

---

This is an electronic reprint of the original article.  
This reprint may differ from the original in pagination and typographic detail.

Aalto, Samuli; Scully, Ziv  
**On the Gittins index for multistage jobs**

*Published in:*  
QUEUEING SYSTEMS

*DOI:*  
[10.1007/s11134-022-09760-z](https://doi.org/10.1007/s11134-022-09760-z)

Published: 01/12/2022

*Document Version*  
Publisher's PDF, also known as Version of record

*Published under the following license:*  
CC BY

*Please cite the original version:*  
Aalto, S., & Scully, Z. (2022). On the Gittins index for multistage jobs. *QUEUEING SYSTEMS*, 102(3-4), 353-371. <https://doi.org/10.1007/s11134-022-09760-z>

---

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.



# On the Gittins index for multistage jobs

Samuli Aalto<sup>1</sup>  · Ziv Scully<sup>2</sup>

Received: 22 March 2021 / Revised: 18 February 2022 / Accepted: 20 February 2022 /  
Published online: 7 April 2022  
© The Author(s) 2022

## Abstract

Optimal scheduling in single-server queueing systems is a classic problem in queueing theory. The Gittins index policy is known to be the optimal nonanticipating policy minimizing the mean delay in the  $M/G/1$  queue. While the Gittins index is thoroughly characterized for ordinary jobs whose state is described by the attained service, it is not at all the case with jobs that have more complex structure. Recently, a class of such jobs, multistage jobs, were introduced, and it was shown that the computation of Gittins index of a multistage job decomposes into separable computations for the individual stages. The characterization is, however, indirect in the sense that it relies on the recursion for an auxiliary function (the so-called SJP—single-job profit—function) and not for the Gittins index itself. In this paper, we focus on sequential multistage jobs, which have a fixed sequence of stages, and prove that, for them, it is possible to compute the Gittins index directly by recursively combining the Gittins indices of its individual stages. In addition, we give sufficient conditions for the optimality of the FCFS and SERPT disciplines for scheduling sequential multistage jobs. On the other hand, we demonstrate that, for nonsequential multistage jobs, it is better to compute the Gittins index by utilizing the SJP functions.

**Keywords** Optimal scheduling · Multistage job · Gittins index · Stochastic optimization ·  $M/G/1$

**Mathematics Subject Classification** 60K25 · 90B22 · 90B36 · 68M20

---

✉ Samuli Aalto  
samuli.aalto@aalto.fi

<sup>1</sup> Department of Communications and Networking, Aalto University, Espoo, Finland

<sup>2</sup> Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA

## 1 Introduction

We consider the optimal scheduling problem of a single-server queue with Poisson arrivals and generally distributed service times, i.e., the  $M/G/1$  queue. The objective is to minimize the mean delay (a.k.a. sojourn time or response time). The characteristics of the optimal scheduling discipline depend on the information available to the scheduler. The scheduler is said to be anticipating if it knows the remaining service times of jobs, while a nonanticipating scheduler does not know the remaining service times of jobs.

The optimal anticipating scheduling policy, shortest-remaining-processing-time (SRPT), minimizes the number of jobs even sample-pathwise for general service times and arrival processes [11, 14]. The optimal *nonanticipating* scheduling policy minimizing the mean delay in the  $M/G/1$  queue is known to be the *Gittins index* policy [2, 3, 7, 8, 12]. Gittins derived this result as a by-product of his ground-breaking results on the *multiarmed bandit*<sup>1</sup> problems [7, 8]. These problems remained unsolved for long, until Gittins proved that the structure of the optimal policy is surprisingly simple based on an index associated with each bandit separately depending just on the state of the bandit [17]. In our scheduling problem, bandits correspond to jobs: When one of them is being served, the states of the other jobs remain frozen.

The Gittins index is thoroughly characterized for ordinary jobs whose state is described by the attained service.<sup>2</sup> For example, if the service time distribution belongs to the new-better-than-used-in-expectation (NBUE) family of distributions, then the Gittins index policy coincides with the well-known first-come-first-served (FCFS) scheduling discipline, or if there are multiple job classes and the service time distributions of all classes belong to the decreasing-mean-residual-lifetime (DMRL) family, then the Gittins index policy coincides with the shortest-expected-remaining-processing-time (SERPT) discipline [2].<sup>3</sup>

While the Gittins index is thoroughly characterized for ordinary jobs, this is not the case with jobs that have more complex structure. Recently, a class of such jobs, *multistage jobs*, were introduced in [13]. A multistage job progresses through a stochastic sequence of stages, each of which has its own independent stochastic service requirement, and the job completes when its last stage completes. The state of an uncompleted multistage job is described by the pair  $(j, a)$ , where  $j$  refers to the current stage and  $a$  to the attained service in the current stage. In [13], it is shown that the computation of Gittins index of a multistage job decomposes into separable computations for the individual stages. The characterization is, however, *indirect* in the sense that it relies on the recursion for an auxiliary function (the so-called SJP function) and not for the Gittins index itself. Thus, it is natural to ask whether the characterization is possible to

---

<sup>1</sup> The multiarmed bandit problem involves a system with a finite number of “bandits.” At every time slot, the gambler chooses to activate one of the bandits, which will then yield a reward and undergo a Markovian state transition, while all the other bandits are passive and their states remain frozen. See [16] for discussion of the relationship between the multiarmed bandit problem and queue scheduling.

<sup>2</sup> Attained service (a.k.a. age) is the amount of service that the job has already received.

<sup>3</sup> Note that SERPT further reduces to FCFS in the single-class NBUE case, because in this case both disciplines prioritize jobs that have started service above those that have not.

do *more directly*: Is it possible to compute the Gittins index by recursively combining the Gittins indices of individual stages?

In this paper, we focus on *sequential* multistage jobs, which have a fixed sequence of stages, and prove that, for such jobs, it is indeed possible to compute the Gittins index directly by recursively combining the Gittins indices of individual stages. In addition, we give sufficient conditions under which FCFS is the optimal nonanticipating policy for sequential multistage jobs, and a corresponding result for SERPT when there are multiple classes of sequential multistage jobs. On the other hand, we demonstrate that, for nonsequential multistage jobs, it is usually better to compute the Gittins index by utilizing the SJP functions. We also discuss the connection between the scheduling problem of multistage jobs and the optimal control problem of tandem queues with flexible servers.

