Operations Research
Vol. 56, No. 3, May–June 2008, pp. 728–744
issn 0030-364X | eissn 1526-5463 | 08 | 5603 | 0728

doi 10.1287/opre.1070.0483

Resource-Sharing Queueing Systems with Fluid-Flow Traffic

Sai Rajesh Mahabhashyam
Oracle Corporation, King of Prussia, Pennsylvania 19406, mahabhashyam@yahoo.com

Natarajan Gautam
Department of Industrial and Systems Engineering, Texas A&M University, College Station, Texas 77843, gautam@tamu.edu

Soundar R. T. Kumara
Department of Industrial and Manufacturing Engineering, Pennsylvania State University, University Park, Pennsylvania 16802, skumara@psu.edu

A system consisting of two buffers, each with independent fluid sources, is considered in this paper. Due to ease of implementation, the output capacities for the two buffers depend on the workload of only one of the buffers that is measured. A threshold-based policy is considered to dynamically assign output capacities for both buffers. Marginal workload distributions for the two buffers need to be evaluated for this policy. The key contribution of this paper is the performance analysis to derive the workload distribution in the two buffers. In addition, the paper also provides some guidelines to choose the output capacities for the two buffers as well as a mathematical program to determine an optimal threshold to dynamically switch between output capacities. Further, various applications of such systems to computer-communication networks are discussed.

Subject classifications: probability: stochastic model applications; queues; information systems: management.
Area of review: Stochastic Models.
History: Received May 2006; revisions received January 2007, May 2007; accepted June 2007.

1. Introduction

This paper considers a queueing system where the entities are not discrete, but are fluids. A bathtub is a good example of a fluid queue where the tap or faucet is a traffic source, the bathtub itself is a buffer, and the drain is the channel (analogous to server) that empties the buffer. The notion of fluid approximations in queues can mean several things. One way to organize them is as follows: zeroth-order fluid queues where flow rate is considered a constant (viz. Dai et al. 1999); first-order fluid queues where the flow rate is piecewise constant for random times (pioneering work by Anick et al. 1982); and second-order queues where the flow follows a Gaussian-like process (viz. Simonian 1991). This paper falls under the first-order fluid queues category with multiple classes of on-off fluid traffic sources. Such sources are excellent models for traffic entering buffers in computer-communication systems.

Single-station queues with multiclass traffic have been well studied in the queueing literature, but usually with only a single scheduler emptying the queues according to some policy. One can broadly classify the approaches that researchers have taken for analytical models of multiclass queues into two categories. The first kind is to choose the best policy to empty a queue given a specific objective function. The second is to choose a scheduling policy up-front and determine various system-level performance measures and fine-tune the policy using complex objective functions, if needed. Problems such as load balancing (see Mirchandany et al. 1990, Kostin et al. 2000, and Tantawi and Towsley 1985) and armed bandits (Darce et al. 1999, Whittle 1988) fall into the first category, whereas polling system analysis (Takagi 1986, Daganzo 1990, Boxma 1984) and its variants fall into the second category. The problem considered in this paper falls into the second category as well—however, under a fluid domain and using two schedulers.

In the literature, there are several articles such as Kesidis (1996) and Daganzo (1990) that have considered variants of polling systems with discrete traffic. However, for first-order fluid traffic, the literature is relatively limited. There are a significant number of articles that have used fluid models (some of the earlier ones include Anick et al. 1982), in which the researchers have assumed a constant output rate and a single class. In recent times, fluid models have been considered with varying service rates under some special conditions and circumstances. For example, Narayanan and Kulkarni (1996) analyze a multiclass fluid model that uses a static priority service policy, and Aggarwal et al. (2004) consider a threshold-based policy where the processing rate is shared between two classes of fluid based on the amount of content waiting to be served. There are a few articles in multiclass fluid models that consider various policies. For example, Elwalid and Mitra (1995) and Kulkarni and Gautam (1997) consider a static priority policy, whereas Gautam and Kulkarni (2000) consider a timed round-robin policy. There are a few other articles (see Gautam 2002 and Kulkarni and Gautam 1996) that consider fluid models, however, with only one scheduler.

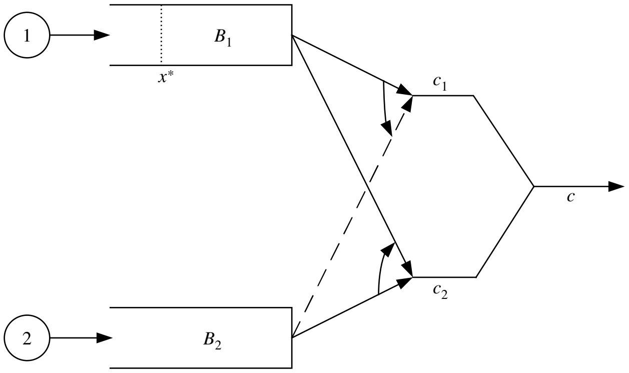
In this paper, a more complex two-class fluid-flow threshold-policy problem is considered in which the output process is semi-Markovian for one of the classes. One of the most popular tools to analyze fluid queues is using asymptotic results based on the large deviations principle (see Botvich and Duffield 1995). Unlike the traditional asymptotic analysis, where tail distributions for queues are obtained, the analysis in this paper uses extensions to those results to obtain entire distribution functions of the queues, not just the tail. However, to use those results, it is required to obtain partial differential equations for the dynamics of the buffer contents (similar to those in Anick et al. 1982). The analysis also leverages upon the idea of using compensating sources (as done in Elwalid and Mitra 1995 for a static priority scheme) and thereby analyzing each class individually.

It is also to be noted that the results obtained in this paper can be applied to a variety of situations, such as mobile ad hoc networks, software agents, and Web servers, where there are multiple classes of traffic and resource sharing. With that understanding, the organization of this paper is as follows. In §2, the two-buffer fluid queueing problem with switching threshold is formally defined along with notation, assumptions, objectives, some preliminary analysis, and a review of the literature. The main contribution of this paper is detailed performance analyses of the two buffers, which form the bulk of the paper. Specifically, §3 is devoted to the analysis of buffer-1 and §4 to the analysis of buffer-2. Section 5 is devoted to some numerical considerations. In particular, a variety of numerical experiments are run to evaluate system performance. Then, an optimization problem is considered to optimally choose the switching threshold. Finally, in §6 concluding remarks are presented, followed by some directions for future work.

2. Problem Description

Consider a system consisting of two buffers with fluid traffic (as shown in Figure 1). Each buffer has a traffic source that stochastically generates fluid into them. Fluid is emptied out of the buffers by schedulers with different capacities (c_1 for buffer-1 and c_2 for buffer-2 in Figure 1). The schedulers switch between the two buffers based on the content of the buffers. The content of buffer-1 is observed, whereas, due to simplicity of implementation, the contents of buffer-2 is not. Whenever the content of the observed buffer (i.e., buffer-1) is below a threshold value (x^* in Figure 1), both schedulers empty out the contents from their respective buffers. Whenever the content of the observed buffer exceeds the threshold value x^*, both of the schedulers empty the content of the observed buffer. Also, whenever the content in the observed buffer goes down to zero, both of the schedulers drain the content from the unobserved buffer. The main objective of this paper is to analyze the performance of such a system. In the remainder of this section, the scenario is explained in detail with notations in §2.1, followed by some applications where such scenarios are common (§2.2). Choice of parameters such as c_1, c_2, and x^* are explained in §2.3, some preliminary performance analysis in §2.4, and buffer content-analysis results from the literature for single buffers with a constant output channel are summarized in §2.5.

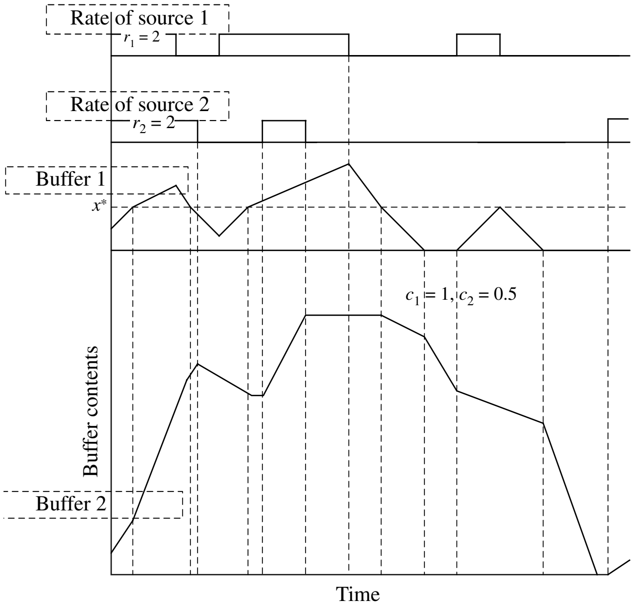
Figure 1. Two-buffer system.



2.1. Scenario

As explained above, the scenario considered here is a two-buffer fluid-flow system (Figure 1). There are two classes of fluid with one buffer for each class. For $j = 1, 2$, fluid enters buffer-j according to an alternating on-off process so that fluid enters continuously at rate r_j for an exponentially distributed time (on times) with mean $1/\alpha_j$, and then no fluid enters (off times) for another exponential time with mean $1/\beta_j$. The on and off times continue alternating one after the other. Let $X_j(t)$ be the amount of fluid in buffer-j (for $j = 1, 2$) at time t. The buffers can hold an infinite amount of fluid; however, the content of only one buffer is observed, namely, buffer-1. There are two schedulers that drain fluid from the two buffers. Scheduler-1 has a capacity of c_1 and scheduler-2 has a capacity of c_2, which are the maximum rates at which the respective schedulers can drain fluid. Fluid is drained from the two buffers in the following fashion. When $X_1(t)$ is nonzero, scheduler-1 serves buffer-1 and when $X_1(t) = 0$, scheduler-1 serves buffer-2. Also, if $X_1(t)$ is less than a threshold x^*, scheduler-2 removes fluid from buffer-2, otherwise it drains out buffer-1. In other words, when $X_1(t) = 0$, both schedulers serve buffer-2; and when $0 < X_1(t) < x^*$, scheduler-1 serves buffer-1 and scheduler-2 serves buffer-2, whereas when $X_1(t) \geqslant x^*$, both schedulers serve buffer-1. Because only $X_1(t)$ is observed, the buffer-emptying scheme depends only on it. If $C_j(t)$ is the capacity available for buffer-j at time t, then $C_1(t) = 0$ when $X_1(t) = 0$, $C_1(t) = c_1$ when $0 < X_1(t) < x^*$, and $C_1(t) = c_1 + c_2$ when $X_1(t) \geqslant x^*$. Capacity available for buffer-2 at any time t is $C_2(t) = c_1 + c_2 - C_1(t)$.

Figure 2. Sample path of buffer content process.



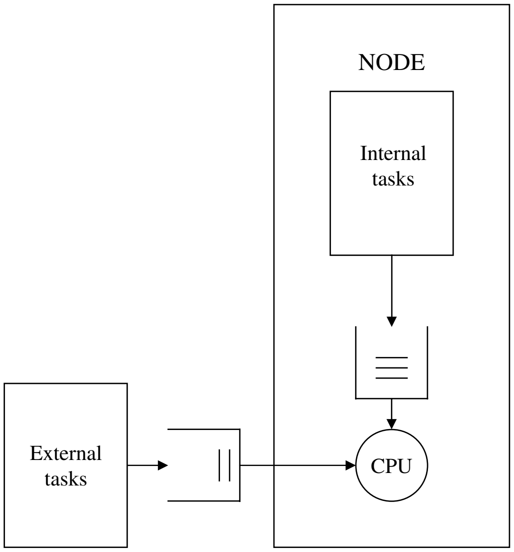
To illustrate the scheduler switching between buffers and their effect on the buffer contents, a sample path of the buffer content process is provided in Figure 2. The numerical values used for generating that sample path are as follows: $r_1 = 2$, $r_2 = 2$, $c_1 = 1$, $c_2 = 0.5$, $\alpha_1 = 1$, $\alpha_2 = 0.5$, $\beta_1 = 1$, and $\beta_2 = 0.25$. As evident from Figure 2, the fluid in buffer-j increases at a rate of $R_j(t) - C_j(t)$ if $X_j(t) > 0$, where $R_j(t)$ is the rate of fluid entering buffer-j at time t. Note that $R_j(t) = r_j$ if source-j is on at time t, and $R_j(t) = 0$ if source-j is off at time t.

