq i$. Hence, one can find a value z_i such that the QoS-constraint for facility i with $\tau = z_i e_i$ is met under the assumption that there are infinitely many resources. By repeating this procedure for all facilities, one finds a box $\prod_{i=1}^n [0, z_i]$ in which the value of τ should lie under the optimal allocation that satisfies all constraints. Now, we can divide this box into a grid \mathcal{G} which serves as our search space for τ . Then the following approach will find A_2^* .

1. Set $A := A_1^*$.
2. Solve the Markov decision process for all values of $\tau \in \mathcal{G}$ until all QoS-constraints are met or all grid points have been searched.
3. If for the value of τ , all constraints are met, set $A := A - 1$ and return to step 2.
4. Return $A_2^* := A + 1$.

Note that in step 2 of the algorithm one needs to solve an infinite-dimensional Markov decision problem. In our numerical experiments we truncate the state space such that we get a finite-dimensional problem that is numerically tractable. In doing so, the truncation is done such that the difference in the outcomes do not differ significantly when the state space is somewhat enlarged by shifting the truncation boundary. We will illustrate this algorithm in Sect. 4.

The algorithm to find A_2^* relies on evaluating the Markov decision problem for all $\tau \in \mathcal{G}$. One might formulate an unconstrained Markov decision problem, in which costs are associated with the queue length, in order to circumvent these evaluations. However, this would lead to formulation (1) with $\tau_i = \lambda_i$ for all i . Since the alternative unconstrained model is a special case of (1), the structural results that are obtained in the next section for the constrained Markov decision problem also hold for the unconstrained problem.

2.3 Service facilities with full flexibility in resource sharing

In this subsection we study the case in which service facilities have full flexibility in the resource allocation policies. Thus, the resource facilities can change the resource allocation during a service of a job and do not have to wait for the job to finish. Since our system has Poisson arrivals and exponential service times, such a situation need only occur at moments an event occurs. Therefore, the only difference with the previous case is that this system allows one to change the allocation at arrival instants.

In this case, the state space is given by $\mathcal{X} = \mathbb{N}_0^N$, where $x \in \mathcal{X}$ denotes that there are x_i customers at facility i for $i = 1, \dots, N$. The action space is given by $\mathcal{A}_x = \{a \in \mathbb{N}_0^N \mid \sum_{i=1}^N a_i \leq A\}$, where action $a \in \mathcal{A}_x$ denotes the number of resources that one can allocate in state $x \in \mathcal{X}$. The transition rates when the system is in state $x \in \mathcal{X}$ and action $a \in \mathcal{A}_x$ is chosen are given by

$$p(x, a, x') = \begin{cases} \lambda_i, & x' = x + e_i \text{ for } i = 1, \dots, N, \\ \mu(a_i), & x' = [x - e_i]^+ \text{ for } i = 1, \dots, N, \\ 0 & \text{otherwise.} \end{cases}$$

Finally, when the system is in state $x \in \mathcal{X}$ and action $a \in \mathcal{A}_x$ has been chosen, the direct costs are $c(x, a) = \sum_{i=1}^N c_i(a_i)$. The tuple $(\mathcal{X}, \mathcal{A}, p, c)$ completely describes the Markov decision process for this problem.

Let $V(x)$ denote the relative value function in this case. Then, the dynamic programming operator acting on V is defined as follows:

$$TV(x) = \sum_{i=1}^N \tau_i \frac{x_i}{\lambda_i} + \sum_{i=1}^N \lambda_i V(x + e_i) + \min_{a \in \mathcal{A}_x} \left[\sum_{i=1}^N \mu(a_i) V([x - e_i]^+) + \left(1 - \sum_{i=1}^N \lambda_i - \sum_{i=1}^N \mu(a_i) \right) V(x) + \sum_{i=1}^N c_i(a_i) \right]$$

$$\begin{aligned}
 &= \sum_{i=1}^N \tau_i \frac{x_i}{\lambda_i} + \sum_{i=1}^N \lambda_i V(x + e_i) + \min_{a \in \mathcal{A}_x} \left[\sum_{i=1}^N \mu(a_i) V([x - e_i]^+) \right. \\
 &\quad \left. + \left(N\mu(A) - \sum_{i=1}^N \mu(a_i) \right) V(x) + \sum_{i=1}^N c_i(a_i) \right]. \tag{2}
 \end{aligned}$$

Note that the algorithm described in the previous subsection also applies to this case to obtain A_3^* . In fact, in step 1 one can choose $A := A_2^*$ for faster convergence.

3 Structural properties of the optimal policy

In the previous section we described the three models and a solution technique to obtain the optimal policy. However, the optimal policy also possesses structural properties that can speed up the solution technique. Instead of searching a full grid for the optimal solution, the structural properties can reduce the search space considerably. Therefore, in this section, we focus on the structural properties of the described systems.

3.1 Allocation strategy for service facilities with dedicated resources

In the case when the service facilities have their own dedicated servers, each facility i should be equipped with A_i^* servers (as defined in Sect. 2.1) to meet the QoS constraint. Since the servers are not shared, the optimal resource allocation strategy is quite simple. When there are $x > 0$ customers at the facility, A_i^* servers are allocated, and when $x = 0$, then no servers are allocated. This policy is also known as a bang-bang control policy (i.e., everything or nothing).

3.2 Allocation strategy for service facilities with limited resource sharing

The structure of the optimal policy for a service facility with limited resource sharing is more intricate than the case with dedicated resources. In order to study the structure, in principle, one needs to solve the optimality equation $g + V = TV$ with TV given by (1). However, the optimality equation is hard to solve analytically in practice. Alternatively, the optimal actions can also be obtained by recursively defining $V_{l+1} = TV_l$ for arbitrary V_0 . As $l \rightarrow \infty$, the maximizing actions converge to the optimal ones (for the existence and convergence of solutions and optimal policies, we refer to Chap. 8 of Puterman [9]). Therefore, we consider the backward recursion equation that is given by

$$\begin{aligned}
 V_{n+1}(x, a) &= \sum_{i=1}^N \tau_i \frac{x_i}{\lambda_i} + \sum_{i=1}^N c_i(a_i) + \sum_{i=1}^N \lambda_i H_n(x + e_i, a) \\
 &\quad + \sum_{i=1}^N \mu(a_i) H_n([x - e_i]^+, a - a_i e_i) \\
 &\quad + \left(N\mu(A) - \sum_{i=1}^N \mu(a_i) \right) V_n(x, a), \tag{3}
 \end{aligned}$$

where the function H_n is given by

$$H_n(x, a) = \min_{b \in \mathcal{A}(x,a)} \{V_n(x, a + b)\}.$$

For ease of notation in the proofs in the sequel, we also define $\arg H_n(x, a)$ by

$$\arg H_n(x, a) = \arg \min_{b \in \mathcal{A}(x,a)} \{V_n(x, a + b)\}.$$

The backward recursion equation allows us to prove structural properties of the relative value function V through induction on n in V_n . It is clear that the objective of the system is to strive for the fewest number of customers in the system as more customers mean higher waiting costs. Therefore, it is intuitive that V is increasing in each component of x , i.e., adding customers to facility i results in higher costs for the system. The following lemma makes this statement more precise.

Lemma 1 (Increasingness) *The relative value function V is increasing in the number of customers, i.e.,*

$$V(x + e_j, a) - V(x, a) \geq 0$$

for all $x \in \mathcal{X}$, for some a with $\sum_{i=1}^N a_i \leq A$, and for $j = 1, \dots, N$.

Proof The proof is by induction on n in V_n . Define $V_0(x, a) = 0$ for all states x and actions a . Then, clearly, $V_0(x, a)$ is increasing in all components of x . Now, assume that $V_n(x + e_j, a) - V_n(x, a) \geq 0$ for some $n \in \mathbb{N}$ and for $j = 1, \dots, N$. Now, we prove that $V_{n+1}(x, a)$ satisfies the increasingness property as well. Therefore, fix $j \in \{1, \dots, N\}$; then

$$\begin{aligned} &V_{n+1}(x + e_j, a) - V_{n+1}(x, a) \\ &= \frac{\tau_j}{\lambda_j} + \sum_{i=1}^N \lambda_i [H_n(x + e_j + e_i, a) - H_n(x + e_i, a)] \\ &\quad + \sum_{i=1}^N \mu(a_i) [H_n([x + e_j - e_i]^+, a - a_i e_i) - H_n([x - e_i]^+, a - a_i e_i)] \\ &\quad + \left[N\mu(A) - \sum_{i=1}^N \mu(a_i) \right] [V_n(x + e_j, a) - V_n(x, a)]. \end{aligned} \tag{4}$$

Note that the first term (τ_j/λ_j) and the last term with $V_n(x + e_j, a) - V_n(x, a)$ are positive. Hence, based on the induction hypothesis, we have

$$\begin{aligned} &V_{n+1}(x + e_j, a) - V_{n+1}(x, a) \\ &\geq \sum_{i=1}^N \lambda_i [H_n(x + e_j + e_i, a) - H_n(x + e_i, a)] \\ &\quad + \sum_{i=1}^N \mu(a_i) [H_n([x + e_j - e_i]^+, a - a_i e_i) - H_n([x - e_i]^+, a - a_i e_i)]. \end{aligned} \tag{5}$$