The rest of the paper is organized as follows. In Sect. 2, we recall some results concerning the Gittins index and the related Gittins index policy for ordinary jobs (consisting of a single stage). In Sect. 3, we present our main result concerning the direct computation of the Gittins index for sequential multistage jobs. As other main results, we give sufficient conditions for the optimality of the FCFS and SERPT disciplines for scheduling sequential multistage jobs. In Sect. 4, we discuss the connection between the scheduling problem of multistage jobs and the optimal control problem of tandem queues with flexible servers. In Sect. 5, we discuss the relationship between our new results and the approach based on the SJP framework. In Sect. 6, we discuss the computation of the Gittins index for nonsequential multistage jobs. Finally, in Sect. 7, we outline how one can generalize our main result to a more general model of sequential multistage jobs.

The original version of this paper, albeit very different and much more complicated due to the alternative approach used, is available in [1].

## 2 Gittins index and the optimal policy for ordinary jobs

In this preliminary section, we first recall some results (Lemmas 1, 2) related to the Gittins index for ordinary single-stage jobs. These results characterize the optimal nonanticipating policy (i.e., the Gittins index policy) for the corresponding  $M/G/1$  scheduling problem in certain cases (Propositions 1, 2).

Consider first a single ordinary job with service time  $S$ . From a nonanticipating scheduler point of view, the state of such an ordinary job is described by its *attained service*  $a$ , which increases at rate 1 whenever served, until the service is completed. As presented in [2, 3, 7, 8, 12], the corresponding *Gittins index*  $G(a)$  at state  $a \geq 0$  can be characterized as follows:

$$G(a) = \sup_{b>a} \frac{P\{S \leq b \mid S > a\}}{E[\min\{S, b\} - a \mid S > a]}. \tag{1}$$

**Remark 1** The intuition behind the Gittins index is as follows. Imagine the state  $b > a$  as a “stopping state,” and suppose we serve the job until it either completes or reaches the stopping state. In (1), the numerator is the probability the job completes

before reaching the stopping state, and the denominator is the expected amount of time we spend serving the job. The fraction in (1) can therefore be thought of as an “expected completion rate” relative to stopping state  $b$ . The Gittins index  $G(a)$  is the thus *maximum expected completion rate* attainable by optimizing the stopping state  $b$ .

The mean residual lifetime (MRL) function  $M(a)$  of a service time distribution is defined as

$$M(a) = E[S - a \mid S > a]. \quad (2)$$

Let NBUE refer to the family of service time distributions with the New-Better-than-Used-in-Expectation property, i.e.,

$$M(a) \leq M(0) \quad \text{for all } a \geq 0. \quad (3)$$

Now, we give a known result, which we utilize later on in this paper.

**Lemma 1** ([2, Prop. 7]) *The service time distribution belongs to NBUE if and only if*

$$G(a) \geq G(0) = \frac{1}{M(0)} \quad \text{for all } a \geq 0. \quad (4)$$

Moreover, if the service time distribution belongs to NBUE, then

$$G(a) \geq \frac{1}{M(a)} \quad \text{for all } a \geq 0.$$

Let then DMRL refer to the family of service time distributions with a decreasing MRL function, i.e.,

$$M(b) \leq M(a) \quad \text{for all } b \geq a. \quad (5)$$

Note that  $\text{DMRL} \subset \text{NBUE}$ . Below we give another known result, which we utilize later on in this paper.

**Lemma 2** ([2, Prop. 6]) *The service time distribution belongs to DMRL if and only if*

$$G(a) = \frac{1}{M(a)} \quad \text{for all } a \geq 0. \quad (6)$$

Thus, if the service time distribution belongs to DMRL, the Gittins index  $G(a) = 1/M(a)$  is an increasing function of  $a$ .

Let us now consider the optimal scheduling problem of an  $M/G/1$  queue loaded with ordinary jobs. Assume that the jobs are classified into  $K \geq 1$  different classes. Let *FCFS* refer to the first-come-first-served scheduling discipline, which serves customers in their arrival order, and *SERPT* to the shortest-expected-remaining-processing-time discipline, where the scheduler always serves the job with the current shortest mean residual lifetime (2). Below we give two known results related to these scheduling disciplines, which are based on the two lemmas above.

**Proposition 1** ([2, Thm. 4]) *Assume that there is just one class of ordinary jobs ( $K = 1$ ). If the service time distribution belongs to NBUE, then FCFS (or any other nonpreemptive work-conserving policy) is the optimal nonanticipating policy.*

**Proposition 2** ([2, Thm. 7]) *Assume that there are  $K$  classes of ordinary jobs ( $K \geq 2$ ). If the service time distributions for all job classes belong to DMRL, then SERPT is the optimal nonanticipating policy.*

### 3 Gittins index and the optimal policy for sequential multistage jobs

In this section, we present our main results (Theorem 1, Corollaries 1, 2, and 3) concerning the computation of the Gittins index for sequential multistage jobs. In addition, based on these results, we characterize the optimal nonanticipating policy (i.e., the Gittins index policy) for the corresponding M/G/1 scheduling problem in certain cases (Theorems 2, 3).

*Multistage jobs* were introduced in [13] as jobs consisting of a finite number of stages, say  $J$ , that form an absorbing acyclic Markov chain. The service times  $S_j$  in each stage  $j$  are assumed to be independent. From a nonanticipating scheduler point of view, the state of such a multistage job is described by the pair  $(j, a)$ , where  $j$  refers to the current stage and  $a$  to the attained service at the current stage. As noted in [13], the Gittins index  $G(j, a)$  for a multistage job is defined by a multidimensional optimization problem with one dimension per stage, which makes the direct computation of the Gittins indices intractable in general. However, below we demonstrate that this is not the case always but the direct computation is, indeed, tractable for a certain type of multistage jobs.

A multistage job with a fixed (i.e., deterministic) sequence of stages is called a *sequential multistage job*. The total service time  $S$  of such a job is given by

$$S = S_1 + \dots + S_J,$$

where  $S_j$  refers to the service time of stage  $j$  and  $J$  is the number of stages for the job. So, the service starts from stage 1, and only when the first stage is completed, the service of stage 2 starts and so on. The whole service is completed when the service of the last stage is completed. The crucial difference between a sequential multistage job and an ordinary job with service time  $S$  is that in the multistage case, the scheduler is aware of which stage  $j$  is currently in progress, as indicated by the multistage job's state being a pair  $(j, a)$ .

