Service Scheduling for Random Requests With Deadlines and Linear Waiting Costs

Ramya Burra, Chandramani Singh, and Joy Kuri

Abstract—**We study service scheduling problems in a slotted system where jobs arrive according to a Bernoulli process and leave within two slots after arrival. Service costs are quadratic in service rates, and there is also a linear waiting cost. We frame the problems as average cost Markov decision processes. While the studied system is a linear system with quadratic costs, it has state-dependent control and a non-standard cost function structure, rendering the optimization problem complex. We obtain explicit optimal policies in the case when all the jobs are of the same size. In particular, we show that the optimal policy is linear or piece-wise linear in the system state, depending on the system parameters. We then consider a scenario where each service request comes from a rational agent interested in optimizing his/her service and waiting cost, and we obtain a symmetric Nash equilibrium. We extend our study to a scenario where job sizes can take distinct values, and job arrivals constitute a Markov chain. Here, we provide an algorithm that yields the optimal policy, but it is of exponential complexity. Finally, we propose an approximate policy of linear complexity for general job size distributions and derive its performance bound.**

Index Terms—**Service Scheduling, Linear waiting costs, Quadratic service cost, Markov Decision Process.**

I. Introduction

SERVICE or job scheduling problems arise in many contexts such as cloud computing, task scheduling in CPUs, traffic routing and scheduling, production scheduling in plants, scheduling charging of *electric vehicles* (EVs), etc. For instance, in cloud computing, server power consumption increases as a convex function of the load [1]. Hence, often delay-tolerant jobs need to be deferred in order to save on long-term average power cost. Similarly, traffic needs to be regulated in transportation networks to ensure that congestion is not excessive at any point in time. Serious concerns have been raised about service costs in many of these problems. For instance, it was estimated that electricity used in global data centers was about 1.1% of the total electricity use in 2010, and it has been increasing since then [2]. In the context of EV charging, the electricity cost of a charging station could rise quite steeply as the load increases. So, it is immensely important to schedule jobs or services in all such cases optimally.

When service costs are convex functions of offered loads, optimal scheduling promotes load balancing over time. However, if deferring loads also incur waiting costs, equalizing loads across time is not optimal. We study service scheduling in slotted systems with Bernoulli service arrivals and service delay guarantees. We consider quadratic service costs and linear waiting costs. In particular, jobs can stay for two slots but incur a waiting cost in the second slot. We see that this service scheduling problem is a special case of constrained linear quadratic control (See [3, Section 4.2]). We provide an optimal service scheduling policy and a symmetric Nash equilibrium policy for the scenario where jobs are interested in minimizing their own costs. Our framework applies to several service scheduling problems like CPU speed scaling, EV charging, etc. For an elaborate discussion, see Section A of supplementary material.

A. Related Work

Job or service scheduling problems have drawn great attention in many domains in the recent past. In the context of EV charging, scheduling is aimed at optimizing charging costs. Electricity generation cost is known to be a quadratic function of the charging rate [4]. The authors in [5] adopt this quadratic cost model. They assume that EV arrival statistics are not available and propose an online scheduling algorithm to minimize the cost. In [6], the authors consider a charging station with multiple outlets and real-time pricing. They assume that EVs have uncertain departure times. They frame the EV scheduling problem as a stochastic optimization problem and provide a closed-form solution. The authors of [7] have formulated a *vehicle to grid* (V2G) control problem with price uncertainty as a Markov decision process (MDP). Furthermore, they also proposed a Q-learning algorithm. In [8], the authors consider the problem of minimizing random processing cost and convex non-completion penalty for jobs that arrive randomly at a service center with multiple service stations. They formulate this problem as a restless multi-arm bandit. They propose an approximate algorithm with asymptotic guarantees.

In the area of smart grids, scheduling aims at *peak-shaving* to save on steep electricity costs at higher loads. In [9], the authors consider renewable energy buying-back schemes with dynamic pricing to achieve energy efficiency for smart grids.

Manuscript received August 21, 2020; revised June 9, 2021; accepted June 15, 2021. Date of publication June 23, 2021; date of current version September 16, 2021. This work was supported in part by the Research Fellowships of Visvesvaraya Ph.D. scheme, awarded to Ramya Burra and Chandramani Singh, and in part by INSPIRE Faculty Research under Grant DSTO-1363. Recommended for acceptance by Dr. Zhen Wang. *(Corresponding author: Ramya Burra.)*

The authors are with the Department of Electronic Systems Engineering, Indian Institute of Science, Bangalore, Karnataka 560012, India (e-mail: burra@iisc.ac.in; chandra@iisc.ac.in; kuri@dese.iisc.ernet.in).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNSE.2021.3091763, provided by the authors.

Digital Object Identifier 10.1109/TNSE.2021.3091763

They pose this problem as convex optimization and design distributed day-ahead pricing algorithms. The authors in [10] find the optimal energy consumption levels for grid subscribers to maximize the aggregate utility of all the subscribers.

In the case of data centers, server scheduling, aimed at minimizing power consumption, is implemented in the form of dynamic voltage and frequency scaling (DVFS). The authors in [11], [12] propose algorithms to dynamically schedule the workload on Internet data center servers while meeting constant service delay guarantees. In [13], the authors assume that the energy curve of data center switches is also a convex function of their transmission speeds and design VM assignment and routing algorithms to minimizing energy consumption.

The authors in [14] consider a finite number of agents competing for a common link to ship their demands to a destination over time. They consider polynomial congestion cost and linear waiting cost to obtain optimal, and equilibrium flows. However, in this setup, every agent has a common arrival and deadline. In [15], the authors consider routing on a ring network in the presence of quadratic congestion cost. However, their formulation does not have a temporal element and is thus a one-shot optimization problem. Scheduling for minimizing energy costs has also been considered in the context of CPU power consumption [16], [17], big data processing [18], production scheduling in plants [19]. In [20], the authors propose an optimal online algorithm for job arrivals with deadline uncertainty. In this work, they consider convex processing cost. They also derive a competitive ratio for the proposed algorithm. None of these studies account for the waiting costs of jobs as considered in our work.

Speed scaling problems have been studied extensively in computer science and systems engineering. In one of the first works, the authors in [16] consider a minimum energy scheduling problem in the context of CPU processing. They assume that instantaneous processing cost is a convex function of the processing speed and provides optimal offline and online algorithms. There have been several follow-up works since then. In [21], the authors provide a formal proof of optimality of the algorithm in [16]. The authors also propose online algorithms to minimize energy and minimize the maximum temperature of the CPU. In the same context, the authors in [22] study the tradeoff between the energy consumption and the total flow time of all jobs. Unlike the above problems, we consider service scheduling in a slotted system and, in addition to energy cost, also include a waiting cost in the objective. Our service scheduling framework applies to several contexts, e.g., CPU speed scaling, EV charging, etc.

For instance, in the context of EV charging, the drivers become jittery as the deadline approaches if they have not received a good amount of charge until then [23]. This justifies the inclusion of waiting costs in our formulation. Our problem can be seen as a discrete version of the speed scaling problem with a waiting cost. More precisely, we minimize the total average cost that comprises energy cost and a waiting cost. Recently, the authors in [24] focus on optimizing a weighted sum of energy cost and tardiness in the context of jobs with soft deadlines. We can see our setting as having a soft deadline of one slot and a hard deadline of two slots. Unlike [24], we consider weighted tardiness cost, weighed by the amount of service not met by the soft deadline. Refer to Table I for an overview of the distinction between our formulation and others.

TABLE I
COMPARISION OF OUR FORMULATION WITH DIFFERENT MODELS

References	Objective	Service cost
Yao et al. [16]	Minimize energy	Convex function
Bansal et al. [21]	Minimize energy and maximum temperature attained	Power function [1]
Albers and Fujiwara [22]	Minimize energy and flow time	Power function
Che et al. [24]	Minimize energy and maximum tardiness in a power-down mechanism	fixed amount of energy when the CPU is on
Our formulation	Minimize energy and waiting cost	Quadratic function (i.e., Power function with 2 as the exponent)

Linear systems with quadratic cost have also been widely studied in control theory. For instance, in infinite horizon unconstrained linear quadratic control, the optimal policy is found to be linear in system state and is given by the *Riccati equation* [25]. The authors in [26] study constrained linear quadratic control and show that the optimal policy is piecewise linear and the value function is piece-wise quadratic. We have a control problem with state-dependent constraints. The problem does not conform to standard assumptions, e.g., positive definiteness of the control weighing matrix.

B. Our Contribution

(1) We study optimal service scheduling for Bernoulli job arrivals, quadratic service costs, linear waiting costs, and maximum sojourn time of two slots. We frame this as a constrained linear quadratic control problem and derive optimal scheduling policy for the case where jobs' service requirements are identical.

(2) We then consider a scenario where each service request comes from a rational agent interested in optimizing his/her service and waiting cost. We obtain a symmetric Nash equilibrium for the associated stochastic game.

(3) We extend the framework to allow two distinct job sizes with Markov service requests. We do not get a closed-form optimal policy, but we provide an algorithm that computes the optimal policy.

(4) We extend the framework to general, i.i.d. service requirements. The algorithm for two distinct job sizes can be extended to address this scenario as well. We also propose a closed-form approximate policy and derive its performance bound.

II. SYSTEM MODEL

We consider a discrete-time (slotted) system with a service facility and dynamically arriving "splittable" service requests. Thus, for a service request in any slot, the service provider may defer a portion of it to future slots. Serving requests incur a cost, and the cost per unit service in a slot depends on the quantum of service delivered in that slot. Here we focus on

[1] The function $f:[0,\infty)\to\mathbb{R}_+$ is called a power function with exponent α, if $f(x)=x^{\alpha},\alpha\geq 1$.

the problems of scheduling of service. Below we present the system model and the scheduling problems formally.

Service request model: Agents with service requests arrive according to an i.i.d. Bernoulli(p) process. All the agents demand ψ amount of service. Further, each request can be met in at most two slots, i.e., a fraction of the demand arriving in a slot could be deferred to the next slot. In Section V, we extend this model to include general service requirements and Markov job arrivals.

Cost model: The cost consists of two components:

(1) *Service cost:* The service cost per unit service in a slot is a linear function of the total service offered in that slot. Thus the total service cost in a slot is square of the total offered service in that slot.

(2) *Waiting cost:* Each service incurs a waiting cost that is a linear function of the portion of the service deferred to the next slot. We use d to denote the waiting cost per deferred unit service. We use linear waiting cost instead of prominently used constant waiting cost for two reasons. First, it encourages the service provider not to defer large fractions of demands. Secondly, in many of the systems of interest, agents can leave the system in between slots also if their service requirements are met. In such systems, assuming the earliest deadline first service discipline, agents' waiting costs would indeed be commensurate with the amount of deferred demands.

Let, for $k \geq 1$, x_k be the remaining demand from slot $k-1$ to slot k; $x_1 = 0$. This demand must be met in slot k. Also, for $k \geq 1$, let v_k be the extra service offered in slot k. Clearly, $v_k \in [0, \psi]$ and is 0 if there is no request in slot k. A *scheduling policy* $\overline{\pi} = (\pi_k, k \geq 1)$ is a sequence of functions $\pi_k : [0, \psi] \rightarrow [0, \psi]$ such that if there is a service request in slot k then $\pi_k(x_k)$ gives the amount of service deferred from slot k to slot $k+1$. In other words,

$$x_{k+1} = \begin{cases} \pi_k(x_k) = \psi - v_k, & \text{if a request arrives in slot } k \\ 0, & \text{otherwise.} \end{cases}$$

We consider the following two scheduling problems.

A. Optimal Scheduling

We aim to minimize the time-averaged cost of the service provider. More precisely, we want to determine the scheduling policy $\overline{\pi}$ that minimizes

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{k=1}^{T} \mathbb{E}[(x_k + v_k)^2 + d x_k]. \tag{1}$$

We obtain the optimal solution in Section III.

At first glance, the optimization problem appears to be a special case of the well-studied *constrained linear quadratic control Markov decision problems*. In particular, if we define binary variables $e_k, k \geq 1$, as

$$e_k = \begin{cases} \psi, & \text{if slot } k \text{ has a request} \\ 0, & \text{otherwise} \end{cases}$$

then (x_k, e_k) can be considered to be the system state in slot k. The total service in slot k, $\bar{u}_k \in [x_k, x_k + e_k]$, and $w_k = e_{k+1}$ can be considered the action and the noise in slot k, respectively. Then state evolution happens as $(x_{k+1}, e_{k+1}) = (x_k + e_k - \bar{u}_k, w_k)$ and the single stage cost is $d x_k + \bar{u}_k^2$. *We see that the actions are subject to state dependent constraints and the single stage costs are not expressible in the form* $(x_k, e_k)^T Q (x_k, e_k) + \bar{u}_k^2$ *with* Q *a positive semidefinite matrix*. Thus the problem does not conform to the standard framework.