Let $b = \arg H_n(x + e_j + e_i, a)$ and $c = \arg H_n([x + e_j - e_i]^+, a - a_i e_i)$. Then,

$$\begin{aligned}
 &V_{n+1}(x + e_j, a) - V_{n+1}(x, a) \\
 &\geq \sum_{i=1}^N \lambda_i [V_n(x + e_j + e_i, a + b) - V_n(x + e_i, a + b)] \\
 &\quad + \sum_{i=1}^N \mu(a_i) [V_n([x + e_j - e_i]^+, a - a_i e_i + c) - V_n([x - e_i]^+, a - a_i e_i + c)] \\
 &\geq 0.
 \end{aligned} \tag{6}$$

Clearly, the inequality above holds because of the induction hypothesis. Hence, we conclude, by taking the limit as $n \rightarrow \infty$, that $V(x, a)$ is increasing in x_j for all $j = 1, \dots, N$. \square

Lemma 1 shows that the costs that the system incurs increase as the number of customers in the system increases. In fact, more can be said about the rate at which the costs increase; the increase in costs is higher when more customers are in the system. Hence, this implies that the relative value function is a convex function. In the sequel we will show that this is indeed true. We do this by first studying the case with one dimension, e.g., $N = 1$. Note that we will adjust the notation for $N = 1$ straightforwardly by omitting the indices of all variables. However, before doing so, we need two preparative lemmas.

Lemma 2 *The value function satisfies the following property:*

$$H(x + 1, 0) - H(x, 0) - V(x, 0) + V(x - 1, 0) < 0$$

for all $x \geq 1$.

Proof Let $x \geq 1$. Then

$$\begin{aligned}
 &V(x, 0) - V(x - 1, 0) \\
 &= \frac{\tau}{\lambda} + \lambda [H(x + 1, 0) - H(x, 0)] + \mu(A) [V(x, 0) - V(x - 1, 0)].
 \end{aligned}$$

Since $\lambda + \mu(A) = 1$, the equation above implies that

$$\lambda [V(x, 0) - V(x - 1, 0)] = \frac{\tau}{\lambda} + \lambda [H(x + 1, 0) - H(x, 0)].$$

Therefore,

$$\lambda [H(x + 1, 0) - H(x, 0) - V(x, 0) + V(x - 1, 0)] = -\frac{\tau}{\lambda}.$$

Thus, $H(x + 1, 0) - H(x, 0) - V(x, 0) + V(x - 1, 0) < 0$, since $-\tau/\lambda < 0$. \square

Lemma 2 is almost the inequality that represents convexity of the value function. This would be the case if H were to be replaced by V . However, for the proof of convexity, we need three additional properties to hold as well. The following lemma makes these properties explicit.

Lemma 3 (Convexity) *For $N = 1$, the following properties hold:*

- (i) $V(x + 1, a) - 2V(x, a) + V(x - 1, a) \geq 0$ for all $x \geq 1$ and $a \geq 0$.
- (ii) $V(x, a) - V([x - 1]^+, a) - H([x - 1]^+, 0) + H([x - 2]^+, 0) \geq 0$ for all $x \geq 0$ and $a > 0$.
- (iii) $\arg H(x, 0) > 0$ for all $x \geq 2$.
- (iv) $H(x + 1, 0) - 2H(x, 0) + H(x - 1, 0) \geq 0$ for all $x \geq 1$.

Proof The proof is by induction on n in V_n . Define $V_0(x, a) = 0$ for all states x and actions a . Then, clearly, $V_0(x, a)$ satisfies all properties (in case (iii), there is an optimal action that satisfies the property). Now suppose that the properties hold for some $n \in \mathbb{N}$. We prove that the properties also hold for $n + 1$. Therefore, we start with convexity first.

Property (i) Let $x \geq 1$ and suppose that $a = 0$. Then,

$$\begin{aligned} &V_{n+1}(x + 1, 0) - 2V_{n+1}(x, 0) + V_{n+1}(x - 1, 0) \\ &= \lambda[H_n(x + 2, 0) - 2H_n(x + 1, 0) + H_n(x, 0)] \\ &\quad + \mu(A)[V_n(x + 1, 0) - 2V_n(x, 0) + V_n(x - 1, 0)] \\ &\geq \lambda[H_n(x + 2, 0) - 2H_n(x + 1, 0) + H_n(x, 0)] \\ &\geq 0. \end{aligned} \tag{7}$$

The equality follows by expanding V_{n+1} into V_n . The first inequality follows by using property (i) of the induction hypothesis. The last inequality follows by using property (iv) of the induction hypothesis.

Now let $x \geq 1$ and suppose that $a > 0$. Then,

$$\begin{aligned} &V_{n+1}(x + 1, a) - 2V_{n+1}(x, a) + V_{n+1}(x - 1, a) \\ &= \lambda[V_n(x + 2, a) - 2V_n(x + 1, a) + V_n(x, a)] \\ &\quad + \mu(a)[H_n(x, 0) - 2H_n([x - 1]^+, 0) + H_n([x - 2]^+, 0)] \\ &\quad + [\mu(A) - \mu(a)][V(x + 1, a) - 2V(x, a) + V(x - 1, a)] \\ &\geq \mu(a)[H_n(x, 0) - 2H_n([x - 1]^+, 0) + H_n([x - 2]^+, 0)] \\ &\geq 0. \end{aligned} \tag{8}$$

The equality follows by expanding V_{n+1} into V_n . The first inequality follows by using property (i) of the induction hypothesis. The last inequality follows by using property (iv) of the induction hypothesis. Thus, for all $x \geq 1$ and $a \geq 0$, $V_{n+1}(x + 1, a) - 2V_{n+1}(x, a) + V_{n+1}(x - 1, a) \geq 0$.

Property (ii) Let $x \geq 0$ and suppose that $a > 0$. Then, based on the optimality equation, we have

$$\begin{aligned} &V_{n+1}(x, a) - V_{n+1}([x - 1]^+, a) \\ &= \frac{\tau}{\lambda} + \lambda[V_{n+1}(x + 1, a) - V_{n+1}(x, a)] \\ &\quad + \mu(a)[H_{n+1}([x - 1]^+, 0) - H_{n+1}([x - 2]^+, 0)] \\ &\quad + [\mu(A) - \mu(a)][V_{n+1}(x, a) - V_{n+1}([x - 1]^+, a)]. \end{aligned}$$

Recall that the uniformization constant $\lambda + \mu(A) = 1$. Thus, the equation above is equivalent to

$$\begin{aligned} &\lambda[V_{n+1}(x, a) - V_{n+1}([x - 1]^+, a)] \\ &= \frac{\tau}{\lambda} + \lambda[V_{n+1}(x + 1, a) - V_{n+1}(x, a)] \\ &\quad + \mu(a)[H_{n+1}([x - 1]^+, 0) - H_{n+1}([x - 2]^+, 0)] \\ &\quad - \mu(a)[V_{n+1}(x, a) - V_{n+1}([x - 1]^+, a)]. \end{aligned}$$

The equation above implies that

$$\begin{aligned} &\mu(a)[V_{n+1}(x, a) - V_{n+1}([x - 1]^+, a) - H_{n+1}([x - 1]^+, 0) + H_{n+1}([x - 2]^+, 0)] \\ &= \frac{\tau}{\lambda} + \lambda[V_{n+1}(x + 1, a) - 2V_{n+1}(x, a) + V_{n+1}([x - 1]^+, a)]. \end{aligned}$$

Hence, by using property (i) of the induction hypothesis, the righthand side of the equation is nonnegative. Hence, $V_{n+1}(x, a) - V_{n+1}([x - 1]^+, a) - H_{n+1}([x - 1]^+, 0) + H_{n+1}([x - 2]^+, 0) \geq 0$.

Property (iii) We prove the property by means of contradiction. Assume that there exists an $x \geq 2$ such that $\arg H(x, 0) = 0$. This, by definition, implies that $V_{n+1}(x, 0) = H_{n+1}(x, 0)$. Therefore,

$$\begin{aligned} &H_{n+1}(x, 0) - H_{n+1}(x - 1, 0) - V_{n+1}(x - 1, 0) + V_{n+1}(x - 2, 0) \\ &\geq V_{n+1}(x, 0) - 2V_{n+1}(x - 1, 0) + V_{n+1}(x - 2, 0) \\ &\geq 0. \end{aligned}$$

The first inequality follows by taking action $a = 0$ in the second term $H_{n+1}(x - 1, 0)$. The second inequality follows by property (i) of the induction hypothesis. However, based on Lemma 2, we know that $H_{n+1}(x, 0) - H_{n+1}(x - 1, 0) - V_{n+1}(x - 1, 0) + V_{n+1}(x - 2, 0) < 0$. Therefore, we conclude that $\arg H(x, 0) > 0$ for $x \geq 2$.