2.2. Applications

The abstract model described above is applicable to certain computer-communication network settings. In particular, a schematic in Figure 3 is considered where a node owns and controls a central processing unit (CPU) residing in it. One function of the CPU is to process the tasks generated internally by the node. However, the node can share the CPU for processing external tasks as well. The CPU is managed by a software inside the node, and real-time state information, due to simplicity of implementation, is obtained only regarding the internal buffer, but not the external buffer. In some sense the traffic in the external buffer is considered lower priority, and to reduce algorithmic complexity its state is not taken into account. However, obtaining statistical characteristics of the external (low-priority) buffer is still of interest. A more detailed treatment is provided for a specific application in §2.2.1.

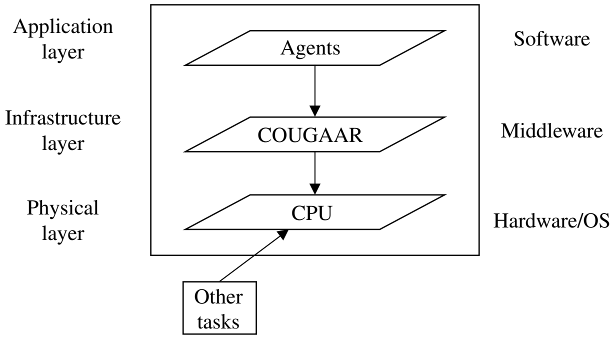
The key idea is that a resource such as CPU (or even bandwidth) can be shared between one's own applications (with observed buffer) and other applications (buffer content not observed in real time). The entire processor capacity or bandwidth of c bytes per second is split into two portions, c_1 and c_2 (such that $c = c_1 + c_2$). The units of r_1 and r_2 (i.e., arrival rates of information when sources are on) are also in bytes per second. Then, for $i = 1, 2$ (denoting internal and external), $X_i(t)$ can be thought of as the total number of bytes (i.e., workload) in buffer-i at time t. The quantities $1/\alpha_i$ and $1/\beta_i$ are assumed to be in seconds. With that mapping in mind, we first describe the motivating example (in §2.2.1) that led us to this research and then briefly state other examples (in §2.2.2) where the scenario depicted in §2.1 can be applied.

Figure 3. Schematic of a typical node.



2.2.1. Motivating Example: Multiagent System with COUGAAR Middleware. The motivation for this paper comes from a large-scale distributed multiagent system for military logistics called *Cognitive Agent Architecture* (COUGAAR; see http://www.cougaar.org). The COUGAAR system comprises several software agents that work in a cooperative manner to accomplish tasks for military logistics planning operations. The society of software agents (or a subset of them) reside in a machine (i.e., computer). At each machine, there is a single CPU that processes tasks submitted not only by the agents on the machine, but also by other applications running on the machine. Especially under wartime conditions when the resources are limited (such as a computer on a ship or aircraft), it becomes crucial for the software agents to share the CPU with other, possibly critical, applications. In that light, our objective is to adaptively control when the software agents capture the CPU and when the agents release a portion of the processing capacity for other applications. Other approaches in the literature that have been used to improve utilization and sharing of resources apply techniques at the kernel (operating system) or the hardware level. These techniques have a limitation in that they cannot be reconfigured easily by the end user to adapt to changing load characteristics or different objectives set for the system. Therefore, the COUGAAR system uses software

Figure 4. Three-layer architecture for COUGAAR.



(sometimes referred to as middleware) to monitor the state of the resources and take control actions.

Adapted from Gnanasambandam et al. (2005), the architecture comprises three layers, as depicted in Figure 4. The society of software agents occupies the application layer and is responsible for military logistics planning operations. COUGAAR is the middleware that resides in the infrastructure layer and is responsible for coordinating and controlling hundreds of agents, as well as the resources such as CPU, over which it has complete authority. It forms a link between the applications and the physical layer (hence the name middleware). Sensors located in this layer collect system-state information, and software codes perform required control actions. The layer where the operating system resides and is closely linked to hardware and CPU is the physical layer. In terms of information sharing, this architecture is based on two key features: (a) avoid real-time information sharing across layers and nodes due to security concerns and performance degradation; and (b) optimally share aggregate information across layers and nodes.

For the above architecture and features, Helsinger et al. (2003) develop sensors and software plug-ins at the COUGAAR middleware for obtaining state information, measuring performance, and undertaking control actions. In fact, the sensors obtain real-time (and relatively trustworthy) state information not by probing the buffer, but by estimating based on information that flows through the middleware. Helsinger et al. (2003) state the reasons for that are: (a) the sheer volume of data generated by multiagent systems causes difficulties for monitoring systems; (b) to ensure accuracy, data retrieval must often be synchronized, and the results time stamped and centrally stored. However, this is expensive—requiring excessive communications bandwidth and central CPU time; (c) any process for measuring the state of a system should not itself impact that state, and in multiagent systems this requires care not to overwhelm communications resources.

Therefore, due to this estimation of state information and the fact that real-time information is not shared across layers, only the buffer inside the node is observed and the buffer outside is not. In addition, the tasks arrive to the CPU in chunks and resemble an on-off fluid source. These aspects motivated us to study an abstract model described in §2.1. In essence, depending on the status of their submitted tasks (i.e., buffer contents), the COUGAAR system can decide if it wants to give up the entire CPU capacity ($c_1 + c_2$), a fraction of the CPU capacity (c_2), or none of the CPU capacity for other applications that run on the node. The COUGAAR system has the capability to implement this type of resource-sharing control, and what is missing is an optimal technique for it. Next, we describe some other systems where such resource sharing occurs.

2.2.2. Other Example Applications. There are other examples in computer-communication networks where the system can be modeled and controlled as described in §2.1. A few such examples that are gaining a lot of attention in the research community are briefly explained below.

- **User-share wireless networks:** In traditional wireless networks, laptop computers, personal data assistants (PDAs), smart cellular phones, etc. typically connect directly to a base station. Sometimes the base station is too far, resulting in the signal being too weak or the device using up excessive battery power. A technology being proposed and implemented is user-share wireless networks, where user information does not directly flow to a base station (single-hop), but through many other users (multihop). Under these circumstances, devices in these multihop wireless networks allocate a portion of the bandwidth (c_1) and offer the remaining bandwidth (c_2) for others that wish to transmit through the device under consideration, which we will call the *main node*. A software in the main node measures the workload remaining and decides whether to use 0, c_1, or $c_1 + c_2$ bytes per second to transmit its own information. Information from other nodes that is transmitted by the main node is collected in a buffer at the network interface, and the buffer state is not observed by the software of the main node. All the CPU does is program the network interface card to determine the destination of the information. The information itself does not enter the node. This system can hence be modeled and controlled as described in §2.1.
- **Sensor networks:** Each sensor in a sensor network acts as a source node and a transshipment node whereby it transmits information that either it generates or another source forwards. Usually, the sensor's processing capacity can be divided into two parts, c_1 and c_2, respectively, for its own transmission and for what other nodes send. Also, the software in the sensor only observes in real time the workload it generates, and not what it transships. This is because local information flows from the application layer where the software resides, but the transshipment information is at the network layer and does not flow down the protocol stack. Based on the workload generated by the sensor, the software in the sensor decides when to use all ($c_1 + c_2$), some (c_1), or none of the processing capacity for its own transmission and leave the remaining capacity to forward

information from other nodes. Therefore, the setting can be modeled and controlled using what is described in §2.1.

- **Grid computing:** An inexpensive way to run massively parallel jobs that traditionally used to be run on supercomputers or clusters is to run them on geographically dispersed workstations. A workstation that is on the grid network processes not only its own tasks, but also those that are submitted by other machines for massively parallel computations. The workstations can agree to devote some of their processing abilities for a prenegotiated fraction of time for processing the parallel computation tasks. However, due to security reasons, real-time information about the external jobs that arrive at a workstation is not available and only aggregate measures are collected. Therefore, for the workstation to share its CPU with externally arriving tasks, a software looks at the workload generated within the workstation (observed buffer) and decides whether to use all (c_1+c_2), some (c_1), or none of its processing power. Hence, this system can also be analyzed using the model described in §2.1.

From a computer-communication systems standpoint, the above three and COUGAAR (§2.2.1) are all very different systems. However, as far as performance analysis is concerned, they can all be modeled and analyzed in a similar way. In addition, on-off processes model the traffic characteristics well because at any time point, either that information flows at the link speed or nothing flows. When there is information, it flows back to back, making first-order fluid models excellent approximations. Therefore, having motivated the application and the appropriateness of the modeling assumptions, the next question to ask is how the controllable parameters c_1, c_2, and x^* are chosen.

2.3. Choosing c_1, c_2, and x^*

In all the applications described in §2.2, the quantity c_1+c_2 is the total capacity (processing power or bandwidth) of the resource that is shared between the two classes of traffic. Clearly, c_1+c_2 is a constant that is denoted as c. Two policies are considered for choosing c_1 and c_2. In the first policy, c_1 and c_2 are chosen in the ratio of the corresponding mean arrival rates into buffer-1 and buffer-2. In the second policy, c_1 and c_2 are chosen in the ratio of the corresponding mean burst size (i.e., the average amount of fluid that is generated when a source is in the on state).

(1) *Policy* 1. The average arrival rate into buffer-1 is $r_1\beta_1/(\alpha_1+\beta_1)$, and the average arrival rate into buffer-2 is $r_2\beta_2/(\alpha_2+\beta_2)$. In this policy, because the capacities for scheduler-1 and scheduler-2 are in the ratio of the mean arrival rates into the buffers, $c_1{:}c_2 = r_1\beta_1/(\alpha_1+\beta_1){:}r_2\beta_2/(\alpha_2+\beta_2)$. Therefore,

$$c_1 = \left(\frac{r_1\beta_1 c}{\alpha_1+\beta_1}\right)\bigg/\left(\frac{r_1\beta_1}{\alpha_1+\beta_1}+\frac{r_2\beta_2}{\alpha_2+\beta_2}\right),$$
$$c_2 = c - c_1.$$

(2) *Policy* 2. For $i=1,2$, the average on time for source-i (for buffer-i) is $1/\alpha_i$. Also, when source-i is on, the arrival rate is r_i. Therefore, the average burst size is r_i/α_i for source-i. In this policy, because the capacities for scheduler-1 and scheduler-2 are in the ratio of mean burst size,

$$c_1 = \left(\frac{r_1 c}{\alpha_1}\right)\bigg/\left(\frac{r_1}{\alpha_1}+\frac{r_2}{\alpha_2}\right),$$
$$c_2 = c - c_1.$$

Clearly, one could use some other policy as well. Once c_1 and c_2 are determined, the next task is to obtain x^*. For this, an optimization problem is suggested and formulated. Let B_1 and B_2 be target buffer contents that are undesirable to exceed for buffer-1 and buffer-2, respectively. The objective is to optimize for x^* so that the weighted sum of the probabilities exceeding B_1 and B_2 in buffer-1 and buffer-2 respectively, in steady state is minimized. If $w_1>0$ and $w_2>0$ are given weights, then the optimization problem is given by

$$\text{Minimize}\quad w_1 \lim_{t\to\infty} P(X_1(t)>B_1) + w_2 \lim_{t\to\infty} P(X_2(t)>B_2) \tag{1}$$

$$\text{subject to}\quad \lim_{t\to\infty} P(X_1(t)>x^*) \leqslant 1-\nu. \tag{2}$$

The constraint in inequality (2) needs explaining. It is a service-level agreement (SLA) that source-2 is guaranteed. The SLA in words is: a capacity of at least c_2 is provided for source-2 with a minimum probability ν. For example, if $\nu=0.95$, then at least 95% of the time, a capacity of at least c_2 will be provided to buffer-2, which in turn means that at least 95% of the time the content of buffer-1 needs to be less than x^*, hence, inequality (2).

To solve the above mathematical program, it is required to obtain the limiting distribution of buffer contents $X_1(t)$ and $X_2(t)$, as $t\to\infty$. In fact, the main objective of this paper, and the key contribution, is in obtaining $\lim_{t\to\infty} P\{X_1(t)>x\}$ and $\lim_{t\to\infty} P\{X_2(t)>x\}$, for which §§3 and 4, respectively, are entirely devoted. Before that, some assumptions and preliminary results are needed, which are described next, followed by some results from the literature on obtaining limiting distributions for the single-buffer case.