The Gittins index  $G(j, a)$  for the whole sequential multistage job at state  $(j, a)$  is defined by the following multidimensional optimization problem [13]:

$$G(j, a) = \sup_{b_j > a, b_{j+1}, \dots, b_J > 0} \frac{P\{S_j \leq b_j, S_{j+1} \leq b_{j+1}, \dots, S_J \leq b_J \mid S_j > a\}}{E\left[\min\{S_j, b_j\} - a + \sum_{n=j+1}^J \prod_{m=j}^{n-1} 1_{\{S_m \leq b_m\}} \min\{S_n, b_n\} \mid S_j > a\right]}. \tag{7}$$

Although the above expression appears complicated, it follows essentially the same intuition as explained for the single-stage case in Remark 1. The difference is that instead of a single stopping state, we have a “stopping set” represented by a multidimensional vector  $(b_j, b_{j+1}, \dots, b_J)$ , where we stop serving the job once it reaches state  $(i, b_i)$  for any stage  $j \leq i \leq J$ . In (7), the numerator is the probability the job completes without hitting the stopping set, and the denominator is the amount of time we spend serving the job until it completes or hits the stopping set.

Below, we show that the multidimensional optimization problem in (7) can be solved recursively stage-by-stage. So, instead of solving one (intractable) multidimensional optimization problem, we can proceed by solving a sequence of (tractable) one-dimensional problems.

**Theorem 1** *For any sequential multistage job with  $J$  stages, the Gittins index can be calculated recursively as follows:*

$$\begin{aligned}
 G(J, a) &= G_J(a) = \sup_{b>a} \frac{P\{S_J \leq b \mid S_J > a\}}{E[\min\{S_J, b\} - a \mid S_J > a]}, \\
 G(j, a) &= \sup_{b>0} \frac{P\{S_J \leq b\}}{\frac{1}{G_{\leq j-1}(j, a)} + E[\min\{S_J, b\}]}, \quad j \in \{1, 2, \dots, J - 1\},
 \end{aligned}
 \tag{8}$$

where  $G_J(a)$  denotes the Gittins index of a single-stage job with service time  $S_J$ , and  $G_{\leq j-1}(j, a)$  denotes the Gittins index at state  $(j, a)$  of a sequential multistage job consisting of stages  $1, 2, \dots, J - 1$ .<sup>4</sup>

**Proof** Consider a sequential multistage job with  $J$  stages. For the last stage  $J$ , the claim  $G(J, a) = G_J(a)$  is trivially true. Let then  $j \in \{1, 2, \dots, J - 1\}$ . According to (7), the Gittins index  $G(j, a)$  at state  $(j, a)$  reads as

$$G(j, a) = \sup_{b_j>a, b_{j+1}, \dots, b_J>0} \frac{P(b_j, b_{j+1}, \dots, b_J \mid a)}{E[b_j, b_{j+1}, \dots, b_J \mid a]},$$

where we have used the following shorthand notations:

$$\begin{aligned}
 &P(b_j, b_{j+1}, \dots, b_J \mid a) \\
 &= P\{S_j \leq b_j, S_{j+1} \leq b_{j+1}, \dots, S_J \leq b_J \mid S_j > a\}, \\
 &E[b_j, b_{j+1}, \dots, b_J \mid a] \\
 &= E[\min\{S_j, b_j\} - a + \sum_{n=j+1}^J \prod_{m=j}^{n-1} 1_{\{S_m \leq b_m\}} \min\{S_n, b_n\} \mid S_j > a].
 \end{aligned}$$

<sup>4</sup> We note that the Gittins index of a job in stage  $j$  depends only on the service time distributions of stage  $j$  and later stages. This means that we could equivalently define the recursive case using stages  $j, j + 1, \dots, J - 1$ , or more generally a sequence starting at any stage  $i \leq j$ , instead of starting at stage 1. We make the latter choice for simplicity of the  $G_{\leq j-1}$  notation.

Since the stages are independent, we have

$$\begin{aligned}
 &P(b_j, b_{j+1}, \dots, b_J \mid a) \\
 &= P(b_j, b_{j+1}, \dots, b_{J-1} \mid a)P\{S_J \leq b_J\}, \\
 &E[b_j, b_{j+1}, \dots, b_J \mid a] \\
 &= E[b_j, b_{j+1}, \dots, b_{J-1} \mid a] + P(b_j, b_{j+1}, \dots, b_{J-1} \mid a)E[\min\{S_J, b_J\}].
 \end{aligned}$$

Thus,

$$G(j, a) = \sup_{b_j > a, b_{j+1}, \dots, b_J > 0} \frac{P(b_j, b_{j+1}, \dots, b_{J-1} \mid a)P\{S_J \leq b_J\}}{E[b_j, b_{j+1}, \dots, b_{J-1} \mid a] + P(b_j, b_{j+1}, \dots, b_{J-1} \mid a)E[\min\{S_J, b_J\}]},$$

which is equivalent to

$$\begin{aligned}
 G(j, a) &= \sup_{b_j > 0} \frac{P\{S_J \leq b_j\}}{\inf_{b_j > a, b_{j+1}, \dots, b_{J-1} > 0} \frac{E[b_j, b_{j+1}, \dots, b_{J-1} \mid a]}{P(b_j, b_{j+1}, \dots, b_{J-1} \mid a)} + E[\min\{S_J, b_j\}]} \\
 &= \sup_{b_j > 0} \frac{P\{S_J \leq b_j\}}{\frac{1}{G_{\leq J-1}(j, a)} + E[\min\{S_J, b_j\}]}.
 \end{aligned}$$

This completes the proof. □

By applying Lemma 3 (presented in Appendix A) to (8) with the choice  $u = G_{\leq J-1}(j, a)$ , we get the following corollary.

**Corollary 1** Consider a sequential multistage job with  $J$  stages for which the service time of the last stage belongs to NBUE. Now, for any  $a \geq 0$ ,

$$\begin{aligned}
 G(J, a) &= G_J(a), \\
 G(j, a) &= \frac{1}{\frac{1}{G_{\leq J-1}(j, a)} + E[S_J]}, \quad j \in \{1, 2, \dots, J - 1\},
 \end{aligned} \tag{9}$$

where  $G_{\leq J-1}(j, a)$  denotes the Gittins index at state  $(j, a)$  of a sequential multistage job consisting of stages  $1, 2, \dots, J - 1$ .