Property (iv) Let $x \geq 1$. Since $x - 1 \geq 0$, we have $x + 1 \geq 2$. Thus, by using property (iii) of the induction hypothesis, we have $a^*(x) := \arg H(x, 0) > 0$. Therefore,

$$\begin{aligned}
 &H_{n+1}(x + 1, 0) - 2H_{n+1}(x, 0) + H_{n+1}(x - 1, 0) \\
 &\geq V_{n+1}(x + 1, a^*(x + 1)) - V_{n+1}(x, a^*(x + 1)) - H_{n+1}(x, 0) + H_{n+1}(x - 1, 0) \\
 &\geq 0.
 \end{aligned}$$

The first inequality follows by taking action $a^*(x + 1)$ in $H_{n+1}(x, 0)$. The second inequality follows by property (ii) of the induction hypothesis.

We conclude the proof by taking the limit as $n \rightarrow \infty$. □

Lemma 3 shows that the relative value function is convex. However, in proving this, one needed three additional properties simultaneously in the proof by induction (property (i) depends on (iv), which depends on (ii) and (iii)). Now we are ready to study monotonicity properties of the optimal policy. The convexity of the relative value function is crucial in this step. Due to the convexity, we have that the optimal policy is a step function. The following theorem formalizes this statement.

Theorem 4 (Monotonicity) *For $N = 1$, let $a^*(x) := \arg H(x, 0)$ for all $x \geq 0$. If the service rate $\mu(a)$ and cost function $c(a)$ are increasing functions in a , then $a^*(x)$ is an increasing function in x .*

Proof For $x = 0$, it is clear that $a^*(0) = 0$, since there are no customers to serve. For $x = 1$, we know that when $a^*(1) = 0$, then $a^*(2) > 0$ [property (iii) of Lemma 3]. Hence, $a^*(2) > a^*(1) \geq a^*(0)$. Now, for $x \geq 1$ and $a > 0$, it suffices to show that the relative value function satisfies an extension of submodularity, namely

$$[V(x, a + k) - V(x, a)] - [V(x + 1, a + k) - V(x + 1, a)] \geq 0 \tag{9}$$

for all $k \geq 0$. If this property holds, then since $V(x + 1, a^*(x + 1) + k) - V(x + 1, a^*(x + 1)) \geq 0$, we have that $V(x, a^*(x + 1) + k) - V(x, a^*(x + 1)) \geq V(x + 1, a^*(x + 1) + k) - V(x + 1, a^*(x + 1)) \geq 0$. Hence, this implies that the minimizing action $a^*(x)$ in state x satisfies $a^*(x) \leq a^*(x + 1)$.

We prove the submodularity property by induction on n in V_n . Let $V_0(x, a) = 0$. Clearly, the submodularity property holds. Now assume that the property holds for some $n \in \mathbb{N}$ and for all $x \geq 0$. We proceed to prove that $V_{n+1}(x, a)$ satisfies the property as well. Therefore, fix $x \geq 1$ and $a > 0$; then

$$\begin{aligned}
 &V_{n+1}(x, a + k) - V_{n+1}(x, a) - V_{n+1}(x + 1, a + k) + V_{n+1}(x + 1, a) \\
 &= \lambda[V_n(x + 1, a + k) - V_n(x + 1, a) - V_n(x + 2, a + k) + V_n(x + 2, a)] \\
 &\quad + [\mu(a + k) - \mu(a)][H_n(x - 1, 0) - H_n(x, 0)] \\
 &\quad + \mu(A)[V_n(x, a + k) - V_n(x, a) - V_n(x + 1, a + k) + V_n(x + 1, a)] \\
 &\quad - \mu(a + k)V_n(x, a + k) + \mu(a)V_n(x, a) \\
 &\quad + \mu(a + k)V_n(x + 1, a + k) - \mu(a)V_n(x + 1, a) \\
 &= \lambda[V_n(x + 1, a + k) - V_n(x + 1, a) - V_n(x + 2, a + k) + V_n(x + 2, a)] \\
 &\quad + [\mu(a + k) - \mu(a)][H_n(x - 1, 0) - H_n(x, 0)]
 \end{aligned}$$

$$\begin{aligned}
 & + \mu(A)[V_n(x, a + k) - V_n(x, a) - V_n(x + 1, a + k) + V_n(x + 1, a)] \\
 & - [\mu(a + k) - \mu(a)]V_n(x, a + k) - \mu(a)V_n(x, a + k) + \mu(a)V_n(x, a) \\
 & + [\mu(a + k) - \mu(a)]V_n(x + 1, a + k) + \mu(a)V_n(x + 1, a + k) \\
 & - \mu(a)V_n(x + 1, a) \\
 = & \lambda[V_n(x + 1, a + k) - V_n(x + 1, a) - V_n(x + 2, a + k) + V_n(x + 2, a)] \\
 & + [\mu(A) - \mu(a)][V_n(x, a + k) - V_n(x, a) - V_n(x + 1, a + k) \\
 & + V_n(x + 1, a)] \\
 & + [\mu(a + k) - \mu(a)][V_n(x + 1, a + k) - V_n(x, a + k) - H_n(x, 0) \\
 & + H_n(x - 1, 0)]. \tag{10}
 \end{aligned}$$

The first equality follows from expanding V_{n+1} into V_n . The second equality follows from adding and subtracting $\mu(a)V_n(x, a + k)$ and $\mu(a)V_n(x + 1, a + k)$. The third equality follows from standard algebraic manipulations. Based on the induction hypothesis, we have

$$\begin{aligned}
 & V_{n+1}(x, a + k) - V_{n+1}(x, a) - V_{n+1}(x + 1, a + k) + V_{n+1}(x + 1, a) \\
 & \geq [\mu(a + k) - \mu(a)] \\
 & \quad \times [V_n(x + 1, a + k) - V_n(x, a + k) - H_n(x, 0) + H_n(x - 1, 0)]. \tag{11}
 \end{aligned}$$

By using property (ii) of Lemma 3, we obtain

$$V_n(x + 1, a + k) - V_n(x, a + k) - H_n(x, 0) + H_n(x - 1, 0) \geq 0. \tag{12}$$

Thus, we have shown that $[V(x, a + k) - V(x, a)] - [V(x + 1, a + k) - V(x + 1, a)] \geq 0$. Consequently, this completes the proof as this property implies that $a^*(x)$ is an increasing function in x . \square

Theorem 4 shows that in the one-dimensional case, i.e., $N = 1$, the optimal policy $a^*(x)$ is a step function in the variable x . The multidimensional case with arbitrary N has the same structure. The proof of this statement fundamentally boils down to the one-dimensional case. The following theorem shows the argument.

Theorem 5 (Step function) *Consider an arbitrary number $N \in \mathbb{N}$ of queues, and let $a_i^*(x, a) := [\arg H(x, a)]_i = [\arg \min_{b \in \mathcal{A}(x,a)} \{V_n(x, a + b)\}]_i$. If the service rate $\mu(a)$ and cost function $c(a)$ are increasing functions in a , then $a_i^*(x, a)$ is an increasing function in x_i .*

Proof Fix $(x, a) \in \mathcal{X}$ and suppose that $(x + e_i, a) \in \mathcal{X}$ as well. If $a^*(x, a)$ is such that $\sum_{n=1}^N (a_n + a_n^*(x, a)) < A$, then there is spare capacity to assign. Hence, in state $(x + e_i, a)$, all queues except queue i need no more capacity, since the number of customers in their system did not increase. Hence, queue i can be viewed in isolation due to the spare capacity. Therefore, Theorem 4 applies, and $a_i^*(x, a) \leq a_i^*(x + e_i, a)$.

In the case where $\sum_{n=1}^N (a_n + a_n^*(x, a)) = A$, there is no spare capacity left. Now, there are two cases. Either the performance of queue i becomes so stringent that capacity is taken away from a different queue, or the capacity allocation does not change at all. In both cases, we have $a_i^*(x, a) \leq a_i^*(x + e_i, a)$. \square

3.3 Allocation strategy for service facilities with full flexibility in resource sharing

In this subsection we shall adopt the same techniques in deriving the structure of the optimal policy as in the previous section, but now for the case of service facilities with full flexibility in resource sharing. Therefore, we focus on the main theorems that characterize the structure of the policy, while we move the lengthiest proof to [Appendix](#).

We start by rewriting (2) for the fully flexible system as a set of backward recursion equations. This set of equations is given by

$$V_{n+1}(x) = \sum_{i=1}^N \tau_i \frac{x_i}{\lambda_i} + \sum_{i=1}^N \lambda_i V_n(x + e_i) + \min_{a \in \mathcal{A}_x} T_a^n(x), \tag{13}$$

where $T_a^n(x)$ is given by

$$T_a^n(x) = \sum_{i=1}^N \mu(a_i) V_n([x - e_i]^+) + \left[N\mu(A) - \sum_{i=1}^N \mu(a_i) \right] V_n(x) + \sum_{i=1}^N c_i(a_i).$$

Rewriting the optimality equations in this way has its advantage in showing structural properties of the relative value function V . We start by showing that the relative value function is increasing in all components of the state. The following lemma makes this statement more precise.

