Towards Maximizing Nonlinear Delay Sensitive Rewards in Queuing Systems

Sushmitha Shree S[*§], Avijit Mandal[§], Avhishek Chatterjee, Krishna Jagannathan

Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai 600036, India

{sushmithasriram, avijitbesu1995}@gmail.com, {avhishek, krishnaj}@ee.iitm.ac.in

Abstract—We consider maximizing the long-term average reward in a single server queue, where the reward obtained for a job is a non-increasing function of its sojourn time. The motivation behind this work comes from multiple applications, including quantum information processing and multimedia streaming. We introduce a new service discipline, shortest predicted sojourn time (SPST), which, in simulations, performs better than well-known disciplines. We also present some limited analytical guarantees for this highly intricate problem.

Index Terms—delay sensitive reward, service discipline, sojourn time.

I. Introduction

JOB scheduling in single server systems is one of the most widely researched areas due to its diverse applications [1]. Historically, the design of service disciplines focused on optimizing the average linear functions of sojourn times (a.k.a response times). Under this performance measure, the discipline that processes the job with the shortest remaining processing time (SRPT) proves to be optimal [2]. However, almost no work considers optimizing nonlinear functions of sojourn times, which have become crucial in many emerging applications, a few of which we briefly discuss.

1) Quantum information processing: The quantum bits (qubits) that are generated for sequential processing by a circuit or for transmission over a channel undergo decoherence while waiting to be processed or transmitted [3]. The effective information extracted out of a stream of bits is the stationary expectation of a non-increasing function of the sojourn time [4].

2) Multimedia streaming: In streaming applications, delayed packets cause stream to break or pause. Hence, the value of a multimedia packet decreases with its delay [5].

3) Delay sensitive online services: In online service platforms like ride-sharing and food delivery, customers' satisfaction and hence, in turn, ratings often depend on the delay in the service. In fact, in many settings, user dissatisfaction due to delays cannot be compensated by better service or other promotional offers [6], [7].

Although optimizing nonlinear functions of sojourn times is crucial for these applications, there is hardly any study aimed at optimizing the average nonlinear functions of sojourn times, even in a single server case. This paper takes a few steps towards this goal, and is motivated by the aforementioned applications.

A. Related work and Motivation

In work conserving single server queuing systems, jobs can arrive arbitrarily. When the service requirements (job sizes) are known, SRPT minimizes the average sojourn time regardless of the arrival and service distributions [2]. Under SRPT, the job in service has the least remaining processing time, and an incoming job preempts the server only if its processing time is shorter than the remaining processing time of the job in service. Specifically, SRPT minimizes the sojourn time for every arrival sequence [2]. In other words, SRPT is said to be sample-path optimal. Schrage in [8] first discussed the proof of optimality of SRPT, followed by Smith in [9]. SRPT gained popularity thereon that prompted the analysis of its performance guarantees [10, Chapter 33], the evaluation of its fairness among jobs [11], its implementation in web servers [12] and its extension to multiple server systems [13], [14].

Unlike classical queuing systems that assume no constraints on the waiting times, jobs do come with fixed deadlines in certain applications [15]. If the server does not process a job within its deadline, it drops off the queue and never returns for service (balking or reneging). The dynamics of these systems have been extensively investigated under multiple settings [16]–[21]. The most common performance measure here is the overall loss fraction that captures the fraction of jobs lost out of the total arrivals to the system. The earliest deadline first (EDF) discipline is shown to be optimal in minimizing the overall loss fraction irrespective of the service requirements [22], [23]. However, minimizing the overall loss fraction does not always guarantee the minimum average sojourn time. Therefore, it is reasonable to associate a reward for each job that captures the trade-off between the fraction of loss and the average sojourn time in the system. In [24], the deadline and reward of jobs are known upon arrival, and the optimal policy that maximizes the rewards per service requirement of served jobs has been studied. [25] and [26] present a similar line of work. Nevertheless, in real-time systems, neither the deadlines nor the rewards of jobs are known to the server.

Our work is inspired by the applications such as quantum information processing and multimedia streaming. In these applications, the information in the jobs (qubits in quantum systems [4] and data packets in multimedia systems [27]) become useless or erased after a certain deadline. Unlike

impatient customers, the jobs do not drop off the queue; however, processing them after their deadline may not be useful to the system.

For instance, in the quantum setting, qubits arrive sequentially at a quantum system and wait in the queue until they are processed. While a qubit waits in the queue, it undergoes decoherence due to its interaction with the environment [3]. The decoherence of a qubit leads to the erasure of its information, and the probability of qubit erasure is modeled as an explicit function of its sojourn time. For example, if a qubit waits for W units of time in the system, then the probability of its erasure is modeled as $p(W) = 1 - \exp(-\kappa W)$ for some $\kappa > 0$, where κ is the characteristic parameter of the quantum system [4]. In other words, a qubit with sojourn time W is associated with a reward of the form $\exp(-\kappa W)$ for some $\kappa > 0$. A similar model is relevant in the areas of multimedia streaming [27] and crowdsourcing [28].

The information capacity of quantum erasure channels has been derived irrespective of the service discipline in [4]. Specifically, this capacity is proportional to $\mathbb{E}[\exp(-\kappa W)]$, where the expectation is over the limiting distribution of the sojourn times. The goal of maximizing the capacity of quantum erasure channels poses an interesting problem and reduces to maximizing the average nonlinear function of sojourn times (rewards). Our work is inspired by such a setting. In particular, this work aims to maximize the average nonlinear functions of the form $\exp(-\kappa W)$ for some $\kappa > 0$ from a scheduling perspective.

B. Contributions

In this work, we consider a work conserving single server queuing system in which the service requirements of the jobs are known upon arrival. Each job is associated with a reward based on its sojourn time. Specifically, the reward of a job is a specified non-increasing function, possibly nonlinear in its sojourn time. This work aims to identify the service discipline that maximizes the long-term average of rewards. Since the rewards are a function of sojourn times, this essentially ensures the maximization of the long-term average of rewards while processing the maximal number of jobs.