By continuing and combining this corollary with Lemma 1 presented in Sect. 2 (and the remark made thereafter), we get another corollary.

**Corollary 2** Consider a sequential multistage job with  $J$  stages for which the service time of each stage belongs to NBUE. Now, for any  $j \in 1, 2, \dots, J$  and  $a \geq 0$ ,

$$G(j, a) \geq G(1, 0) = \frac{1}{E[S]}. \tag{10}$$



By further continuing and combining Corollary 1 with Lemma 2 presented in Sect. 2, we get still another corollary.

**Corollary 3** *Consider a sequential multistage job with  $J$  stages for which the service time of each stage belongs to DMRL. Now, for any  $j \in 1, 2, \dots, J$  and  $a \geq 0$ ,*

$$G(j, a) = \frac{1}{M(j, a)}, \quad (11)$$

where  $M(j, a)$  is the MRL function for the sequential multistage job at state  $(j, a)$ ,

$$M(j, a) = E[S_j - a \mid S_j > a] + E[S_{j+1}] + \dots + E[S_J]. \quad (12)$$

Let us now consider the optimal scheduling problem of an M/G/1 queue loaded with sequential multistage jobs. Assume that the jobs are classified into  $K \geq 1$  different classes, and service time of a class- $k$  job consists of  $J_k$  stages,

$$S_k = S_{k,1} + \dots + S_{k,J_k}.$$

Below we give two novel results, which follow immediately from the two last corollaries above.

**Theorem 2** *Assume that there is just one class of sequential multistage jobs ( $K = 1$ ). If the service time distributions of all stages belong to NBUE, then FCFS (or any other nonpreemptive work-conserving policy) is the optimal nonanticipating policy.*

**Theorem 3** *Assume that there are  $K$  classes of sequential multistage jobs ( $K \geq 2$ ). If the service time distributions of all stages in each job class belong to DMRL, then SERPT is the optimal nonanticipating policy.*

We conclude this section with two remarks about the above results. First, the  $K = 1$  case of Theorem 3 is a special case of Theorem 2, because DMRL is a subset of NBUE. Second, under the preconditions of Theorem 2, namely in the  $K = 1$  NBUE case, the SERPT policy, like Gittins, also coincides with FCFS. This is because in this case, SERPT, like Gittins, prioritizes jobs that have started service above those that have not.

## 4 Connection to the optimal control of tandem queues

In this section, we discuss the connection between the scheduling problem of multistage jobs and the optimal control problem of tandem queues with flexible servers.

Assume that there is just one class of sequential multistage jobs ( $K = 1$ ). In this case, the scheduling problem of multistage jobs is in fact equivalent to preemptively assigning a single server to a job at a station in a tandem queue consisting of one server,  $K$  stations, and infinite buffers at each station. Or, as well, it is equivalent to preemptively assigning multiple servers that are *fully flexible* (i.e., able to server any job at any station) and *collaborative* (i.e., able to serve simultaneously the same job).

Such optimal control problems of tandem queues have been studied in [4–6, 9, 10, 15].

In [5, 10, 15], the aim is to find the optimal policy  $\pi$  that maximizes the departure process  $D^\pi(t)$  (from the last station) either stochastically or pathwise. Note that maximizing the departure process is equivalent to minimizing the total number of jobs,  $X^\pi(t)$ , for any  $t$ , since  $X^\pi(t) = A(t) - D^\pi(t)$ , where  $A(t)$  refers to the arrival process of jobs at the first station, which is naturally independent of the control policy applied. In all these papers, the optimal policy, under certain additional assumptions, proves to be such that all the jobs arriving in the first station are served nonpreemptively one-by-one through all the stations so that there is, at any time, at most one job in service in the whole system. In [10], this optimal policy is called CEF (closest-to-exit-first), while, in [15], it is named the expedite policy, and LBFS (last-buffer-first-served) in [5]. Clearly this optimal policy, called Expedite below, corresponds to FCFS for sequential multistage jobs. Thus, our Theorem 2 together with the well-known Little's formula gives the following optimality result.

**Corollary 4** *Consider a tandem queue with Poisson arrivals and fully flexible and collaborative servers. Expedite minimizes the mean total delay (i.e., the total sojourn time through all the stations), as well as the mean total number of jobs in the system, if the service times in all stations belong to NBUE.*

This result does not follow from the previous ones, since the related results in [10, Theorem 2] and [5, Theorem 4.3] are restricted to exponential service times in all stations, and optimization in [15, Theorem 1] is restricted to the subset of policies for which the departure order of jobs is required to be the same as their arrival order.

In [4, 6, 9], the objective is to minimize the mean holding costs of jobs, possibly together with switching costs of servers. Our multistage job scheduling problem covers the models of these papers in the special case where holding costs remain the same in each station and there are no switching costs nor any switching delays. It follows from [9, Proposition 3] that Expedite is optimal in this special case if  $K = 2$  and optimization is restricted to the subset of nonpreemptive policies. When preemptive policies and multiple stations  $K \geq 2$  are allowed, [6, Theorem 1] implies that, in this special case, the last station  $K$  is served in a greedy and exhaustive manner, which is in line with Expedite. A similar characterization is presented in [4, Theorem 3.2] for exponential service times and  $K = 2$ . On the other hand, our Theorem 2 has now the following implication, which does not follow from the previous results.

**Corollary 5** *Consider a tandem queue with Poisson arrivals and fully flexible and collaborative servers, where holding costs remain the same in each station and there are no switching costs nor any switching delays. In this special case, Expedite minimizes the mean holding costs if the service times in all stations belong to NBUE.*

Finally, we note that, for any stationwise service time distributions and even when there are multiple classes of jobs, the Gittins index policy, in which all servers are assigned to the job with the highest index, is optimal in a tandem queue with Poisson arrivals and fully flexible and collaborative servers minimizing (among all nonanticipating control policies)

- (i) the mean total delay and the mean total number of jobs in any case;
- (ii) the mean holding costs in the special case mentioned above in Corollary 5.

In addition, our Theorem 1 describes how to recursively compute the corresponding Gittins index.