B. Equilibrium for Selfish Agents

Here, we consider rational agents, each determining how much of its demand should be deferred. Further, each agent is aiming at minimizing his/her own service and waiting costs. *For example, in the context of EV charging, EV owners being rational agents would be interested in minimizing their own cost instead of the total system cost*. In such scenarios we can frame service-scheduling problem as a non-cooperative dynamic game among the agents. In this context, let us refer to π_k as a strategy of the agent who arrives in slot k (if there is one) and $\overline{\pi} = (\pi_k, k \geq 1)$ as a strategy profile.[2] The expected cost of an agent who arrives in slot k, if it sees a remaining demand x, is

$$\begin{aligned} c_k(x, \overline{\pi}) = & (\psi - \pi_k(x))(\psi - \pi_k(x) + x) + \pi_k(x)(\pi_k(x) \\ & + p(\psi - \pi_{k+1}(\pi_k(x))) + d), \end{aligned} \tag{2}$$

A strategy profile $\overline{\pi}$ is called a *Nash equilibrium* if

$$c_k(x, \overline{\pi}) \leq c_k(x, (\mu, \overline{\pi}_{-k}))$$

for all $k \geq 1$, $x \in [0, \psi]$ and strategies $\mu : [0, \psi] \rightarrow [0, \psi]$.[3] We focus on symmetric Nash equilibria of the form $(\pi, \pi, \ldots)$ and obtain one such equilibrium in Section IV.

The proposed framework can be used to model many scheduling and speed scaling problems, e.g., traffic scheduling, speed scaling in CPUs, scheduling of charging of EVs etc. Please see Section A in supplementary material for a detailed discussion.

III. Optimal Scheduling

We first show that the optimal scheduling problem can be transformed into a stochastic shortest path problem. Let $A_i, i \geq 1$ be the successive slots that have service requests but do not have service requests in the preceding slots. More precisely,

$$A_i = \begin{cases} \min\{k : \text{slot } k \text{ has a request}\}, & \text{if } i = 1 \\ \min\{k > A_{i-1} : \text{slot } k \text{ has a request but} \\ \qquad k-1 \text{ does not}\}, & \text{if } i \geq 2. \end{cases}$$

Then $A_i, i \geq 1$ can be seen to be *renewal instants*of a renewal process. We show below that the mean renewal life times, $\mathbb{E}[A_{i+1} - A_i]$, do not depend on the scheduling policy.

Lemma 3.1: $\mathbb{E}(A_{i+1} - A_i) = \frac{1}{p(1-p)}$.

[2] Notice that π consists of a strategy for each slot but there may not be any agent in a slot to use the corresponding strategy.

[3] $(\mu, \overline{\pi}_{-k}) \triangleq (\pi_1, \ldots, \pi_{k-1}, \mu, \pi_{k+1}, \ldots)$.

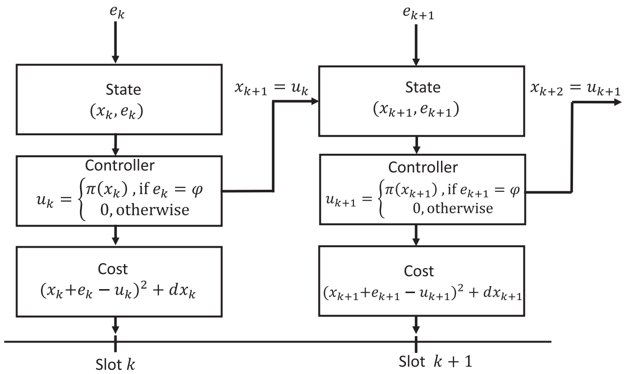

Fig. 1. The system evolution: The noise variables e_ks are i.i.d. Bernoulli(p). The state (x_k, e_k) and action u_k at slot k determine the cost at slot k and also the state at slot $k+1$.

Proof: See Section B in supplementary material. ■

Hence, from the *Renewal Reward Theorem*,

$$\lim_{T\to\infty} \frac{1}{T} \sum_{k=1}^{T} \mathbb{E}[(x_k + v_k)^2 + dx_k]$$
$$= \frac{\mathbb{E}\left[\sum_{k=A_i}^{A_{i+1}-1} \left((x_k + v_k)^2 + dx_k\right)\right]}{\mathbb{E}[A_{i+1} - A_i]}$$
$$= p(1-p)\mathbb{E}\left[\sum_{k=A_i}^{A_{i+1}-1} \left((x_k + v_k)^2 + dx_k\right)\right],$$

where the last equality follows from Lemma 3.1. So, we can focus on minimizing the aggregate cost over a "renewal lifetime" $A_{i+1} - A_i$. But we do not incur any cost after service completion of the last customer in this lifetime. We can thus frame the problem as *stochastic shortest path problem* where terminal state corresponds to absence of request in a slot.

Remark 3.1: Average cost optimality problems have equivalent stochastic shortest path formulations and we often solve the latter ones to get a solution to the former ones [3]. We obtain a simpler connection in the service scheduling problem as renewal cycle length does not depend on policy.

Stochastic shortest path formulation. We let x_k be the system state at any slot k and t be a special *terminal state* which is hit if there is no new request in a slot. Let x_{k+1} also denote the action in slot k. Clearly, the single stage cost before hitting the terminal state is $(x_k + \psi - x_{k+1})^2 + dx_k$ (see Fig. 1). Given the state-action pair in slot k, (x_k, x_{k+1}), the next state is the terminal state with probability $1-p$ and the terminal cost is $x_{k+1}(x_{k+1} + d)$.

Let $J : [0, \psi] \to \mathbb{R}_+$ be the optimal cost function for the problem. It is the solution of the following Bellman's equation: For all $x \in [0, \psi]$,

$$J(x) = \min_{u\in[0,\psi]} \Big\{ (\psi - u + x)^2 + dx$$
$$+ pJ(u) + (1-p)u(u+d) \Big\}.$$

The optimal cost is attained by a stationary policy of the form $(\pi^*, \pi^*, \ldots)$ where $\pi^*(x)$ minimizes the right hand side in the above equation for all x. For brevity, we use π^* to refer to this policy. Let us define the "k-stage problem" as the one that allows at most $k+1$ service requests. More precisely, here the system is *forced to enter*the terminal state after $k+1$ service requests if it has not already done so. Let $J_k(\cdot)$ be the optimal cost function of the k-stage problem. Clearly,

$$J_0(x) = \min_u \Big\{ (\psi - u + x)^2 + dx + u(u+d) \Big\} \quad (3)$$

and for $k \geq 1$,

$$J_k(x) = \min_u \Big\{ (\psi - u + x)^2 + dx + pJ_{k-1}(u)$$
$$+ (1-p)u(u+d) \Big\}. \quad (4)$$

We can express $J(\cdot)$ as the limit of $J_k(\cdot)$ as k approaches infinity. Furthermore, we can express the desired optimal policy also as the limit of the optimal controls of k-stage problems (i.e., optimal actions in (3)-(4)). This is the approach we follow to arrive at the optimal scheduling policy.

A. Optimal Policy

We show that the optimal policy is either piece-wise linear or linear in the pending service depending on the parameters ψ, p and d. If $d > 2\psi(1-p)$, the former case arises, and the successive constituent line segments are characterized by sequences $a_i^*, b_i^*, i \geq 0$ defined below. These sequences converge to a_∞ and b_∞, respectively (see Lemma 3.2(a) and (b)). If $d \leq 2\psi(1-p)$, the optimal policy is linear and is characterized by a_∞ and b_∞. For $i \geq 0$, let us define the following sequences

$$a_i^* = \begin{cases} 1, & \text{if } i = 0 \\ 1 - \frac{p}{1+a_{i-1}^*}, & \text{otherwise} \end{cases} \quad (5)$$

$$b_i^* = \begin{cases} 2p\psi + d, & \text{if } i = 0 \\ \frac{p(2a_{i-1}^*\psi + b_{i-1}^*)}{1+a_{i-1}^*} + d, & \text{otherwise.} \end{cases} \quad (6)$$

Let us further define

$$x_i = \begin{cases} \frac{b_0^*}{2} - \psi, & \text{if } i = 0 \\ \frac{2(1+a_i^*)x_{i-1} + b_i^*}{2} - \psi, & \text{otherwise.} \end{cases} \quad (7)$$

If $d > 2\psi(1-p)$, the sequence, $x_i, i \geq 0$, is strictly increasing (see Lemma 3.3(b)). It consists of the points where the optimal policy changes it slope. We begin with formally stating the properties of the above sequences.

Lemma 3.2: (a) The sequence $a_k^*, k \geq 0$ converges to $a_\infty := \sqrt{1-p}$.

(b) The sequence $b_k^*, k \geq 0$ converges to

$$b_\infty := \frac{2p\psi}{1+\sqrt{1-p}} + \frac{d}{\sqrt{1-p}}.$$

(c) If $2\psi(1-p) \geq d$, then $b_i^* \leq 2p\psi + d \leq 2\psi$ for all $i \geq 0$ and, so $b_\infty \leq 2\psi$.

(d) If $2\psi(1-p) < d$, then $b_i^* \geq 2p\psi + d > 2\psi$ for all $i \geq 0$ and, so $b_\infty \geq 2\psi$.

Proof: See Section C in supplementary material. ■

Lemma 3.3: (a) $\frac{x+\psi-\frac{b_\infty}{2}}{(1+a_\infty)} < \psi$ for all $0 \leq x \leq \psi$.

(b) If $2\psi(1-p) < d$, then $x_0 > 0$ and the sequence $x_k, k \geq 0$ is strictly increasing. Furthermore, there exists a $k \geq 0$ such that $x_k > \psi$.

Proof: See Section D in supplementary material. ■

The optimal scheduling policy is as follows.

Theorem 3.1: (a) If $d \leq 2\psi(1-p)$,

$$\pi^*(x) = \frac{x+\psi-\frac{b_\infty}{2}}{(1+a_\infty)}.$$

(b) If $2\psi(1-p) < d$,

$$\pi^*(x) = \begin{cases} 0, & \text{if } x \in [0, x_0] \\ \frac{2(x+\psi)-b_l^*}{2(1+a_l^*)}, & \text{if } x \in [x_l, x_{l+1}], l = 0, \ldots, K-1 \\ \frac{2(x+\psi)-b_K^*}{2(1+a_K^*)}, & \text{if } x \in [x_K, \psi] \end{cases}$$

where $K = \min\{k : x_{k+1} > \psi\}$.[4]

Proof: See Appendix A-A. ■

Remark 3.2:

(1) When $2\psi(1-p) \geq d$, the optimal scheduling policy is a linear function.

(2) When $2\psi(1-p) < d$, the optimal policy is a piecewise linear function with progressively increasing slopes, $0, \frac{1}{1+a_0^*}, \ldots, \frac{1}{1+a_K^*}$.

(3) When $2\psi(1-p) < d$, assuming that the first user sees zero pending service, it defers zero service to the next slot. So, the second user also, irrespective of when it arrives, sees zero pending service, and behaves similarly. We can repeat this argument to conclude that no user defers any service in this case.

(4) In the deterministic case (i.e., when $p = 1$), $2\psi(1-p) < d$ for any $d > 0$. Hence, the optimal scheduling policy is a piecewise linear function. However, as explained above, no user defers any service in this case.

The second remark implies that the optimal policy is convex and increasing in the pending service. The following theorem shows that the optimal cost is also a convex increasing function. The proof consists of inductively showing that $J_k(\cdot), k \geq 0$, are convex increasing and so is their limit.

Theorem 3.2: The optimal cost function $J(x)$ is an increasing convex function.

Proof: See Appendix A-B. ■

IV. Nash Equilibrium for Selfish Agents

In this section we provide a Nash equilibrium for the non-cooperative game among the selfish agents (see Section II). Specifically, we look at symmetric Nash equilibria where each agent's strategy is a piece-wise linear function of the remaining demand of the previous player.

Let $C : [0, \psi] \to \mathbb{R}_+$ give the optimal cost for a player as a function of the pending demand given that all other players use strategy $\pi' : [0, \psi] \to [0, \psi]$. Clearly, $C(x)$ is given by the following equation for all $x \in [0, \psi]$.

$$C(x) = \min_{u \in [0,\psi]} \{(\psi - u)(\psi - u + x) + u(u + d + p(\psi - \pi'(u)))\}$$

We call $\bar{\pi}' = (\pi,'\pi,' \ldots)$ a symmetric nash equilibrium if $\pi'(x)$ attain the optimal cost in the above optimization problem for all x, i.e., if

$$\pi'(x) \in \arg\min_{u \in [0,\psi]} \{(\psi - u)(\psi - u + x) + u(u + d + p(\psi - \pi'(u)))\},$$

for all $x \in [0, \psi]$. We characterize one such nash equilibrium in the following. As in section III we define k-stage problems, where the tagged player has atmost k service requests after it, before the terminal state is hit. Let $C_k(\cdot)$ be the tagged users optimal cost in the k-stage problem and $\pi'_k(\cdot)$ be the corresponding optimal strategy. Then

$$C_0(x) = \min_{u \in [0,\psi]} \{(\psi - u)(\psi - u + x) + u(u + d)\} \quad (8)$$

and for all $k \geq 1$,

$$C_k(x) = \min_{u \in [0,\psi]} \{(\psi - u)(\psi - u + x) + u(u + d + p(\psi - \pi'_{k-1}(u)))\}. \quad (9)$$

We can see $C(x)$ as the limit of $C_k(x)$ as k approaches infinity. Furthermore, the limit of the optimal strategy of k-stage problems yield a symmetric Nash equilibrium.