Lemma 6 (Increasingness) *The relative value function $V(x)$ is increasing in all components of the state x , i.e., $V(x + e_j) - V(x) \geq 0$ for $j = 1, \dots, N$.*

Proof The proof is by induction on n in $V_n(x)$. Let $V_0(x) = 0$ for all states $x \in \mathcal{X}$. Then, clearly, $V_0(x)$ satisfies the increasingness property. Now, assume that $V_n(x)$ is an increasing function in x . We proceed to prove that $V_{n+1}(x)$ is also increasing in x . First, we have

$$\begin{aligned} V_{n+1}(x + e_j) - V_{n+1}(x) &= \frac{\tau_j}{\lambda_j} + \sum_{i=1}^N \lambda_i [V_n(x + e_i + e_j) - V_n(x + e_i)] \\ &\quad + \min_{a \in \mathcal{A}_x} \{T_a^n(x + e_j)\} - \min_{a \in \mathcal{A}_x} \{T_a^n(x)\} \\ &\geq \min_{a \in \mathcal{A}_x} \{T_a^n(x + e_j)\} - \min_{a \in \mathcal{A}_x} \{T_a^n(x)\}. \end{aligned}$$

The inequality holds because the first term is nonnegative, and the second expression between the brackets is also nonnegative due to the induction hypothesis. Let $a^* =$

$\arg \min_{a \in \mathcal{A}_x} \{T_a^n(x + e_j)\}$. Then we have

$$\begin{aligned} V_{n+1}(x + e_j) - V_{n+1}(x) &\geq T_{a^*}^n(x + e_j) - \min_{a \in \mathcal{A}_x} \{T_a^n(x)\} \\ &\geq T_{a^*}^n(x + e_j) - T_{a^*}^n(x) \\ &= \sum_{i=1}^N \mu(a_i^*) [V_n(x - e_i + e_j) - V_n(x - e_i)] \\ &\quad + \left[N\mu(A) - \sum_{i=1}^N \mu(a_i^*) \right] [V_n(x + e_i) - V_n(x)] \\ &\geq 0. \end{aligned}$$

Therefore, by induction, we have shown that $V_{n+1}(x + e_j) - V_{n+1}(x) \geq 0$. Hence, by taking the limit as n approaches infinity, we get that $V(x)$ is increasing in x_j for $j = 1, \dots, N$. □

We are now ready to pose our main theorem for the model with full flexibility. The theorem characterizes the structure of the optimal policy to be a nondecreasing function in the components of the state, i.e., if the number of customers in queue i increases, then the allocation of the number of servers to queue i is nondecreasing. The following theorem provides a rigorous proof to this statement.

Theorem 7 Consider an arbitrary number $N \in \mathbb{N}$ of queues, and let

$$\begin{aligned} a_i^*(x) &:= \left[\arg \min_{a \in \mathcal{A}_x} \{T_a(x)\} \right]_i \\ &= \left[\arg \min_{a \in \mathcal{A}_x} \left\{ \sum_{i=1}^N \mu(a_i) V([x - e_i]^+) + \left[N\mu(A) - \sum_{i=1}^N \mu(a_i) \right] V(x) \right. \right. \\ &\quad \left. \left. + \sum_{i=1}^N c_i(a_i) \right\} \right]_i. \end{aligned}$$

If the service rate $\mu(a)$ and cost function $c(a)$ are increasing functions in a , then $a_i^*(x)$ is an increasing function in x_i , while $a_j^*(x)$ is nonincreasing in x_i for $j \neq i$.

Proof Fix $x \in \mathcal{X}$ and suppose that $x + e_i \in \mathcal{X}$ as well. If $a^*(x)$ is such that $\sum_{i=1}^N a_i^*(x) < A$, then there is sufficient spare capacity to assign. Hence, in state $x + e_i$, all queues except queue i need no more capacity, since the number of customers in their system did not increase. Hence, queue i can be viewed in isolation to the spare capacity. The queue in isolation satisfies that conditions of Theorem 1 of [14]. From this theorem it follows that $a_i^*(x + e_i) \geq a_i^*(x)$ and $a_j^*(x + e_i) = a_j^*(x)$ for $j \neq i$.

Now suppose that $\sum_{i=1}^N a_i^*(x) = A$, i.e., all servers have been allocated. Denote $a = a^*(x + e_i)$ and $b = a^*(x)$. Note that $T_a(x) - T_b(x) \geq 0$ and $T_a(x + e_i) - T_b(x +$

$e_i) \leq 0$. Therefore, $Z = T_a(x) - T_b(x) - T_a(x + e_i) + T_b(x + e_i) \geq 0$. Since

$$\begin{aligned}
 & T_a(x) - T_b(x) \\
 &= \left[\sum_{j=1}^N c_j(a_j) - \sum_{j=1}^N c_j(b_j) \right] + [\mu(b_i) - \mu(a_i)][V(x) - V(x - e_i)] \\
 & \quad + \sum_{j \neq i}^N [\mu(b_j) - \mu(a_j)][V(x) - V(x - e_j)] \tag{14}
 \end{aligned}$$

and

$$\begin{aligned}
 & T_a(x + e_i) - T_b(x + e_i) \\
 &= \left[\sum_{j=1}^N c_j(a_j) - \sum_{j=1}^N c_j(b_j) \right] + [\mu(b_i) - \mu(a_i)][V(x + e_i) - V(x)] \\
 & \quad + \sum_{j \neq i}^N [\mu(b_j) - \mu(a_j)][V(x + e_i) - V(x + e_i - e_j)], \tag{15}
 \end{aligned}$$

we have

$$\begin{aligned}
 Z &= T_a(x) - T_b(x) - T_a(x + e_i) + T_b(x + e_i) \\
 &= [\mu(a_i) - \mu(b_i)][V(x - e_i) - 2V(x) + V(x + e_i)] \\
 & \quad + \sum_{j \neq i}^N [\mu(a_j) - \mu(b_j)][V(x + e_i) - V(x + e_i - e_j) - V(x) + V(x - e_j)]. \tag{16}
 \end{aligned}$$

Now, we proceed to prove the structure of the policy. We distinguish between four cases:

- (1) $a_j > b_j$ for $j = 1, \dots, N$.
- (2) $a_i < b_i$ and $a_j > b_j$ for $j \neq i$.
- (3) $a_j < b_j$ for $j = 1, \dots, N$.
- (4) $a_i \geq b_i$ and $a_j \leq b_j$ for $j \neq i$.

Note that case 1 cannot occur, since we assumed that our starting point was a state in which all capacity was assigned already. Hence, one cannot assign even more capacity. Cases 2 and 3 do not occur either. Intuitively, assigning fewer servers to queue i , while increasing the number of jobs in queue i leads to degraded performance. To improve readability, the rigorous proofs are given by Lemmas 8 and 9 in the Appendix. The proofs are by contradiction: assuming that the statements of cases 2 or 3 are true, we derive that $Z \leq 0$. However, above we have shown that $Z \geq 0$. Knowing that $Z = 0$ cannot occur, we are left with case 4, which completes the proof. \square

4 Numerical experiments

In this section we will illustrate the monotonicity results of the previous sections. First, we will show how variability in the time constraints affects the processor allocation for the three different models: (I) service facilities with dedicated resources, (II) service facilities with limited resource sharing, and (III) service facilities with full flexibility in resource sharing. Then, we will study the differences between the optimal policies for the three different models; in particular, we will study the effect of having more flexibility in the system versus the reduction in the number of allocated processors. We will run our experiments under two systems that consist of 2 and 3 facilities, respectively. The parameters used in the experiment are defined as follows:

- λ_i : the arrival rate at facility i .
- $\mu(a_i)$: the service rate of using a_i processors at facility i .
- $c_i(a_i)$: the costs of using a_i processors at facility i .
- α_i : the time constraint of facility i .

First, we show the experimental results of the system consisting of two facilities. The parameter values are set as follows:

- $\lambda_1 = \lambda_2 = 0.5$.
- $\mu(a_i) = \sqrt{a_i}\mu$ and $\mu(a_i) = a_i^2\mu/5$ with $\mu = 1.2$.
- $c_i(a_i) = a_i$, $c_i(a_i) = a_i^2$, and $c_i(a_i) = \sqrt{a_i}$.
- $\alpha_1 = 0.5$ and $\alpha_2 \in \{0.25, 0.35, 0.5, 0.75, 1, 1.25, 1.5\}$.

Based on these values, the minimum number of processors required by the system to meet all time constraints of the different facilities are illustrated in Tables 1(a) and (b). From these tables we observe that (1) when the time constraint is less strict, the minimum number of processors required is nonincreasing, and (2) as the system is more flexible, the number of processors needed decreases.

The properties stated above also hold for the experiment with three facilities, and its results are shown in Tables 1(c) and (d). The results of the system with three facilities are based on the parameter values below.

- $\lambda_1 = \lambda_2 = \lambda_3 = 0.5$.
- $\mu(a_i) = \sqrt{a_i}\mu$ and $\mu(a_i) = a_i^2\mu/5$ with $\mu = 1.2$.
- $c_i(a_i) = a_i$, $c_i(a_i) = a_i^2$, $c_i(a_i) = \sqrt{a_i}$.
- $\alpha_1 = \alpha_2 = 0.5$, and $\alpha_3 \in \{0.25, 0.35, 0.5, 0.75, 1, 1.25, 1.5\}$.