We view this problem for two arrival models. Firstly, we consider batch arrival models in which an arbitrary number of jobs arrive at the server at the same instant. In this model, we show that processing the jobs with the shortest service requirements maximizes the long-term average rewards of the system. In addition, we show that this result holds for all monotonic functions of sojourn times.

Next, we analyze a more realistic arrival model in which jobs arrive according to a stochastic process. It is well-known that SRPT maximizes linear rewards [2] for all arrival sequences and service distributions; however, it is unclear if SRPT maximizes nonlinear rewards. For a single server system with a unit service rate, simulations show that SRPT does not perform better for some arrival and job size distributions. Indeed, we find that identifying a discipline that maximizes any monotonic function of sojourn times poses a difficult problem. This is mainly because the performance of the

service disciplines has a complex dependence on the i) arrival and service distributions, ii) job sizes, and iii) function of sojourn times. Certainly, the simulation of the performance of existing disciplines shows that there is no clear winner for all arrival sequences and functions of sojourn times. To reduce the complex dependency on the function of sojourn times, we focus only on rewards of the form $\exp(-\kappa W)$ for some $\kappa > 0$, where W represents the sojourn time. These functions have practical implications in applications such as quantum information systems and multimedia streaming, as mentioned before.

In this work, we introduce a service discipline, *shortest predicted sojourn time (SPST)* and analyze its performance in this setting. According to SPST, a job in service has the least predicted sojourn time. Through simulations, we infer that the performance of SPST is promising for all arrival and job size distributions. However, analytically proving this for all arrival distributions and job sizes is still a hard problem. Therefore, we assume a simple model where jobs of the same size arrive at the server with stochastic interarrival times. Due to the combinatorial intricacies, we compare the performance of SPST with only the first come first serve (FCFS) discipline for this model. In particular, we show that the long-term average reward under SPST is higher than that under FCFS for $\kappa \geq \log_e 2$. Moreover, it is evident from this result that there is no optimal service discipline that maximizes the long-term average of rewards of the form $\exp(-\kappa W)$ for all κ.

C. Organization

The rest of the paper is organized as follows: Section II gives an overview of the system with batch arrivals and stochastic arrivals. Section III and IV discuss the main results for these two scenarios respectively. Under stochastic arrivals, the simulations of the performance of SPST and other disciplines are discussed in section IV-B. Followed by the analytical findings of the performance comparison of SPST with FCFS that are covered in section IV-C. Proofs are detailed in the Appendix.

II. SYSTEM MODEL

We consider a discrete-time work conserving single server queue with unit service rate. The jobs with integer sizes $\{S_i, i \in \mathbb{N}\}$ arrive randomly at the server. These jobs are indexed by positive integers according to their arrivals, with the ties broken arbitrarily. At the beginning of every time slot, the server can change its service from one job to another based on the service discipline. Each job waits in the queue before being served, and the total time it spends in the system is known as its sojourn time. For a job indexed by i, W_i represents its sojourn time, and $f(W_i)$ is the associated reward, where $f(\cdot)$ is a non-increasing function. This work aims to find a service discipline that maximizes the long-term average of rewards.

In this work, we consider two scenarios: (i) batch arrivals and arbitrary job sizes and (ii) stochastic arrivals and stochastic job sizes. In the first scenario, as the name suggests, n jobs arrive at time 0 and their sizes are $\{S_i : 1 \leq i \leq n\}$. In this

context, our goal is to find a service discipline that maximizes the accumulated reward, $\sum_{i=1}^{n} f(W_i)$, for any positive integer n and $\{S_i : 1 \leq i \leq n\}$.

In the second setting, jobs arrive according to some point process with i.i.d. positive inter-arrival times $Y_1, Y_2, \ldots$. Job sizes $\{S_i\}$ are also i.i.d. positive random variables. In this scenario, the goal is to find a *stationary* service discipline π under which the long-term average reward, $\lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} f(W_i)_\pi$, is maximum. Note that whenever $\mathbb{E}[Y_1] > \mathbb{E}[S_1]$, this limit exists almost surely for any work conserving stationary service discipline.

III. BATCH ARRIVALS AND ARBITRARY JOB SIZES

In any queue setup, jobs are assumed to arrive singly at a server. However, this is not the case in all real-world scenarios. Jobs do come in batches of fixed or random sizes [29] as in the case of cloud-based data processing. This section characterizes the service discipline that maximizes the accumulated reward in a queuing system with a single batch of job arrivals.

Definition 1 (Shortest job first (SJF) [10, Chapter 31]). *Under this non-preemptive service discipline, whenever the server frees up, it serves the job with the shortest service requirement to completion. That is, at any time t, the index of the job in service is $k = \underset{i}{\arg\min} \, S_i$. Ties are broken arbitrarily.*

In the case of a single batch of arrivals, the jobs are served in increasing order of their sizes under SJF.

Theorem 1. *For any batch size n and any service requirements $\{S_i : 1 \leq i \leq n\}$, SJF maximizes $\sum_{i=1}^{n} f(W_i)$.*

The proof of theorem 1 is a direct consequence of the following lemma. Consider that a bunch of n jobs arrive at an arbitrary time t. Let $\{S_k, k \in [1, n]\}$ denote their sizes and $\{J_i, i \in [1, n]\}$ be the job labels in increasing order of their sizes i.e., if, for J_i, J_j such that $i < j$, then $S_i \leq S_j \quad \forall i, j \in [1, n]$. Let A_1 be the service discipline that serves the jobs in the order $\{J_1, J_2, \ldots, J_k, J_{k+1}, \ldots, J_n\}$. Consider another discipline A_2 with order of service $\{J_1, J_2, \ldots, J_{k+1}, J_k, \ldots, J_n\}$. Let R_π denote the accumulated reward under service discipline π. Here, $R_\pi = \sum_{i=1}^{n} f(W_i)_\pi$.