A symmetric Nash equilibrium: We show that, similar to the optimal policy, a Nash equilibrium policy is also either piece-wise linear (not necessarily continuous) or linear in the pending service. If $d > (2-p)\psi$, the former case arises, and the successive constituent line segments are characterized by sequences $a'_i, b'_i, i \geq -1$ defined below. These sequences converge to a'_∞ and b'_∞, respectively (see Lemma 4.1(a) and (b)). If $d \leq (2-p)\psi$, the Nash equilibrium policy is $a'_\infty x + b'_\infty$, where x is the pending service. For $k \geq -1$,

$$a'_k = \begin{cases} 0, & \text{if } k = -1 \\ \frac{1}{4-2pa'_{k-1}}, & \text{otherwise} \end{cases} \quad (10)$$

$$b'_k = \begin{cases} 0, & \text{if } k = -1 \\ \frac{(2-p)\psi - d + pb'_{k-1}}{4-2pa'_{k-1}}, & \text{otherwise} \end{cases} \quad (11)$$

We first state a few properties of the above sequences.

[4] The cases $x \in [x_l, x_{l+1}]$ and $x \in [x_K, \psi]$ do not arise if $x_0 > \psi$ and the former does not arise also when $x_0 \leq \psi < x_1$.

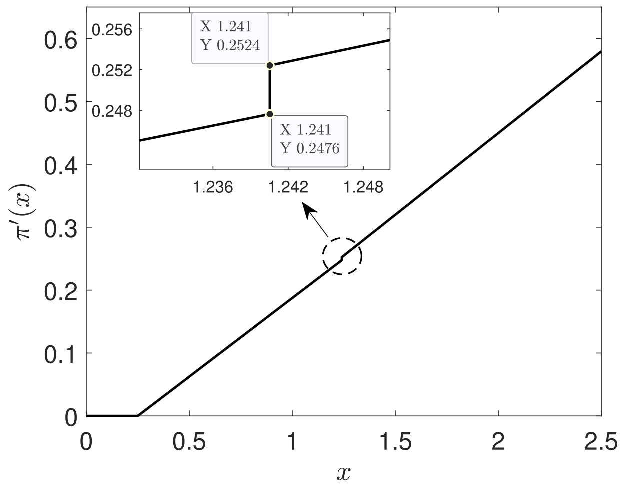

Fig. 2. The Nash equilibrium policy for $p = 0.3, d = 4.5, \psi = 2.5$. The Nash equilibrium policy is piece-wise linear, and is discontinuous at $x = 1.241$ (see the magnified graph).

Lemma 4.1: (a) The sequence $a'_k, k \geq -1$ converges to

$$a'_\infty := \frac{1}{p} - \frac{\sqrt{4-2p}}{2p}.$$

Also, $a'_\infty < \frac{1}{2}$.

(b) The sequence $b'_k, k \geq -1$ converges to

$$b'_\infty := \frac{a'_\infty((2-p)\psi - d)}{1 - a'_\infty p}.$$

Proof: See Section F in supplementary material. ∎

Further, let us define $x_k, k \geq 0$ as

$$x_k = \begin{cases} d - (2-p)\psi, \text{if } k = 0 \\ (4 - 2pa'_{k-1})x_{k-1} - (2-p)\psi + d - pb'_{k-1}, \text{otherwise.} \end{cases} \tag{12}$$

If $(2-p)\psi < d$, the sequence, $x_i, i \geq 0$, is strictly increasing (see Lemma 4.2(b)). It consists of the points where the Nash policy changes its slope. We begin with formally stating the properties of the above sequences.

Lemma 4.2: (a) $a'_\infty x + b'_\infty < \psi$ for all $0 \leq x \leq \psi$.

(b) If $(2-p)\psi < d$ then $x_0 > 0$ and the sequence $x_k, k \geq 0$ is strictly increasing. Furthermore, there exists a $k \geq 0$ such that $x_k > \psi$.

Proof: See Section G in supplementary material. ∎

Let us also define functions $Q'_k : [0,\psi] \times [0,\psi] \to \mathbb{R}_+, k \geq 0$ as follows:

$$Q'_0(x,u) = (\psi + x - u)(\psi - u) + u(u + d)$$

and for all $k \geq 1$,

$$Q'_k(x,u) = (\psi + x - u)(\psi - u) + u(u + d + p(\psi - a'_{k-1}u - b'_{k-1})).$$

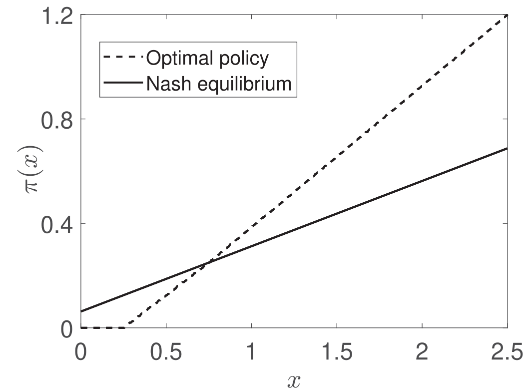

Fig. 3. Optimal and Nash policies for $p = 0.3, \psi = 2.5, d = 5$. Here, the optimal policy is piece-wise linear where as the Nash equilibrium policy is linear. The Nash equilibrium policy defers more service than the optimal policy for small x and vice-versa for large x.

Clearly,

$$C_k(x) = \min_{u \in [0,\psi]} Q'_k(x,u) \; \forall k \geq 0.$$

Finally, we define a another sequence $x'_k, k \geq 0$ as follows: $x'_0 = x_0$ and for each $k \geq 1$, x'_k is the largest root of

$$Q'_{k-1}(x, a'_{k-1}x + b'_{k-1}) = Q'_k(x, a'_k x + b'_k). \tag{13}$$

We show that $x'_k \geq x_k \; \forall k \geq 0$ (See Section H in supplementary material). Hence, following Lemma 4.2(b), there exists a $K \geq 0$ such that $x'_k \leq \psi$ for all $k \leq K$ and $x'_k > \psi$ for all $k > K$. Following theorem gives a symmetric Nash equilibrium.

Theorem 4.1: $\bar{\pi}' = (\pi,' \pi,' \ldots)$ is a symmetric Nash equilibrium where

(a) If $d \leq (2-p)\psi$,

$$\pi'(x) = a'_\infty x + b'_\infty, \forall x \in [0,\psi],$$

(b) If $d > (2-p)\psi$,

$$\pi'(x) = \begin{cases} 0, & \text{if } x \in [0, x'_0] \\ a'_l x + b'_l, & \text{if } x \in [x'_l, x'_{l+1}], l = 0, 1, \ldots, K-1 \\ a'_K x + b'_K, & \text{if } x \in [x'_K, \psi] \end{cases}$$

Proof: See Section H in supplementary material. ∎

We illustrate the Nash equilibrium policy via an example in Fig. 2.

Remark 4.1:

(1) When $(2-p)\psi < d$, the Nash equilibrium policy can be discontinuous as seen in Fig. 2.

(2) When $(2-p)\psi < d$, assuming that the first user sees zero pending service, it defers zero service to the next slot. This argument can be repeated to conclude that no user defers any service in this case. Recall that the optimal policy does not defer demands when $2\psi(1-p) < d$. Hence, when $2\psi(1-p) < d < (2-p)\psi$, the optimal policy will not defer demands whereas the Nash equilibrium policy will (Refer Fig. 3).

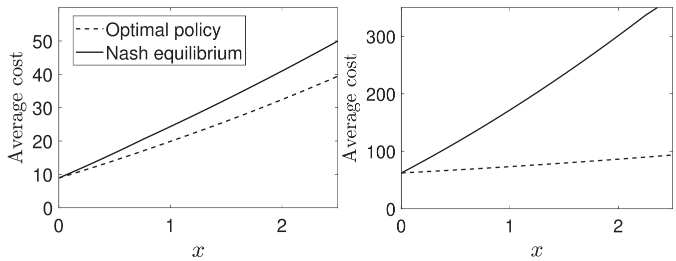

Fig. 4. Average cost vs pending service, x for optimal and Nash policies. We use $p = 0.3, \psi = 2.5, d = 5$ for the left sub plot and $p = 0.9, \psi = 2.5, d = 5$ for the right sub plot. Note that, the difference between the costs of the optimal and Nash equilibrium policies increase with x. Also, the differences are high for the higher value of p.

We compare the optimal and Nash equilibrium policies in Fig. 3. We also compare cost for optimal policy and Nash equilibrium policy with various parameters in Fig. 4. It can be observed that the cost gap between optimal cost and Nash equilibrium increases with an increase in p (refer plots in Fig. 4).

V. Distinct Service Requirements

A. Two Distinct Service Requirements With Markov Arrivals

Let us recall that we assumed i.i.d. job arrivals in Section II. Now we assume that jobs arrive according to a Markov process. We also assume that the service requirements can take two distinct values, ψ_1 and ψ_2. We define $\psi_0 = 0$; a service request ψ_0 in a slot indicates no job arrival in that slot. The transition probability matrix governing the Markov chain is a 3×3 matrix with elements being $p_{i,j}, i, j \in \{0, 1, 2\}$. In other words, if the current slot has an arrival $\psi_i, i \in \{0, 1, 2\}$, then the arrival in next slot is ψ_j with probability $p_{i,j}$. As before, each request can be met in at most two slots, i.e., a fraction of the demand arriving in a slot could be deferred to the next slot. We consider $\psi_2 > \psi_1$ without loss of generality. The cost structure is similar to that in Section II.

We now describe state evolution and the optimal scheduling problem. Let x_k and v_k be defined as in Section II. Let us also redefine e_k to be the demand in slot k. Clearly here, $e_k \in \{\psi_0, \psi_1, \psi_2\}$ and $v_k \in [0, e_k], \forall k \geq 1$. A *scheduling policy* $\overline{\pi} = (\pi_k, k \geq 1)$ is a sequence of functions $\pi_k : \{\psi_0, \psi_1, \psi_2\} \times [0, \psi_2] \to [0, \psi_2]$ such that $\pi_k(e_k, x_k)$ gives differed service from slot k to slot $k+1$. Clearly, $\pi_k(e_k, x_k) \in [0, e_k]$ and

$$x_{k+1} = \pi_k(e_k, x_k) = e_k - v_k$$

The scheduling cost to be minimized is given by (1) in Section II.

Again, we can obtain an equivalent shortest path formulation of the problem as in Section III, we let (e_k, x_k) be the system state at slot k and t be a special *terminal state* which is hit if there is no new request in a slot. Let x_{k+1} and e_{k+1} also denote the action and noise, respectively, in slot k(i.e., an arrival of ψ_0). Let $e_k = \psi_i$, given state action tuple (e_k, x_k, x_{k+1}), the next state is (ψ_1, x_{k+1}) with probability $p_{i,1}$, (ψ_2, x_{k+1}) with probability $p_{i,2}$ and t with probability $1 - p_{i,1} - p_{i,2}$. The single stage cost before hitting the terminal state is $(x_k + e_k - x_{k+1})^2 + dx_k$. Given the state-action pair in slot k, (e_k, x_k, x_{k+1}), the next state is the terminal state with probability $1 - p_{i,1} - p_{i,2}$ and the terminal cost is $x_{k+1}(x_{k+1} + d)$.

Let $J : \{\psi_1, \psi_2\} \times [0, \psi_2] \to \mathbb{R}_+$ be the optimal cost function for the problem. It is the solution of the following Bellman's equation: For all $x \in [0, \psi_2], i \in \{1, 2\}$,

$$J(\psi_i, x) = \min_{u \in [0, \psi_i]} \Big\{ (\psi_i - u + x)^2 + dx + p_{i,1} J(\psi_1, u) + p_{i,2} J(\psi_2, u) + (1 - p_{i,1} - p_{i,2}) u(u + d) \Big\}.$$

Let us define the k-stage problem as in Section III and let $J_k(\cdot, \cdot)$ be the optimal cost function for this problem. Clearly,

$$J_0(\psi_i, x) = \min_{u \in [0, \psi_i]} \Big\{ (\psi_i - u + x)^2 + dx + u(u + d) \Big\} \quad (14)$$

and

$$J_k(\psi_i, x) = \min_{u \in [0, \psi_i]} \Big\{ (\psi_i - u + x)^2 + dx + p_{i,1} J_{k-1}(\psi_1, u) \times + p_{i,2} J_{k-1}(\psi_2, u) + (1 - p_{i,1} - p_{i,2}) u(u + d) \Big\}$$

for $k \geq 1$. We express $J_k(\psi_i, x)$ as

$$J_k(\psi_i, x) = \min_{u \in [0, \psi_i]} \{ (\psi_i - u + x)^2 + dx + a_k^i u^2 + b_k^i u + c_k \}. \quad (15)$$

In the following, we observe that optimal controls for k-stage problems, $\pi_k(\psi_i, x), k \geq 0$, are piecewise linear functions and $J_k(\psi_i, x), k \geq 0$ are piecewise quadratic functions. In particular for all $k \geq 0$, $[0, \psi_2]$ is divided into intervals such that (a_k^i, b_k^i) assume different values in different intervals. We use I_k to denote the total number of such intervals. We use x_k (not to be confused with x_i in Section III) to represent the intervals: $x_{k,0} = 0, x_{k,I_k} = \psi_2$ and $x_{k,j}, 0 < j < I_k$ are the cross-over points. In order to obtain the optimal policy $\pi(\psi_i, x)$ we iteratively compute $\pi_k(\psi_i, x), k \geq 0$.