From Table 1 we observe that the difference in the required number of processors between the system with dedicated servers and the system with limited resource sharing is quite large (models I versus model II). However, the system with limited and full resource sharing have quite similar performance (model II versus model III). Note that we did not specify the cost functions that have been used in these experiments. This is due to the fact that the calculation of the minimum required number of processors A_i^* , such that the time constraints are satisfied, is independent of the cost functions.

Now we focus our attention to the structure of the optimal resource allocation policy for a system consisting of two and three facilities, respectively. To illustrate

Table 1 Minimum number of processors required to meet service level constraints

(a) System with two facilities and $\mu(a_i) = \sqrt{a_i}\mu$ with $\mu = 1.2$							
α_2	0.25	0.35	0.5	0.75	1	1.25	1.5
dedicated resources	20	13	10	8	7	7	6
limited resource sharing	16	10	6	6	6	6	5
full resource sharing	15	9	6	5	5	5	5
(b) System with two facilities and $\mu(a_i) = a_i^2\mu/5$ with $\mu = 1.2$							
α_2	0.25	0.35	0.5	0.75	1	1.25	1.5
dedicated resources	9	8	8	7	7	7	7
limited resource sharing	5	4	4	4	4	4	4
full resource sharing	5	4	4	4	4	4	4
(c) System with three facilities and $\mu(a_i) = \sqrt{a_i}\mu$ with $\mu = 1.2$							
α_3	0.25	0.35	0.5	0.75	1	1.25	1.5
dedicated resources	25	18	15	13	12	12	11
limited resource sharing	17	11	8	7	7	7	7
full resource sharing	15	9	6	6	6	6	6
(d) System with three facilities and $\mu(a_i) = a_i^2\mu/5$ with $\mu = 1.2$							
α_3	0.25	0.35	0.5	0.75	1	1.25	1.5
dedicated resources	13	12	12	11	11	11	11
limited resource sharing	5	5	4	4	4	4	4
full resource sharing	5	5	4	4	4	4	4

Table 2 Parameter choices for $c_i(\cdot)$ and $\mu(\cdot)$ with $\mu = 1.2$

Experiment 1	$c_i(a_i) = a_i$	$\mu(a_i) = \sqrt{a_i}\mu$
Experiment 2	$c_i(a_i) = a_i^2$	$\mu(a_i) = \sqrt{a_i}\mu$
Experiment 3	$c_i(a_i) = \sqrt{a_i}$	$\mu(a_i) = a_i^2\mu/5$

the structure of the policies, we vary the structure of the functions for $c_i(\cdot)$, and $\mu(\cdot)$ as given in Table 2. We start by showing the experimental results of the system with two facilities. The parameter values used in the three experiments are set as follows.

- $\lambda_1 = \lambda_2 = 0.5$.
- $\alpha_1 = 0.5$ and $\alpha_2 = 0.25$.

In model I of service facilities with dedicated resources, the number of resources A_i^* allocated to facility i in the system is a constant given by $A_i^* = \lceil \mu^{-1}(\lambda_i + 1/\alpha_i) \rceil$. For example, for experiment 1, the expression gives that 5 processors should be used for facility 1 and 15 for facility 2.

We now turn our attention to model II of service facilities with limited resource sharing. Based on Theorem 7, if both $\mu(a_i)$ and $c_i(a_i)$ are increasing functions in the

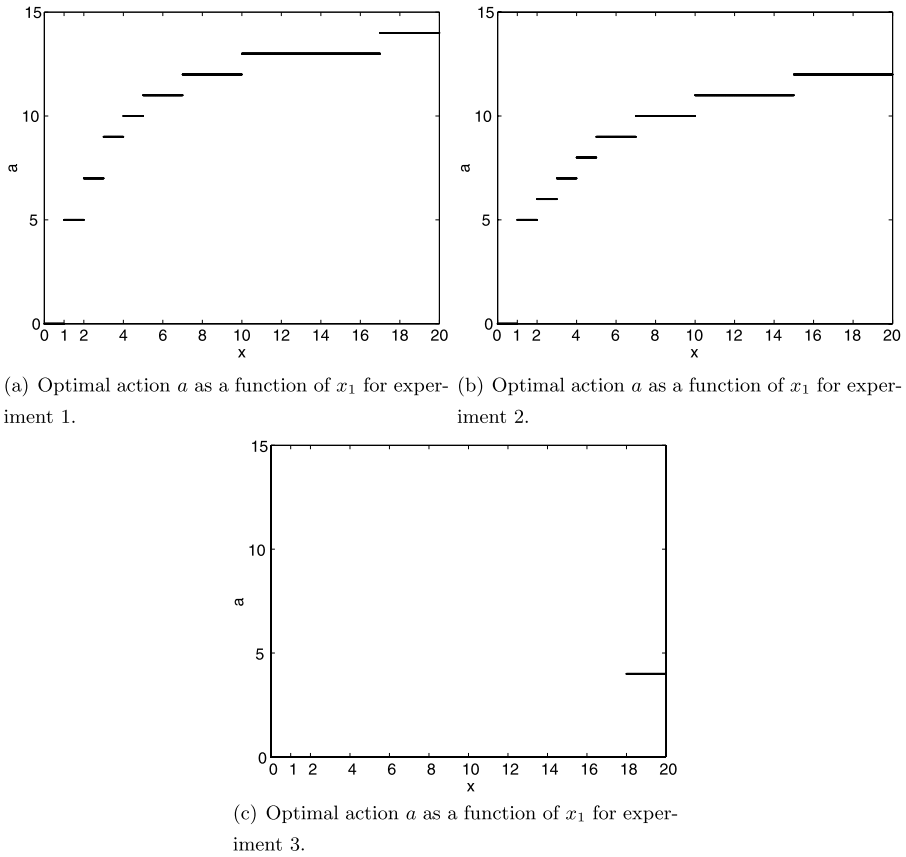


Fig. 1 Experiments of the system with two facilities: optimal policy of facility 1 for the limited resource sharing model, given $(x_2, a_2) = (2, 1)$

number of allocated resources a_i , then the optimal allocation policy to any facility j is a nondecreasing function in the number of customers at that facility, given that the number of customers at all other facilities and the number of resources allocated to all other facilities are fixed. Given that the number of customers at facility 2 is 2 and the number of resources assigned to facility 2 is 1, Fig. 1 illustrates the structure of the optimal policy for facility 1. These three figures correspond to the three experiments shown in Table 2.

We finally discuss model III with full flexibility in resource sharing. The structure of the optimal policy is quite similar to that of the previous model (with limited resource sharing). Consider again the optimal policy for facility 1. The corresponding experimental results are shown in Fig. 2. All figures illustrate that the optimal policy is a nondecreasing function in number of customers at the facility, given that the number of customers at facility 2 is $x_2 = 2$.

Figures 1 and 2 show the structure of the optimal policy for the system with two facilities. The experiments with three (and more) facilities show a similar structure. Combining the experimental results of the system with two and three facilities, we

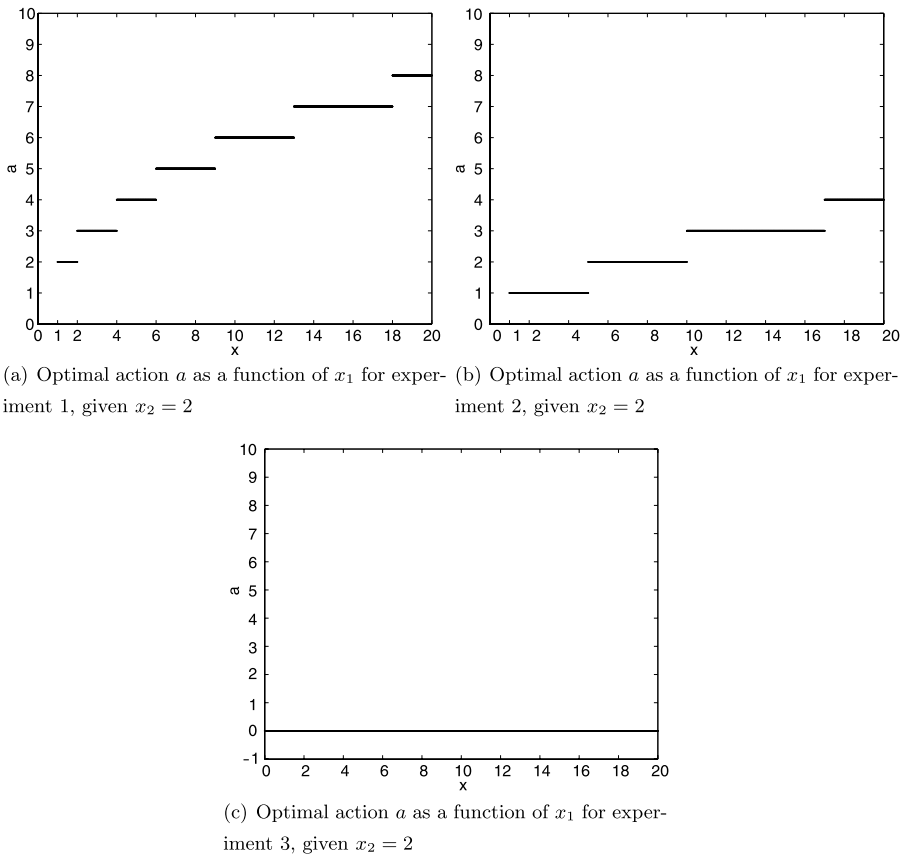


Fig. 2 Experiments of the system with two facilities: optimal policy to facility 1 in case of service facility with full flexibility in resource sharing

conclude that the optimal policy for an arbitrary facility in model I with dedicated resources is a constant, since there is no resource sharing; in case of model II with limited resource sharing and model III with full flexibility in resource sharing, the optimal policy for each facility is a nondecreasing function in the number of customers at that facility.