Lemma 1. *For a non-increasing function f, $R_{A_1} \geq R_{A_2}$.*

Proof of lemma 1. In a work-conserving system with order of service $\{l_i, i \in [1, n]\}$, the sojourn time of job at index l_k, $W_{l_k} = W_{l_{k-1}} + S_{l_k}$. Equivalently, $W_{l_k} = \sum_{i=1}^{k} S_{l_i}$. Clearly,

$$f(W_{l_i})_{A_1} = f(W_{l_i})_{A_2} \quad \forall i \neq k, k+1. \tag{1}$$

So, it is sufficient to compare $f(W_{l_k}) + f(W_{l_{k+1}})$ under A_1 and A_2.

$$f(W_{l_{k+1}})_{A_1} = f\Big(\sum_{i=1}^{k-1} S_i + S_k + S_{k+1}\Big)$$
$$= f\Big(\sum_{i=1}^{k-1} S_i + S_{k+1} + S_k\Big)$$

$$= f(W_{l_{k+1}})_{A_2}. \tag{2}$$

Now, $f(W_{l_k})_{A_1} = f\Big(\sum_{i=1}^{k-1} S_i + S_k\Big)$. Since f is non-increasing in its argument, we have

$$f(W_{l_k})_{A_1} \geqslant f\Big(\sum_{i=1}^{k-1} S_i + S_{k+1}\Big) = f(W_{l_k})_{A_2}. \tag{3}$$

From (1), (2) and (3), we have $R_{A_1} \geqslant R_{A_2}$. $\qquad \square$

We observe that an arbitrary order of service is a permutation of the servicing order A_2 and that lemma 1 can be extended to all such orders of service in place of A_2. More generally, lemma 1 states that any work-conserving discipline that serves the jobs in increasing order of their sizes yields higher rewards. Examples of such service discipline include SRPT and preemptive shortest job first (PSJF) also.

IV. STOCHASTIC ARRIVALS

We now focus on the scenario with stochastic job arrivals. The goal here is quite different from that for batch arrivals. We cannot extend the results in section III to this scenario as lemma 1 does not hold here. Furthermore, the well-known service disciplines perform differently depending on the job sizes and the arrival rates. For instance, consider that the jobs of same size, j, arrive with interarrival times $\{Y_i, i \in \mathbb{N}\}$, where

$$Y_i = \begin{cases} j_1 = j + 1 - \delta & \text{w.p } \frac{1}{2} \\ j_2 = j + 1 + \delta & \text{otherwise} \end{cases}$$

for any $\delta > 0$. Note that the system is stable with $\{Y_i, i \in \mathbb{N}\}$. Let $f(W_i) = \exp(-\kappa W_i) \, \forall i$ for some $\kappa > 0$.
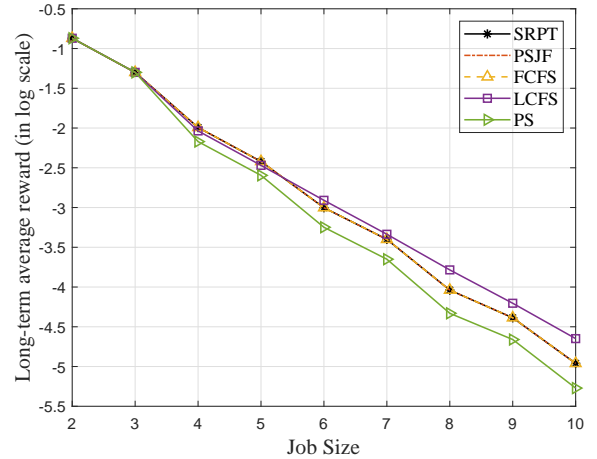

Fig. 1: Job size vs. long-term average reward for $\kappa = 1$

For this arrival sequence with $\delta = \lfloor \frac{j}{2} \rfloor$ and $\kappa = 1$, figure 1 shows the performance of well-known disciplines: SRPT, PSJF, FCFS, last come first serve (LCFS), and processor sharing (PS). Since the jobs are of same size j, some disciplines perform the same. However, this is not the case for all arrival sequences. For $j = 4$, FCFS, SRPT, and PSJF yield higher long-term average rewards, whereas LCFS

dominates for $j \geq 6$. It is therefore evident from figure 1 that even for a fixed κ, the performance of the aforementioned disciplines varies according to the job sizes. Next, we propose a new service discipline named shortest predicted sojourn time (SPST), which performs better than FCFS, LCFS, PSJF, SRPT, and PS in simulation. We also provide an analytical comparison with FCFS.

A. Shortest predicted sojourn time (SPST)

The server of a work conserving queue cycles between idle and busy periods, i.e., the periods when the queue is empty and when it is not, respectively. On a given sample path of the arrival process and a given realization of the job size sequence, the positions and duration of the busy and idle periods are the same for all work conserving policies. Moreover, for an arrival process with i.i.d inter-arrival times, the beginning of a busy period is a renewal (or regenerative) epoch. Thus, by the renewal reward theorem [30], for maximizing the long-term average reward, it is enough to maximize the average total reward in a renewal cycle.

For a fast decaying $f(\cdot)$, the total reward in a renewal cycle is dominated by the jobs with the shortest sojourn time. Thus, the two main factors that ensure high total reward in a cycle are the minimum sojourn time across all jobs in that cycle and the number of jobs whose sojourn time is equal to or close to that.

As the future arrivals and job sizes are not known while making the service decision, intuitively, the best one can do is to serve the job whose completion would result into the shortest sojourn time among the existing jobs. This may increase the sojourn times of other jobs. However, as they are not the dominating terms in the total reward, the overall reward would be high.

Based on the above insights, we design the following policy, which we call shortest predicted sojourn time (SPST).