1) Optimal Policy Computation: We start with computing $\pi_0(\psi_i, x)$. For any $k \geq 1$, given $\pi_{k-1}(\psi_i, x)$, we first compute x_k, then we compute (a_k^i, b_k^i) and then $\pi_k(\psi_i, x)$. We now illustrate this process for $k = 0, 1, 2$. $\underline{x_0, a_0^i, b_0^i}$. From (14), $a_0^i = 1, b_0^i = d$ and $c_0^i = 0, \forall i \in \{1, 2\}$. These values remain constant for all $x \in [0, \psi_2]$. Here $x_0 = \{0, \psi_2\}$. $\underline{\pi_0(\psi_i, x)}$. The solution to (14) is

$$\pi_0(\psi_i, x) = \left[\frac{2(x + \psi_i) - b_0^i}{2(1 + a_0^i)} \right]_0^{\psi_i}.$$

Since, $[a_0^2 = 1$ and $b_0^2 = d$, $\pi_0(\psi_2, x) < \psi_2, \forall x \in [0, \psi_2]$. On the other hand $\pi_0(\psi_1, x) = \psi_1, \forall x \geq a_0^1 \psi_1 + \frac{b_0^1}{2}$; there will be no such point in $[0, \psi_2]$ if $a_0^1 \psi_1 + \frac{b_0^1}{2} > \psi_2$. $\underline{x_1, a_1^i, b_1^i}$. The optimal cost of 0-stage problem can be written as

$$J_0(\psi_2, x) = \begin{cases} (\psi_2 + x)^2 + dx, & \text{if } x \in [0, x^2_{1,0}] \\ \frac{a_0^2}{1+a_0^2} x^2 + \left(\frac{2\psi_2 a_0^2 + b_0^2}{1+a_0^2} + d\right) x & \\ + \frac{a_0^2 \psi_2^2 + b_0^2 \psi_2 - \frac{(b_0^2)^2}{4}}{(1+a_0^2)} + c_0^2, & \text{if } x \in [x^2_{1,0}, \psi_2] \end{cases} \tag{16}$$

$$J_0(\psi_1, x) = \begin{cases} (\psi_1 + x)^2 + dx, & \text{if } x \in [0, x^1_{1,0}] \\ \frac{a_0^1}{1+a_0^1} x^2 + \left(\frac{2\psi_1 a_0^1 + b_0^1}{1+a_0^1} + d\right) x & \\ + \frac{a_0^1 \psi_1^2 + b_0^1 \psi_1 - \frac{(b_0^1)^2}{4}}{(1+a_0^1)} + c_0^1, & \text{if } x \in [x^1_{1,0}, x^1_{1,1}] \\ x^2 + dx + \psi_1(\psi_1 + d), & \text{if } x \in [x^1_{1,1}, \psi_2] \end{cases} \tag{17}$$

where $x^2_{1,0} = \frac{b_0^2}{2} - \psi_2, x^1_{1,0} = \frac{b_0^1}{2} - \psi_1, x^1_{1,1} = a_0^1\psi_1 + \frac{b_0^1}{2}$. We arrange $\{x^i_{1,j}, i = 1, 2, j \geq 0\}$ in ascending order; note that the order is $x^2_{1,0} < x^1_{1,0} < x^1_{1,1}$. These points divide real line into intervals. Using (16) and (17) in (15) with $k = 1$, we obtain,

$$a_1^i = \begin{cases} a^i_{1,0} = 1, & \text{if } u \in [0, x^2_{1,0}] \\ a^i_{1,1} = 1 - \frac{p_{i,2}}{1+a_0^2}, & \text{if } u \in (x^2_{1,0}, x^1_{1,0}] \\ a^i_{1,2} = 1 - \frac{p_{i,1}}{1+a_0^1} - \frac{p_{i,2}}{1+a_0^2}, & \text{if } u \in (x^1_{1,0}, x^1_{1,1}] \\ a^i_{1,3} = 1 - \frac{p_{i,2}}{1+a_0^2}, & \text{if } u \in (x^1_{1,1}, \psi_2] \end{cases}$$

$$b_1^i = \begin{cases} b^i_{1,0} = 2(p_{i,1}\psi_1 + p_{i,2}\psi_2) + d, & \text{if } u \in [0, x^2_{1,0}] \\ b^i_{1,1} = \frac{p_{i,2}(2a_0^2\psi_2 + b_0^2)}{1+a_0^2} + 2p_{i,1}\psi_1 + d, & \text{if } u \in (x^2_{1,0}, x^1_{1,0}] \\ b^i_{1,2} = \frac{p_{i,2}(2a_0^2\psi_2 + b_0^2)}{1+a_0^2} + \frac{p_{i,1}(2a_0^1\psi_2 + b_0^1)}{1+a_0^1} \quad +d, & \\ & \text{if } u \in (x^1_{1,0}, x^1_{1,1}] \\ b^i_{1,3} = \frac{p_{i,2}(2a_0^2\psi_2 + b_0^2)}{1+a_0^2} + d \quad , & \text{if } u \in (x^1_{1,1}, \psi_2] \end{cases}$$

We now discard values in the above ordered sequence that are not in $[0, \psi_2]$ and add 0 and ψ_2 to obtain sequence $x_1 = \{x_{1,j}, 0 \leq j \leq I_1\}$. Notice that each $x_{1,j}$ $(j < I_1)$ has corresponding $a^i_{1,j}$ and $b^i_{1,j}$ that determine the optimal action in the interval $(x_{1,j}, x_{1,j+1}]$ for every $i \in \{1, 2\}$ in $\pi_1(\psi_i, x)$. $\pi_1(\psi_i, x)$. The solution to the 1-stage problem can be written as

$$\pi_1(\psi_i, x) = \begin{cases} 0, & \text{if } x \leq x^i_{2,0} \\ \left[\frac{2(x+\psi_i) - b^i_{1,j}}{2(1+a^i_{1,j})}\right]^{\psi_i}, & \text{if } x \in (x^i_{2,j}, x^i_{2,j+1}] \\ & 0 \leq j < I_1 \end{cases} \tag{18}$$

where $x^i_{2,j}, j < I_1$ are given by

$$\frac{2(x^i_{2,j} + \psi_i) - b^i_{1,j}}{2(1 + a^i_{1,j})} = x_{1,j},$$

and $x^i_{2,I_1} = \psi_2$.

The following lemma states that $\pi_1(\psi_2, x) < \psi_2, \forall x \in [0, \psi_2]$, which implies that, if the current request is ψ_2, it is not entirely deferred to the next slot. We use this property in deriving the optimal policy for 2-stage problem. We later argue that the optimal policies for k-stage problems for all k have this property.

Lemma 5.1: $\pi_1(\psi_2, x) < \psi_2, \forall x \in [0, \psi_2]$.

Proof: See Section I in supplementary material. ∎

x_2, a^i_2, b^i_2 . We next investigate whether $\pi_1(\psi_1, x) = \psi_1$ for some $x \in [0, \psi_2]$, i.e., whether $\{x : \pi_1(\psi_1, x) = \psi_1\}$ is non-empty. If this set is non-empty we define $\hat{x}$ to be the least value in the set. Let $x^1_{2,\bar{l}}$ be the largest value in $\{x^1_{2,j}, j \geq 0\}$ smaller than $\hat{x}$. We then set $x^1_{2,\bar{l}+1} = \hat{x}$, $x^1_{2,\bar{l}+2} = \psi_2$ and discard $x^1_{2,j}, j > \bar{l} + 2$. From Lemma 5.1 $\pi_1(\psi_2, x) < \psi_2, \forall x \in [0, \psi_2]$, so we leave the sequence $\{x^2_{2,j}, j \geq 0\}$ unchanged. Next, we arrange $x^i_{2,j}, j \geq 0$ in ascending order. As for the case $k = 1$, these ordered points divide real line in intervals. For any two consecutive points $x^i_{2,j}$ and $x^{i'}_{2,j'}$. there exist unique intervals $[x^1_{2,l}, x^1_{2,l+1}]$ and $[x^2_{2,m}, x^2_{2,m+1}]$ (we define $x^i_{2,-1} := 0$) containing $[x^i_{2,j}, x^{i'}_{2,j'}]$.

$$a_2^i = \begin{cases} 1, & \text{if } l = -1, m = -1 \\ 1 - \frac{p_{i,2}}{1+a^2_{1,m}}, & \text{if } l = -1, m \geq 0 \\ 1 - \frac{p_{i,1}}{1+a^1_{1,l}} - \frac{p_{i,2}}{1+a^2_{1,m}}, & \text{if } 0 \leq l \leq \bar{l}, m \geq 0 \\ 1 - \frac{p_{i,2}}{1+a^2_{1,m}}, & \text{if } l = \bar{l} + 1, m \geq 0 \end{cases}$$

$$b_2 = \begin{cases} 2(p_{i,1}\psi_1 + p_{i,2}\psi_2) + d, & \text{if } l = -1, m = -1 \\ \frac{p_{i,2}(2a^2_{1,m}\psi_2 + b^2_{1,m})}{1+a^2_{1,m}} + 2p_{i,1}\psi_1 + d, & \text{if } l = -1, m \geq 0 \\ \frac{p_{i,2}(2a^2_{1,m}\psi_2 + b^2_{1,m})}{1+a^2_{1,m}} + \frac{p_{i,1}(2a^1_{1,m}\psi_2 + b^1_{1,m})}{1+a^1_{1,m}} + d, & \\ & \text{if } 0 \leq l \leq \bar{l}, m \geq 0 \\ \frac{p_{i,2}(2a^2_{1,m}\psi_2 + b^2_{1,m})}{1+a^2_{1,m}} + d, & \text{if } l = \bar{l} + 1, m \geq 0 \end{cases}$$

We again discard values in the above ordered sequence that are not in $[0, \psi_2]$ and add 0 and ψ_2 to obtain another ordered sequence $x_2 = \{x_{2,j}, 0 \leq j \leq I_2\}$. For illustration, we show x_1, x_2, π_0 and π_1 for an example with i.i.d. arrivals in Fig. 5. Next, we determine the solution to the 2-stage problem, $\pi_2(\psi_i, x), i = 1, 2$. We continue this iterative process for $k \geq 3$. We can stop at a $k \geq 1$ if (x_k, a^i_k, b^i_k) are identical to $(x_{k-1}, a^i_{k-1}, b^i_{k-1})$. The limiting policy, $\pi_k(\psi_i, x), i = 1, 2$, is the desired optimal policy. We formalize this process in Algorithm 1. We find that $\pi_k(\psi_i, x), i = 1, 2$ gets closer to the optimal policy as k increases. So, to obtain an approximately optimal policy, we can stop at a sufficiently large k. We illustrate optimal policies for two examples with i.i.d. arrivals in Fig. 6.

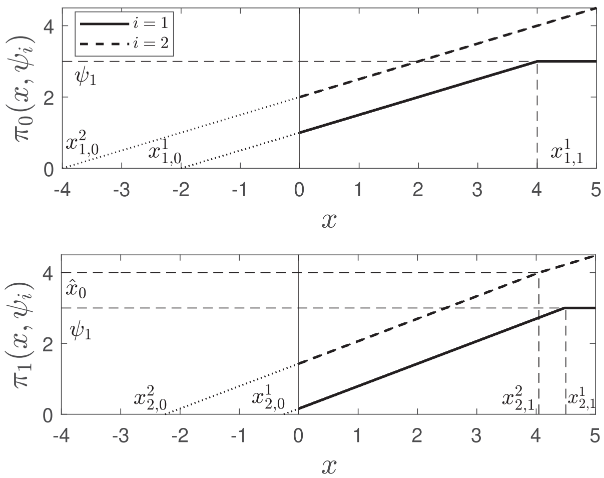

Fig. 5. Illustration of optimal policy for 0-stage and 1-stage problem with i.i.d. arrivals. $\pi_0(\psi_i, x)$ vs x and $\pi_1(\psi_i, x)$ vs x for $p_1 = 0.8, p_2 = 0.05, \psi_1 = 3, \psi_2 = 5$ and $d = 0.5$. Note that $x^2_{1,0}, x^1_{1,0}, x^2_{2,0}, x^1_{2,0}$ are negative. Hence $x_1 = \{0, x^1_{1,1}, \psi_2\}$ and $x_2 = \{0, x^2_{2,1}, x^1_{2,1}, \psi_2\}$.

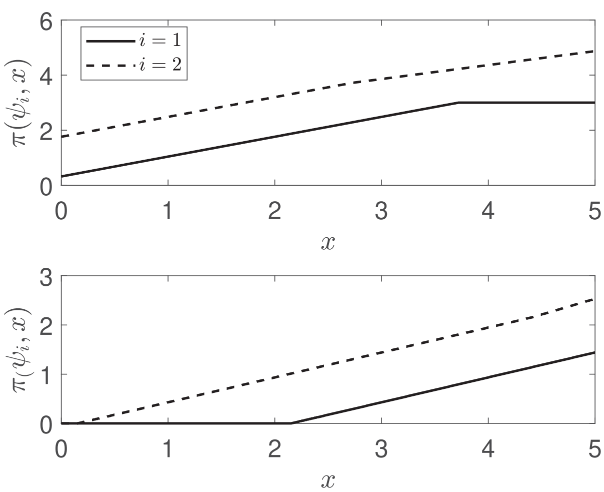

Fig. 6. Illustration of optimal policies for i.i.d arrivals. We use $p_1 = 0.8, p_2 = 0.05, \psi_1 = 3, \psi_2 = 5$. $d = 0.5$ for the upper sub-plot and $d = 5$ for the lower sub-plot.