2.4. Preliminary Analysis

Consider the problem defined in §2.1. The main objective and contribution of this paper is to develop a methodology to obtain closed-form limiting distributions of $X_1(t)$ and $X_2(t)$, i.e., as $t\to\infty$, $P\{X_i(t)>x\}$ for $i=1,2$. For that, as a first step, the condition for stability is stated in the following theorem.

Theorem 1. *The stability condition for the two-buffer system in Figure* 1 *is given by*

$$\frac{r_1\beta_1}{\alpha_1+\beta_1}+\frac{r_2\beta_2}{\alpha_2+\beta_2} < c_1+c_2.$$

Proof. Because the system is work conserving, the limiting distributions of the buffer contents exist when the mean input rate is smaller than the processing capacity (see Kulkarni and Rolski 1994). The mean input rate for source-1 is $r_1\beta_1/(\alpha_1+\beta_1)$, and the mean input rate of source-2 is $r_2\beta_2/(\alpha_2+\beta_2)$. The processing capacity is the sum of capacities of the schedulers, i.e., c_1+c_2. Thus, the result follows. □

Before proceeding, it is crucial to make a few remarks.

Remark 1. The objective is to obtain marginal distributions of $X_1(t)$ and $X_2(t)$ in steady state (not joint distributions or distributions of $X_1(t)+X_2(t)$, which can be obtained using other techniques than what is described in this paper).

The above remark is made to set the stage for the analysis to follow.

Remark 2. In terms of the parameters, a few assumptions are made:

$$r_1 > c_1+c_2,$$
$$r_2 > c_1+c_2,$$
$$B_1 > x^*.$$

Although the above assumptions are not required, they are there purely for practical considerations and nontriviality. Using Theorem 1 above and two remarks, the next step is to obtain limiting marginal distributions of contents for buffer-1 (§3) and buffer-2 (§4). However, the analysis builds upon the results for *standard single buffers*, described next.

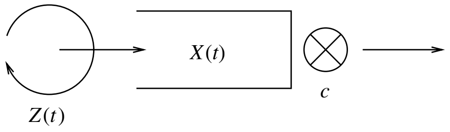
2.5. Standard Single-Buffer Analysis

The analyses in §§3 and 4 are partially based on some results for the *standard single-buffer model*. Hence, we present a summary of the literature pertaining to the standard single-buffer model illustrated in Figure 5. Input to the infinite-sized buffer is driven by a random environment process $\{Z(t), t \geqslant 0\}$. When the environment is in state $Z(t)$, fluid enters the buffer at rate $r(Z(t))$. The output capacity is c. Let $X(t)$ be the amount of fluid in the buffer at time t. We are interested in the limiting distribution of $X(t)$, i.e.,

$$\lim_{t\to\infty} P\{X(t) > x\} = P\{X > x\}.$$

The literature for computing the above distribution can be classified into asymptotic analysis (when $x\to\infty$) and full distribution (when $x \in [0,\infty)$).

Figure 5. Standard single-buffer system.



- *Asymptotic Analysis or Tail Distribution* ($x\to\infty$).

If the environment process is a continuous-time Markov chain (Elwalid and Mitra 1993) or a Markov regenerative process (Kulkarni 1997), approximations for the tail probabilities of $P\{X > x\}$ can be derived of the form

$$P\{X > x\} \approx e^{-\eta x}.$$

The above asymptotic analysis was further fine-tuned using the Chernoff dominant eigenvalue approximation (Elwalid et al. 1995, Elwalid and Mitra 1995) as

$$P\{X > x\} \approx L\, e^{-\eta x},$$

where L is an adjusted factor described in the above references.

- *Full Distribution* ($x \in [0,\infty)$).

If the environment process is a continuous-time Markov chain (Anick et al. 1982, Elwalid and Mitra 1991, 1992), derive exact expressions (not approximations) for the full distribution

$$P\{X > x\} = \sum_i l_i\, e^{-\eta_i x}.$$

These results are used in §3 for a specific problem where the parameters l_i and η_i are described in closed form.

If the environment process is an alternating renewal process (Palmowski and Rolski 1996) or a semi-Markov process (Gautam et al. 1999), bounds for $P\{X > x\}$ can be computed using martingale inequalities. The bounds are of the form

$$C_* e^{-\eta x} \leqslant P\{X > x\} \leqslant C^* e^{-\eta x}, \quad x \geqslant 0.$$

Although the η value is the same as the asymptotic analysis, the bounds are valid for all $x \geqslant 0$. These results are further developed and used for a specific problem instant in §4, and the parameters C_*, C^*, and η are derived there.

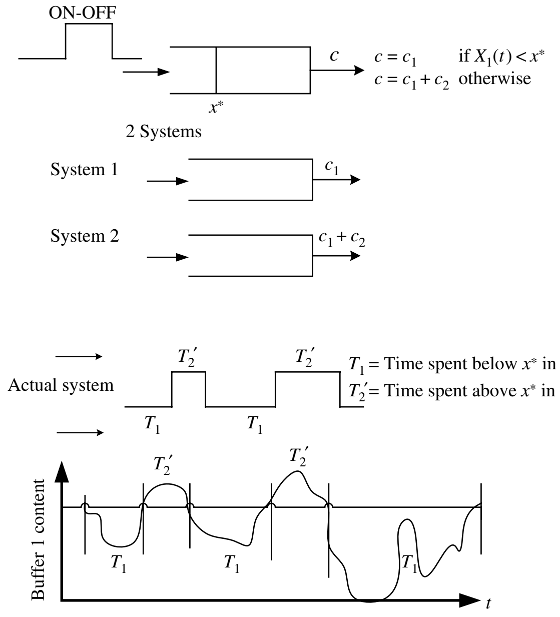
3. Buffer-1 Performance Analysis

We now revert back to the scenario described in §2.1. In this section, the aim is to compute the limiting distribution (as $t\to\infty$) of $X_1(t)$, the amount of fluid in buffer-1 at time t. For that, we use the definition of a standard single buffer as described in §2.5. However, note that buffer-1 in this paper is not a standard single buffer because the channel output capacity is zero when $X_1(t)=0$, it is c_1 when $X_1(t) < x^*$, and it is c_1+c_2 when $X_1(t) \geqslant x^*$. For performance analysis, the channel output capacity does not matter when $X_1(t)=0$. Therefore, when $X_1(t) < x^*$, the buffer-1 system can be thought of as a standard single buffer with an on-off source with rate r_1 when on (for exponential time with mean $1/\alpha_1$) and rate zero when off (for exponential time with mean $1/\beta_1$) with a constant output capacity c_1. Similarly, when $X_1(t) \geqslant x^*$, the buffer-1 system can be thought of as a standard single buffer with the same exponential on-off source, but with an output capacity $c_1 + c_2$. Using that, the limiting buffer content distribution can be obtained, as explained next in §3.1. It is crucial to point out that a similar system with arrival rates dependent on buffer contents via a threshold and a constant service rate is considered in Elwalid and Mitra (1994), where the authors solve a partial differential equation with appropriate boundary conditions to obtain the buffer content distribution.

3.1. Combining Standard Single-Buffer Results

As described above, buffer-1 can be analyzed by considering two standard single buffers with an exponential on-off source for two different output capacities separately and combining them appropriately to obtain an exact closed-form expression for the steady-state distribution for buffer-1. Refer to Figure 6. System-1 is a standard single buffer with an exponential on-off source with a constant output of c_1. Likewise, system-2 is a standard single buffer with an exponential on-off source with a constant output of $c_1 + c_2$. The actual buffer-1 system looks like system-1 when $x < x^*$, whereas it functions like system-2 when $x \geqslant x^*$. Let T_1 be the time that buffer content in system-1 takes to reach x^* going down from x^* for the first time. Let T_2 be the time that buffer content in system-1 takes to reach x^* going up from x^* for the first time. Similarly, define T_1' and T_2' for system-2. Now, as shown in Figure 6, it can be seen that in the actual buffer-1 system, the time taken for the content to reach x^* (going down from x^*) for the first time is identical to that of system-1 (i.e., T_1) and the time taken for the contents to reach x^* (going up from x^*) for the first time is identical to that of system-2 (i.e., T_2'). Let X_1 be a random variable such that when $t \to \infty$, $X_1(t) \to X_1$.

Figure 6. Buffer-1 model.


ON-OFF
c
$c = c_1$ if $X_1(t) < x^*$
$c = c_1 + c_2$ otherwise
x^*
2 Systems
System 1
c_1
System 2
$c_1 + c_2$
Actual system
T_2'
T_2'
T_1
T_1
T_1 = Time spent below x^* in system 1
T_2'= Time spent above x^* in system 2
Buffer 1 content
T_2'
T_2'
T_1
T_1
T_1
t

Note that the aim is to obtain an expression for $P(X_1 > x)$ when $x \geqslant x^*$. For that, consider system-2 (in Figure 6) when the buffer contents exceed x^*. Let the buffer content in system-2 at time t be denoted by $X_{12}(t)$ (the notation X_{12} is due to the fact that it is the limiting buffer content for the buffer-1 problem in system-2). From the results (see Elwalid and Mitra 1991 and §2.5 for a generic form for $P\{X > x\}$) for a standard single buffer with exponential on-off source,

$$P(X_{12} > a) = \frac{r_1\beta_1}{(\alpha_1+\beta_1)(c_1+c_2)} e^{-\eta a}, \tag{3}$$

where

$$\eta = \frac{(c_1+c_2)(\alpha_1+\beta_1) - \beta_1 r_1}{(c_1+c_2)[r_1-(c_1+c_2)]}.$$

To use the above result in deriving an expression for $P(X_1 > x)$, when $x \geqslant x^*$, the following lemma for a conditional distribution of X_{12} is needed.

Lemma 1.

$$P(X_{12} > x - x^* \mid X_{12} > 0) = e^{-\eta(x-x^*)}. \tag{4}$$

Proof. By conditioning and unconditioning $P(X_{12} > x - x^*)$ on X_{12},

$$\begin{aligned} P(X_{12} > x - x^*) &= P(X_{12} > x - x^* \mid X_{12} > 0) P(X_{12} > 0) \\ &\quad + P(X_{12} > x - x^* \mid X_{12} = 0) P(X_{12} = 0). \end{aligned}$$

The term $P(X_{12} > x - x^* \mid X_{12} = 0)$ is zero because $x \geqslant x^*$. Therefore, from the above equation and Equation (3),

$$\begin{aligned} &P(X_{12} > x - x^* \mid X_{12} > 0) \\ &\quad = P(X_{12} > x - x^*)/P(X_{12} > 0) \\ &\quad = \frac{r_1\beta_1}{(\alpha_1+\beta_1)(c_1+c_2)} e^{-\eta(x-x^*)} \Bigg/ \frac{r_1\beta_1}{(\alpha_1+\beta_1)(c_1+c_2)} \\ &\quad = e^{-\eta(x-x^*)}. \end{aligned}$$

Hence, the lemma is proved. □

The above lemma is used to prove the following theorem that describes the buffer content distribution of buffer-1.

Theorem 2. *The probability that in steady state the content of buffer-*1 *exceeds* x (*for* $x > x^*$) *is*

$$P(X_1 > x) = \frac{E[T_2']}{(E[T_1] + E[T_2'])} e^{-\eta(x-x^*)}.$$

Proof. Consider the sample path depicted in Figure 6 that shows how the content of buffer-1 varies with time. Recall that T_1 is the time spent with buffer content below x^* before reaching x^*, and T_2' is the time spent with buffer content above x^* before reaching x^*. Also, T_1 is the time spent below x^* for a standard single buffer with on-off source with c_1 output, and T_2' is the time spent above x^* for a standard single buffer with an on-off source with (c_1+c_2) output.