5 Conclusion and further discussion

In this paper we have derived a characterization of the optimal policy in three different models. We have shown that directional monotonicity is not sufficient to derive the structure of the optimal policy. In addition to directional convexity, the structure of the problem also requires submodularity of the relative value function. In general, this is hard to derive for multidimensional systems, since one needs to compare different states that differ in multiple components simultaneously. The extensive nu-

merical experiments reveal several fundamental insights of the relative effectiveness the optimal policies.

There are several interesting avenues for further research. First, one may suspect that the Poisson assumption of the arrival process can be relaxed. The proof of submodularity shows that the service rates are the dominant factor for the properties of the relative value function. This suggests that there is room for more generality in the arrival process for which the structure of the policies remain valid. Secondly, from an application point of view, generalization to nonexponential service times is practically relevant. It is an open question to what extent the policies are still optimal for, for example, phase-type service distributions. Finally, user-perceived service quality often requires more detailed information than the mean processing time only. The level of quality can be highly dependent on, for example, variance and/or tail probabilities in the processing times. This requires a new approach to handle such service requirements, opening up new challenging areas in research.

Acknowledgements The authors would like to thank Dennis Roubos for fruitful discussions which have helped in the details of the proofs and for help in the simulation code.

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

Appendix: Proofs of Sect. 3.3

In this section we provide proofs of statements made on two cases that appear in Theorem 7. The first case deals with the situation in which $a_i < b_i$ and $a_j > b_j$ for $j \neq i$, i.e., the case in which the queue with one additional customer is allocated strictly fewer processors, whereas the other queues are potentially allocated more processors. The other case concerns $a_j < b_j$ for $j = 1, \dots, N$, i.e., all queues, including the one with an additional customer, are allocated strictly fewer processors. In Theorem 7, the claim is that neither case can occur (which is intuitively plausible). In Lemmas 8 and 9, we show that this claim indeed holds.

Lemma 8 *Let $a = a^*(x + e_i)$ and $b = a^*(x)$. If $a_i < b_i$ and $a_j > b_j$ for $j \neq i$, then the following properties hold:*

1. $V(x)$ is convex in i :

$$V(x + e_i) - 2V(x) + V(x - e_i) \geq 0.$$

2. $V(x)$ is submodular in (x_i, x_j) :

$$\begin{aligned}
 &V(x) - V(x - e_j) - V\left(x + e_i - \sum_{p=1, p \neq j, p \neq i}^N d_p \cdot e_p\right) \\
 &+ V\left(x + e_i - e_j - \sum_{p=1, p \neq j, p \neq i}^N d_p \cdot e_p\right) \geq 0
 \end{aligned}$$

for $j \in \{1, 2, \dots, N\} \setminus \{i\}$ and $d_p \geq 0$.

Proof The proof is by induction on V_n . First, for $n = 0$, define $V_0(x) = 0$ for all x . Clearly, the function $V_0(x)$ is convex in x_i and submodular in (x_i, x_j) . Secondly, assume that the two properties hold for $n = k$. Then, we proceed to prove that $V_{n+1}(x)$ is also convex in x_i and submodular in (x_i, x_j) under the conditions stated in the theorem. Let $c = a^*(x - e_i)$ and recall that $a = a^*(x + e_i)$ and $b = a^*(x)$. By the conditions of the theorem, we have that $a_i < b_i < c_i$ and $a_j > b_j > c_j$ for all $j \neq i$. Now, we are ready to prove the convexity.

Convexity

$$\begin{aligned}
 & V_{n+1}(x + e_i) - 2V_{n+1}(x) + V_{n+1}(x - e_i) \\
 &= \left[\sum_{l=1}^N \lambda_l V_n(x + e_l + e_i) - 2 \sum_{l=1}^N \lambda_l V_n(x + e_l) + \sum_{l=1}^N \lambda_l V_n(x + e_l - e_i) \right] \\
 &\quad + \left[\min_{a \in \mathcal{A}_x} \{T_a^k(x + e_i)\} - 2 \min_{a \in \mathcal{A}_x} \{T_a^k(x)\} + \min_{a \in \mathcal{A}_x} \{T_a^k(x - e_i)\} \right] \\
 &= \sum_{l=1}^N \lambda_l [V_n(x + e_l + e_i) - 2V_n(x + e_l) + V_n(x + e_l - e_i)] \\
 &\quad + \left[\min_{a \in \mathcal{A}_x} \{T_a^k(x + e_i)\} - 2 \min_{a \in \mathcal{A}_x} \{T_a^k(x)\} + \min_{a \in \mathcal{A}_x} \{T_a^k(x - e_i)\} \right] \\
 &\geq \min_{a \in \mathcal{A}_x} \{T_a^k(x + e_i)\} - 2 \min_{a \in \mathcal{A}_x} \{T_a^k(x)\} + \min_{a \in \mathcal{A}_x} \{T_a^k(x - e_i)\}.
 \end{aligned}$$

The inequality holds because the first expression between the brackets is nonnegative due to the induction hypothesis. Now, by expanding the operator T , we derive

$$\begin{aligned}
 & V_{n+1}(x + e_i) - 2V_{n+1}(x) + V_{n+1}(x - e_i) \\
 &\geq T_a^k(x + e_i) - T_a^k(x) - T_c^k(x) + T_c^k(x - e_i) \\
 &\geq \left[\sum_{l=1}^N c_l(a_l) - \sum_{l=1}^N c_l(a_l) - \sum_{l=1}^N c_l(c_l) + \sum_{l=1}^N c_l(c_l) \right] \\
 &\quad + \left[\sum_{l=1}^N \mu(a_l) V_n(x - e_l + e_i) - \sum_{l=1}^N \mu(a_l) V_n(x - e_l) \right. \\
 &\quad \left. - \sum_{l=1}^N \mu(c_l) V_n(x - e_l) + \sum_{l=1}^N \mu(c_l) V_n(x - e_l - e_i) \right] \\
 &\quad + \left[\left(N\mu(A) - \sum_{l=1}^N \mu(a_l) \right) V_n(x + e_i) - \left(N\mu(A) - \sum_{l=1}^N \mu(a_l) \right) V_n(x) \right. \\
 &\quad \left. - \left(N\mu(A) - \sum_{l=1}^N \mu(c_l) \right) V_n(x) + \left(N\mu(A) - \sum_{l=1}^N \mu(c_l) \right) V_n(x - e_i) \right].
 \end{aligned}$$

Note that the first expression between the brackets is equal to 0 (with a slight abuse of notation, where we use $c_l(\cdot)$ for the cost function and c_l as variable for the optimal allocation). Now, we rearrange the terms and put all terms together. Then we get

$$V_{n+1}(x + e_i) - 2V_{n+1}(x) + V_{n+1}(x - e_i) \geq S_i + \sum_{l \neq i, l=1}^N S_l,$$

where

$$\begin{aligned} S_i := & [\mu(a_i)V_n(x - e_l + e_i) - \mu(a_l)V_n(x - e_l) - \mu(c_l)V_n(x - e_l) \\ & + \mu(c_l)V_n(x - e_l - e_i)] \\ & + (\mu(A) - \mu(a_l))V_n(x + e_i) - (\mu(A) - \mu(a_l))V_n(x) \\ & - (\mu(A) - \mu(c_l))V_n(x) + (\mu(A) - \mu(c_l))V_n(x - e_i) \end{aligned}$$

for $l = 1, \dots, N$. Clearly, if all $S_l \geq 0$, then $V_{n+1}(x + e_i) - 2V_{n+1}(x) + V_{n+1}(x - e_i) \geq 0$. Thus, as the next step, we prove that this statement holds. We first discuss this for S_i , and then we deal with the case S_l for $l \neq i$. We first add $-\mu(c_i)[V_n(x) - V_n(x - e_i)] + \mu(c_i)[V_n(x) - V_n(x - e_i)]$ and $[-\mu(a_i) + \mu(a_i)][V_n(x) - V_n(x - e_i)]$ to S_i . Then, we get

$$\begin{aligned} S_i = & [\mu(a_i) - \mu(c_i)][V_n(x) - V_n(x - e_i)] \\ & + \mu(c_i)V_n(x) - 2\mu(c_i)V_n(x - e_i) + \mu(c_i)V_n(x - 2e_i) \\ & + [\mu(A) - \mu(a_i)][V_n(x + e_i) - V_n(x)] \\ & - [\mu(A) - \mu(a_i) + \mu(a_i) - \mu(c_i)][V_n(x) - V_n(x - e_i)] \\ = & [\mu(a_i) - \mu(c_i)][V_n(x) - V_n(x - e_i)] \\ & + \mu(c_i)V_n(x) - 2\mu(c_i)V_n(x - e_i) + \mu(c_i)V_n(x - 2e_i) \\ & + [\mu(A) - \mu(a_i)][V_n(x + e_i) - V_n(x)] \\ & - [\mu(A) - \mu(a_i)][V_n(x) - V_n(x - e_i)] \\ & - [\mu(a_i) - \mu(c_i)][V_n(x) - V_n(x - e_i)] \\ = & \mu(c_i)[V_n(x) - 2V_n(x - e_i) + V_n(x - 2e_i)] \\ & + [\mu(A) - \mu(a_i)][V_n(x + e_i) - 2V_n(x) + V_n(x - e_i)] \\ \geq & 0. \end{aligned}$$