B. Computational Complexity of Algorithm 1

Algorithm 1 is an iterative algorithm that iteratively calculates $(x_k, a^i_k, b^i_k),\ k \geq 0$, where (x_k, a^i_k, b^i_k) characterize the piece-wise linear optimal policy of the k-stage problem. Recall that $I_k,\ k = 0, 1, \ldots$ denote the maximum number of line segments in the optimal policy of the $k-$stage problem. Note that $I_0 = 2$ and, for $k \geq 1$,

$$I_k \leq 2(I_{k-1} + 1).$$

We get an inequality as "order" function may remove a few values that are outside $[0, \psi_2]$. We thus see that

$$I_k \leq \sum_{i=1}^{k+1} 2^i, \tag{19}$$

for all $k \geq 0$. In k-th iteration of the algorithm $O(I_{k-1})$ computation is needed obtain x_k, and further $O(I_k)$ computation is needed to obtain a^i_k and b^i_k. So, if we run the algorithm for K iterations (see Algorithm 1 for the stopping criteria), then the complexity is $O(\sum_{k=1}^{K}(I_{k-1} + I_k))$. Consequently, using (19), the worst case computational complexity of the algorithm $O(2^{K+1})$. If instead of two we have N distinct service

---

Algorithm 1. (Two Distinct Arrivals)

Input: $p_{1,1}, p_{1,2}, p_{2,1}, p_{2,2}, \psi_1, \psi_2, d$
$a^i_{n,-1} = \infty, b^i_{n,-1} = 0\ \forall n \geq 0, i \in \{1, 2\}$
$n = 0$
$x_{0,0} = 0, x_{0,1} = \psi_2, I_0 = 1$
$a^i_{0,0} = 1, b^i_{0,0} = d, i \in \{1, 2\}$
do
$n = n + 1$
for $i = 1 : 2$ **do**
  for $j = 0 : I_{n-1} - 1$ **do**

$$x^i_{n,j} = \frac{2(1 + a^i_{n-1,j})x_{n-1,j} + b^i_{n-1,j}}{2} - \psi_i$$

  end for
end for

$$\bar{l} = \max\{j : x_{n-1,j} < \psi_1\}$$
$$x^1_{n,\bar{l}+1} = \frac{2(1 + a^1_{n-1,\bar{l}})\psi_1 + b^1_{n-1,\bar{l}}}{2} - \psi_1$$
$$(x_{n,0}, \ldots, x_{n,I_n}) = \text{order}(x^1_{n,0}, \ldots, x^1_{n,\bar{l}+1}, x^2_{n,0}, \ldots, x^2_{n,I_{n-1}-1}, 0, \psi_2)$$

▷This function removes the values outside $[0, \psi_2]$ and puts the remaining in ascending order.
for $j = 0 : I_n - 1$ **do**
  for $i = 1 : 2$ **do**

$$j_i = \begin{cases} -1, & \text{if } x^i_{n,0} > x_{n,j} \\ \max\{l : x^i_{n,l} \leq x_{n,j}\}, & \text{otherwise} \end{cases}$$

  end for
  for $i = 1 : 2$ **do**

$$a^i_{n,j} = 1 - \frac{p_{i,1}}{1 + a^1_{n-1,j_1}} 1_{\{j_1 \leq \bar{l}\}} - \frac{p_{i,2}}{1 + a^2_{n-1,j_2}}$$
$$b^i_{n,j} = \frac{p_{i,1}(2\psi_1 a^1_{n-1,j_1} + b^1_{n-1,j_1})}{1 + a^1_{n-1,j_1}} 1_{\{j_1 \leq \bar{l}\}} + \frac{p_{i,2}(2\psi_2 a^2_{n-1,j_2} + b^2_{n-1,j_2})}{1 + a^2_{n-1,j_2}} + d$$

  end for
end for
do while$(x_n, a^j_n, b^j_n) \neq (x_{n-1}, a^j_{n-1}, b^j_{n-1}), \forall j \in \{1, 2\}$
Output

$$\pi(\psi_i, x) = \begin{cases} 0, & \text{if } x \leq x^i_{n,0} \\ \left[\frac{2(x+\psi_i) - b^i_{n,j}}{2(1 + a^i_{n,j})}\right]^{\psi_i}, & \text{if } x \in (x^i_{n,j}, x^i_{n,j+1}] \\ & 0 \leq j < I_n \end{cases}$$

---

requirements then the computation complexity increases to $O(N2^{K+1})$.

In view of the exponential complexity of Algorithm 1, we propose an approximate policy in Section V-C1 and also provide its performance bound. The computational complexity of this approximate policy is $O(NK)$ for N distinct arrivals case.

C. General Service Requirements

We now further extend the system model in Section V-A to allow more than two distinct job sizes. We assume that, in each slot an agent with demand ψ_i $(i = 1, 2, \ldots, N)$ arrives with probability p_i and there is no arrival with probability $1 - \bar{p}$ where $\bar{p} := \sum_{i=1}^{N} p_i$. Without loss of generality we assume that ψ_is are monotonically increasing.

Let us see the stochastic shortest path formulation of this problem. Let $J : \{\psi_1, \ldots, \psi_N\} \times [0, \psi_N] \to \mathbb{R}_+$ be the optimal cost function and $\pi : \{\psi_1, \ldots, \psi_N\} \times [0, \psi_N] \to [0, \psi_N]$ be the optimal policy for the problem ($\pi(\psi_i, \cdot) \in [0, \psi_i]$ for all i). The optimal cost function is solution of the following Bellman's equation: For all $x \in [0, \psi_N], i \in \{1, 2, .., N\}$,

$$J(\psi_i, x) = \min_{u \in [0, \psi_i]} \left\{ (\psi_i - u + x)^2 + dx + \sum_{j=1}^{N} p_j J(\psi_j, u) + (1 - \bar{p}) u(u + d) \right\}$$

Let us also define the k-stage problems with $J_k(\cdot, \cdot)$s being the optimal cost functions and $\pi_k(\cdot, \cdot)$s being the optimal controls as in Section III. Clearly,

$$J_0(\psi_i, x) = \min_{u \in [0, \psi_i]} \left\{ (\psi_i - u + x)^2 + dx + u(u + d) \right\} \quad (20)$$

and

$$J_k(\psi_i, x) = \min_{u \in [0, \psi_i]} \left\{ (\psi_i - u + x)^2 + dx + \sum_{j=1}^{N} p_j J_{k-1}(\psi_j, u) + (1 - \bar{p}) u(u + d) \right\}. \quad (21)$$

for $k \geq 1$. The following lemma shows that the optimal cost is a convex increasing function in both of its arguments. We use convexity of the optimal cost in Lemma 5.3 to derive a lower bound on it.

Lemma 5.2: For all $k \geq 0$, $J_k(\psi_i, x)$ is convex in both its arguments.

Proof: The proof follows exactly same arguments as the proof of Theorem 3.2. ■

As in Section V-A, we can iteratively obtain $\pi_k(\cdot, \cdot), k \geq 0$, yielding $\pi(\cdot, \cdot)$. In other words, we can extend the arguments in Section V-A and Algorithm 1 for handling any finite number of distinct job sizes. As the Algorithm 1 is of exponential complexity, instead, we propose a closed form approximate policy $\bar{\pi}(\cdot, \cdot)$ and also derive its performance bound.

1) An Approximately Optimal Policy: Let us define $\bar{\psi} := (\sum_{j=1}^{N} p_j \psi_j)/\bar{p}$ and consider a fictitious problem wherein an agent with demand $\bar{\psi}$ arrives with probability $\bar{p}$ and there is no arrival with probability $1 - \bar{p}$. The optimal cost function for this fictitious problem, $J'(\cdot)$, is solution of the following Bellman's equation.

$$J'(x) = \min_{u \in [0, \bar{\psi}]} \left\{ (\bar{\psi} - u + x)^2 + dx + \bar{p} J'(\bar{\psi}, u) + (1 - \bar{p}) u(u + d) \right\}$$

Moreover, following Theorem 3.1, the optimal policy for this problem, say $\pi'(\cdot)$, is: If $d \leq 2\bar{\psi}(1 - \bar{p})$,

$$\pi'(x) = \frac{x + \bar{\psi} - \frac{b_\infty}{2}}{(1 + a_\infty)}.$$

else

$$\pi'(x) = \begin{cases} 0, & \text{if } x \in [0, x_0] \\ \frac{2(x+\bar{\psi}) - b_l^*}{2(1+a_l^*)}, & \text{if } x \in [x_l, x_{l+1}], l = 0, \ldots, K-1 \\ \frac{2(x+\bar{\psi}) - b_K^*}{2(1+a_K^*)}, & \text{if } x \in [x_K, \bar{\psi}], \end{cases}$$

where a_i^*s, b_i^*s and x_is are as in (5), (6) and (7), respectively, with ψ and p replaced by $\bar{\psi}$ and $\bar{p}$, respectively. Also, $a_\infty = \lim_{i \to \infty} a_i^*$, $b_\infty = \lim_{i \to \infty} b_i^*$ and $K = \min\{k : x_{k+1} > \bar{\psi}\}$. Our approximate policy for general service requirements is motivated by $\pi'(\cdot)$. Notice that the cost function $J'(\cdot)$ and the policy $\pi'(\cdot)$ implicitly assume that all the agents have demand $\bar{\psi}$ ($\pi'(\cdot) \in [0, \bar{\psi}]$). But, motivated by $J'(\cdot)$, we can define functions

$$J'(\psi_i, x) = \min_{u \in [0, \psi_i]} \left\{ (\psi_i - u + x)^2 + dx + \bar{p} J'(u) + (1 - \bar{p}) u(u + d) \right\}$$

for all $i = 1, \ldots, N, x \in [0, \psi_N]$. The optimal actions in the above expressions, $\bar{\pi}(\psi_i, x)$, are: For $i = 1, \ldots, N$,

(1) if $d \leq 2\bar{\psi}(1 - \bar{p})$,

$$\bar{\pi}(\psi_i, x) = \left[\frac{x + \psi_i - \frac{b_\infty}{2}}{(1 + a_\infty)} \right]_0^{\psi_i},$$

(2) if $2\bar{\psi}(1 - \bar{p}) < d$,

$$\bar{\pi}(\psi_i, x) = \begin{cases} \left[\frac{2(x+\psi_i) - b_0^*}{2(1+a_0^*)} \right]_0^{\psi_i}, & \text{if } x \in [0, x_1], \\ \left[\frac{2(x+\psi_i) - b_l^*}{2(1+a_l^*)} \right]_0^{\psi_i}, & \text{if } x \in [x_l, x_{l+1}], \; l = 1, \ldots, K-1 \\ \left[\frac{2(x+\psi_i) - b_K^*}{2(1+a_K^*)} \right]_0^{\psi_i}, & \text{if } x \in [x_K, \psi_N], \end{cases}$$

where $K = \min\{k : x_{k+1} > \psi_N\}$.

We propose to use $\bar{\pi}(\cdot, \cdot)$ as an approximate policy for our problem.

2) Performance Bound: Let $\bar{J} : \{\psi_1, \ldots, \psi_N\} \times [0, \psi_N] \to \mathbb{R}_+$ be the cost function of the policy $\bar{\pi}(\cdot, \cdot)$. It satisfies the fixed point equation

$$\bar{J}(\psi_i, x) = (\psi_i - \bar{\pi}(\psi_i, x) + x)^2 + dx + \sum_{l=1}^{N} p_l \bar{J}(\psi_l, \bar{\pi}(\psi_i, x)) + (1 - \bar{p})\bar{\pi}(\psi_i, x)(\bar{\pi}(\psi_i, x) + d).$$

We would like to bind $\bar{J} - J$. But we have not been able to assess the optimal cost function J. We instead show that $J'(\cdot,\cdot)$ is a lower bound on $J(\cdot,\cdot)$ and bind $\bar{J} - J'$.

Lemma 5.3: For all $i = 1, \ldots, N, x \in [0, \psi_N]$,

$$\bar{J}(\psi_i, x) \geq J(\psi_i, x) \geq J'(\psi_i, x)$$

Proof: See Section J in supplementary material. ■

Lemma 5.2 immediately implies that

$$\bar{J}(\psi_i, x) - J(\psi_i, x) \leq \bar{J}(\psi_i, x) - J'(\psi_i, x).$$

We now focus on binding $\bar{J} - J'$. Let us define

$$D := \sum_j p_j |\psi_j - \bar{\psi}|, \quad (22)$$

$$B := \sum_j p_j (\psi_j - \bar{\psi})^2, \quad (23)$$

$$\text{and } A := \sum_{j:\psi_j < \bar{\psi}} p_j (\bar{\psi} - \psi_j)(2\psi_N + \psi_j - \bar{\psi}). \quad (24)$$

Following is the desired bound.