The total time is divided into two parts, one when buffer content is less than or equal to x^*, and the other when buffer content is more than x^*. In the long run, the average time that the buffer content is below x^* is $E[T_1]$ and the average time that the buffer content is above x^* is $E[T_2']$. By conditioning and unconditioning on whether the buffer content is below or above x^*,

$$P(X_1>x)=P(X_1>x\mid X_1<x^*)P(X_1<x^*)\\ \qquad +P(X_1>x\mid X_1>x^*)P(X_1>x^*).$$

In the above equation, $P(X_1>x\mid X_1<x^*)=0$ because buffer content can never exceed x when $X_1<x^*$ and $x>x^*$. Also, $P(X_1>x^*)$ is the fraction of time that the buffer content is above x^*, i.e., $E[T_2']/(E[T_1]+E[T_2'])$. Therefore,

$$P(X_1>x)=0P(X_1<x^*)+P(X_1>x\mid X_1>x^*)P(X_1>x^*)\\ =P(X_1>x\mid X_1>x^*)\frac{E[T_2']}{E[T_1]+E[T_2']}.$$

Here, $P(X_1-x^*>x-x^*\mid X_1-x^*>0)$ is identical to $P(X_{12}>(x-x^*)\mid X_{12}>0)$ because this is the case when buffer content is above x^*. Therefore,

$$P(X_1>x)=\frac{E[T_2']}{E[T_1]+E[T_2']}P(X_{12}>(x-x^*)\mid X_{12}>0). \tag{5}$$

From Equations (4) and (5),

$$P(X_1>x)=\frac{E[T_2']}{(E[T_1]+E[T_2'])}e^{-\eta(x-x^*)}. \quad \square \tag{6}$$

Note that in Equation (6), both $E[T_2']$ and $E[T_1]$ are unknown. In the next two sections, expressions for $E[T_2']$ and $E[T_1]$ will be derived in that order.

3.2. Derivation of $E[T_2']$

A closed-form algebraic expression for $E[T_2']$ is provided in the next theorem.

Theorem 3. *The expression for $E[T_2']$ is given by*

$$E[T_2']=\frac{r_1}{(c_1+c_2)(\alpha_1+\beta_1)-r_1\beta_1}. \tag{7}$$

Proof. Clearly, $E[T_2']$ is the average time for buffer content to go up from x^* (with the source on) and return to x^* again for the first time in system-2. It is identical to the average busy period in a standard single buffer with exponential on-off source and output capacity c_1+c_2. Using the busy period distribution in Narayanan and Kulkarni (1996) for a standard single buffer, Equation (7) can be obtained. □

Although the expression for $E[T_2']$ was relatively straightforward, it is quite a task to obtain $E[T_1]$, which is done in the next section.

3.3. Derivation of $E[T_1]$

Consider system-1, defined in §3.1. In this section, the average time for the buffer to go down from x^* and come back to x^* (i.e., $E[T_1]$) is derived. Similar to $X_{12}(t)$ in §3.1, define $X_{11}(t)$ as the buffer content in system-1 at time t (the notation X_{11} is due to the fact that it is the limiting buffer content for buffer-1 in system-1). Define the first passage time T as

$$T=\inf\{t>0\colon X_{11}(t)=x^*\}.$$

This is the time that it takes for the buffer content to reach x^* for the first time in the future. Let $Z_1(t)$ be the state of the source at time t (subscript 1 denotes source-1). When the source is on, $Z_1(t)=1$, and when off, $Z_1(t)=0$. Define

$$H_{ij}(x,t)=P\{T\leqslant t, Z_1(T)=j\mid X_{11}(0)=x, Z_1(0)=i\}$$

for $i,j=0,1$; $0\leqslant x\leqslant x^*$ and $t\geqslant 0$. Then, $E[T_1]$ is given by the negative differential value of Laplace Stieltjes Transform (LST) of $H_{01}(x,t)$ at zero. Mathematically,

$$E[T_1]=-\frac{d}{dw}\widetilde{H}_{01}(x^*,w)|_{w=0}, \tag{8}$$

where $\widetilde{H}_{01}(x^*,w)$ is the LST of $H_{01}(x^*,t)$, i.e., $\widetilde{H}_{01}(x^*,w)=\int_0^\infty e^{-wt}(\partial H_{01}(x,t)/\partial t)\,dt$. Therefore, to obtain $E[T_1]$, an expression for $H_{01}(x,t)$ needs to be derived.

Note that $H_{i0}(x,t)=0\ \forall\, i$ because the source has to be on for the buffer content to reach x^* from below, i.e., the first passage time T cannot end when the source is off. Define the vector

$$H_1(x,t)=\begin{bmatrix}H_{01}(x,t)\\ H_{11}(x,t)\end{bmatrix},$$

the rate matrix

$$R=\begin{bmatrix}-c_1 & 0\\ 0 & r_1-c_1\end{bmatrix},$$

and the generator matrix

$$Q=\begin{bmatrix}-\beta_1 & \beta_1\\ \alpha_1 & -\alpha_1\end{bmatrix}.$$

The following theorem shows the dynamics of $H_1(x,t)$.

Theorem 4. *The dynamics of $H_1(x,t)$ are governed by the following partial differential equation:*

$$\frac{\partial H_1(x,t)}{\partial t}-R\frac{\partial H_1(x,t)}{\partial x}=QH_1(x,t) \tag{9}$$

with initial conditions

$$H_{01}(x,0)=0 \quad \text{for } x\geqslant 0,$$

$$H_{11}(x,0)=0 \quad \text{for } 0\leqslant x<x^*,$$

and boundary conditions (in the LST domain, which is all that is needed)

$$\widetilde{H}_{11}(x^*,w)=1,$$

$$\widetilde{H}_{01}(0,w)=\frac{\beta_1}{\beta_1+w}\widetilde{H}_{11}(0,w).$$

Proof. Consider the term $H_{01}(x, t+h)$, where h is a small positive real number. One can write $H_{01}(x, t+h)$ as

$$\begin{aligned} &H_{01}(x, t+h) \\ &\quad = P\{T \leqslant t+h, Z_1(T) = 1 \mid X_{11}(0) = x, Z_1(0) = 0\}. \end{aligned}$$

By conditioning and unconditioning on $Z_1(h)$, it is possible to show that

$$\begin{aligned} H_{01}(x, t+h) = {}& H_{01}(x - hc_1, t)(1 - \beta_1 h) \\ & + H_{11}(x - hc_1, t)\beta_1 h + O(h), \end{aligned}$$

where $O(h)$ represents higher-order terms of h. Rearranging terms,

$$\begin{aligned} &\frac{H_{01}(x, t+h) - H_{01}(x, t)}{h} \\ &\quad = \frac{H_{01}(x - hc_1, t) - H_{01}(x, t)}{h} - \beta_1 H_{01}(x - hc_1, t) \\ &\qquad + \beta_1 H_{11}(x - hc_1, t) + O(h)/h. \end{aligned}$$

Taking the limit as $h \to 0$, because $O(h)/h \to 0$,

$$\frac{\partial H_{01}(x, t)}{\partial t} = -c_1 \frac{\partial H_{01}(x, t)}{\partial x} - \beta_1 H_{01}(x, t) + \beta_1 H_{11}(x, t).$$

Similarly, one can derive

$$\frac{\partial H_{11}(x,t)}{\partial t} = (r_1 - c_1)\frac{\partial H_{11}(x,t)}{\partial x} + \alpha_1 H_{01}(x,t) - \alpha_1 H_{11}(x,t).$$

Combining the above two and writing in a matrix form results in Equation (9). The initial conditions are

$$H_{01}(x, 0) = 0 \quad \text{for } x \geqslant 0, \tag{10}$$

$$H_{11}(x, 0) = 0 \quad \text{for } 0 \leqslant x < x^*. \tag{11}$$

The first initial condition (i.e., Equation (10)) means that the probability of the first passage time occurring instantaneously with a change in state (source going from off to on) for any nonnegative value of x is zero. This is true because the probability that a source changes states (off to on) at time $t = 0$ is zero. The second initial condition (i.e., Equation (11)) means that the probability of the first passage time occurring instantaneously when the source is on for any value of x such that $0 \leqslant x < x^*$ is zero. This is true because the buffer content cannot instantaneously reach x^* (because $x < x^*$) or go to zero (because the source is on).

The boundary conditions are as follows:

$$\widetilde{H}_{11}(x^*, w) = 1, \tag{12}$$

$$\widetilde{H}_{01}(0, w) = \frac{\beta_1}{\beta_1 + w}\widetilde{H}_{11}(0, w). \tag{13}$$

The first boundary equation (i.e., Equation (12)) means that when $X_{11}(0) = x^*$, the probability of reaching x^* instantaneously is one. This is true because at time $t = 0$, the first passage time occurred. The second boundary equation (i.e., Equation (13)) is a representation of the time taken for the source to go from off to on states, which is exponentially distributed with mean $1/\beta_1$. □

The next step is to solve the partial differential equation (PDE) in Equation (9). The following theorem transforms the PDE into an ordinary differential equation (ODE) in terms of the LST of $H_1(x, t)$, namely, $\widetilde{H}_1(x, w)$.

Theorem 5. *The PDE in Equation* (9) *along with the initial and boundary conditions results in the following ODE of the LSTs*:

$$R\frac{d\widetilde{H}_1(x, w)}{dx} = (wI - Q)\widetilde{H}_1(x, w).$$

Proof. Taking the LSTs on both sides of Equation (9),

$$w\widetilde{H}_1(x, w) - H_1(x, 0) - R\frac{\partial \widetilde{H}_1(x, w)}{\partial x} = Q\widetilde{H}_1(x, w).$$

Substituting the initial condition equations and rearranging terms, it is possible to show that

$$R\frac{d\widetilde{H}_1(x, w)}{dx} = (wI - Q)\widetilde{H}_1(x, w). \quad \square$$

To obtain $E[T_1]$ in Equation (8), the above ODE is solved below by using a spectral decomposition technique. The solution of the ODE is given by

$$\widetilde{H}_1(x, w) = a_1 e^{s_1(w)x}\phi_1(x) + a_2 e^{s_2(w)x}\phi_2(x),$$

where $s_i(w)$ and $\phi_i(x)$ are, respectively, eigenvalues and right eigenvectors that can be obtained by solving

$$Rs_i(w)\phi_i(w) = (wI - Q)\phi_i(w) \quad \text{for } i = 1, 2, \tag{14}$$

and a_i can be solved by using the boundary conditions. Therefore, the closed-form algebraic expressions for $s_i(w)$ are

$$\begin{aligned} s_1(w) &= \frac{-b - \sqrt{b}^2 + 4w(w + \alpha_1 + \beta_1)c(r_1 - c)}{2c(r - c)}, \\ s_2(w) &= \frac{-b + \sqrt{b^2 + 4w(w + \alpha_1 + \beta_1)c(r_1 - c)}}{2c(r_1 - c)}, \end{aligned} \tag{15}$$

where $b = ((r - 2c)w + (r - c)\beta_1 - c\alpha_1)$, $w > (2\sqrt{c\alpha_1\beta_1(r_1 - c)} - r_1\beta_1 - c\alpha_1 + c\beta_1)/r_1$, and $c = c_1 + c_2$. Also,

$$\phi_i(w) = \begin{bmatrix} k_i \\ 1 \end{bmatrix} \tag{16}$$

and

$$a_1 = \frac{k_2(\beta_1 + w) - \beta_1}{(k_2(\beta_1 + w) - \beta_1)e^{s_1(w)x^*} - (k_1(\beta_1 + w) - \beta_1)e^{s_2(w)x^*}},$$

$$a_2 = \frac{-k_1(\beta_1 + w) - \beta_1}{(k_2(\beta_1 + w) - \beta_1)e^{s_1(w)x^*} - (k_1(\beta_1 + w) - \beta_1)e^{s_2(w)x^*}},$$

where $k_i = (w + \alpha_1 - s_i(w)(r - c))/\alpha_1$ (which is also equal to $\beta_1/(w + \beta_1 + s_i(w))$).

Therefore, once $\widetilde{H}_{01}(x, w)$ is known, one can use Equation (8) to obtain $E[T_1]$. Because $E[T_2']$ is derived in Equation (7), one should combine them (in Equation (6)) to obtain the probability $P\{X_1 > x\}$ for $x > x^*$.

