http://pubsonline.informs.org/journal/stsy

STOCHASTIC SYSTEMS
Articles in Advance, pp. 1–2
ISSN 1946-5238 (online)

Special Section: Open Problems in Applied Probability

Open Problem—M/G/1 Scheduling with Preemption Delays

Ziv Scully[a]

[a] Computer Science Department, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213
Contact: zscully@cs.cmu.edu, https://orcid.org/0000-0002-8547-1068 (ZS)

Published Online in Articles in Advance:
September 17, 2019

https://doi.org/10.1287/stsy.2019.0047

History: This paper was accepted for the *Stochastic Systems* Special Section on Open Problems in Applied Probability, presented at the 2018 INFORMS Annual Meeting in Phoenix, Arizona, November 4–7, 2018.

Funding: Z. Scully was supported by an ARCS Foundation scholarship and the National Science Foundation Graduate Research Fellowship Program [Grants DGE-1745016 and DGE-125222] and partially supported by the National Science Foundation [Grants CSR-1763701 and XPS-1629444] and a 2018 Microsoft Faculty Award via his advisor Mor Harchol-Balter.

Keywords: stochastic networks • control • optimization • scheduling • M/G/1

The problem of preemptively scheduling jobs in the M/G/1 queue to minimize mean response time is a classic problem in computer systems. Typically, one assumes that the time it takes to preempt a job is negligible, in which case the *shortest remaining processing time* (SRPT) policy is optimal when job sizes are known (Schrage 1968), and the *Gittins index* (GI) policy is optimal when job sizes are unknown (von Olivier 1972, Gittins et al. 2011).

Unfortunately, the assumption that preemption takes negligible time is not always reasonable. For example, switching between data-intensive jobs may require moving a significant amount data between RAM (short-term memory) and disk (long-term data storage). This motivates the following question: what scheduling policy minimizes mean response time of an M/G/1 queue in which *preemption takes nonnegligible time*? This is an open problem for both known and unknown sizes.

Known Job Sizes

In the case of known job sizes, the natural approach is to somehow adapt SRPT to incorporate preemption delays. This approach was studied by Goerg (1986), who considers SRPT with constant preemption time v and modifies SRPT using a constant *preemption gap* $p > v$. Specifically, in this modified SRPT, if the job in service has remaining size r_1 while another has remaining size r_2, we preempt the job in service only if $r_1 > r_2 + p$. This biases the scheduler away from preempting the job in service. Goerg (1986) analyzes SRPT with a constant preemption gap, showing that the optimal gap p depends on the system load.

It is still an open question to determine the optimal policy in this setting, and it is open whether SRPT with a constant preemption gap is competitive with the optimal policy.

Unknown Job Sizes

In the case of unknown job sizes, a natural approach is to try to generalize GI to incorporate preemption delays. To review, the GI policy works by independently assigning each job a priority, called its *Gittins index*, based on its age, and always serving the job of best Gittins index. Remarkably, a job's Gittins index depends only on its own age and not on any other aspect of the system state.

The origin of the GI policy is work on the multiarmed bandit problem (Gittins and Jones 1974, Gittins et al. 2011), and there has been work on generalizing multiarmed bandits and similar problems to settings with switching costs (Glazebrook 1980, Asawa and Teneketzis 1996, Jun 2004). One approach, introduced by Asawa and Teneketzis (1996), is to give each bandit an ordinary Gittins index, which is its priority while it is being served, and a *switching cost index*, which is its priority while another bandit is being served. The switching index has a worse priority than the continuation index, indicating a preference for avoiding switching costs. This is analogous to the previously discussed variant of SRPT (Goerg 1986).

However, policies for multiarmed bandits with switching costs do not necessarily immediately apply to M/G/1 scheduling with preemption delays. This is because of the difference between *costs*, which are a lump sum instantly paid, and *delays*, which takes nonzero time. One can think of a delay as a cost that varies depending on the number of other jobs in the system. Thus, in M/G/1 scheduling, one would expect a job's switching cost index to depend on the arrival rate of jobs, among other factors.