### 5 Relationship to the SJP framework

While the original method to compute the Gittins index for ordinary jobs (as presented in Sect. 2) was based on the work of Gittins himself, an interesting alternative method has recently been presented in [13], where the so-called *SJP (single-job profit) framework* is introduced. Given attained service  $a$  and an arbitrary parameter  $r \geq 0$ , one defines the *SJP function* as

$$V(a; r) = \sup_{b>a} \left( r P\{S \leq b \mid S > a\} - E[\min\{S, b\} - a \mid S > a] \right). \tag{13}$$

As a supremum of convex increasing functions of  $r$ , the SJP function has a (partial) inverse with respect to its second argument:

$$V^{-1}(a; u) = \inf\{r \geq 0 : V(a; r) > u\}.$$

One can define the Gittins index in terms of the inverse SJP function as

$$G(a) = \frac{1}{V^{-1}(a; 0)}. \tag{14}$$

The SJP framework can also be utilized to compute the Gittins index for multistage jobs. In this section, we discuss the relationship between the SJP approach and our new approach.

Consider a sequential multistage job that consists of  $J$  stages. According to [13], the Gittins index  $G(j, a)$  at state  $(j, a)$  satisfies

$$G(j, a) = \frac{1}{V^{-1}(j, a; 0)}, \tag{15}$$

where  $V^{-1}(j, a; \cdot)$  is the partial inverse of  $V(j, a; \cdot)$ , the SJP function at state  $(j, a)$ , which is in turn defined by the following multidimensional optimization problem:

$$\begin{aligned}
 V(j, a; r) = & \sup_{b_j>a, b_{j+1}, \dots, b_J>0} \\
 & \left( r P\{S_j \leq b_j, S_{j+1} \leq b_{j+1}, \dots, S_J \leq b_J \mid S_j > a\} \right. \\
 & \left. - E \left[ \min\{S_j, b_j\} - a + \sum_{n=j+1}^J \prod_{m=j}^{n-1} 1_{\{S_m \leq b_m\}} \min\{S_n, b_n\} \mid S_j > a \right] \right). \tag{16}
 \end{aligned}$$

In [13], a *composition law* is proved, which reduces the computation of the SJP function (and, thus, the Gittins index, as well) for *any* multistage job into separable SJP computations for the individual stages. Below we formulate this composition law for sequential multistage jobs.

**Proposition 3** ([13, Thm. 4.2]) *For any sequential multistage job with  $J$  stages, the SJP function can be calculated recursively as follows:*

$$\begin{aligned} V(J, a; r) &= V_J(a; r), \\ V(j, a; r) &= V_{\leq J-1}(j, a; V_J(0; r)), \quad j \in \{1, 2, \dots, J - 1\}, \end{aligned} \tag{17}$$

where  $V_{\leq J-1}(j, a; \cdot)$  refers to the SJP function for a sequential multistage job consisting of stages  $1, 2, \dots, J - 1$  at state  $(j, a)$ , and  $V_J(0; \cdot)$  refers to the SJP function for a single-stage job consisting just of stage  $J$  at state  $0$ .

By iterating Proposition 3 and taking inverses, one can transform (15) into nested single-stage inverse SJP functions:

$$\begin{aligned} G(j, a) &= \frac{1}{V_J^{-1}(0; V_{J-1}^{-1}(0; \dots V_{j+1}^{-1}(0; V_j^{-1}(a; 0)) \dots))} \\ &= \frac{1}{V_J^{-1}(0; V_{J-1}^{-1}(0; \dots V_{j+1}^{-1}(0; 1/G_j(a)) \dots))}. \end{aligned} \tag{18}$$

While (18) characterizes the Gittins index for a sequential multistage job *indirectly* via the inverse of the SJP function, our novel result presented in Theorem 1 gives a *more direct* computation method: It enables computing the Gittins index for a sequential multistage job by solving a single one-dimensional optimization problem for each stage.

The alternative perspectives offered by (18) and Theorem 1 can be reconciled. Underlying Theorem 1 is a direct way of computing the inverse SJP function of a single-stage job. Specifically, using reasoning similar to the proof of Theorem 1, one can show

$$V^{-1}(a; u) = \inf_{b > a} \frac{u + E[\min\{S, b\} - a \mid S > a]}{P\{S \leq b \mid S > a\}}.$$

For completeness, we provide a proof in Proposition 4 (presented in Appendix B).

### 6 Gittins index for nonsequential multistage jobs

In this section, we discuss whether it is useful or at least possible to generalize Theorem 1 to *any* multistage jobs. There is one type of nonsequential job, or more accurately sequential jobs with possible skips, to which Theorem 1 easily applies (Corollary 6). But in general, generalizing Theorem 1 is possible but not that useful as demonstrated via the examples given below (Examples 1, 2).

Let us define a *sequential job with possible skips* as follows: It consists of stages  $1, \dots, J$ , but each stage  $j$  is included only with probability  $p_j$  and skipped with probability  $1 - p_j$ , as revealed to the scheduler upon reaching stage  $j$ . Now, we can apply Theorem 1 to such a job because we can think of it as an ordinary sequential job: Simply give each stage  $j$  probability  $1 - p_j$  of having service time equal to 0. Thus, we have the following corollary of Theorem 1.

**Corollary 6** *For any sequential multistage job with possible skips, the Gittins index can be calculated recursively by applying Eq. (8) as follows: Replace the original service times  $S_j$  with the modified service times  $\tilde{S}_j$  that take value 0 with probability  $1 - p_j$  and follow the distribution of the original service time  $S_j$  with probability  $p_j$ .*

**Example 1** (*Random number of stages*) Consider now a stochastic multistage job that has a random number of stages,  $Z \in \{1, 2, 3\}$ , where the random variable  $Z$  is independent of everything. Let  $q_j$  denote the probability that the job consists of exactly first  $j$  stages  $1, \dots, j$ , where  $j \in \{1, 2, 3\}$ . Clearly, the Gittins index  $G(j, a)$  for stage 3 is just the same as the Gittins index (1) for the corresponding single-stage job:

$$G(3, a) = G_3(a) = \sup_{b>a} \frac{P\{S_3 \leq b \mid S_3 > a\}}{E[\min\{S_3, b\} - a \mid S_3 > a]}.$$

For the second stage  $j = 2$ , the Gittins index  $G(2, a)$  is defined by the following multidimensional optimization problem:

$$G(2, a) = \sup_{b_2>a, b_3>0} \frac{P\{Z = 2, S_2 \leq b_2 \mid Z \geq 2, S_2 > a\} + P\{Z = 3, S_2 \leq b_2, S_3 \leq b_3 \mid Z \geq 2, S_2 > a\}}{E[\min\{S_2, b_2\} - a + 1_{\{Z=3\}} 1_{\{S_2 \leq b_2\}} \min\{S_3, b_3\} \mid Z \geq 2, S_2 > a]}.$$