4. Buffer-2 Performance Analysis

Having obtained $P(X_1 > x)$, the next step is to derive an expression for $P(X_2 > x)$. For buffer-2, the output capacity is not only variable, but also inherently dependent on the contents of buffer-1. Therefore, it is not a trivial task to compute limiting buffer content distribution for buffer-2. If buffer-2 is considered separately, the input is from an exponential on-off source but the output capacity varies from zero to $(c_1 + c_2)$ depending on the buffer content in buffer-1. The variation of output capacity over time (say $\hat{O}(t)$) with respect to the content of buffer-1 is as follows:

$$\hat{O}(t) = \begin{cases} 0 & \text{when } X_1(t) \geqslant x^*, \\ c_2 & \text{when } 0 < X_1(t) < x^*, \\ c_1 + c_2 & \text{when } X_1(t) = 0. \end{cases}$$

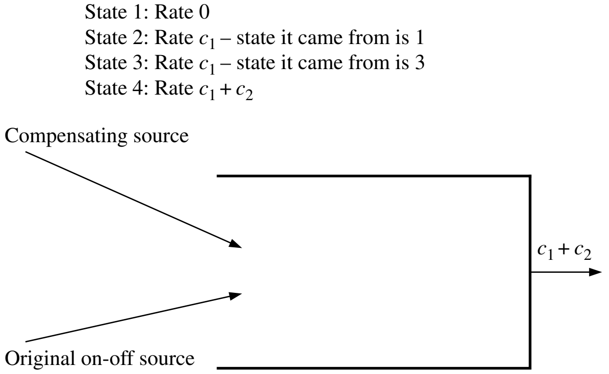
To model buffer-2, which has a time-varying output process, a compensating source (extending the idea developed in Elwalid and Mitra 1995 for fluid queues with priorities) is used, as explained in the next section.

4.1. Compensating Source

Consider the following two cases:

- *Case* A. Fluid enters buffer-2 according to an exponential on-off process with rates r_2 when on and zero when off. The output capacity is $\hat{O}(t)$ as defined above and is time varying. Note that this is identical to what is actually going on in buffer-2.
- *Case* B. A fictitious queueing system (as depicted in Figure 7) where there are two input streams and a server with a constant output capacity $O_2(t) = c_1 + c_2$. The first input stream is a *compensating source* where fluid enters the queue at rate $c_1 + c_2 - \hat{O}(t)$ at time t. Note that this compensating source is not an on-off source. The second input stream is identical to source-2, where fluid enters according to an exponential on-off process with rates r_2 when on and zero when off. Note that this is a standard single buffer with constant output capacity.

Figure 7. Buffer-2 with compensating source.


State 1: Rate 0
State 2: Rate c_1 – state it came from is 1
State 3: Rate c_1 – state it came from is 3
State 4: Rate $c_1 + c_2$
Compensating source
Original on-off source
$c_1 + c_2$

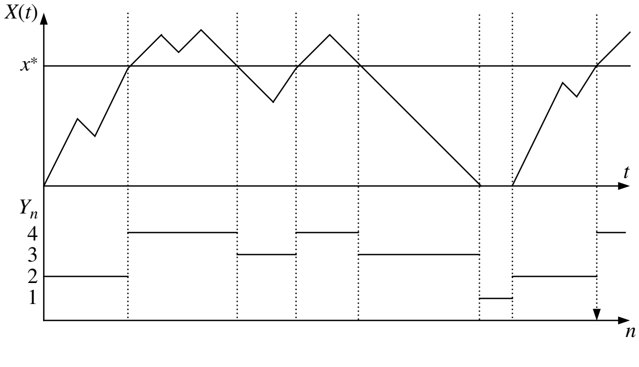
The critical observation to make is that the amount of fluid in buffer-2 in both cases above are identical at all times. This is because at time t the output capacity is $\hat{O}(t)$ in Case A. In Case B, the output capacity is constant at $c = c_1 + c_2$ and an additional $c - \hat{O}(t)$ amount of fluid flows in, nullifying the extra $c - \hat{O}(t)$ capacity that is available. Because both Case A and Case B have the same buffer content at all times, a system with Case B using results extending those for standard single-buffer models is studied.

The environment process that drives traffic generation for the compensating source can be modeled as a 4-state semi-Markov process (SMP). Let $Z_1(t)$ be the environment process denoting the on-off source for buffer-1. If source-1 is on at time t, $Z_1(t) = 1$; and if source-1 is off at time t, $Z_1(t) = 0$. Consider the Markov regenerative sequence $\{(Y_n, S_n), n \geqslant 0\}$, where S_n is the nth regenerative epoch, corresponding to $X_1(t)$ equaling either x^* or 0, and Y_n is the state immediately following the nth Markov regenerative epoch such that

$$Y_n = \begin{cases} 1 & \text{if } X_1(S_n) = 0 \text{ and } Z_1(S_n) = 0, \\ 2 & \text{if } X_1(S_n) = 0 \text{ and } Z_1(S_n) = 1, \\ 3 & \text{if } X_1(S_n) = x^* \text{ and } Z_1(S_n) = 0, \\ 4 & \text{if } X_1(S_n) = x^* \text{ and } Z_1(S_n) = 1. \end{cases}$$

A sample path of Y_n is provided in Figure 8. At time $t = 0$, the buffer is empty—i.e., $X_1(t) = 0$. Let source-1 be on at $t = 0$. Therefore, $Z_1(0) = 1$ and $S_0(0) = 0$. From time $t = 0$ until the time when $X_1(t)$ reaches x^* (as in Figure 8), $Y_0 = 2$. This time is S_1. Because $Z_1(S_1) = 1$ and $X_1(S_1) = x^*$, we have $Y_1 = 4$. The next Markov regenerative epoch S_2 happens when the buffer content comes back to x^*. Then, $Z_1(S_1) = 0$, $X_1(S_1) = x^*$, and so $Y_2 = 3$. Then, the buffer content goes below x^* and comes back to x^* at time S_3. Because $Z_1(S_2) = 1$ and $X_1(S_2) = x^*$, $Y_2 = 3$. Going along the same lines, it can be seen that $Y_3 = 4$, $Y_4 = 3$, $Y_5 = 1$, and so on. The state $Y_5 = 1$ because $X_1(S_5) = 0$ and $Z_1(S_5) = 0$. In the next section, the compensating source is modeled using an SMP.

Figure 8. Illustration of a Markov regenerative sequence.



4.2. Semi-Markov Process Model

The compensating source is governed by an underlying environment process, which is an SMP with four states $\{1, 2, 3, 4\}$ as described in the previous section. Fluid is generated at rates 0, c_1, c_1, and $c_1 + c_2$ when the SMP is, respectively, in states 1, 2, 3, and 4. Let $r(i)$ be the rate of fluid generation when the SMP is in state i. Therefore, $r(1) = 0$, $r(2) = c_1$, $r(3) = c_1$, and $r(4) = c_1 + c_2$. Recall that Y_n takes the value of 1, 2, 3, and 4 (denoting the compensating source with fluid generation rates of 0, c_1, c_1, and $c_1 + c_2$, respectively) and S_n the time when Y_n changes. The kernel of the SMP $G(t) = [G_{ij}(t)]$ will be computed first, which is given by

$$G_{ij}(t) = P\{Y_1 = j, S_1 \leqslant t \mid Y_o = i\}.$$

Because $X_1(t) = 0$ in state 1 and $X_1(t) > x^*$ in state 4, there cannot be a direct transition from state 1 to state 4 and vice versa without actually going through states 2 or 3 (where $0 < X_1(t) < x^*$). Therefore, $G_{14}(t) = G_{41}(t) = 0$. Also, clearly, $G_{ii}(t) = 0$ for $i = 1, 2, 3, 4$. In addition, according to the definitions of states 2 and 3, $G_{23}(t) = G_{32}(t) = G_{42}(t) = G_{13}(t) = 0$.

The kernel matrix $G(t) = [G_{ij}(t)]$ of the SMP is given by

$$G(t) = \begin{bmatrix} 0 & G_{12}(t) & 0 & 0 \\ G_{21}(t) & 0 & 0 & G_{24}(t) \\ G_{31}(t) & 0 & 0 & G_{34}(t) \\ 0 & 0 & G_{43}(t) & 0 \end{bmatrix}.$$

The expressions $G_{12}(t)$, $G_{21}(t)$, $G_{24}(t)$, $G_{31}(t)$, $G_{34}(t)$, and $G_{43}(t)$ need to be obtained. Two of them are relatively straightforward to obtain, namely, $G_{12}(t)$ and $G_{43}(t)$. First consider $G_{12}(t)$. This is the probability that Y_n changes from 1 to 2 before time t, which is the same as the probability of the source-1 going from off to on. So, $G_{12}(t)$ is given by

$$G_{12}(t) = 1 - e^{-\beta_1 t}.$$

Next, consider $G_{43}(t)$. This is the probability that the buffer-2 content goes up from x^* and reaches x^* in time t. This is identical to the probability that the buffer content starting at zero goes up and comes back to zero within time t. The LST of $G_{43}(t)$ can be obtained by substituting appropriate terms in Aggarwal et al. (2004) (see Equations (12), (13), and (14) there), given by

$$\widetilde{G}_{43}(w) = \begin{cases} \dfrac{w + \beta_1 + cs_0(w)}{\beta_1} e^{x^* S_0(w)} & \text{if } w > w^*, \\ \infty & \text{otherwise,} \end{cases}$$

where

$$S_0(w) = \frac{-b - \sqrt{b^2 + 4w(w + \alpha_1 + \beta_1)c(r_1 - c)}}{2c(r_1 - c)},$$

$$b = ((r_1 - 2c)w + (r_1 - c)\beta_1 - c\alpha_1),$$

$$w^* = (2\sqrt{c\alpha_1\beta_1(r_1 - c)} - r_1\beta_1 - c\alpha_1 + c\beta_1)/r_1,$$

and

$$c = c_1 + c_2.$$

For this paper, this LST $\widetilde{G}_{43}(w)$ need not be inverted.

To obtain expressions for the remaining terms in the kernel of the SMP, viz. $G_{21}(t)$, $G_{24}(t)$, $G_{31}(t)$, and $G_{34}(t)$, define the following. Let $T' = \inf\{t > 0\colon X_1(t) = 0$ or $x^*\}$—i.e., the smallest time for $X_1(t)$ to reach either 0 or x^*. For $i = 0, 1$ and $j = 0, 1$, define

$$H'_{ij}(x, t) = P\{T' \leqslant t, Z_1(T') = j \mid X_1(0) = x, Z_1(0) = i\} \quad \text{for } 0 \leqslant x \leqslant x^*.$$

Note that $H'(x, t)$ should not be confused with $H(x, t)$ in §3.1. Now define

$$H'(x, t) = \begin{bmatrix} H'_{00}(x, t) & H'_{01}(x, t) \\ H'_{10}(x, t) & H'_{11}(x, t) \end{bmatrix}.$$

From the definition of $H'(x, t)$, it can be seen that $G_{21}(t)$, $G_{24}(t)$, $G_{31}(t)$, and $G_{34}(t)$ can be expressed as $H'_{10}(0, t)$, $H'_{11}(0, t)$, $H'_{00}(x^*, t)$, and $H'_{01}(x^*, t)$, respectively. So, now the kernel can be written as

$$G(t) = \begin{bmatrix} 0 & 1 - e^{-\beta_1 t} & 0 & 0 \\ H'_{10}(0, t) & 0 & 0 & H'_{11}(0, t) \\ H'_{00}(x^*, t) & 0 & 0 & H'_{01}(x^*, t) \\ 0 & 0 & G_{43}(t) & 0 \end{bmatrix}.$$

To get all the remaining elements in the kernel $G(t)$, one needs to obtain $H'(x, t)$. Recall that the rate matrix

$$R = \begin{bmatrix} -c_1 & 0 \\ 0 & r_1 - c_1 \end{bmatrix}$$

and the generator matrix

$$Q = \begin{bmatrix} -\beta_1 & \beta_1 \\ \alpha_1 & -\alpha_1 \end{bmatrix}.$$

The following theorem shows the dynamics of $H'(x, t)$.

Theorem 6. *The dynamics of $H'(x, t)$ is governed by the following PDE*:

$$\frac{\partial H'(x, t)}{\partial t} - R\frac{\partial H'(x, t)}{\partial x} = QH'(x, t),$$

with initial conditions

$$\begin{aligned} &H_{00}(x, 0) = 0 \quad \textit{for } 0 < x < x^*, \\ &H_{01}(x, 0) = 0 \quad \textit{for } x \geqslant 0, \\ &H_{10}(x, 0) = 0 \quad \textit{for } x \geqslant 0, \\ &H_{11}(x, 0) = 0 \quad \textit{for } 0 < x < x^*, \end{aligned} \qquad (17)$$

and boundary conditions

$$H'_{11}(x^*, t) = 1, \tag{18}$$

$$H'_{00}(0, t) = 1, \tag{19}$$

$$H'_{01}(0, t) = 0, \tag{20}$$

$$H'_{10}(x^*, t) = 0. \tag{21}$$

Proof. Deriving the PDE is identical to that in Theorem 4. However, the derivation of the initial and boundary conditions needs to be explained.