Theorem 5.1: For all $i = 1, \ldots, N, x \in [0, \psi_N]$,

$$\bar{J}(\psi_i, x) - J'(\psi_i, x) \leq \frac{A + B + \bar{\psi} D}{1 - \bar{p}}.$$

Proof: See Appendix B-A. ■

Remark 5.1:

(1) Note that B and D are measures of variability of the service requests. Moreover, A also depends on variability of the service requests. All these parameters reduce to 0 for single service requirement problem of Section III. This is expected as the approximately optimal policy is identical to the optimal policy in this special case.

(2) Note that the bound on $\bar{J}(\psi_i, x) - J'(\psi_i, x)$ diverges as $\bar{p}$ approaches 1. But the time-averaged scheduling cost of any policy is $\bar{p}(1 - \bar{p}) \sum_{i=1}^{N} p_i J(\psi_i, x)$ (see the first paragraph of Section III). Hence the time-averaged service cost of policy $\bar{\pi}(\cdot,\cdot)$ exceeds the optimal time averaged service cost by at most $(A + B + \bar{\psi} D)\bar{p}^2$. From the previous remark, this quantity reduces as variability of the service requests reduces.

We illustrate the average optimal costs, their lower bounds and the average costs of the approximate policies for two examples in Fig. 7. Here we keep ψ_1 fixed and vary ψ_2 from ψ_1 to $10\psi_1$. We observe the costs of using the optimal policy and approximate policy are close for smaller values of ψ_2. For larger values of ψ_2 also, the bound suggested by Theorem 5.1 is pretty loose and the actual costs of the approximate policy are much lower.

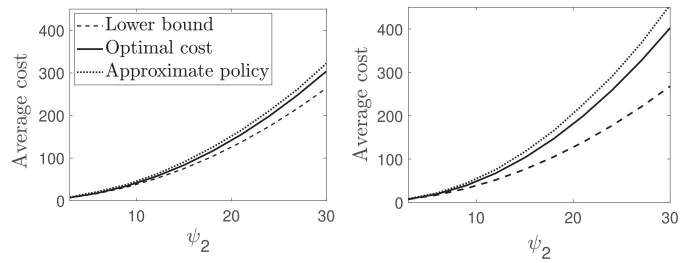

Fig. 7. Optimal average costs, their lower bounds and the costs of approximate policies for two examples. For both the examples, we fix $\psi_1 = 3, p_1 = 0.4, p_2 = 0.5$ and vary ψ_2 from ψ_1 to $10\psi_1$. We consider $d = 1$ for the first example (left sub-plot) and $d = 150$ for the second example (right sub-plot).

VI. Tuning System Parameters

Given the parameters ψ, p and d, optimal scheduling results in certain maximum offered service per slot, average service cost, and average deferred service. The service provider may want to regulate each of these. For instance, it may want to limit the maximum amount of service in any slot owing to practical considerations or keep the average deferred service to a prescribed value. It can achieve these goals by suitably tuning d, the waiting cost per unit service. For instance, increasing d may lower the average deferred service.

In the non-cooperative game paradigm, the agents dictate the amount of service they should receive based on their respective service and waiting costs. Expectedly, the aggregate service and waiting costs in the non-cooperative setting are higher than those in the optimal scheduling scenario. However, the service provider can attain optimal service cost and average delay by tuning a few parameters, as is widely done. For instance, the authors in [27] propose a punishment mechanism to enhance trust and cooperative behavior among the agents. In [28], they model cluster identification in an e-commerce system as a Stackelberg game in which a leader group tunes the payoff functions to elicit cluster information from the followers.

In our formulation, the service provider can also scale the service costs, say by c, or offer incentives for deferring the service, say δ per unit of service, or can do both to induce desired service deferral behavior by the agents. In particular, exercising both the above options makes the effective waiting cost per unit of service $\frac{(d-\delta)}{c}$. The service providers can suitably choose c and δ, attaining optimal service and waiting costs.

VII. Conclusion

We studied service scheduling in slotted systems with Bernoulli request arrivals, quadratic service costs, linear waiting costs, and service delay guarantee of two slots. We derived optimal policy in closed form in case of requests with identical service requirements in Theorem 3.1. In competing requests, all with identical service requirements, we also gave a symmetric Nash equilibrium. We gave an algorithm (Algorithm 1) to compute the optimal policy in the case of Markov arrivals with two different service requirements. We proposed an approximate policy for general service requirements and derived its performance bound.

We consider that all the jobs have same maximum allowed sojourn time of two slots which is quite restrictive. Allowing arbitrary limits on sojourn times requires us to keep track of service requirements with diffident deadlines in each slot. This leads to the so called curse of dimensionality. We leave this extension for our future work. Our future work also entails examining service scheduling problems with convex, but not necessarily quadratic, service costs. We would also like to consider scenario with arbitrary job arrivals, service requirements and permissible sojourn times, i.e., where statistics of these variables are unknown.

VIII. Appendix A

A. Proof of Theorem 3.1

Let us first recall the notions of k-stage problems and k-stage optimal cost functions J_k. For all $k \geq 0$, we will express J_k as

$$J_k(x) = \min_{u \in [0,\psi]} \left\{ (\psi - u + x)^2 + dx + a_k u^2 + b_k u + c_k \right\}. \quad (25)$$

Comparing with (3), $a_0 = 1, b_0 = d, c_0 = 0$. For $k \geq 1$, we consider the following two cases separately.

(a) $2\psi(1-p) \geq d$: We iteratively show that $b_k \leq 2\psi$ for all $k \geq 0$. For $k = 0$, $b_0 \leq 2\psi(1-p) \leq 2\psi$, so the claim trivially holds. Assume that it holds for some $k \geq 0$. Considering the form of J_k in (25), the optimal policy for the k-stage problem

$$\pi_k(x) = \min\left\{ \max\left\{ \frac{2(x+\psi) - b_k}{2(1+a_k)}, 0 \right\}, \psi \right\}. \quad (26)$$

We need the following lemma (for proof see Section E in the supplementary material).

Lemma A.-A.1: If $d \leq 2\psi(1-p)$ then, $\pi_k(x) < \psi, \forall k \geq 0$.

Using Lemma A-A.1, (26) can be written as

$$\pi_k(x) = \max\left\{ \frac{2(x+\psi) - b_k}{2(1+a_k)}, 0 \right\},$$

and hence

$$J_k(x) = \begin{cases} \left(\frac{2a_k(x+\psi)+b_k}{2(1+a_k)}\right)^2 + dx + a_k\left(\frac{2(x+\psi)-b_k}{2(1+a_k)}\right)^2 \\ \qquad + b_k\left(\frac{2(x+\psi)-b_k}{2(1+a_k)}\right) + c_k, \text{if } 2(x+\psi) > b_k \\ (\psi + x)^2 + dx + c_k, \qquad \text{otherwise.} \end{cases}$$

Therefore, using (4),

$$a_{k+1} = \begin{cases} a_{k+1,0} = 1, & \text{if } 2(u+\psi) \leq b_k \\ a_{k+1,1} = 1 - \frac{p}{a_k+1}, & \text{otherwise} \end{cases} \quad (27)$$

$$b_{k+1} = \begin{cases} b_{k+1,0} = 2p\psi + d, & \text{if } 2(u+\psi) \leq b_k \\ b_{k+1,1} = \frac{p(2a_k\psi + b_k)}{1+a_k} + d, & \text{otherwise.} \end{cases} \quad (28)$$

Since we have assumed $b_k \leq 2\psi$, a_{k+1} and b_{k+1} only assume values $a_{k+1,1}$ and $b_{k+1,1}$, respectively. Further, using exactly same argument as in the proof of Lemma 3.2(c), $b_{k+1} \leq 2p\psi + d \leq 2\psi$. Thus, we can conclude by induction that $b_k \leq 2\psi$ for all $k \geq 0$. We can now see the following.

$a_k \to a_\infty$ *as* $k \to \infty$*:* For this observe that the sequence $a_k, k \geq 0$ is identical to the sequence $a_k^*, k \geq 0$ in Lemma 3.2(a) (see (27)).

$b_k \to b_\infty^*$ *as* $k \to \infty$*:* Observe that $b_0 = d \neq b_0^*$ but for all $k \geq 1$, b_k depends on b_{k-1} in exactly same way in which b_k^* depends on b_{k-1}^* (see (28)). Hence the claim holds following the proof of Lemma 3.2(b).

Further, $b_\infty < 2\psi$ from Lemma 3.2(c) and $\frac{x+\psi-\frac{b_\infty}{2}}{(1+a_\infty)} < \psi$ for all $0 \leq x \leq \psi$ from Lemma 3.3. Hence, the optimal policy is as in Theorem 3.1(a).

(b) $2\psi(1-p) < d$: Let us first recall $b_{k,1}, k \geq 1$, defined earlier (see (28)). Define $\bar{k} = \min\{k : b_{k,1} > 2\psi\}$. From Lemma 3.2$(d)$, $\lim_{k\to\infty} b_{k,1} = b_\infty > 2\psi$. Hence $\bar{k} < \infty$. Clearly, in this case also, a_k and b_k assume unique values for all $k \leq \bar{k}$; $a_0 = 1, b_0 = d$ and $a_k = a_{k,1}, b_k = b_{k,1}$ for $1 \leq k \leq \bar{k}$. But

$$\pi_{\bar{k}}(x) = \begin{cases} 0, & \text{if } x \in [0, x_0'] \\ \frac{2(x+\psi)-b_{\bar{k}}}{2(1+a_{\bar{k}})}, & \text{if } x \in [x_0', \psi] \end{cases}$$

where $x_0' = \frac{b_{\bar{k}}}{2} - \psi$. Notice that

$$b_{\bar{k}} = \frac{p(2a_{\bar{k}-1}\psi + b_{\bar{k}-1})}{1 + a_{\bar{k}-1}} + d \leq 2p\psi + d = b_0^*.$$

Hence $x_0 > x_0'$. So, if $x_0' > \psi$, the case $x \in [x_0', \psi]$ does not arise, and the desired result is obtained. But, if $x_0' \leq \psi$, $\pi_{\bar{k}}(x)$ is a piecewise linear function and $J_{\bar{k}}(x)$ is a piecewise quadratic function. Moreover,

$$\pi_{\bar{k}+1}(x) = \arg\min_{u\in[0,\psi]} \left\{ (\psi - u + x)^2 + dx + a_{\bar{k}+1}u^2 + b_{\bar{k}+1}u + c_{\bar{k}+1} \right\},$$

where

$$a_{\bar{k}+1} = \begin{cases} a_{\bar{k}+1,0} = a_0^*, & \text{if } u \leq x_0' \\ a_{\bar{k}+1,1} = 1 - \frac{p}{a_{\bar{k}}+1}, & \text{otherwise,} \end{cases}$$

$$b_{\bar{k}+1} = \begin{cases} b_{\bar{k}+1,0} = b_0^*, & \text{if } u \leq x_0' \\ b_{\bar{k}+1,1} = \frac{p(2a_{\bar{k}}\psi + b_{\bar{k}})}{1+a_{\bar{k}}} + d, & \text{otherwise.} \end{cases}$$

We now argue that

$$\pi_{\bar{k}+1}(x) = \begin{cases} 0, & \text{if } x \in [0, x_0] \\ \frac{2(x+\psi)-b_0^*}{2(1+a_0^*)}, & \text{if } x \in [x_0, x_1'] \\ \frac{2(x+\psi)-b_{\bar{k}+1,1}}{2(1+a_{\bar{k}+1,1})}, & \text{if } x \in [x_1', \psi] \end{cases}$$

where x_0 is as in (7) and $x_1' = \frac{2(1+a_{\bar{k}+1,1})x_0'+b_{\bar{k}+1,1}}{2} - \psi$. It suffices to show that

$$\left.\frac{2(x+\psi)-b_0^*}{2(1+a_0^*)}\right|_{x=x_0} = 0 \tag{29}$$

$$\text{and } \left.\frac{2(x+\psi)-b_0^*}{2(1+a_0^*)}\right|_{x=x_1'} = \left.\frac{2(x+\psi)-b_{\bar{k}+1,1}}{2(1+a_{\bar{k}+1,1})}\right|_{x=x_1'} = x_0'. \tag{30}$$

Note that (29) follows from the definition of x_0 and equality of the last two terms in (30) follows from the definitions of x_0, x_0' and x_1'. Equality of the first two terms in (30) is equivalent to

$$2x_0'(1+a_{\bar{k}+1,1}) + b_{\bar{k}+1,1} = 2x_0'(1+a_0^*) + b_0^*,$$
$$\text{or, } b_{\bar{k}+1,1} - b_0^* = 2x_0'(a_0^* - a_{\bar{k}+1,1}),$$

which also clearly holds from definitions of a_0^*, b_0^* and x_0'. Further, (30) implies that $x_0 < x_1' < x_1$ (see the definitions of x_0 and x_1). If $x_1' > \psi$, then the case $x \in [x_1', \psi]$ does not arise, and again the desired result is obtained.