Since all random variables are independent, it follows, after some manipulation, that

$$\begin{aligned} G(2, a) &= \sup_{b_3>0} \frac{P\{Z = 2 \mid Z \geq 2\} + P\{Z = 3 \mid Z \geq 2\}P\{S_3 \leq b_3\}}{\inf_{b_2>a} \frac{E[\min\{S_2, b_2\} - a \mid S_2 > a]}{P\{S_2 \leq b_2 \mid S_2 > a\}} + P\{Z = 3 \mid Z \geq 2\}E[\min\{S_3, b_3\}]} \\ &= \sup_{b_3>0} \frac{P\{Z = 2 \mid Z \geq 2\} + P\{Z = 3 \mid Z \geq 2\}P\{S_3 \leq b_3\}}{\frac{1}{G_2(a)} + P\{Z = 3 \mid Z \geq 2\}E[\min\{S_3, b_3\}]} \end{aligned}$$

Thus, the computation of the Gittins index for stage 2 reduces to a sequence of separate one-dimensional optimization problems.

Consider then the first stage  $j = 1$ , for which the Gittins index  $G(1, a)$  is defined by the following multidimensional optimization problem:

$$G(1, a) = \sup_{b_1>a, b_2, b_3>0} \frac{\sum_{n=1}^3 P\{Z = n, S_1 \leq b_1, \dots, S_n \leq b_n \mid S_1 > a\}}{E[\min\{S_1, b_1\} - a + \sum_{n=2}^3 1_{\{Z \geq n\}} \prod_{m=1}^{n-1} 1_{\{S_m \leq b_m\}} \min\{S_n, b_n\} \mid S_1 > a]}.$$

Since all random variables are independent, it follows, after some manipulation, that

$$\begin{aligned}
 G(1, a) &= \sup_{b_2, b_3 > 0} \frac{\sum_{n=1}^3 P\{Z = n\} \prod_{m=2}^n P\{S_m \leq b_m\}}{\inf_{b_1 > a} \frac{E[\min\{S_1, b_1\} - a | S_1 > a]}{P\{S_1 \leq b_1 | S_1 > a\}} + \sum_{n=2}^3 P\{Z = n\} \prod_{m=2}^{n-1} P\{S_m \leq b_m\} E[\min\{S_n, b_n\}]} \\
 &= \sup_{b_2, b_3 > 0} \frac{\sum_{n=1}^3 P\{Z = n\} \prod_{m=2}^n 1_{\{S_m \leq b_m\}}}{\frac{1}{G_1(a)} + \sum_{n=2}^3 P\{Z = n\} \prod_{m=2}^{n-1} P\{S_m \leq b_m\} E[\min\{S_n, b_n\}]}
 \end{aligned}$$

As we see from above, unlike for stages 2 and 3, the computation of the Gittins index for stage 1 does *not* reduce to separate one-dimensional optimization problems, but, in one optimization problem, there are as many dimensions as there are remaining possible stages.

On the other hand, by utilizing the SJP framework and the related composition law developed in [13], we show below that the computation of the Gittins index for stage 1 can be reduced to separate one-dimensional optimization problems, which makes the computation much more tractable.

According to [13], the Gittins index  $G(1, a)$  at state  $(1, a)$  satisfies

$$G(1, a) = \frac{1}{V^{-1}(1, a; 0)} = \frac{1}{\inf\{r \geq 0 : V(1, a; r) > 0\}},$$

where  $V(1, a; r)$  denotes the SJP function at state  $(1, a)$  defined by the following multidimensional optimization problem:

$$\begin{aligned}
 V(1, a; r) &= \sup_{b_1 > a, b_2, b_3 > 0} \left( r \sum_{n=1}^3 P\{Z = n, S_1 \leq b_1, \dots, S_n \leq b_n \mid S_1 > a\} \right. \\
 &\quad \left. - E \left[ \min\{S_1, b_1\} - a + \sum_{n=2}^3 1_{\{Z \geq n\}} \prod_{m=1}^{n-1} 1_{\{S_m \leq b_m\}} \min\{S_n, b_n\} \mid S_1 > a \right] \right).
 \end{aligned}$$

However, by [13, Cor. 4.5], we have the following composition for  $V(1, a; r)$ , which modifies this multidimensional optimization problem into a sequence of separate one-dimensional optimization problems:

$$\begin{aligned}
 V(1, a; r) &= V_1(a; P\{Z = 1\}r + P\{Z \geq 2\}V_2(0; P\{Z = 2 \mid Z \geq 2\}r \\
 &\quad + P\{Z = 3 \mid Z \geq 2\}V_3(0; r)),
 \end{aligned}$$

where  $V_j(a; \cdot)$  refers to the SJP function for a single-stage job consisting just of stage  $j$ ,  $j \in \{1, 2, 3\}$ . □

**Example 2** (*Parallel stages with random routing*) Consider now another multistage job, which always starts with stage 1 but thereafter comes either stage 2 or 3 depending on a random and independent routing decision  $Z \in \{2, 3\}$ . Clearly, the Gittins index  $G(j, a)$  for stages  $j \in \{2, 3\}$  is just the same as the Gittins index (1) for the corresponding single-stage job:

$$G(j, a) = G_j(a) = \sup_{b>a} \frac{P\{S_j \leq b \mid S_j > a\}}{E[\min\{S_j, b\} - a \mid S_j > a]}, \quad j \in \{2, 3\}.$$

But, for the first stage  $j = 1$ , the Gittins index  $G(1, a)$  is defined by the following multidimensional optimization problem:

$$G(1, a) = \sup_{b_1>a, b_2, b_3>0} \frac{\sum_{n=2}^3 P\{Z = n, S_1 \leq b_1, S_n \leq b_n \mid S_1 > a\}}{E[\min\{S_1, b_1\} - a + \sum_{n=2}^3 1_{\{Z=n\}} 1_{\{S_1 \leq b_1\}} \min\{S_n, b_n\} \mid S_1 > a]}.$$

Since the stages and the routing decision are independent, it follows, after some manipulation, that

$$\begin{aligned} G(1, a) &= \sup_{b_2, b_3>0} \frac{\sum_{n=2}^3 P\{Z = n\} P\{S_n \leq b_n\}}{\inf_{b_1>a} \frac{E[\min\{S_1, b_1\} - a \mid S_1 > a]}{P\{S_1 \leq b_1 \mid S_1 > a\}} + \sum_{n=2}^3 P\{Z = n\} E[\min\{S_n, b_n\}]} \\ &= \sup_{b_2, b_3>0} \frac{\sum_{n=2}^3 P\{Z = n\} P\{S_n \leq b_n\}}{\frac{1}{G_1(a)} + \sum_{n=2}^3 P\{Z = n\} E[\min\{S_n, b_n\}]} \end{aligned}$$

As we see from above, unlike for stages 2 and 3, the computation of the Gittins index for stage 1 does *not* reduce to separate one-dimensional optimization problems, but, in one optimization problem, there are as many dimensions as there are routing options after the first stage.