Definition 2 (Predicted sojourn time). *Predicted sojourn time of a job at index i at time $t \geq 0$ under a service discipline π, denoted by $P_\pi^{(t,i)}$, is its sojourn time if it is chosen by the server at time t and is run to completion without preemption.*

Definition 3 (Shortest predicted sojourn time (SPST)). *Under SPST, at every time instant, the job in service is the one with the shortest predicted sojourn time. That is, at any time t, the index of the job in service is*

$$k = \underset{i}{\arg\min} \, P_{SPST}^{(t,i)}.$$

In case of a tie, the job with the least arrival time is prioritized.

B. Performance of SPST and other disciplines

In this subsection, the performance of SPST is compared with that of other well-known service disciplines. The long-term average rewards are plotted on a log scale for better visualization. We consider that the reward associated with each job is of the form $f(W) = \exp(-\kappa W)$ for some $\kappa > 0$. Figures 2 and 3 depict the performance of disciplines when jobs of same size arrive with interarrival times $\{Y_i : i \geq 0\}$.

We consider $\delta = \lfloor \frac{j}{2} \rfloor$ for the simulations. It is noted that SPST performs better than the existing disciplines for all job sizes for $\kappa = 1$.
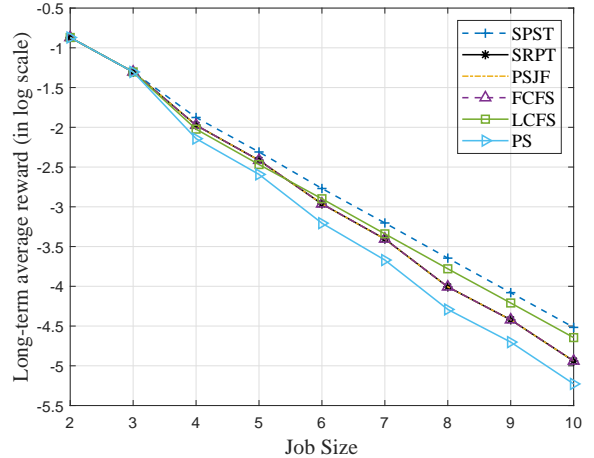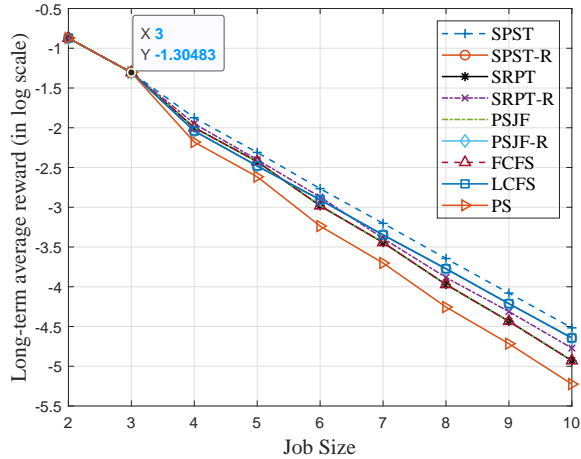

Fig. 2: Job size vs. long-term average reward for $\kappa = 1$. An illustration of the performance of SPST discipline.

By convention, the job with the least arrival time is prioritized for service in case of a tie under any service discipline. However, in our reward-based queue setup with $f(W) = \exp(-\kappa W)$, the tie-breaking criterion has to be suitably chosen to exploit the contribution of smaller jobs to the accumulated reward of the system. So, we also simulate the disciplines with a tie-breaker that prioritizes the most recent job for service. The suffix *-R* represents the discipline with this tie-breaker. e.g., SPST-R.
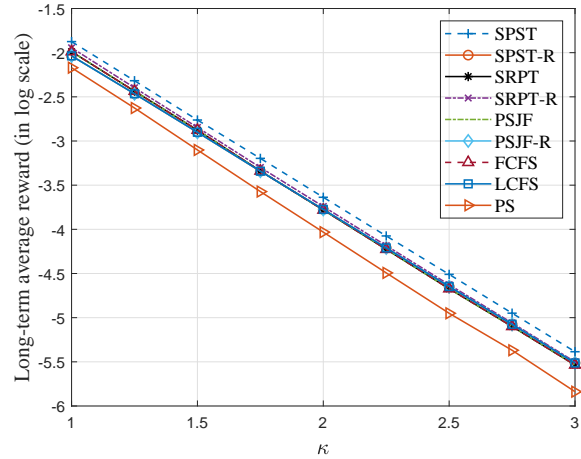
Figure 3a depicts the performance of the disciplines along with their tie-breaking variant. With $j = 2$, any busy period is 2 irrespective of the service discipline, and hence, their long-term average rewards are the same. In addition, for any i, Y_i is either 2 or 4 with equal probability, which is why the long-term average reward is 0.135. The same applies for $j = 3$, in which case the long-term average reward is 0.0497. It is noted that SPST still yields rewards higher than that of any of its contenders for $\kappa = 1$. In particular, for $\kappa \geq 1$, SPST is a clear winner for all sample paths regardless of j as shown in 3b.

Figures 4a and 4b illustrate a more general case of Bernoulli arrivals with jobs of fixed size j. To ensure stability of the queue, we take arrival rate to be $\frac{1}{j+1}$. It can be seen that, even in this case, the performance of SPST is clearly better than the other policies for $\kappa \geq 1$.