Initial conditions: The initial conditions (i.e., Equations (17)) are due to the facts that: (i) the first passage time cannot occur instantaneously when $0 < x < x^*$, and (ii) the source cannot be in different states at the same time point for any $x \geqslant 0$.

Boundary conditions: The first boundary equation (i.e., Equation (18)) is so because if the initial buffer-1 content is x^* and source-1 is on, then essentially the first passage time has occurred. Therefore, the probability that the first passage time occurs before time t and the source is on when it occurred is one. The second boundary equation (Equation (19)) is based on the fact that if the initial buffer-1 content is zero and source-1 is off, then the first passage time is zero. Hence, the probability that the first passage time occurs before time t and the source is off when it occurred is one. The third boundary equation (i.e., Equation (20)) is due to the fact that although the first passage time is zero, the probability that the source is on when the first passage time occurs is zero (because at time $t = 0$, the source is off). For exactly the same reason, the last boundary equation (Equation (21)) is the way it is—i.e., the first passage time is zero, but it cannot occur when the source is off, given that the source was on at time $t = 0$. □

To solve the PDE, take the LST across the PDE w.r.t t. That reduces to the following:

$$R\frac{d\widetilde{H}(x, w)}{dx} = (wI - Q)\widetilde{H}(x, w).$$

Let ϕ_1 and ϕ_2 be the right eigenvectors and $S_1(w)$ and $S_2(w)$ be the corresponding eigenvalues. The solution to the above ODE is given by

$$\begin{bmatrix} \widetilde{H}'_{00}(x, w) \\ \widetilde{H}'_{10}(x, w) \end{bmatrix} = a_{00}e^{S_1(w)x}\phi_1(x) + a_{10}e^{S_1(w)x}\phi_2(x),$$

$$\begin{bmatrix} \widetilde{H}'_{01}(x, w) \\ \widetilde{H}'_{11}(x, w) \end{bmatrix} = a_{01}e^{S_1(w)x}\phi_1(x) + a_{11}e^{S_1(w)x}\phi_2(x),$$

where $S_1(w)$, $S_2(w)$, ϕ_1, and ϕ_2 are similar to Equations (15) and (16) obtained in §3.1. The closed-form algebraic expressions for $S_i(w)$ and $\phi_i(w)$ are

$$S_1(w) = \frac{-b - \sqrt{b^2 + 4w(w + \alpha_1 + \beta_1)c(r_1 - c)}}{2c(r_1 - c)},$$

$$S_2(w) = \frac{-b + \sqrt{b^2 + 4w(w + \alpha_1 + \beta_1)c(r_1 - c)}}{2c(r_1 - c)},$$

where $b = ((r_1 - 2c)w + (r_1 - c)\beta_1 - c\alpha_1)$, $w > (2\sqrt{c\alpha_1\beta_1(r_1 - c)} - r_1\beta_1 - c\alpha_1 + c\beta_1)/r_1$, and $c = c_1 + c_2$. Also,

$$\phi_i(w) = \begin{bmatrix} \dfrac{w + \alpha_1 - S_i(w)(r - c)}{\alpha_1} \\ 1 \end{bmatrix} = \begin{bmatrix} \dfrac{\beta_1}{w + \beta_1 + S_i(w)} \\ 1 \end{bmatrix}.$$

Solving for a_{00}, a_{10}, a_{01}, and a_{11} using the LST of the boundary conditions $\widetilde{H}'_{11}(x^*, w) = 1$, $\widetilde{H}'_{00}(0, w) = 1$, $\widetilde{H}'_{10}(x^*, w) = 0$, and $\widetilde{H}'_{01}(0, w) = 0$, results in

$$\begin{aligned} a_{00} &= e^{S_2(0)x^*}/D, \\ a_{01} &= -\phi_2(0)/D, \\ a_{10} &= -e^{S_1(0)x^*}/D, \\ a_{11} &= \phi_1(0)/D, \end{aligned} \tag{22}$$

where $D = e^{S_2(0)x^*}\phi_1(0) - e^{S_1(0)x^*}\phi_2(0)$.

Thus, all the terms for the LST of the kernel, i.e.,

$$\widetilde{G}(w) = \int_0^\infty e^{-wt}\, dG(t),$$

have been derived. The LST cannot be inverted easily to get back $G(t)$ except for specific numerical examples. However, for this paper, only the LST of the kernel is required and there is no need to derive $G(t)$ explicitly.

Having completely characterized $\widetilde{G}(w)$, the LST of the kernel of the SMP, the next step is to obtain the limiting probabilities for the states of the SMP as well as their sojourn times. Let τ_i be the expected sojourn time in state i of the SMP (i.e., the average time the SMP spends in state i before transitioning to another state). For $i = 1, 2, 3, 4$,

$$\tau_i = -\sum_{j=1}^{4} \widetilde{G}'_{ij}(0),$$

where $\widetilde{G}'_{ij}(0)$ is the derivative of $\widetilde{G}_{ij}(w)$ taken at $w = 0$. Note that $\{Y_n, n \geqslant 0\}$ is a discrete-time Markov chain (DTMC) embedded in the SMP. Let P be the transition probability matrix of the embedded DTMC such that P is a matrix of p_{ij} values and $P = G(\infty) = \widetilde{G}(0)$. Let $\pi = [\pi_1 \;\; \pi_2 \;\; \pi_3 \;\; \pi_4]$ be the steady-state probabilities of the embedded DTMC. Clearly, $\pi = \pi P$ and $\pi_1 + \pi_2 + \pi_3 + \pi_4 = 1$. After solving for π and τ, the fraction of time the SMP spends in state i is p_i which can be derived as

$$p_i = \frac{\pi_i \tau_i}{\sum_{j=1}^{4} \pi_j \tau_j}.$$

Therefore, the SMP is completely characterized and can be used to analyze the contents of buffer-2 as explained in the next section.

4.3. Buffer Content Analysis Using a Compensating Source

Consider Case B described in §4.1 and depicted in Figure 7. A compensating source is used to make the output capacity a constant, thereby converting the system into a standard single buffer (defined in §2.5). Recall that the buffer content at any time would be identical between Case A and Case B. Let X_2 be a random variable such that when $t \to \infty$, $X_2(t) \to X_2$. It is not possible to obtain a closed-form expression for $P(X_2 > x)$ unless the compensating source is a continuous-time Markov chain (CTMC). However, because the compensating source is an SMP, one can leverage upon some results in Gautam et al. (1999), which describes how to obtain bounds for standard single buffers as shown in §2.5 for SMP environment processes. Using the results in Gautam et al. (1999), one can obtain bounds for $P(X_2 > x)$. It is important to note that the bounds for $P(X_2 > x)$ are valid for any x, although traditionally this type of analysis is applied only for the asymptotic case of $x \to \infty$—i.e., the tail of the distribution.

Note that in Case B, there are two sources that input traffic into a standard single buffer with output capacity $c_1 + c_2$. The compensating source is considered first, and the original source-2 is considered next. For both sources, the aim is to obtain their "effective bandwidths," which in some sense captures the source characteristics. For any positive v, the effective bandwidth of a source that generates $A(t)$ amount of fluid in time t is (see Elwalid and Mitra 1993)

$$eb(v) = \lim_{t \to 0} \frac{1}{vt} \log E[e^{vA(t)}].$$

It is fairly straightforward to compute the effective bandwidth if the source is a CTMC; however, the analysis is a little more involved if the source is an SMP.

The effective bandwidth of the compensating source can be computed as follows (adapted from Gautam et al. 1999). For a given v such that $v > 0$, define the matrix $\chi(v, u)$ such that

$$\chi(v,u) = \begin{bmatrix} 0 & \widetilde{G}_{12}(vu) & 0 & 0 \\ \widetilde{G}_{21}(vu - c_1 v) & 0 & 0 & \widetilde{G}_{24}(vu - c_1 v) \\ \widetilde{G}_{31}(vu - c_1 v) & 0 & 0 & \widetilde{G}_{34}(vu - c_1 v) \\ 0 & 0 & \widetilde{G}_{43}(vu - c_1 v - c_2 v) & 0 \end{bmatrix}.$$

Let $e(A)$ be the Perron-Frobenius eigenvalue of a square matrix A. The effective bandwidth of the compensating source for a particular positive value of v (i.e., $eb_c(v)$) is given by the unique solution to

$$e(\chi(v, eb_c(v))) = 1.$$

Having obtained the effective bandwidth of the compensating source, the next step is to obtain that for the other source. Because the original source-2 has a CTMC as the environment process, its effective bandwidth (i.e., $eb_2(v)$) can be obtained in closed form. In particular, the effective bandwidth of the exponential on-off source-2 is (see Elwalid and Mitra 1993)

$$eb_2(v) = \frac{r_2 v - \alpha_2 - \beta_2 + \sqrt{(r_2 v - \alpha_2 - \beta_2)^2 + 4\beta_2 r_2 v}}{2v}. \quad (23)$$

The effective bandwidth of a source formed by aggregating two or more independent flows is equal to the sum of the effective bandwidth of the independent flows. Therefore, because the compensating source and source-2 are independent, if the fluid queue is stable, let η be the unique solution to

$$eb_2(\eta) + eb_c(\eta) = c_1 + c_2. \quad (24)$$

For notational convenience, use the following:

$$\gamma_2 = eb_2(\eta),$$
$$\gamma_c = eb_c(\eta).$$

Define $\Phi(\eta) = \chi(\eta, eb_c(\eta))$ such that $\phi_{ij}(\eta)$ is the ijth element of $\Phi(\eta)$. Let h be the left eigenvector of $\Phi(\eta)$ corresponding to the eigenvalue of one, i.e.,

$$h = h\Phi(\eta).$$

Also note that $h = [h_1\ h_2\ h_3\ h_4]$. Next, using η, h, γ_2, γ_c and other parameters defined earlier, the bounds for $P(X_2 > x)$ will be obtained.

4.4. Bounds for Distribution of X_2

The following theorem gives the lower and upper bounds on the probability that X_2 exceeds a value x.

Theorem 7. *Using $\Psi_{\max}(i,j)$ and $\Psi_{\min}(i,j)$, one gets*

$$K_* e^{-\eta x} \leqslant P(X_2 > x) \leqslant K^* e^{-\eta x},$$

where

$$K^* = \frac{\left\{\frac{r_2\beta_2}{\gamma_2(\alpha_2+\beta_2)}\right\}\sum_{i=1}^4 \left\{\frac{h_i}{\eta(r(i)-\gamma_c)}\left(\left[\sum_{j=1}^4 \phi_{ij}(\eta)\right]-1\right)\right\}}{\min_{(i,\,j):\,p_{ij}>0} \Psi_{\min}(i,j)}$$

and

$$K_* = \frac{\left\{\frac{r_2\beta_2}{\gamma_2(\alpha_2+\beta_2)}\right\}\sum_{i=1}^4 \left\{\frac{h_i}{\eta(r(i)-\gamma_c)}\left(\left[\sum_{j=1}^4 \phi_{ij}(\eta)\right]-1\right)\right\}}{\max_{(i,\,j):\,p_{ij}>0} \Psi_{\max}(i,j)},$$

with $\Psi_{\min}(i,j)$ and $\Psi_{\max}(i,j)$ defined in the appendix.

Proof. From Gautam et al. (1999), it is relatively straightforward to obtain the numerators for K^* and K_*, which in fact are identical. The expressions for the denominators for K^* and K_* need to be explained, and this is done in the appendix. □

Table 1. Input parameters.

Scenario	r_1	r_2	α_1	α_2	β_1	β_2	c_1	c_2
A: $r_1 < r_2, \alpha_1 < \alpha_2$, policy 1	12.48	16.25	11	12	1	0.5	1.6	1.0
B: $r_1 > r_2, \alpha_1 > \alpha_2$, policy 1	12.48	1.625	11	0.75	1	0.5	1.6	1.0
C: $r_1 < r_2, \alpha_1 > \alpha_2$, policy 1	12.48	16.25	11	1.2	1	0.05	1.6	1.0
D: $r_1 > r_2, \alpha_1 < \alpha_2$, policy 1	12.48	1.625	11	12	1	8	1.6	1.0
E: $m_1 < c_1$, policy 2	12.48	16.25	11	12	1	0.5	1.19	1.41
F: $m_1 > c_1$, policy 2	12.48	16.25	11	1.2	1	0.05	0.2	2.4

5. Evaluations

After deriving a closed-form expression for $P(X_1 > x)$ and bounds for $P(X_2 > x)$, the next question to ask is what they look like for various numerical test cases. In particular, of importance is their effect on burstiness, frequency of alternating between on and off states, policies for selecting c_1 and c_2 as described in §2.3, etc. This is done in §5.1. Further, in §5.2, numerical experiments to design an optimal x^* are provided consistent with the optimization problem described in §2.3. However, first a set of numerical values that are used as "input" to the analyses is explained.