The second and third equalities above follow from the standard calculus. The last inequality holds because of the induction hypothesis. Next, we proof that $S_l \geq 0$ for $l \neq i$. By rearranging the terms of S_l , we derive

$$\begin{aligned} S_l = & \mu(a_l)[V_n(x) + V_n(x - e_l + e_i) - V_n(x + e_i) - V_n(x - e_l)] \\ & + \mu(c_l)[V_n(x) + V_n(x - e_l - e_i) - V_n(x - e_i) - V_n(x - e_l)] \\ & + \mu(A)[V_n(x + e_i) - 2V_n(x) + V_n(x - e_i)]. \end{aligned}$$

Recall that $a_l > c_l$ for $l \neq i$. In addition, due to the induction hypothesis, $V_n(x)$ is submodular in (x_i, x_l) for all $d_p \geq 0$ and for $l \neq i$. This implies that

$$\begin{aligned}
 &V_n(x) - V_n(x - e_l) - V_n\left(x + e_i - \sum_{p=1, p \neq i, p \neq l}^N d_p \cdot e_p\right) \\
 &+ V_n\left(x + e_i - e_l - \sum_{p=1, p \neq i, p \neq l}^N d_p \cdot e_p\right) \geq 0.
 \end{aligned}$$

Set $d_p = 0$ for all $p \in \{1, \dots, N\} \setminus \{i, l\}$. Then it follows that $V_n(x) + V_n(x - e_l + e_i) - V_n(x + e_i) - V_n(x - e_l) \geq 0$ for $l \neq i$. Therefore, using $a_l > c_l$, we derive

$$\begin{aligned}
 S_l &\geq \mu(c_l)[V_n(x) + V_n(x - e_l + e_i) - V_n(x + e_i) - V_n(x - e_l) \\
 &+ V_n(x) + V_n(x - e_l - e_i) - V_n(x - e_i) - V_n(x - e_l)] \\
 &+ \mu(A)[V_n(x + e_i) - 2V_n(x) + V_n(x - e_i)] \\
 &\geq \mu(c_l)[V_n(x - e_l + e_i) - 2V_n(x - e_l) + V_n(x - e_l - e_i)] \\
 &+ [\mu(A) - \mu(c_l)][V_n(x + e_i) - 2V_n(x) + V_n(x - e_i)] \\
 &\geq 0.
 \end{aligned}$$

Hence, we conclude that $V_{n+1}(x + e_i) - 2V_{n+1}(x) + V_{n+1}(x - e_i) \geq 0$. Now, we proceed to prove that $V_{n+1}(x)$ is submodular in (x_i, x_j) . To this purpose, define $P = \{1, \dots, N\} \setminus \{i, j\}$.

Submodularity Let $d_p \geq 0$ for all $p \in P$. Then,

$$\begin{aligned}
 &V_{n+1}(x) - V_{n+1}(x - e_j) - V_{n+1}\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \\
 &+ V_{n+1}\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \\
 &= \sum_{l=1}^N \lambda_l \left[V_n(x + e_l) - V_n(x - e_j + e_l) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p + e_l\right) \right. \\
 &+ \left. V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p + e_l\right) \right] \\
 &+ \left[\min_{a' \in \mathcal{A}_x} \{T_{a'}^k(x)\} - \min_{a' \in \mathcal{A}_x} \{T_{a'}^k(x - e_j)\} - \min_{a' \in \mathcal{A}_x} \left\{ T_{a'}^k\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \right\} \right. \\
 &+ \left. \min_{a' \in \mathcal{A}_x} \left\{ T_{a'}^k\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right\} \right]
 \end{aligned}$$

$$\begin{aligned} &\geq \min_{a' \in \mathcal{A}_x} \{T_{a'}^k(x)\} - \min_{a' \in \mathcal{A}_x} \{T_{a'}^k(x - e_j)\} - \min_{a' \in \mathcal{A}_x} \left\{ T_{a'}^k \left(x + e_i - \sum_{p \in P} d_p \cdot e_p \right) \right\} \\ &\quad + \min_{a' \in \mathcal{A}_x} \left\{ T_{a'}^k \left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p \right) \right\}. \end{aligned}$$

The inequality holds because the first expression between the brackets is nonnegative due to the induction hypothesis. Let $\tilde{a} = a^*(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p)$ and recall that $b = a^*(x)$. Then, it follows that

$$\begin{aligned} &V_{n+1}(x) - V_{n+1}(x - e_j) - V_{n+1} \left(x + e_i - \sum_{p \in P} d_p \cdot e_p \right) \\ &\quad + V_{n+1} \left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p \right) \\ &\geq T_b^k(x) - T_b^k(x - e_j) - T_{\tilde{a}}^k \left(x + e_i - \sum_{p \in P} d_p \cdot e_p \right) \\ &\quad + T_{\tilde{a}}^k \left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p \right) \\ &= \left[\sum_{l=1}^N \mu(b_l) V_n(x - e_l) + \left(N\mu(A) - \sum_{l=1}^N \mu(b_l) \right) V_n(x) \right] \\ &\quad - \left[\sum_{l=1}^N \mu(b_l) V_n(x - e_j - e_l) + \left(N\mu(A) - \sum_{l=1}^N \mu(b_l) \right) V_n(x - e_j) \right] \\ &\quad - \left[\sum_{l=1}^N \mu(\tilde{a}_l) V_n \left(x + e_i - \sum_{p \in P} d_p \cdot e_p - e_l \right) \right. \\ &\quad \left. + \left(N\mu(A) - \sum_{l=1}^N \mu(\tilde{a}_l) \right) V_n \left(x + e_i - \sum_{p \in P} d_p \cdot e_p \right) \right] \\ &\quad + \left[\sum_{l=1}^N \mu(\tilde{a}_l) V_n \left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p - e_l \right) \right. \\ &\quad \left. + \left(N\mu(A) - \sum_{l=1}^N \mu(\tilde{a}_l) \right) V_n \left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p \right) \right]. \end{aligned}$$

Now we rearrange the terms to simplify the inequality. Then,

$$\begin{aligned}
 & V_{n+1}(x) - V_{n+1}(x - e_j) - V_{n+1}\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \\
 & \quad + V_{n+1}\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \\
 & \geq \sum_{l=1}^N \mu(b_l) [V_n(x - e_l) - V_n(x) - V_n(x - e_j - e_l) + V_n(x - e_j)] \\
 & \quad - \sum_{l=1}^N \mu(\tilde{a}_l) \left[V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p - e_l\right) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \right. \\
 & \quad \left. - V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p - e_l\right) + V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right] \\
 & \quad + N\mu(A) \left[V_n(x) - V_n(x - e_j) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \right. \\
 & \quad \left. + V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right].
 \end{aligned}$$