On the other hand, by utilizing the SJP framework and the related composition law developed in [13], we show below that the computation of the Gittins index for stage 1 can be reduced to separate one-dimensional optimization problems, which makes the computation much more tractable.

According to [13], the Gittins index  $G(1, a)$  at state  $(1, a)$  satisfies

$$G(1, a) = \frac{1}{V^{-1}(1, a; 0)} = \frac{1}{\inf\{r \geq 0 : V(1, a; r) > 0\}},$$

where  $V(1, a; r)$  denotes the SJP function at state  $(1, a)$  defined by the following multidimensional optimization problem:

$$\begin{aligned}
 V(1, a; r) = & \sup_{b_1 > a, b_2, b_3 > 0} \\
 & \left( r \sum_{n=2}^3 P\{Z = n, S_1 \leq b_1, S_n \leq b_n \mid S_1 > a\} \right. \\
 & \left. - E \left[ \min\{S_1, b_1\} - a + 1_{\{S_1 \leq b_1\}} \sum_{n=2}^3 1_{\{Z=n\}} \min\{S_n, b_n\} \mid S_1 > a \right] \right).
 \end{aligned}$$

However, by [13, Cor. 4.5], we have the following composition for  $V(1, a; r)$ , which modifies this multidimensional optimization problem into a sequence of separate one-dimensional optimization problems:

$$V(1, a; r) = V_1(a; P\{Z = 2\}V_2(0; r) + P\{Z = 3\}V_3(0; r)),$$

where  $V_j(a; \cdot)$  refers to the SJP function for a single-stage job consisting just of stage  $j$ ,  $j \in \{1, 2, 3\}$ . □

To conclude, for sequential multistage jobs (even with possible skips), Theorem 1 gives a direct way to compute the Gittins index, but, for the other types of multistage jobs, it is better to do the computation by utilizing the SJP framework.

### 7 Generalizing to the case of complex dynamics within each stage

In [12], a Markov-process job model was proposed that generalizes the multistage job model studied in this paper. Below, we outline how one can generalize Theorem 1 to this more general job model.

We begin with a brief discussion of the Markov-process job model. In the model, each job is an absorbing Markov process  $X(t)$  on a general state space. A job’s state evolves only during service, and a job completes when it reaches an absorbing state  $*$ . One can define the Gittins index of a state  $x$  following the same intuition as in Remark 1. The main difference is that the stopping set is a general set of states  $Y$ . Letting  $\tau^*(Y)$  denote the hitting time of  $Y \cup \{*\}$ , we have, roughly speaking,<sup>5</sup>

$$G(x) = \sup_Y \frac{P\{X(\tau^*(Y)) = * \mid X(0) = x\}}{E[\tau^*(Y) \mid X(0) = x]}. \tag{19}$$

Consider now a sequential multistage job where each stage, instead of being an ordinary job, is an absorbing Markov process. Let  $X_j(t)$  be the  $j$ th stage’s Markov process. The state of a multistage job is then a pair  $(j, x_j)$ , denoting being in state  $x_j$  of the  $j$ th stage’s Markov process. When a job would reach a state  $(j, *)$ , it instead jumps to the (possibly random) initial state  $(j + 1, X_{j+1}(0))$  of the next stage, unless  $j = J$ , in which case the job completes.

<sup>5</sup> We elide some measure-theoretic technicalities.



Our main result, Theorem 1, can be generalized to sequential multistage jobs with Markov-process stages:

$$G(J, x_J) = G_J(x_J)$$

$$G(j, x_j) = \sup_{Y_j} \frac{P\{X_J(\tau_j^*(Y_j)) = *\}}{\frac{1}{G_{\leq j-1}(j, x_j)} + E[\tau_j^*(Y_j)]}, \quad j \in \{1, 2, \dots, J-1\},$$

where  $Y_j$  and  $\tau_j^*(Y_j)$  denote sets of states and hitting times, respectively, for stage  $J$ 's Markov process, and, as usual,  $G_{\leq j-1}(j, x_j)$  denotes the Gittins index of  $(j, x_j)$  in the sequential multistage job with stages  $1, 2, \dots, J-1$ . The proof is essentially the same as that of Theorem 1, with tuples of real numbers  $(b_j, b_{j+1}, \dots, b_J)$  replaced by tuples of sets of states  $(Y_j, Y_{j+1}, \dots, Y_J)$ .

The main takeaway is that one can recursively compute the Gittins index of any job which can be decomposed into multiple sequential stages, even if those stages have more complicated dynamics than ordinary jobs. However, the practical usefulness of this more general result rests on the ability to compute Gittins indices of the individual stages. Such computation of Gittins indices requires optimizing over subsets of states  $Y$ , see (19), which is likely difficult to do in general.

**Acknowledgements** Ziv Scully was supported by NSF Grant Nos. CMMI-1938909 and CSR-1763701. We thank the anonymous reviewer who pointed out the connection between the scheduling problem of sequential multistage jobs and the optimal control problem of tandem queues with flexible servers.

**Funding** Open Access funding provided by Aalto University.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## A Auxiliary result for single-stage jobs with NBUE service times

This appendix presents an auxiliary lemma which implies Corollary 1.

Consider a single-stage job with service time denoted by positive random variable  $S$ . In this appendix, we consider the case where  $S$  belongs to NBUE and the job is in state 0 (i.e., it has not yet entered service).

Define a function  $\tilde{G}(u)$  as follows:

$$\tilde{G}(u) = \sup_{b>0} \frac{P\{S \leq b\}}{u + E[\min\{S, b\}]}, \quad u \geq 0. \quad (20)$$

The significance of  $\tilde{G}(\cdot)$  is that it is the “outer part” of the recursive Gittins index formula presented in Theorem 1. In the terminology of Sect. 5,  $\tilde{G}(\cdot)$  is the reciprocal of the inverse SJP function (see also Appendix B).