Since the jobs are of the same size in either case of arrival sequences, some disciplines perform equally. As seen in figures 3 and 4, SPST-R, PSJF-R, and LCFS show similar performance. Likewise, the performances of SRPT, PSJF, and FCFS are similar. In addition, it is evident that the performance of PS is worse than that of SPST for $\kappa \geq 1$. This could mainly be due to the time-sharing of jobs under PS. It is also observed that SRPT-R and PSJF-R show better performance when compared to their respective conventional variants. However, under SPST, when same-sized jobs arrive in a sequence, there can never be a tie between two jobs waiting in the queue based
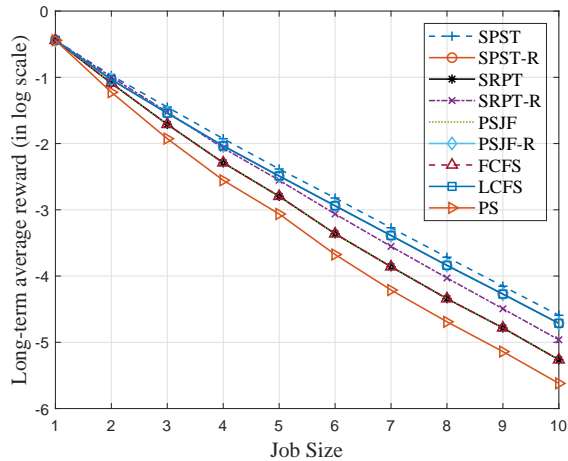
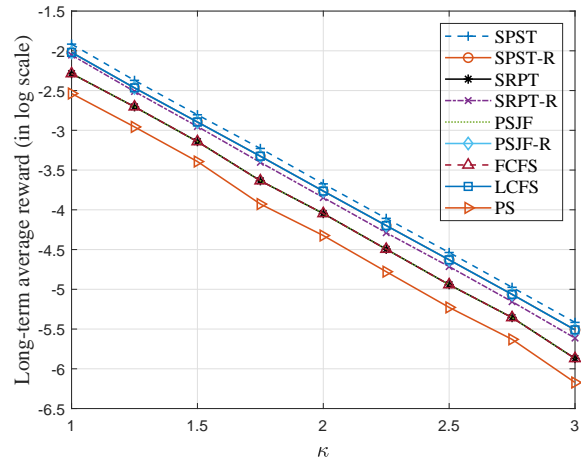(a) Job size vs. long-term average reward for $\kappa = 1$

(b) κ vs. long-term average reward for $j = 4$.

Fig. 3: An illustration for the case of job arrivals with $\{Y_i, i \in \mathbb{N}\}$ for $\delta = \lfloor \frac{j}{2} \rfloor$.

(a) Job size vs. long-term average reward for $\kappa = 1$

(b) κ vs. long-term average reward for $j = 4$.

Fig. 4: An illustration for the case of Bernoulli arrivals with probability of arrival $\frac{1}{j+1}$.

on their predicted sojourn times. Only the job in service and a job in the queue are tied on this basis, in which case priority to the job in service yields better rewards. On the other hand, for more realistic arrival models with different job sizes, the server can choose a tie-breaker under SPST depending on the secondary performance measure such as expected slowdown [10, Chapter 28].

Although simulations suggest that SPST is better and may even be an optimal policy for all arrival sequences, proving such guarantees are extremely hard. In the next section, we analytically prove that SPST performs better than FCFS. It will be evident that even this comparison is quite challenging due to intricate combinatorial structures.

C. Analytical guarantee

For theoretical analysis, we consider a queuing system in which jobs of same size, j, arrive with interarrival times $\{Y_i :$

$i \geq 0\}$. Recall

$$Y_i = \begin{cases} j_1 = j + 1 - \delta & \text{w.p } \frac{1}{2} \\ j_2 = j + 1 + \delta & \text{otherwise} \end{cases}$$

We consider that the reward associated with each job is of the form $f(W) = \exp(-\kappa W)$ for $\kappa > 0$. Under a stationary service discipline π, let $\bar{r}_\pi := \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} f(W_i)_\pi$, i.e., the long-term average reward. The following proposition is the main result of this section.

Proposition 1. *For the defined queuing system with $\delta \leq \frac{j}{2}$ and $f(W) = \exp(-\kappa W)$, $\bar{r}_{SPST} \geq \bar{r}_{FCFS}$ for all $\kappa \geq \log_e 2$.*

Proposition 1 is a direct consequence of theorem 2. A better understanding of this relationship requires the following definitions.

Definition 4 (Busy period). *The time from when the server is busy until it becomes idle.*

Definition 5 (Busy period length). *The number of jobs in a busy period is called its length, L.*

Definition 6 (Idle period). *The time from when the server is idle until it becomes busy.*

Let R_π denote the accumulated reward in an arbitrary busy period under service discipline π. That is, for a busy period of length n, $R_\pi = \sum_{i=1}^{n} f(W_i)_\pi$. We denote the number of arrivals till t by $\mathcal{A}(t)$. Then, $\bar{r}_\pi = \lim_{t \to \infty} \frac{1}{\mathcal{A}(t)} \sum_{j=1}^{\mathcal{A}(t)} f(W_j)_\pi$.

Theorem 2. *For the defined queuing system with $\delta \leq \frac{j}{2}$ and $f(W) = \exp(-\kappa W)$, $R_{SPST} \geq R_{FCFS}$ for $\kappa \geq \log_e 2$.*

By renewal reward theorem, we have

$$\bar{r}_\pi = \frac{\mathbb{E}[R_\pi]}{\lambda \mathbb{E}[\text{busy period} + \text{idle period}]}$$

where $\lambda = \frac{1}{j+1}$ denotes the arrival rate of the jobs. We note that $\mathbb{E}[\text{busy period} + \text{idle period}]$ is the same for all work conserving disciplines. Therefore, by theorem 2, we have $\bar{r}_{SPST} \geq \bar{r}_{FCFS}$ for all $\kappa \geq \log_e 2$.

V. Proof of Theorem 2

Recall definitions 4 and 5. The following are the observations with respect to a busy period for the case of job arrivals with $\{Y_i, i \in \mathbb{N}\}$ defined earlier.

(i) If the first inter-arrival time, Y_1, is j_2, then the busy period is j. In this case, any work conserving discipline yields the same reward, $\exp(-\kappa j), \kappa > 0$.

(ii) If $Y_1 = j_1$, then $L > 1$ for $\delta > 1$.

(iii) A busy period has ended if $(k_1 + k_2)j \leq k_1 j_1 + k_2 j_2$ for $k_1, k_2 \geq 0$. This is because, in any work conserving discipline, the total work in a busy period cannot be greater than the busy period itself.