Now let the optimal policy for the $\bar{k}+j$-stage problem be

$$\pi_{\bar{k}+j}(x) = \begin{cases} 0, & \text{if } x \in [0, x_0] \\ \frac{2(x+\psi)-b_l^*}{2(1+a_l^*)}, & \text{if } x \in [x_l, x_{1+1}], \\ & l = 0, \ldots, j-2 \\ \frac{2(x+\psi)-b_{j-1}^*}{2(1+a_{j-1}^*)}, & \text{if } x \in [x_{j-1}, x_j'] \\ \frac{2(x+\psi)-b_{\bar{k}+j,j}}{2(1+a_{\bar{k}+j,j})}, & \text{if } x \in [x_j', \psi] \end{cases}$$

where

$$\frac{2(x_{j-1}+\psi)-b_{j-2}^*}{2(1+a_{j-2}^*)} = \frac{2(x_{j-1}+\psi)-b_{j-1}^*}{2(1+a_{j-1}^*)} = x_{j-2} \tag{31}$$

$$\frac{2(x_j'+\psi)-b_{j-1}^*}{2(1+a_{j-1}^*)} = \frac{2(x_j'+\psi)-b_{\bar{k}+j,j}}{2(1+a_{\bar{k}+j,j})} = x_{j-1}', \tag{32}$$

and $x_{j-1} < x_j' < x_j$. If $x_j' > \psi$, we readily have the desired result.[5] If $x_j' \le \psi$, we can follow similar arguments as above to show that

$$\pi_{\bar{k}+j+1}(x) = \begin{cases} 0, & \text{if } x \in [0, x_0] \\ \frac{2(x+\psi)-b_l^*}{2(1+a_l^*)}, & \text{if } x \in [x_l, x_{l+1}], \\ & l = 0, \ldots, j-1 \\ \frac{2(x+\psi)-b_j^*}{2(1+a_j^*)}, & \text{if } x \in [x_j, x_{j+1}'] \\ \frac{2(x+\psi)-b_{\bar{k}+j+1,j+1}}{2(1+a_{\bar{k}+j+1,j+1})}, & \text{if } \in [x_{j+1}', \psi] \end{cases}$$

[5] In this case K is $j-2$ if $x_{j-1} > \psi$ and $j-1$ if $x_{j-1} \le \psi$.

where

$$a_{\bar{k}+j+1,j+1} = 1 - \frac{p}{a_{\bar{k}+j,j}},$$
$$b_{\bar{k}+j+1,j+1} = \frac{p(2a_{\bar{k}+j,j}\psi + b_{\bar{k}+j,j})}{1+a_{\bar{k}+j,j}} + d,$$
$$\text{and } x_{j+1}' = \frac{2(1+a_{\bar{k}+j+1,j+1})x_j' + b_{\bar{k}+j+1,j+1}}{2} - \psi.$$

Indeed it suffices to show that

$$\left.\frac{2(x+\psi)-b_{j-1}^*}{2(1+a_{j-1}^*)}\right|_{x=x_j} = \left.\frac{2(x+\psi)-b_j^*}{2(1+a_j^*)}\right|_{x=x_j} = x_{j-1}. \tag{33}$$

and

$$\left.\frac{2(x+\psi)-b_j^*}{2(1+a_j^*)}\right|_{x=x_{j+1}'} = \left.\frac{2(x+\psi)-b_{\bar{k}+j+1,j+1}}{2(1+a_{\bar{k}+j+1,j+1})}\right|_{x=x_{j+1}'} = x_j'. \tag{34}$$

Equality of the last two terms in (33) follows from the definition of x_j. Then, equality of the first two terms is equivalent to

$$2x_{j-1}(a_{j-1}^* - a_j^*) = b_j^* - b_{j-1}^*,$$

$$\text{or } 2x_{j-1}\left(\frac{1}{1+a_{j-1}^*} - \frac{1}{1+a_{j-2}^*}\right) = \frac{2a_{j-1}^*\varphi + b_{j-1}^*}{1+a_{j-1}^*} - \frac{2a_{j-2}^*\varphi + b_{j-2}^*}{1+a_{j-2}^*},$$

$$\text{or } 2x_{j-1}(a_{j-2}^* - a_{j-1}^*) = 2\varphi(a_{j-1}^* - a_{j-2}^*) + (1+a_{j-2}^*)b_{j-1}^* - (1+a_{j-1}^*)b_{j-2}^*,$$

or

$$2(x_{j-1}+\varphi)(a_{j-2}^* - a_{j-1}^*) = (1+a_{j-2}^*)b_{j-1}^* - (1+a_{j-1}^*)b_{j-2}^*,$$

or

$$(2(1+a_{j-1}^*)x_{j-2} + b_{j-1}^*)a_{j-2}^* - (2(1+a_{j-2}^*)x_{j-2} + b_{j-2}^*) a_{j-1}^* = (1+a_{j-2}^*)b_{j-1}^* - (1+a_{j-1}^*)b_{j-2}^*, \tag{35}$$

or

$$2x_{j-2}(a_{j-2}^* - a_{j-1}^*) = b_{j-1}^* - b_{j-2}^*, \tag{36}$$

where we arrive at (35) using (31). Furthermore, (36) also follows from (31). Similarly, in (34), equality of the last two terms follows from the definition of x_{j+1}', and equality of the first two terms can be shown using (32) and following similar steps as above. Moreover, (33) and (34) imply that $x_j < x'_{j+1} < x_{j+1}$ (see the definition of x_{j+1}). If $x'_{j+1} > \psi$, the policy $\pi_{\bar{k}+j+1}(\cdot)$ is the desired optimal policy. We need to continue the above iteration up to the least j such that $x'_j > \psi$; this j equals $K+2$ if $x_{j-1} > \psi$ and $K+1$ if $x_{j-1} \le \psi < x_j$.

Finally, observe that the so obtained policy is a piecewise linear function with progressively increasing slopes, $0, \frac{1}{1+a_0^*}, \ldots, \frac{1}{1+a_K^*}$. All these slopes are less than 1 implying that $0 \le \pi_{\bar{k}+j}(x) < \psi$ for all $j \ge 0$ and $x \in [0, \psi]$. Hence the optimal policy is as claimed in Theorem 3.1(b).

B. Proof of Theorem 3.2

We inductively show that $J_k(x), k \ge 0$ are convex and increasing. Let us first define functions $Q_k : [0, \psi] \times [0, \psi] \to \mathbb{R}_+, k \ge 0$ as follows:

$$Q_0(x, u) = (\psi - u + x)^2 + dx + u(u + d),$$

and for all $k \ge 1$,

$$Q_k(x, u) = (\psi - u + x)^2 + dx + pJ_{k-1}(u) + (1 - p)u(u + d).$$

Clearly,

$$J_k(x) = \min_{u\in[0,\psi]} Q_k(x, u).$$

for all $k \ge 0$. All the three terms in $Q_0(x, u)$ are convex (see [29, Section 3.2.2] for convexity of the first term). Therefore, $Q_0(x, u)$ is convex in both its arguments. As partial minimization preserves convexity [29, Section 3.2.5], $J_0(x)$ is also convex. Since $Q_0(x, u)$ is increasing in x, so is $J_0(x)$.

Let us now assume that $J_k(x)$ is convex for some $k \ge 0$. Then $Q_{k+1}(x, u)$ is convex in both its argument and increasing in x, and following the earlier arguments, $J_{k+1}(x)$ is also convex and increasing. This completes the induction step. We thus see that, for all $k \ge 0$, $J_k(x)$ are convex and increasing. As limits of convex and increasing functions are convex and increasing (see [30, Theorem 10.8] for convexity), $J(x)$ is also convex and increasing.

XI. Appendix B

A. Proof of Theorem 5.1

Let us consider k-stage problems that allow at most $k+1$ service requests (similar to those defined in earlier sections). Let $\bar{J}_k(\cdot,\cdot), k \ge 0$ be the cost functions of these problems on application of policies $\bar{\pi}_k(\cdot,\cdot)$. For brevity, we use following notation in rest of the proof:

$$x_k^i := \bar{\pi}_k(\psi_i, x)\ \forall i, k \text{ and } x.$$

Then

$$\bar{J}_0(\psi_i, x) = (\psi_i - x_0^i + x)^2 + dx + x_0^i(x_0^i + d) \tag{37}$$

and for all $k \ge 1$,

$$\bar{J}_k(\psi_i, x) = (\psi_i - x_k^i + x)^2 + dx + \sum_{j=1}^{N} p_j \bar{J}_{k-1}(\psi_j, x_k^i) + (1 - \bar{p})x_k^i(x_k^i + d). \tag{38}$$

Notice that $\lim_{k\to\infty} \bar{J}_k(\psi_i, x) = \bar{J}(\psi_i, x)$ for all i and $x \in [0, \psi_N]$. Also, for all $x \in [0, \psi_N]$, $J'_0(\psi_i, x) = J_0(\psi_i, x)$ and for $k \ge 1$,

$$J'_k(\psi_i, x) = \min_{u\in[0,\psi_i]} \{(\psi_i - u + x)^2 + dx + \bar{p}J'_{k-1}(u). + (1 - \bar{p})u(u + d)\}, \tag{39}$$

where $J'_k(\cdot), k \ge 0$ are the optimal cost functions associated with k-stage versions of the fictitious problem introduced in Section V-C1. We inductively derive bounds for $\bar{J}_k(\psi_i, x) - J'_k(\psi_i, x)$ for all $k \ge 1$. We then take $k \to \infty$ to obtain a bound on $\bar{J}(\psi_i, x) - J'(\psi_i, x)$.

To begin with,

$$\begin{aligned}
&\bar{J}_1(\psi_i, x) - J'_1(\psi_i, x)\\
&= \sum_{j=1}^{N} p_j \left[\bar{J}_0(\psi_j, x_1^i) - J'_0(x_1^i)\right]\\
&= \sum_{j=1}^{N} p_j \left[J'_0(\psi_j, x_1^i) - J'_0(x_1^i)\right]\\
&= \sum_{j=1}^{N} p_j \left[\min_{u\in[0,\psi_j]} \left\{(\psi_j - u + x_1^i)^2 + dx_1^i + u(u + d)\right\}\right.\\
&\qquad \left. - \min_{u\in[0,\bar{\psi}]} \left\{(\bar{\psi} - u + x_1^i)^2 + dx_1^i + u(u + d)\right\}\right]\\
&= \sum_{j=1}^{N} p_j \left[\min_{u\in[0,\psi_j]} \left\{(\psi_j - u + x_1^i)^2 + u(u + d)\right\}\right.\\
&\qquad \left. - \min_{u\in[0,\bar{\psi}]} \left\{(\bar{\psi} - u + x_1^i)^2 + u(u + d)\right\}\right]\\
&= E_1 + E_2,
\end{aligned}$$

where

$$E_1 := \sum_{j=1}^{N} p_j \left[\min_{u\in[0,\psi_j]} \left\{(\psi_j - u + x_1^i)^2 + u(u + d)\right\} - \min_{u\in[0,\bar{\psi}]} \left\{(\psi_j - u + x_1^i)^2 + u(u + d)\right\}\right]$$

and

$$E_2 := \sum_{j=1}^{N} p_j \Big[\min_{u\in[0,\bar\psi]} \Big\{ (\psi_j - u + x_1^i)^2 + u(u+d) \Big\} - \min_{u\in[0,\bar\psi]} \Big\{ (\bar\psi - u + x_1^i)^2 + u(u+d) \Big\} \Big].$$

The first equality follows from (39) and (38) whereas the second one follows since $\bar J_0(\cdot,\cdot) = J_0'(\cdot,\cdot)$. We now bind E_1 and E_2 separately.

Upper bound on E_1: For all j such that $\psi_j \ge \bar\psi$,

$$\min_{u\in[0,\psi_j]} \Big\{ (\psi_j - u + x_1^i)^2 + u(u+d) \Big\} - \min_{u\in[0,\bar\psi]} \Big\{ (\psi_j - u + x_1^i)^2 + u(u+d) \Big\} \le 0. \quad (40)$$

For all j such that $\psi_j < \bar\psi$, (a) if $\bar\pi_0(\psi_j, x_1^i) < \psi_j$ then

$$\min_{u\in[0,\psi_j]} \Big\{ (\psi_j - u + x_1^i)^2 + u(u+d) \Big\} - \min_{u\in[0,\bar\psi]} \Big\{ (\psi_j - u + x_1^i)^2 + u(u+d) \Big\} = 0. \quad (41)$$

(b) if $\bar\pi_0(\psi_j, x_1^i) = \psi_j$ then

$$\min_{u\in[0,\psi_j]} \Big\{ (\psi_j - u + x_1^i)^2 + u(u+d) \Big\} = {x_1^i}^2 + \psi_j(\psi_j + d)$$

and

$$\min_{u\in[0,\bar\psi]} \Big\{ (\psi_j - u + x_1^i)^2 + u(u+d) \Big\} \ge (\psi_j + x_1^i - \bar\psi)^2 + \psi_j(\psi_j + d).$$

Hence

$$\min_{u\in[0,\psi_j]} \Big\{ (\psi_j - u + x_1^i)^2 + u(u+d) \Big\} - \min_{u\in[0,\bar\psi]} \Big\{ (\psi_j - u + x_1^i)^2 + u(u+d) \Big\} \le (\bar\psi - \psi_j)(2x_1^i + \psi_j - \bar\psi). \quad (42)$$

Combining (40), (41) and (42) and observing that $x_1^i \le \psi_N$ we see that $E_1 \le A$ where A is as in (24).