Six sets of numerical values are considered representing scenarios A, B, C, D, E, and F. Note that in all the scenarios, the average arrival rates of both sources are kept constant. Scenario A is called the baseline case. All five of the other scenarios (B to F) are modifications made to scenario A. The six scenarios are tabulated in Table 1. Scenarios A to D use policy 1 in §2.3 to determine c_1 and c_2 when $c = 2.6$ with the key difference being how r_1 and r_2, as well as α_1 and α_2, compare against one another. However, scenarios E and F use policy 2 with $m_1 = r_1\beta_1/(\alpha_1 + \beta_1)$, with the key difference being how m_1 compares against c_1.

Before looking at the numerical results, an approximation is needed for $P(X_2 > x)$. This is because the objective function in the optimization problem is in terms of $P(X_2 > B_2)$ (which may not have been required for some other objective or if $P(X_2 > B_2)$ appeared as a constraint). From extensive simulations, it has been observed that the actual limiting probability $(P(X_2 > x))$ is a lot closer to the lower bound (i.e., $K_* e^{-\eta x}$) than the upper bound. Based on the simulations, an approximation for $P(X_2 > x)$ that works reasonably well is

$$P(X_2 > x) \approx 0.9 K_* e^{-\eta x} + 0.1 K^* e^{-\eta x}, \tag{25}$$

where $K_* e^{-\eta x}$ and $K^* e^{-\eta x}$ are, respectively, the lower and upper bounds for $P(X_2 > x)$. In the tables and graphs, LB denotes the lower bound for $P(X_2 > B_2)$, which is $K_* e^{-\eta B_2}$; and UB denotes the upper bound for $P(X_2 > B_2)$, which is $K^* e^{-\eta B_2}$. Also, if $P(X_2 > B_2)$ alone is stated, then it is from the approximation in Equation (25).

5.1. Numerical Experiments

For all the numerical experiments in this section, the following parameter values are chosen: $B_1 = 3$, $B_2 = 8$, and $x^* = 1.5$. For the six scenarios A to F in Table 1, the probabilities of the buffer contents (in buffer-1 and buffer-2) exceeding B_1 and B_2, respectively, are tabulated in Table 2. The third, fourth, and fifth columns are, respectively, lower bound, upper bound, and approximation for $P(X_2 > B_2)$. Also, the last column denotes $P(X_1 > x^*)$, which is used in the constraint of the optimization problem in §2.3.

Refer to Table 2. Because X_1 depends only on the parameters of source-1 and c_1, the result for $P(X_1 > B_1)$ remains unchanged for scenarios A to D (which also have the same r_1, α_1, β_1, and c_1 values). In scenario A, as well as scenario E, $m_1 < c_1$, and therefore $P(X_1 > B_1)$, is not very different, unlike scenario F. However, m_1 is closer to c_1 in scenario E as compared to scenario A. In going from scenarios A to E to F, $P(X_1 > x^*)$ increases because when $X_1 \leqslant x^*$, the ratio of average arrival rate to service rate increases. Thereby, as per intuition, the probability $P(X_1 > B_1)$ also increases correspondingly.

Next, consider buffer-2 results. The trends for lower bound, upper bound, and approximation are identical. Therefore, the results are analyzed for the approximation (i.e., the fifth column in Table 2). Comparing the burstiness of source-1 and source-2 in scenarios A and B, the burstiness of source-2 is less in scenario B and almost equal in scenario A. It can be seen that $P(X_2 > B_2)$ is slightly more for scenario B than that of scenario A. Although the burstiness of source-2 is less, the ratio r_2/α_2 (average traffic generated when source is on) is slightly more than r_1/α_1, causing a slightly higher $P(X_2 > B_2)$. In scenario C, the burstiness of source-2 is more than source-1 (i.e., $r_1 < r_2$), but $\alpha_1 > \alpha_2$. Therefore, the ratio r_2/α_2 is a lot more than r_1/α_1, causing a lot higher $P(X_2 > B_2)$ for scenario C. In scenario D, although from a burstiness standpoint alone it should have had a much lower loss probability because the on and off times of source-2 are much closer (and

Table 2. Buffer content probabilities.

Scenario	$P(X_1 > B_1)$	LB of $P(X_2 > B_2)$	UB of $P(X_2 > B_2)$	Appr. of $P(X_2 > B_2)$	$P(X_1 > x^*)$
A	0.0572	0.0271	0.0680	0.0312	0.1706
B	0.0572	0.0354	0.0843	0.0403	0.1706
C	0.0572	0.2699	0.3073	0.2736	0.1706
D	0.0572	0.0507	0.1170	0.0573	0.1706
E	0.0651	0.0286	0.0850	0.0342	0.1942
F	0.1173	0.2840	0.4647	0.3021	0.3501

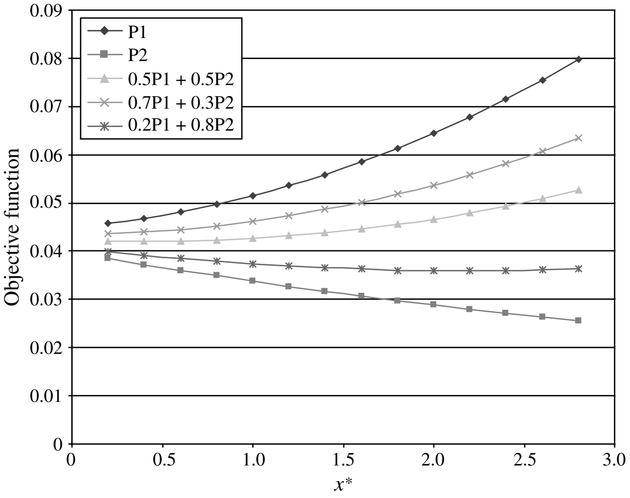
faster), the variability in buffer content is higher. It appears that if c_2 is higher (as observed going from scenario A to E to F), $P(X_2 > B_2)$ should go higher. However, the results are just the opposite, which can be attributed to the fact that $P(X_1 > x^*)$ is also higher, which implies that a higher fraction of time c_2 is not available.

5.2. Design Optimization

Having described the numerical findings for $P(X_1 > B_1)$ and $P(X_2 > B_2)$, the next step is to solve the optimization problem in §2.3 to choose an optimal x^* value. For all experiments, $B_1 = 3$ and $B_2 = 8$. In all previous experiments, $x^* = 1.5$, a constant. Here, x^* is optimized where the objective function and the constraint of the optimization problem are given by Equations (1) and (2), respectively. Various values for w_1 and w_2 are considered and the variation of the optimal value of x^* is illustrated. For this example, scenario A in Table 1 is used. For the numerical experiments, the performance measures are $P(X_1 > B_1)$, the approximate value of $P(X_2 > B_2)$ (substituting appropriate values in Equation (25)), the objective function values (denoted by obj. val.) for three different combinations of w_1 and w_2 (to plug into Equation (1)) and $P(X_1 > x^*)$. The results are tabulated in Table 3.

A graph is plotted (see Figure 9, where P1 and P2 are, respectively, $P(X_1 > B_1)$ and $P(X_2 > B_2)$) between x^* and a different set of weights as given in Table 3. The SLA value ν is set as 85%, i.e., $P(X_1 > x^*) \leqslant 0.15$. Consider the first set of weights ($w_1 = 0.5$, $w_2 = 0.5$). The objective function decreases a little and then increases (easier to see in Table 3 than in Figure 9). One can obtain x^* where $P(X_1 > x^*) = 0.15$ because the x^* where the function minima occurs ($x^* = 0.4$) is not feasible. It is a very similar case for the second set of weights ($w_1 = 0.7$, $w_2 = 0.3$) (except that the objective function is monotonically increasing), where one can obtain x^* as the value where $P(X_1 > x^*) = 0.15$. However, in the third case, where $w_1 = 0.2$, $w_2 = 0.8$, the objective function reduces monotonically up to a point and starts increasing again. Because the value of x^* where the slope of the objective function is zero is less than the point where $P(X_1 > x^*) = 0.15$, the optimal x^* is at the unconstrained minimum ($x^* = 2.1$ approximately). If the SLA constraint (inequality (2)) is changed, then the optimal value of x^* also changes appropriately. Thus, the optimal value of x^* can be obtained and can be used as a design value for a given set of input parameters.

Figure 9. Variation of the objective function and probabilities with x^*.



6. Concluding Remarks and Future Work

In several computer-communication systems, and possibly in other systems as well, a resource such as bandwidth or processor capacity is sometimes available in excess and could be shared with other users in the system. This is an important step even from an energy conservation viewpoint. In this paper, a two-buffer queue with fluid on-off sources for each buffer is considered. Buffer-1 is the application that usually needs only a capacity c_1

Table 3. Variation of probabilities with x^*.

x^*	$P(X_1 > B_1)$	$P(X_2 > B_2)$	$P(X_1 > x^*)$	Obj. val. $w_1 = 0.5, w_2 = 0.5$	Obj. val. $w_1 = 0.7, w_2 = 0.3$	Obj. val. $w_1 = 0.2, w_2 = 0.8$
0.2	0.0457	0.0385	0.3516	0.0421	0.0435	0.0399
0.4	0.0468	0.0372	0.311	0.0420	0.0439	0.0391
0.6	0.0481	0.0360	0.2765	0.0421	0.0445	0.0384
0.8	0.0497	0.0349	0.2469	0.0423	0.0452	0.0378
1	0.0515	0.0338	0.2213	0.0426	0.0462	0.0373
1.2	0.0536	0.0327	0.1990	0.0431	0.0473	0.0369
1.4	0.0559	0.0317	0.1795	0.0438	0.0486	0.0365
1.6	0.0585	0.0307	0.1623	0.0446	0.0502	0.0362
1.8	0.0613	0.0297	0.1471	0.0455	0.0518	0.0360
2	0.0644	0.0288	0.1335	0.0466	0.0537	0.0359
2.2	0.0678	0.0279	0.1214	0.0479	0.0558	0.0359
2.4	0.0715	0.0271	0.1106	0.0493	0.0582	0.0360
2.6	0.0754	0.0263	0.1009	0.0509	0.0607	0.0361
2.8	0.0797	0.0256	0.0922	0.0526	0.0635	0.0364

of the resource and shares the remaining c_2 with other applications (i.e., buffer-2). Because only the workload in buffer-1 is observed, if it becomes too high (greater than x^*), it takes capacity c_2 from buffer-2. However, if the workload in buffer-1 becomes zero, the entire capacity $c_1 + c_2$ is given to buffer-2.

The main objective of this paper is to obtain the marginal distribution of the buffer contents in steady state for each buffer—i.e., $P(X_1 > x)$ and $P(X_2 > x)$. Exact closed-form algebraic expressions were obtained for $P(X_1 > x)$ by solving a partial differential equation for the buffer content dynamics. For $P(X_2 > x)$, bounds were obtained by first modeling buffer-2 using two sources, one the usual source-2 and the other a compensating source so that the output capacity can be a constant. One of the key contributions of this paper is to model the compensating source using a semi-Markov process and to obtain its kernel. The paper also suggests techniques to assign c_1 and c_2 as well as to solve an optimization problem to select an optimal x^*. It is critical to note that other techniques could have been used as well and a different optimization problem could have been formulated.

In the future, an aim would be to relax the exponential assumption for the on-off sources. Although for source-2 this would not change too much, but the Markovian structure would be lost if source-1 is not exponential. In that case, only tail distributions (i.e., for $x \to \infty$) of $P(X_i > x)$ for $i = 1, 2$ can be obtained. In addition, by modeling the traffic as Levy processes and using analysis as described in Bertoin (1996), similar systems with general on-time distributions can be analyzed. Another future consideration is a random delay in getting the state information, i.e., the buffers are polled from time to time and its contents are revealed a random time after they are polled. In addition, for wireless communication especially, the channel capacity does not stay fixed, i.e., $c_1 + c_2$ is also time varying. Also, in the future, buffer-2 does not have to be restricted to one source. There could be several independent sources that generate traffic into buffer-2, consistent with the applications in mind. Similarly, there are some other interesting options to explore in the future.