Remark 1. *Following observation (iii), before a busy period ends, k_2 cannot be larger than k_1. However $k_1 \leq k_2$ is only a sufficient condition for a busy period to end.*

We use the following lemmas to prove theorem 2.

Definition 7 (Priority job). *A job of size j is called a priority job under any discipline if its sojourn time is j. In other words, a priority job neither waits nor is preempted until it is run to completion.*

Lemma 2. *Under SPST, there are at least $\lceil \frac{n}{2} \rceil$ priority jobs for $\delta \leq \frac{j}{2}$.*

The following definitions are instrumental to understanding lemma 2 and the subsequent lemmas. Proof of the lemmas are given in the Appendix.

Definition 8 (Block A). *The consecutive jobs that follow the interarrival time j_1 form a block A.*

Definition 9 (Block B). *The consecutive jobs that follow the interarrival time j_2 form a block B.*

Let n_A and n_B denote the number of blocks A and B in the busy period respectively. We consider that A_k denotes the kth block A, $n(A_k)$ denotes the number of jobs in the kth block A and A_k^i denotes the ith job in the kth block A, with the similar interpretation for block B. Let n_{j_1} and n_{j_2} represent the total number of jobs in blocks A and B respectively. i.e., $\sum_{i=1}^{n_A} n(A_i) = n_{j_1}$ and $\sum_{i=1}^{n_B} n(B_i) = n_{j_2}$. Let n_π^P represent the number of priority jobs in the busy period under a service discipline π.

Lemma 3. *For any A_i with odd $n(A_i)$, there exists a job whose sojourn time under SPST is $j + \delta - 1$.*

Lemma 4. *If, in a busy period, n is even and $n(A_i)$ is even for every i, then $n_{SPST}^P \geq \frac{n}{2} + 1$.*

Lemma 5. *Under FCFS,*

a) *There is only one priority job.*
b) *All other jobs have $W \geq j + 1$.*
c) *For $n \geq 3$, at least one job has $W \geq j + \delta$.*

Proof of theorem 2. Let T denote the busy period of length n. For $n < 3$, $R_{SPST} = R_{FCFS}$. For $n \geq 3$, from lemma 5,

$$R_{FCFS} \leq \exp(-\kappa j) + \exp(-\kappa(j + \delta))$$
$$+ (n - 2)\exp(-\kappa(j + 1)) \quad (4)$$
$$\leq \exp(-\kappa j) + (n - 1)\exp(-\kappa(j + 1)). \quad (5)$$

If n is odd, from lemma 2,

$$R_{SPST} \geq \left\lceil \frac{n}{2} \right\rceil \exp(-\kappa j) + \left(n - \left\lceil \frac{n}{2} \right\rceil\right)\exp(-\kappa T). \quad (6)$$

Using (5) and (6),

$$R_{SPST} - R_{FCFS}$$
$$\geq \left(\left\lceil \frac{n}{2} \right\rceil - 1\right)\exp(-\kappa j) + \left(n - \left\lceil \frac{n}{2} \right\rceil\right)\exp(-\kappa T)$$
$$- (n - 1)\exp(-\kappa(j + 1))$$
$$\geq \left(\left\lceil \frac{n}{2} \right\rceil - 1\right)\exp(-\kappa j) - (n - 1)\exp(-\kappa(j + 1))$$
$$\geq \left\lfloor \frac{n}{2} \right\rfloor \exp(-\kappa j)(1 - 2\exp(-\kappa)). \quad (7)$$

If n is even, following lemmas 3 and 4,

$$R_{SPST} \geq \frac{n}{2}\exp(-\kappa j) + \exp(-\kappa(j + \delta - 1))$$
$$+ \left(n - \frac{n}{2} - 1\right)\exp(-\kappa T). \quad (8)$$

Using (4) and (8),

$$R_{SPST} - R_{FCFS}$$
$$\geq \left[\frac{n}{2} - 1\right]\exp(-\kappa j) + \exp(-\kappa(j + \delta))(\exp(\kappa) - 1)$$
$$+ \left(n - \frac{n}{2} - 1\right)\exp(-\kappa T) - (n - 2)\exp(-\kappa(j + 1))$$
$$\geq \left[\frac{n}{2} - 1\right]\exp(-\kappa j) - (n - 2)\exp(-\kappa(j + 1))$$
$$\geq \left[\frac{n}{2} - 1\right]\exp(-\kappa j)(1 - 2\exp(-\kappa)). \quad (9)$$

From equations (7) and (9), $R_{SPST} \geq R_{FCFS}$ for $\kappa \geq \log_e 2$. $\square$

Proposition 1 states that the long-term average reward under SPST is more than that under FCFS for $\kappa \geq \log_e 2$. However it is also clear from (4), (6), and (8) that $\log_e 2$ is not a sharp threshold and obtaining a tight lower bound on κ is far from simple.

VI. Conclusion

In this paper, we studied the problem of maximizing the average nonlinear functions of sojourn times in work conserving single server queuing systems and characterized the performance of some well-known service disciplines. We argued that identifying a single service discipline that outperforms other disciplines for all arrival distributions and job sizes appears to be a highly nontrivial problem. Indeed, an optimal policy could depend on the specific functional form of the nonlinear reward function. We also introduced a service discipline, shortest predicted sojourn time (SPST), and provided analytical guarantees under specific settings. Numerical experiments suggest that SPST performs well across multiple settings, although it may not be optimal for all job sizes, arrival distributions, and reward functions. As such, the general problem setting remains largely open for further analytical investigations.

Acknowledgement

Appendix

Recall the definitions and notations discussed in section IV. The following claim is required for the construction of the proof of lemmas 2 to 4.