As the next step, we write the final expression above as $S_i + S_j + \sum_{l=1, l \neq i, l \neq j}^N S_l$, where

$$\begin{aligned}
 S_k &= \mu(b_k) [V_n(x - e_k) - V_n(x) - V_n(x - e_j - e_k) + V_n(x - e_j)] \\
 & \quad - \mu(\tilde{a}_k) \left[V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p - e_k\right) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \right. \\
 & \quad \left. - V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p - e_k\right) + V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right] \\
 & \quad + \mu(A) \left[V_n(x) - V_n(x - e_j) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \right. \\
 & \quad \left. + V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right]
 \end{aligned}$$

for $k = 1, \dots, N$. In the remainder of the proof, we will show that all $S_k \geq 0$. First, we focus on S_j . Consider the first term between brackets in S_j . By setting $x' = x - e_i$, we see that this term is equal to $V_n(x') - V_n(x' - e_j) - V_n(x' + e_i) + V_n(x' + e_i - e_j) \geq 0$,

since $V_n(x)$ is submodular. In addition, because of the conditions of the theorem, we have $b_i \geq \tilde{a}_i$. Therefore,

$$\begin{aligned}
 S_3 &\geq \mu(\tilde{a}_i) [V_n(x - e_i) - V_n(x) - V_n(x - e_j - e_i) + V_n(x - e_j)] \\
 &\quad - \mu(\tilde{a}_i) \left[V_n \left(x - \sum_{p \in P} d_p \cdot e_p \right) - V_n \left(x + e_i - \sum_{p \in P} d_p \cdot e_p \right) \right. \\
 &\quad \left. - V_n \left(x - e_j - \sum_{p \in P} d_p \cdot e_p \right) + V_n \left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p \right) \right] \\
 &\quad + \mu(A) \left[V_n(x) - V_n(x - e_j) - V_n \left(x + e_i - \sum_{p \in P} d_p \cdot e_p \right) \right. \\
 &\quad \left. + V_n \left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p \right) \right] \\
 &= \mu(\tilde{a}_i) \left[V_n(x - e_i) - V_n(x - e_j - e_i) - V_n \left(x - \sum_{p \in P} d_p \cdot e_p \right) \right. \\
 &\quad \left. + V_n \left(x - e_j - \sum_{p \in P} d_p \cdot e_p \right) \right] \\
 &\quad + [\mu(A) - \mu(\tilde{a}_i)] \left[V_n(x) - V_n(x - e_j) - V_n \left(x + e_i - \sum_{p \in P} d_p \cdot e_p \right) \right. \\
 &\quad \left. + V_n \left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p \right) \right] \\
 &\geq 0.
 \end{aligned}$$

The second equality above follows from standard calculus, and the last inequality follows from submodularity of the induction hypothesis. Now, we proceed to study S_j . Since $V_n(x)$ is convex, the first expression between the brackets of S_j is nonpositive. Combining this result with the condition of the theorem, $\tilde{a}_j \geq b_j$ that is equivalent to $-b_j \geq -\tilde{a}_j$, we get

$$\begin{aligned}
 S_j &\geq \mu(\tilde{a}_j) [V_n(x - e_j) - V_n(x) - V_n(x - 2e_j) + V_n(x - e_j)] \\
 &\quad - \mu(\tilde{a}_j) \left[V_n \left(x + e_i - \sum_{p \in P} d_p \cdot e_p - e_j \right) - V_n \left(x + e_i - \sum_{p \in P} d_p \cdot e_p \right) \right. \\
 &\quad \left. - V_n \left(x + e_i - 2e_j - \sum_{p \in P} d_p \cdot e_p \right) + V_n \left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p \right) \right]
 \end{aligned}$$

$$\begin{aligned}
 & + \mu(A) \left[V_n(x) - V_n(x - e_j) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \right. \\
 & \left. + V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right] \\
 \geq & \mu(\tilde{a}_j) \left[V_n(x - e_j) - V_n(x - 2e_j) - V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right. \\
 & \left. + V_n\left(x + e_i - 2e_j - \sum_{p \in P} d_p \cdot e_p\right) \right] \\
 & + [\mu(A) - \mu(\tilde{a}_j)] \left[V_n(x) - V_n(x - e_j) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \right. \\
 & \left. + V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right] \\
 \geq & 0.
 \end{aligned}$$

Finally, we prove that S_l for $l \neq i$ and $l \neq j$ is also nonnegative. Let $x' = x - e_j$. Then the first term of S_l is equal to $V_n(x') - V_n(x' - e_l) - V_n(x' + e_j) + V_n(x' + e_j - e_l)$. Since $V_n(x)$ is submodular in all (x_i, x_j) with $j \neq i$, $V_n(x')$ is submodular in (x_j, x_l) . Therefore, using $\mu(A) \geq \mu(\tilde{a}_l)$, we derive that

$$\begin{aligned}
 S_l \geq & \mu(b_l) [V_n(x - e_j) - V_n(x - e_j - e_l) - V_n(x) + V_n(x - e_l)] \\
 & - \mu(\tilde{a}_l) \left[V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p - e_l\right) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \right. \\
 & \left. - V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p - e_l\right) + V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right] \\
 & + \mu(\tilde{a}_l) \left[V_n(x) - V_n(x - e_j) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p\right) \right. \\
 & \left. + V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p\right) \right] \\
 \geq & \mu(\tilde{a}_l) \left[V_n(x) - V_n(x - e_j) - V_n\left(x + e_i - \sum_{p \in P} d_p \cdot e_p - e_l\right) \right. \\
 & \left. + V_n\left(x + e_i - e_j - \sum_{p \in P} d_p \cdot e_p - e_l\right) \right].
 \end{aligned}$$

Now, let $d' = d + e_l$. Then,

$$\begin{aligned}
 S_l &\geq \mu(\tilde{a}_l) \left[V_n(x) - V_n(x - e_j) - V_n\left(x + e_i - \sum_{p \in P} d'_p \cdot e_p\right) \right. \\
 &\quad \left. + V_n\left(x + e_i - e_j - \sum_{p \in P} d'_p \cdot e_p\right) \right] \\
 &\geq 0.
 \end{aligned}$$

Hence, $V_{n+1}(x)$ is submodular in (x_i, x_j) . Thus, by taking the limit as $n \rightarrow \infty$, we conclude that $V(x)$ is convex and submodular under the conditions given in the theorem. □

Lemma 8 shows that the relative value function V satisfies convexity and submodularity in case $a_i < b_i$ and $a_j > b_j$ for $j \neq i$. Note that this leads to a contradiction in Theorem 7, since this implies that $Z = T_a(x) - T_b(x) - T_a(x + e_i) + T_b(x + e_i) \leq 0$. However, it was shown in the theorem that $Z \geq 0$. Hence, this case cannot occur. Similarly, the case $a_j < b_j$ for all j also cannot occur. The next lemma provides a rigorous statement for this.

Lemma 9 *Let $a = a^*(x + e_i)$ and $b = a^*(x)$. If $a_j < b_j$ for $j = 1, \dots, N$, then the following properties hold:*

1. $V(x)$ is convex in i :

$$V(x + e_i) - 2V(x) + V(x - e_i) \geq 0.$$

2. $V(x)$ is supermodular in (x_i, x_j) :

$$\begin{aligned}
 &V\left(x + e_i - \sum_{p=1, p \neq j, p \neq i}^N d_p \cdot e_p\right) - V\left(x + e_i - e_j - \sum_{p=1, p \neq j, p \neq i}^N d_p \cdot e_p\right) \\
 &\quad - V(x) + V(x - e_j) \geq 0
 \end{aligned}$$

for $j \in \{1, 2, \dots, N\} \setminus \{i\}$ and $d_p \geq 0$.

Proof The proof of the lemma is by induction on V_n . Note that the proof of convexity and supermodularity is completely analogous to the proof of convexity and submodularity in Lemma 8. The only difference is that in Lemma 8 we have $a_l > c_l$ for $l \neq i$, whereas in this case, $a_l < c_l$. However, with supermodularity (instead of submodularity) the signs of the inequalities turn out to be correct for proving convexity and supermodularity. □

References

1. Altman, E.: Constrained Markov Decision Processes. Chapman and Hall, London (1999)

2. Bhulai, S., Koole, G.M.: A queueing model for call blending in call centers. *IEEE Trans. Autom. Control* **48**, 1434–1438 (2003)
3. Borst, S.C., Seri, P.: Robust algorithms for sharing agents with multiple skills. Technical Report, Bell Laboratories, Murray Hill, NJ (2000)
4. Gans, N., Zhou, Y.: A call-routing problem with service-level constraints. *Oper. Res.* **51**, 255–271 (2003)
5. Kleinrock, L.: A delay-dependent queue discipline. *Nav. Res. Logist. Q.* **11**, 59–73 (1964)
6. Kleinrock, L., Finkelstein, R.P.: Time dependent priority queues. *Oper. Res.* **15**, 104–116 (1967)
7. Koole, G.: Monotonicity in Markov reward and decision chains: the theory and applications. *Found. Trends Stoch. Syst.* **1** (2006)
8. Perry, M., Nilsson, A.: Performance modeling of automatic call distributors: Assignable grade of service staffing. In: XIV International Switching Symposium, pp. 294–298 (1992)
9. Puterman, M.L.: *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, New York (1994)
10. Shumsky, R.A.: Approximation and analysis of a queueing system with flexible and specialized servers. *OR Spectrum* **26** (2004)
11. Stanford, D.A., Grassmann, W.K.: Bilingual server call centres. In: McDonald, D.R., Turner, S.R.E. (eds.) *Call Centres, Traffic and Performance*, vol. 28, pp. 31–48. Springer, Berlin (2000)
12. Veatch, M.H., Wein, L.M.: Monotone control of queueing networks. *Queueing Syst.* **12**, 393–408 (1992)
13. Weber, R.R., Stidham, S.: Optimal control of service rates in networks of queues. *Adv. Appl. Probab.* **19**(1), 202–218 (1987)
14. Yang, R., Bhulai, S., van der Mei, R., Seinstra, F.: Optimal resource allocation for time-reservation systems. Technical report, VU University Amsterdam (2010)