Upper bound on E_2: Notice that

$$E_2 = \sum_{j=1}^{N} p_j \Big[\min_{u\in[0,\bar\psi]} \Big\{ (\bar\psi - u + x_1^i)^2 + u(u+d) + (\psi_j - u + x_1^i)^2 - (\bar\psi - u + x_1^i)^2 \Big\} - \min_{u\in[0,\bar\psi]} \Big\{ (\bar\psi - u + x_1^i)^2 + u(u+d) \Big\} \Big]$$
$$\le \sum_{j=1}^{N} p_j \Big[\max_{u\in[0,\bar\psi]} \Big\{ (\psi_j - u + x_1^i)^2 - (\bar\psi - u + x_1^i)^2 \Big\} \Big].$$

Further, for all j such that $\psi_j \ge \bar\psi$, $(\psi_j - u + x_1^i)^2 - (\bar\psi - u + x_1^i)^2$ is decreasing in u, implying that

$$\max_{u\in[0,\bar\psi]} \Big\{ (\psi_j - u + x_1^i)^2 - (\bar\psi - u + x_1^i)^2 \Big\} = (\psi_j + x_1^i)^2 - (\bar\psi + x_1^i)^2 = (\psi_j - \bar\psi)(\psi_j + \bar\psi + 2x_1^i). \quad (43)$$

For all j such that $\psi_j < \bar\psi$, $(\psi_j - u + x_1^i)^2 - (\bar\psi - u + x_1^i)^2$ is increasing in u, implying that

$$\max_{u\in[0,\bar\psi]} \Big\{ (\psi_j - u + x_1^i)^2 - (\bar\psi - u + x_1^i)^2 \Big\} = (\psi_j - \bar\psi + x_1^i)^2 - {x_1^i}^2 = (\psi_j - \bar\psi)(\psi_j - \bar\psi + 2x_1^i). \quad (44)$$

Combining (43) and (44) we see that

$$\begin{aligned} E_2 \le & \sum_{j:\psi_j \ge \bar\psi} p_j(\psi_j - \bar\psi)(\psi_j + \bar\psi + 2x_1^i) + \sum_{j:\psi_j < \bar\psi} p_j(\psi_j - \bar\psi)(\psi_j - \bar\psi + 2x_1^i) \\ =& \sum_{j=1}^{N} p_j\psi_j(\psi_j - \bar\psi) + \bar\psi \sum_{j=1}^{N} p_j|\psi_j - \bar\psi| \\ =& \sum_{j=1}^{N} p_j(\psi_j - \bar\psi)^2 + \bar\psi \sum_{j=1}^{N} p_j|\psi_j - \bar\psi| \\ =& B + \bar\psi D, \end{aligned}$$

where D and B are as defined in (22) and (23), respectively. Combining the bounds on E_1 and E_2 we obtain

$$\bar J_1(\psi_i, x) - J_1'(\psi_i, x) \le A + B + \bar\psi D.$$

We prove via induction that

$$\bar J_k(\psi_i, x) - J_k'(\psi_i, x) \le \sum_{l=0}^{k-1} \bar p^l (A + B + \bar\psi D). \quad (45)$$

for all $k \ge 1$. We have already shown that (45) holds $k = 1$. Let us assume that it holds for some $k \ge 1$. Then

$$\begin{aligned} &\bar J_{k+1}(\psi_i, x) - J_{k+1}'(\psi_i, x) \\ &= \sum_{j=1}^{N} p_j \big[\bar J_k(\psi_j, x_{k+1}^i) - J_k'(x_{k+1}^i) \big] \\ &= \sum_{j=1}^{N} p_j \big[\bar J_k(\psi_j, x_{k+1}^i) - J_k'(\psi_j, x_{k+1}^i) \big] \\ &\quad + \sum_{j=1}^{N} p_j \big[J_k'(\psi_j, x_{k+1}^i) - J_k'(x_{k+1}^i) \big], \end{aligned}$$

where, as before, the first equality follows from (39) and (38). Now using induction hypothesis,

$$\sum_{j=1}^{N} p_j\left[\bar{J}_k(\psi_j, x^i_{k+1}) - J'_k(\psi_j, x^i_{k+1})\right]$$
$$\leq \bar{p}\sum_{l=0}^{k-1}\bar{p}^l(A+B+\bar{\psi}D)$$
$$= \sum_{l=1}^{k}\bar{p}^l(A+B+\bar{\psi}D). \quad (46)$$

On the other hand, using similar arguments as for bounding

$$\sum_{j=1}^{N} p_j\left[J'_0(\psi_j, x^i_1) - J'_0(x^i_1)\right]$$

we can see that (we also use the fact that $J'_{k-1}(u)$ is increasing in u)

$$\sum_{j=1}^{N} p_j\left[J'_k(\psi_j, x^i_{k+1}) - J'_k(x^i_{k+1})\right] \leq A+B+\bar{\psi}D. \quad (47)$$

Combining (46) and (47) we obtain

$$\bar{J}_{k+1}(\psi_i, x) - J'_{k+1}(\psi_i, x) \leq \sum_{l=0}^{k}\bar{p}^l(A+B+\bar{\psi}D),$$

which completes the induction step. We thus establish (45) for all $k \geq 1$. Finally, taking $k \to \infty$ in both the sides in (45),

$$\bar{J}(\psi_i, x) - J'(\psi_i, x) \leq \frac{(\mathrm{A}+\mathrm{B}+\bar{\psi}\mathrm{D})}{(1-\bar{p})}.$$

References

[1] S. T. Maguluri, R. Srikant, and L. Ying, “Stochastic models of load balancing and scheduling in cloud computing clusters,” in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 702–710.

[2] J. Koomey, “White paper: Growth in data center electricity use 2005 to 2010.” [Online]. Available: https://www.missioncriticalmagazine.com/ext/resources/MC/Home/Files/PDFs/Koomey_Data_Center.pdf

[3] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. I*, 3rd ed. Athena Scientific, 2007.

[4] D. Kothari and I. Nagrath, *Modern Power System Analysis*. Tata McGraw-Hill Publishing Company, 2003. [Online]. Available: https://books.google.co.in/books?id=AgU0nwEACAAJ

[5] W. Tang, S. Bi, and Y. J. Zhang, “Online coordinated charging decision algorithm for electric vehicles without future information,” *IEEE Trans. Smart Grid*, vol. 5, no. 6, pp. 2810–2824, Nov. 2014.

[6] H. Mohsenian-Rad and M. ghamkhari, “Optimal charging of electric vehicles with uncertain departure times: A closed-form solution,” *IEEE Trans. Smart Grid*, vol. 6, no. 2, pp. 940–942, Mar. 2015.

[7] Wenbo Shi and V. W. S. Wong, “Real-time vehicle-to-grid control algorithm under price uncertainty,” in *Proc. IEEE Int. Conf. Smart Grid Commun.*, Oct. 2011, pp. 261–266.

[8] Z. Yu, Y. Xu, and L. Tong, “Deadline scheduling as restless bandits,” *IEEE Trans. Autom. Control*, vol. 63, no. 8, pp. 2343–2358, Aug. 2018.

[9] T. Chiu, Y. Shih, A. Pang, and C. Pai, “Optimized day-ahead pricing with renewable energy demand-side management for smart grids,” *IEEE Internet Things J.*, vol. 4, no. 2, pp. 374–383, Apr. 2017.

[10] P. Samadi, A. Mohsenian-Rad, R. Schober, V. W. S. Wong, and J. Jatskevich, “Optimal real-time pricing algorithm based on utility maximization for smart grid,” in *Proc. 1st IEEE Int. Conf. Smart Grid Commun.*, Oct. 2010, pp. 415–420.

[11] J. Luo, L. Rao, and X. Liu, “Temporal load balancing with service delay guarantees for data center energy cost optimization,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 3, pp. 775–784, Mar. 2014.

[12] P. Arroba, J. M. Moya, J. L. Ayala, and R. Buyya, “Dynamic voltage and frequency scaling-aware dynamic consolidation of virtual machines for energy efficient cloud data centers,” *Concurr. Comput.: Pract. Exper.*, vol. 29, no. 10, 2017.

[13] L. Wang *et al.*, “GreendCN: A general framework for achieving energy efficiency in data center networks,” *IEEE J. Sel. Areas Commun.*, vol. 32, no. 1, pp. 4–15, Jan. 2014.

[14] M. K. Hanawal, E. Altman, R. El-Azouzi, and B. J. Prabhu, “Spatio-temporal control for dynamic routing games,” in *Game Theory for Networks*, R. Jain and R. Kannan, Eds. Berlin, Germany: Springer, 2012, pp. 205–220.

[15] R. Burra, C. Singh, J. Kuri, and E. Altman, *Routing on a Ring Network*. Cham: Springer International Publishing, 2019, pp. 25–36. [Online]. Available: https://doi.org/10.1007/978-3-319-93058-9_3

[16] F. Yao, A. Demers, and S. Shenker, “A scheduling model for reduced CPU energy,” in *Proc. IEEE 36th Annu. Found. Comput. Sci.*, Oct. 1995, pp. 374–382.

[17] K. Choi, W. Lee, R. Soma, and M. Pedram, “Dynamic voltage and frequency scaling under a precise energy model considering variable and fixed components of the system power dissipation,” in *Proc. IEEE/ACM Int. Conf. Comput. Aided Des.*, Nov. 2004, pp. 29–34.

[18] J. V. Gautam, H. B. Prajapati, V. K. Dabhi, and S. Chaudhary, “A survey on job scheduling algorithms in big data processing,” in *Proc. IEEE Int. Conf. Electrical, Comput. Commun. Technol.*, Mar. 2015, pp. 1–11.

[19] Y. Wang, X. Wu, Y. Yu, and W. Li, “Manufacturing chain and it’s production scheduling problem,” in *Proc. IEEE Int. Conf. Control Automat.*, May 2007, pp. 1435–1439.

[20] G. Reddy and R. Vaze, “Robust online speed scaling with deadline uncertainty,” in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2018*, Aug. 20-22, 2018 - Princeton, NJ, USA, 2018, pp. 22:1–22:17. [Online]. Available: https://doi.org/10.4230/LIPIcs.APPROX-RANDOM.2018.22

[21] N. Bansal, T. Kimbrel, and K. Pruhs, “Speed scaling to manage energy and temperature,” *J. ACM*, vol. 54, no. 1, pp. 1–39, Mar. 2007.

[22] S. Albers and H. Fujiwara, “Energy-efficient algorithms for flow time minimization,” *ACM Trans. Algorithms*, vol. 3, no. 4, pp. 49–es, Nov. 2007.

[23] Q. Wang, X. Liu, J. Du, and F. Kong, “Smart charging for electric vehicles: A survey from the algorithmic perspective,” *IEEE Commun. Surv. Tut.*, vol. 18, no. 2, pp. 1500–1517, Apr.-Jun. 2016.

[24] A. Che, X. Wu, J. Peng, and P. Yan, “Energy-efficient bi-objective single-machine scheduling with power-down mechanism,” *Comput. Operations Res.*, vol. 85, pp. 172–183, 2017.

[25] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. II*, 3rd ed. Athena Scientific, 2007.

[26] T. E. Duncan and B. Pasik-Duncan, “Discrete time linear quadratic control with arbitrary correlated noise,” *IEEE Trans. Autom. Control*, vol. 58, no. 5, pp. 1290–1293, May 2013.

[27] H.-J. Li, Q. Wang, S. Liu, and J. Hu, “Exploring the trust management mechanism in self-organizing complex network based on game theory,” *Phys. A: Statist. Mechanics its Appl.*, vol. 542, 2020, Art. no. 123514.

[28] H.-J. Li, Z. Bu, Z. Wang, and J. Cao, “Dynamical clustering in electronic commerce systems via optimization and leadership expansion,” *IEEE Trans. Ind. Inform.*, vol. 16, no. 8, pp. 5327–5334, Aug. 2020.

[29] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge Univ. Press, 2004.

[30] R. T. Rockafellar, *Convex analysis*, ser. Princeton Mathematical Series. Princeton, NJ.: Princeton Univ. Press, 1970.

Ramya Burra received the B.Tech. degree in 2012. She is currently working toward the Ph.D. degree with the Department of ESE, Indian Institute of Science, Bangalore, India. She currently works on optimal resource allocation and their game theoretic solutions. She specifically use techniques and results from Markov Decision Process literature to solve resource allocation problems.

Chandramani Singh received the M.E. and Ph.D. degrees in electrical communication engineering from the Indian Institute of Science, Bangalore, India, in 2005 and 2012, respectively. He is currently an Assistant Professor with the Department of ESE, Indian Institute of Science. His interests include communication networks, data centers, and smart grids. From 2005 to 2006, he was with ESQUBE Communication Solutions Pvt. Ltd., Bangalore, India. From 2012 to 2013, he was a Research Engineer with TREC, a joint research team between INRIA Rocquencourt and ENS de Paris, Paris, France, and from 2013 to 2014, a Postdoctoral Research Associate with CSL, University of Illinois at Urbana Champaign, Champaign, IL, USA.

Joy Kuri is currently a Professor with the Department of Electronic Systems Engineering, Indian Institute of Science, Bangalore, India. He is the coauthor of two widely cited books and has authored or coauthored 120 articles in peer-reviewed journals and conferences. Over the last two decades, his research and teaching interests included stochastic modelling, analysis, design and control of networks arising in various application contexts, including communication, information dissemination, security, transportation, and power. During June 2014–July 2016, he was an Associate Editor for the IEEE/ACM TRANSACTIONS ON NETWORKING.