Claim 1. *Under SPST discipline,*

a) *For a busy period of length n, $n_{j_1} + n_{j_2} = n - 1$.*
b) *A busy period with $L > 1$ always starts with block A. Also, every block B is preceded by a block A. That is, $n_A - n_B \in \{0, 1\}$.*
c) *For $\delta \leq \frac{j}{2}$, the jobs in the even index of block A are priority jobs under SPST.*
d) *Every job of block B is a priority job under SPST.*

Proof of claim 1.

a) The first job in a busy period does not constitute either of the blocks.
b) Follows observation (ii) and the construction of the blocks.
c) Follows the construction of the blocks and for $\delta \leq \frac{j}{2}$, $Y_{k-1} + Y_k > j$ for any $k > 1$.
d) Follows the construction of blocks B. □

A. Proof of lemma 2

Following claim 1, $n_{SPST}^P = \sum_{i=1}^{n_A} \lfloor \frac{n(A_i)}{2} \rfloor + n_{j_2} + 1$.

Case 1 ($n_A = n_B$).

$$
\begin{aligned}
n_{SPST}^P &\geq \sum_{i=1}^{n_A} \frac{n(A_i) - 1}{2} + n_{j_2} + 1 \\
&\geq \frac{n_{j_1} - n_A}{2} + n_{j_2} + 1 \\
&\geq \frac{n_{j_1} - n_{j_2}}{2} + n_{j_2} + 1 && (\because n_B \leq n_{j_2}) \\
&\geq \frac{n - 1}{2} + 1 && \text{(from claim 1)}
\end{aligned}
$$

which gives $n_{SPST}^P \geq \left\lceil \frac{n}{2} \right\rceil$.

Case 2 ($n_A = n_B + 1$).
When n is even,

$$
\begin{aligned}
n_{SPST}^P &\geq \sum_{i=1}^{n_A} \frac{n(A_i) - 1}{2} + n_{j_2} + 1 \\
&\geq \frac{n_{j_1} - n_A}{2} + n_{j_2} + 1 \\
&\geq \frac{n_{j_1} - (n_{j_2} + 1)}{2} + n_{j_2} + 1 && (\because n_B \leq n_{j_2}) \\
&\geq \frac{n - 2}{2} + 1 && \text{(from claim 1)} \\
&\geq \frac{n}{2}.
\end{aligned}
$$

When n is odd, there are four possible subcases as follows.

Subcase 1 (n_A is odd, n_{j_1} is odd). It is noted that n_B is even and n_{j_2} is odd (from claim 1). This implies that at least one block B has even number of jobs. Therefore, $n_B \leq n_{j_2} - 1$.

$$
\begin{aligned}
n_{SPST}^P &\geq \sum_{i=1}^{n_A} \frac{n(A_i) - 1}{2} + n_{j_2} + 1 \\
&\geq \frac{n_{j_1} - n_A}{2} + n_{j_2} + 1 \\
&\geq \frac{n_{j_1} - n_{j_2}}{2} + n_{j_2} + 1 && (\because n_B \leq n_{j_2} - 1) \\
&\geq \frac{n - 1}{2} + 1 && \text{(from claim 1)} \\
&\geq \left\lceil \frac{n}{2} \right\rceil.
\end{aligned}
$$

Subcase 2 (n_A is even, n_{j_1} is odd). Here n_B is odd and n_{j_2} is odd. This implies that at least one block A has even number of jobs, Say, one such block is $A_{k'}$.

$$
\begin{aligned}
n_{SPST}^P &\geq \frac{n(A_{k'})}{2} + \sum_{i=1}^{n_B} \frac{n(A_i) - 1}{2} + n_{j_2} + 1 \\
&\geq \frac{n_{j_1} - n_B}{2} + n_{j_2} + 1 \\
&\geq \frac{n_{j_1} - n_{j_2}}{2} + n_{j_2} + 1 && (\because n_B \leq n_{j_2}) \\
&\geq \frac{n - 1}{2} + 1 && \text{(from claim 1)} \\
&\geq \left\lceil \frac{n}{2} \right\rceil.
\end{aligned}
$$

Subcase 3 (n_A is even, n_{j_1} is even). In this case, there is at least one block B that has even number of jobs. The lower bound for n_{SPST}^P follows subcase 1 giving $n_{SPST}^P \geq \left\lceil \frac{n}{2} \right\rceil$.

Subcase 4 (n_A is odd, n_{j_1} is even). There is at least one block A that has even number of jobs and hence, the lower bound for n_{SPST}^P in this case follows subcase 2 giving $n_{SPST}^P \geq \left\lceil \frac{n}{2} \right\rceil$.

From cases 1 and 2, it is proved that there are at least $\left\lceil \frac{n}{2} \right\rceil$ priority jobs under SPST discipline for $\delta \leq \frac{j}{2}$. $\qquad \square$

B. Proof of lemma 3

For any i, if $n(A_i)$ is odd, following claim 1, $A_i^{n(A_i)-1}$ is a priority job. Although block A_i might be followed by block B_i, since $j_1 + j_2 > 2j$, the sojourn time of $A_i^{n(A_i)}$ is governed only by its preceding priority job, $A_i^{n(A_i)-1}$, and hence, its waiting time in the queue is $\delta - 1$. Therefore, the sojourn time of $A_i^{n(A_i)}$ is $j + \delta - 1$. $\qquad \square$

C. Proof of lemma 4

For a busy period, assume that n is even and there is at least one block A_j with even $n(A_j)$. By the analysis of cases 1 and 2 as in lemma 2, we get $n_{SPST}^P \geq \frac{n}{2} + 1$. Let m_e denote the number of blocks A_k with even $n(A_k)$. Extending the above-mentioned argument to all $m_e \geq 2$ such blocks, there are at least $\frac{n}{2} + \frac{m_e}{2}$ priority jobs in the busy period. In other words, except for one even block A, the presence of all other even blocks A improves the bound by $\frac{1}{2}$. $\qquad \square$

D. Proof of lemma 5

a) Under FCFS, the first job in a busy period is not pre-empted by any of the subsequent arrivals.
b) Follows lemma 5a).
c) Let W_k^{FCFS} denote the sojourn time of the kth job arrival in the busy period under FCFS. From lemma 5a), we know that $W_1^{FCFS} = j$ and for $k \geq 2$, $W_k^{FCFS} = W_{k-1}^{FCFS} - Y_{k-1} + j \geq j + 1$. Owing to claim 1, for $n \geq 3$, there exists at least one job that follows the interarrival time j_1. Therefore, for such jobs, $k \geq 3$,

$$W_k^{FCFS} \geq 2j - Y_{k-1} + 1 \geq j + \delta$$

$\qquad \square$

REFERENCES

[1] C. N. Potts and V. A. Strusevich, "Fifty years of scheduling: A survey of milestones," *The Journal of the Operational Research Society*, vol. 60, pp. s41–s68, 2009.