Ensuring Finite Mean Response Time

When the job size distribution X of an M/G/1 has infinite variance, the mean response time is infinite for any *nonpreemptive* policy because of an $\mathbf{E}[X^2]$ term (Harchol-Balter 2013). Thus, if the distribution X is unknown, preemptions are necessary to ensure finite mean response time. In the traditional case in which preemption is free, the *processor sharing* (PS) and *preemptive last come, first served* (PLCFS) policies both guarantee finite mean response time for any job size distribution. Specifically, both policies have mean response time $\mathbf{E}[X]/(1-\rho)$. Here $\rho = \lambda\mathbf{E}[X]$ is the system load, in which λ is the arrival rate. To ensure stability, we must have $\rho < 1$.

However, in a setting with preemption delays, having $\rho < 1$ is not enough to ensure stability: we must additionally bound the number of preemptions. Specifically, if our average preemption rate is λ_P and preemptions take time t_P on average, we must have $\rho + \lambda_P t_P < 1$. Neither PS nor PLCFS guarantee this in general. This prompts a weaker version of the question with which we started: is there a scheduling policy that *ensures finite mean response time* for an M/G/1 queue with preemption costs and arbitrary job size distribution for all loads $\rho < 1$?

We conjecture that the following adaptation of PLCFS guarantees to always achieve finite mean response time: always attempt to serve the job that most recently arrived, but when there are n jobs in the system, treat all jobs of size less than cn as nonpreemptible, where c is a constant.[1]

Prior Work on Polling Systems

We briefly mention a similar problem in polling systems. A *polling system* has a single server serving jobs from multiple queues. The server visits one queue at a time, occasionally switching between queues, and may have various restrictions on its service and queue-switchover policies. The problems of analyzing and optimizing response time in polling systems with switchover delays have been extensively studied (Van Oyen et al. 1992, Borst and Boxma 1997, Wierman et al. 2007).

Our problem of scheduling an M/G/1 with preemption costs appears similar to this prior work in polling systems, but there are key differences. We incur a preemption delay every time we switch between jobs, whereas polling systems incur delay only when switching between queues. Moreover, we allow preempting jobs, whereas polling systems typically do not. It is possible that viewing an M/G/1 as a type of polling system in which each job has its own queue could be a helpful perspective, but even with this view, the two settings do not immediately become equivalent.

Endnote

[1] We thank Isaac Grosof for helpful discussions about this policy.

References

Asawa M, Teneketzis D (1996) Multi-armed bandits with switching penalties. *IEEE Trans. Automatic Control* 41(3):328–348.

Borst SC, Boxma OJ (1997) Polling models with and without switchover times. *Oper. Res.* 45(4):536–543.

Gittins JC, Jones DM. (1974) A dynamic allocation index for the sequential design of experiments. Gani J, ed., *Progress in Statistics* (North-Holland, Amsterdam), 241–266.

Gittins JC, Glazebrook KD, Weber R (2011) *Multi-armed Bandit Allocation Indices* (John Wiley & Sons, Hoboken, NJ).

Glazebrook KD (1980) On stochastic scheduling with precedence relations and switching costs. *J. Appl. Probab.* 17(4):1016–1024.

Goerg C (1986) Evaluation of the optimal SRPT strategy with overhead. *IEEE Trans. Comm.* 34(4):338–344.

Harchol-Balter M (2013) *Performance Modeling and Design of Computer Systems: Queueing Theory in Action*, 1st ed. (Cambridge University Press, New York).

Jun T (2004) A survey on the bandit problem with switching costs. *Economist* 152(4):513–541.

Schrage L (1968) A proof of the optimality of the shortest remaining processing time discipline. *Oper. Res.* 16(3):687–690.

Van Oyen MP, Pandelis DG, Teneketzis D (1992) Optimality of index policies for stochastic scheduling with switching penalties. *J. Appl. Probab.* 29(4):957–966.

von Olivier G (1972) Cost-minimum priorities in queueing systems of type M/G/1. *Elektronische Rechenanlagen* 14(1–6):262–271.

Wierman A, Winands EMM, Boxma OJ (2007) Scheduling in polling systems. *Performance Evaluation* 64(9–12):1009–1028.