We now characterize  $\tilde{G}(u)$  in the case where  $S$  belongs to NBUE.

**Lemma 3** *If the service time distribution  $S$  belongs to NBUE, then, for all  $u \geq 0$ ,*

$$\tilde{G}(u) = \frac{1}{u + E[S]}. \tag{21}$$

**Proof** Assume that the service time distribution belongs to NBUE. Let  $u \geq 0$ , and define an auxiliary function  $\tilde{J}(b; u)$  as follows:

$$\tilde{J}(b; u) = \frac{P\{S \leq b\}}{u + E[\min\{S, b\}]}. \quad b > 0, \tag{22}$$

In addition, let  $\tilde{J}(\infty; u)$  denote the limit

$$\tilde{J}(\infty; u) = \lim_{b \rightarrow \infty} \tilde{J}(b; u) = \frac{1}{u + E[S]}. \tag{23}$$

Since  $\tilde{G}(u) = \sup_{b>0} \tilde{J}(b; u)$ , it is sufficient to prove that

$$\tilde{J}(b; u) \leq \tilde{J}(\infty; u) \quad \text{for all } b > 0, \tag{24}$$

which follows because  $u \geq 0$  and

$$E[S] = M(0) \geq M(b) = E[S - b \mid S > b] = \frac{E[S] - E[\min\{S, b\}]}{P\{S > b\}}.$$

The inequality above is justified by the NBUE assumption. □

### B Direct formula for the inverse SJP function

As discussed at the end of Sect. 5, one way to view Theorem 1 is as providing a direct way of computing a single-stage job’s inverse SJP function. We formalize this perspective in Proposition 4.

Recall from Sect. 5 the definition of the SJP function  $V(a; r)$  of a single-stage job:

$$V(a; r) = \sup_{b>a} \left( r P\{S \leq b \mid S > a\} - E[\min\{S, b\} - a \mid S > a] \right). \tag{25}$$

According to [13, Lemma 3.2], the SJP function  $V(a; r)$  is a convex and nondecreasing function of  $r$ , so it has an inverse  $V^{-1}(0; u)$  defined for all  $u \geq 0$  by

$$V^{-1}(a; u) = \inf\{r \geq 0 : V(a; r) > u\}. \tag{26}$$

We show below that the inverse SJP function can be computed by solving a one-dimensional optimization problem.

**Proposition 4** For all  $a, u \geq 0$ ,

$$V^{-1}(a; u) = \inf_{b>a} \frac{u + E[\min\{S, b\} - a \mid S > a]}{P\{S \leq b \mid S > a\}}.$$

**Proof** Let  $r \geq 0$ . It suffices to show that

$$r > \tilde{V}^{-1}(a; u) \quad (27)$$

holds if and only if

$$r > \inf_{b>a} \frac{u + E[\min\{S, b\} - a \mid S > a]}{P\{S \leq b \mid S > a\}}. \quad (28)$$

First, (28) is, by definition, equivalent to the condition that there exists  $b > 0$  such that

$$r > \frac{u + E[\min\{S, b\} - a \mid S > a]}{P\{S \leq b \mid S > a\}}.$$

This is clearly equivalent to the condition that there exists  $b > 0$  such that

$$rP\{S \leq b\} - E[\min\{S, b\} \mid S > a] > u,$$

which, by definition, is equivalent to (27).  $\square$

## References

1. Aalto, S.: Characterization of the Gittins index for sequential multistage jobs. [arXiv:2103.10646v1](https://arxiv.org/abs/2103.10646v1) (2021)
2. Aalto, S., Ayesta, U., Righter, R.: On the Gittins index in the M/G/1 queue. *Queueing Syst.* **63**, 437–458 (2009)
3. Aalto, S., Ayesta, U., Righter, R.: Properties of the Gittins index with application to optimal scheduling. *Probab. Eng. Inf. Sci.* **25**, 269–288 (2011)
4. Ahn, H.-S., Duenyas, I., Lewis, M.E.: The optimal control of a two-stage tandem queueing system with flexible servers. *Probab. Eng. Inf. Sci.* **16**, 453–469 (2002)
5. Ahn, H.-S., Righter, R.: Dynamic load balancing with flexible workers. *Adv. Appl. Probab.* **38**, 621–642 (2006)
6. Duenyas, I., Gupta, D., Olsen, T.: Control of a single server tandem queueing system with setups. *Oper. Res.* **46**, 218–230 (1998)
7. Gittins, J.: *Multi-armed Bandit Allocation Indices*. Wiley, New York (1989)
8. Gittins, J., Glazebrook, K., Weber, R.: *Multi-armed Bandit Allocation Indices*, 2nd edn. Wiley, New York (2011)
9. Iravani, S.M.R., Posner, M.J.M., Buzacott, J.A.: A two-stage tandem queue attended by a moving server with holding and switching costs. *Queueing Syst.* **26**, 203–228 (1997)
10. Johri, P.K., Katehakis, M.N.: Scheduling service in tandem queues attended by a single server. *Stoch. Anal. Appl.* **6**, 279–288 (1988)

11. Schrage, L.: A proof of the optimality of the shortest remaining processing time discipline. *Oper. Res.* **16**, 687–690 (1968)
12. Scully, Z., Harchol-Balter, M.: The Gittins policy in the  $M/G/1$  queue. In: *Proceedings of WiOpt (2021)*
13. Scully, Z., Harchol-Balter, M., Scheller-Wolf, A.: Optimal scheduling and exact response time analysis for multistage jobs. [arXiv:1805.06865v2](https://arxiv.org/abs/1805.06865v2) (2018)
14. Smith, D.: A new proof of the optimality of the shortest remaining processing time discipline. *Oper. Res.* **26**, 197–199 (1978)
15. Van Oyen, M.P., Gel, E.G.S., Hopp, W.J.: Performance opportunity for workforce agility in collaborative and noncollaborative work systems. *IIE Trans.* **33**, 761–777 (2001)
16. Whittle, P.: Applied probability in Great Britain. *Oper. Res.* **50**, 227–239 (2002)
17. Whittle, P.: Tax problems in the undiscounted case. *J. Appl. Probab.* **42**, 754–765 (2005)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.