[2] L. E. Schrage and L. W. Miller, "The queue M/G/1 with the shortest remaining processing time discipline," *Operations Research*, vol. 14, no. 4, pp. 670–684, 1966.

[3] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010, ch. 8.

[4] K. Jagannathan, A. Chatterjee, and P. Mandayam, "Qubits through queues: The capacity of channels with waiting time dependent errors," in *2019 National Conference on Communications (NCC)*, 2019, pp. 1–6.

[5] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*, 6th ed. Pearson, 2012, ch. 7.

[6] J. R. Daugherty and G. L. Brase, "Taking time to be healthy: Predicting health behaviors with delay discounting and time perspective," *Personality and Individual Differences*, vol. 48, no. 2, pp. 202–207, 2010.

[7] S. Dewan and H. Mendelson, "User delay costs and internal pricing for a service facility," *Management Science*, vol. 36, no. 12, pp. 1502–1517, 1990.

[8] L. Schrage, "A proof of the optimality of the shortest remaining processing time discipline," *Operations Research*, vol. 16, no. 3, pp. 687–690, 1968.

[9] D. R. Smith, "Technical note - a new proof of the optimality of the shortest remaining processing time discipline," *Oper. Res.*, vol. 26, pp. 197–199, 1978.

[10] M. Harchol-Balter, *Performance Modeling and Design of Computer Systems: Queueing Theory in Action*, 1st ed. Cambridge University Press, 2013.

[11] N. Bansal and M. Harchol-Balter, "Analysis of SRPT scheduling: Investigating unfairness," in *Proceedings of the 2001 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '01. New York, NY, USA: Association for Computing Machinery, 2001, p. 279–290.

[12] M. Harchol-Balter, N. Bansal, B. Schroeder, and M. Agrawal, "Implementation of SRPT scheduling in web servers," 04 2001.

[13] I. Grosof, Z. Scully, and M. Harchol-Balter, "SRPT for multiserver systems," *Performance Evaluation*, vol. 127-128, pp. 154–175, 2018.

[14] R. Vaze and J. Nair, "Multiple server SRPT with speed scaling is competitive," *IEEE/ACM Transactions on Networking*, vol. 28, no. 4, pp. 1739–1751, 2020.

[15] D. Y. Barrer, "Queuing with impatient customers and ordered service," *Operations Research*, vol. 5, no. 5, pp. 650–656, 1957.

[16] D. J. Daley, "General customer impatience in the queue GI/G/1," *Journal of Applied Probability*, vol. 2, pp. 186–205, 1965.

[17] F. Baccelli, P. Boyer, and G. Hebuterne, "Single-server queues with impatient customers," *Advances in Applied Probability*, vol. 16, no. 4, p. 887–905, 1984.

[18] M. Kargahi and A. Movaghar, "A method for performance analysis of earliest-deadline-first scheduling policy," in *International Conference on Dependable Systems and Networks, 2004*, 2004, pp. 826–834.

[19] P. Moyal, "On queues with impatience: Stability, and the optimality of earliest deadline first," *Queueing Syst. Theory Appl.*, vol. 75, no. 2–4, p. 211–242, nov 2013.

[20] M. Ahmadi, M. Golkarifard, A. Movaghar, and H. Yousefi, "Processor sharing queues with impatient customers and state-dependent rates," *IEEE/ACM Transactions on Networking*, vol. 29, no. 6, pp. 2467–2477, 2021.

[21] K. Gardner, S. Borst, and M. Harchol-Balter, "Optimal scheduling for jobs with progressive deadlines," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, 2015, pp. 1113–1121.

[22] C. L. Liu and J. W. Layland, "Scheduling algorithms for multiprogramming in a hard-real-time environment," *J. ACM*, vol. 20, no. 1, p. 46–61, jan 1973.

[23] S. S. Panwar, D. Towsley, and J. K. Wolf, "Optimal scheduling policies for a class of queues with customer deadlines to the beginning of service," *J. ACM*, vol. 35, no. 4, p. 832–844, oct 1988.

[24] L.-O. Raviv and A. Leshem, "Maximizing service reward for queues with deadlines," *IEEE/ACM Transactions on Networking*, vol. 26, no. 5, pp. 2296–2308, 2018.

[25] E. Hyon and A. Jean-Marie, "Optimal control of admission in service in a queue with impatience and setup costs," *Performance Evaluation*, vol. 144, p. 102134, 2020.

[26] Z. Yu, Y. Xu, and L. Tong, "Deadline scheduling as restless bandits," *IEEE Transactions on Automatic Control*, vol. PP, pp. 1–1, 02 2018.

[27] S. Draper, M. Trott, and G. Wornell, "A universal approach to queuing with distortion control," *IEEE Transactions on Automatic Control*, vol. 50, no. 4, pp. 532–537, 2005.

[28] A. Chatterjee, D. Seo, and L. R. Varshney, "Capacity of systems with queue-length dependent service quality," *IEEE Transactions on Information Theory*, vol. 63, no. 6, pp. 3950–3963, 2017.

[29] M. Chaudhry and J. Templeton, *A First Course in Bulk Queues*, ser. A Wiley-interscience publication. Wiley, 1983, ch. 2.

[30] R. G. Gallager, *Stochastic Processes: Theory for Applications*. Cambridge University Press, 2013, ch. 5.