Appendix

The main aim of this appendix is to describe $\Psi_{\max}(i, j)$ and $\Psi_{\min}(i, j)$ as indicated in Theorem 7. In that process, the proof of Theorem 7 would also be complete. For that, first a few definitions need to be given, as well as a lemma that needs to be stated and proved. Refer to §4.3, as well as the latter part of §4.2, for notation.

Consider the P matrix (i.e., $\widetilde{G}(0)$) of the SMP driving the compensating source. For all (i, j) such that $p_{ij} > 0$, the failure-rate function is defined as

$$\lambda_{ij}(t) = \frac{G'_{ij}(t)}{G_{ij}(\infty) - G_{ij}(t)} = \frac{G'_{ij}(t)}{p_{ij} - G_{ij}(t)}. \tag{26}$$

Two things are needed for this analysis: the first is if $\lambda_{ij}(t)$ is increasing (called increasing failure rate or IFR) or decreasing (called decreasing failure rate or DFR) w.r.t. t; the second is an expression for $\lambda_{ij}(\infty)$. The actual failure-rate function is not required explicitly. In terms of the monotonicity of the failure-rate functions, the following are some realizations: (i) $\lambda_{12}(t)$ is a constant; (ii) $\lambda_{21}(t)$, $\lambda_{34}(t)$, and $\lambda_{43}(t)$ are decreasing functions of t (due to similarity with busy-period hazard-rate functions in queues), hence they are DFRs; (iii) $\lambda_{24}(t)$ and $\lambda_{32}(t)$ are increasing functions of t (due to similarity with hazard rates of hitting times), and hence they are IFR.

As discussed above, the next item is to derive an expression for $\lambda_{ij}(\infty)$, which is provided in the following lemma.

Lemma 2. *The failure-rate function as $t \to \infty$ can be expressed as*

$$\lambda_{ij}(\infty) = -\frac{\widetilde{G}_{ij}(0)}{\widetilde{G}'_{ij}(0)}.$$

Proof. For any function $F(t)$, its LST $\widetilde{F}(s)$, and its derivative $f(t) = F'(t)$, the following results are well known:

$$\begin{aligned} &\lim\nolimits_{t\to\infty} F(t) = \lim\nolimits_{s\to 0} \widetilde{F}(s),\\ &\tilde{f}(s) = s\widetilde{F}(s) - sF(0), \end{aligned} \tag{27}$$

where $\tilde{f}(s)$ is the LST of $f(t)$. From Equation (26), by taking the limit as $t \to \infty$, both numerator and denominator tend to zero, thus, L'Hospital's rule needs to be applied. Taking the derivative of the denominator, one gets $-G'_{ij}(t)$. In the numerator it is $G'_{ij}(t)$. From Equation (27), substituting $G_{ij}(t)$ for $F(t)$ and taking derivative on both sides w.r.t s,

$$\widetilde{G}''_{ij}(s) = s\widetilde{G}'_{ij}(s) + \widetilde{G}_{ij}(s) - G_{ij}(0).$$

Therefore, the failure-rate function in the limit becomes

$$\lim_{t\to\infty} \lambda_{ij}(t) = \lim_{t\to\infty} \frac{-G''_{ij}(t)}{G'_{ij}(t)} = \lim_{s\to 0} \frac{-\widetilde{G}''_{ij}(s)}{\widetilde{G}'_{ij}(s)}. \tag{28}$$

From Equations (28) and (27),

$$\lim_{t\to\infty} \lambda_{ij}(t) = \lim_{s\to 0} \frac{s\widetilde{G}'_{ij}(s) + \widetilde{G}_{ij}(s) - G_{ij}(0)}{-\widetilde{G}'_{ij}(s)} = \lim_{s\to 0} \frac{\widetilde{G}_{ij}(s)}{-\widetilde{G}'_{ij}(s)}.$$

Thus, using L'Hospital's rule and the fact that $G_{ij}(0) = 0$, one can derive $\lambda_{ij}(\infty)$ as

$$\lambda_{ij}(\infty) = -\frac{\widetilde{G}_{ij}(0)}{\widetilde{G}'_{ij}(0)}. \quad \square$$

Now, using the results for whether $\lambda_{ij}(t)$ is IFR, DFR, or constant (where both IFR and DFR would yield the same result), as well as for $\lambda_{ij}(\infty)$ for each (i, j), $\Psi_{\max}(i, j)$ and

Table A.4. Table of $\Psi_{\max}$ and $\Psi_{\min}$ values.

	IFR & $r(i) > \gamma_c$	IFR & $r(i) \leqslant \gamma_c$	DFR & $r(i) > \gamma_c$	DFR & $r(i) \leqslant \gamma_c$
$\Psi_{\max}(i, j)$	$\dfrac{\phi_{ij}(-\eta(r(i)-\gamma_c))\tau_i h_i}{p_{ij}p_i}$	$\dfrac{\tau_i h_i \lambda_{ij}(\infty)}{p_i(\lambda_{ij}(\infty)-\eta(r(i)-\gamma_c))}$	$\dfrac{\tau_i h_i \lambda_{ij}(\infty)}{p_i(\lambda_{ij}(\infty)-\eta(r(i)-\gamma_c))}$	$\dfrac{\tilde{\phi}_{ij}(-\eta(r(i)-\gamma_c))\tau_i h_i}{p_{ij}p_i}$
$\Psi_{\min}(i, j)$	$\dfrac{\tau_i h_i \lambda_{ij}(\infty)}{p_i(\lambda_{ij}(\infty)-\eta(r(i)-\gamma_c))}$	$\dfrac{\tilde{\phi}_{ij}(-\eta(r(i)-\gamma_c))\tau_i h_i}{p_{ij}p_i}$	$\dfrac{\tilde{\phi}_{ij}(-\eta(r(i)-\gamma_c))\tau_i h_i}{p_{ij}p_i}$	$\dfrac{\tau_i h_i \lambda_{ij}(\infty)}{p_i(\lambda_{ij}(\infty)-\eta(r(i)-\gamma_c))}$

$\Psi_{\min}(i, j)$ can be derived using the values in Table A.4 and used in Theorem 7.

Acknowledgments

The authors thank the area editor as well as the anonymous associate editor and reviewers for their comments and suggestions that led to considerable improvements in the content and presentation of this paper. This research was partially supported by NSF grants ACI-0325056 and ANI-0219747 under the ITR initiative, as well as by the DARPA grant MDA 972-01-1-0038.

References

Aggarwal, V., N. Gautam, S. R. T. Kumara, M. Greaves. 2004. Stochastic fluid flow models for determining optimal switching thresholds with an application to agent task scheduling. *Performance Evaluation* **59**(1) 19–46.

Anick, D., D. Mitra, M. M. Sondhi. 1982. Stochastic theory of a data handling system with multiple sources. *Bell System Tech. J.* **61** 1871–1894.

Bertoin, J. 1996. *Levy Processes*. Cambridge University Press, Melbourne, NY.

Botvich, D., N. Duffield. 1995. Large deviations, the shape of the loss curve, and economies of scale in large multiplexers. *Queueing Systems* **20** 293–320.

Boxma, O. J. 1984. Two symmetric queues with alternating service and switching times. *Proc. 10th Internat. Sympos. Models Comput. System Performance*, Paris, France, 409–431.

Daganzo, C. F. 1990. Some properties of polling systems. *Queueing Systems: Theory Appl.* **6** 137–154.

Dai, J. G., J. J. Hasenbein, J. H. VandeVate. 1999. Stability of a three-station fluid network. *Queueing Systems* **33**(4) 293–325.

Darce, M., K. D. Glazebrook, J. Nino-Mora. 1999. The achievable region approach to the optimal control of stochastic systems. *J. Roy. Statist. Soc.* **B61** 741–791.

Elwalid, A. I., D. Mitra. 1991. Analysis and design of rate-based congestion control of high speed networks, part I: Stochastic fluid models, access regulation. *Queueing Systems: Theory Appl.* **9** 29–64.

Elwalid, A. I., D. Mitra. 1992. Fluid models for the analysis and design of statistical multiplexing with loss priorities on multiple classes of bursty traffic. *IEEE Trans. Comm.* **11**(11) 2989–3002.

Elwalid, A. I., D. Mitra. 1993. Effective bandwidth of general Markovian traffic sources and admission control of high speed networks. *IEEE/ACM Trans. Networking* **1**(3) 329–343.

Elwalid, A. I., D. Mitra. 1994. Statistical multiplexing with loss priorities in rate-based congestion control of high-speed networks. *IEEE Trans. Comm.* **42**(11) 2989–3002.

Elwalid, A. I., D. Mitra. 1995. Analysis, approximations and admission control of a multi-service multiplexing system with priorities. *IEEE INFOCOM*, Vol. 2, 463–472.

Elwalid, A. I., D. Heyman, T. V. Lakshman, D. Mitra, A. Weiss. 1995. Fundamental bounds and approximations for ATM multiplexers with applications to video teleconferencing. *IEEE J. Selected Areas Comm.* **13**(6) 1004–1016.

Gautam, N. 2002. Buffered and unbuffered leaky bucket policing: Guaranteeing QoS, design and admission control. *Telecomm. Systems* **21**(1) 35–63.

Gautam, N., V. G. Kulkarni. 2000. Applications of SMP bounds to multiclass traffic in high-speed networks. *Queueing Systems: Theory Appl.* **36** 351–379.

Gautam, N., V. G. Kulkarni, Z. Palmowski, T. Rolski. 1999. Bounds for fluid models driven by semi-Markov inputs. *Probab. Engrg. Informational Sci.* **13**(4) 429–475.

Gnanasambandam, S., S. Lee, S. R. T. Kumara, N. Gautam, W. Peng, V. Manikonda, M. Brinn, M. Greaves. 2005. Survivability of a distributed multi-agent application—A performance control perspective. *IEEE 2nd Sympos. Multi-Agent Security and Survivability*, Washington, D.C., 21–30.

Helsinger, A., R. Lazarus, W. Wright, J. Zinky. 2003. Tools and techniques for performance measurement of large distributed multiagent systems. *Second Internat. Joint Conf. Autonomous Agents and Multiagent Systems*, Melbourne, Australia, 843–850.

Kesidis, G. 1996. *ATM Network Performance*. Kluwer Academic Publishers, Boston.

Kostin, A. E., I. Aybay, G. Oz. 2000. A randomized contention-based load-balancing protocol for a distributed multiserver queueing system. *IEEE Trans. Parallel Distributed Systems* **11**(12) 1252–1273.

Kulkarni, V. G. 1997. Effective bandwidths for Markov regenerative sources. *Queueing Systems* **24** 137–153.

Kulkarni, V. G., N. Gautam. 1996. Leaky buckets: Sizing and admission control. *Proc. 35th IEEE Conf. Decision and Control*, Kobe, Japan, 785–790.

Kulkarni, V. G., N. Gautam. 1997. Admission control of multi-class traffic with service priorities in high speed networks. *Queueing Systems: Theory Appl.* **27**(1–2) 79–97.

Kulkarni, V. G., T. Rolski. 1994. Fluid model driven by an Ornstein-Uhlenbeck process. *Probab. Engrg. Inform. Sci.* **8** 403–417.

Mirchandany, R., D. Towsley, J. A. Stankovic. 1990. Adaptive load sharing in heterogenous distributed systems. *J. Parallel and Distributed Comput.* **9**(4) 331–346.

Narayanan, A., V. G. Kulkarni. 1996. First passage times in fluid models with an application to two-priority fluid systems. *IEEE Internat. Comput. Performance and Dependability Sympos.*, Urbana-Champaign, 166–175.

Palmowski, Z., T. Rolski. 1996. A note on martingale inequalities for fluid models. *Statist. Probab. Lett.* **31**(1) 13–21.

Simonian, A. 1991. Stationary analysis of a fluid queue with input rate varying as an Ornstein-Uhlenbeck process. *SIAM J. Appl. Math.* **51** 823–842.

Takagi, H. 1986. *Analysis of Polling Systems*. MIT Press, Cambridge, MA.

Tantawi, A. N., D. Towsley. 1985. Optimal static load balancing in distributed computer systems. *J. ACM* **32**(2) 445–465.

Whittle, P. 1988. Restless bandits: Activity allocation in a changing world. *J. Appl. Probab.* **A25** 287–298.