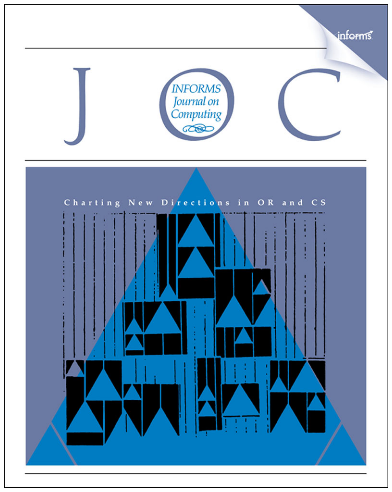

INFORMS Journal on Computing
Vol. 29, No. 1, Winter 2017, pp. 1–17
ISSN 1091-9856 (print) | ISSN 1526-5528 (online)

https://doi.org/10.1287/ijoc.2016.0702

Sampling the Functional Kolmogorov Forward Equations for Nonstationary Queueing Networks

Jamol Pender
School of Operations Research and Information Engineering, Cornell University, Ithaca, New York 14850,
jjp274@cornell.edu

Nonstationary queueing networks are often difficult to approximate. Recent novel methods for approximating the moments of nonstationary queues use the functional version of the Kolmogorov forward equations in conjunction with orthogonal polynomial expansions. However, these methods require closed form expressions for the expectations that appear in the functional Kolmogorov forward equations. When closed form expressions cannot be easily derived, these methods cannot be used. In this paper, we present a new sampling algorithm to overcome this difficulty; our sampling algorithm accurately estimates the expectations using simulation. We apply our algorithm to priority queues, which are useful for modeling hospital triage systems. We show that our sampling algorithm accurately estimates the mean and variance of the priority queue without spending significantly more computational time than integrating ordinary differential equations. Last, we compare our sampling algorithm to the closed form analytical approximations for the Erlang-A queueing model and find that our method is comparable in time and accuracy.

Keywords: sampling; simulation; nonstationary queues; forward equations; Markov processes; healthcare; priority queues; triage; closure approximations
History: Accepted by Winfried Grassmann, Area Editor for Computational Probability and Analysis; received November 2014; revised September 2015; accepted February 2016. Published online November 8, 2016.

1. Introduction

In many applied settings, life is nonstationary and stochastic. Thus, it is natural to use nonstationary stochastic models to gain insight about real world stochastic systems. To gain insight on the behavior of nonstationary queueing models, many authors use asymptotic expansion methods such as uniform acceleration and singular perturbations, see for example Massey (1985) or Massey and Whitt (1998). Another method of analysis is the heavy traffic limit theory, which provides a sample path analysis of the queueing process. One approach uses strong approximations such as those in the work of Mandelbaum and Massey (1995) and Mandelbaum et al. (1998). Another approach is to use the traditional continuous mapping approach in under- and overloaded regions like in the work of Liu and Whitt (2014). All of these approaches are important because they allow one to use simple, but well known processes such as Brownian motion with drift to approximate various performance measures of the queueing process. However, one major drawback of these methods is that they are asymptotic in nature; these approaches typically apply to stochastic systems with large rates. Thus, the theory does not necessarily apply to systems with small or moderate rates.

More recently, Massey and Pender (2011, 2013, 2016), Pender (2014b, c), and Engblom and Pender (2014) have used the functional Kolmogorov forward equations to model the dynamics of Markovian queueing processes. Unfortunately, the functional Kolmogorov forward equations are not an autonomous system when the queue length process is not an infinite server queue, or a very simple queueing process. Thus, to numerically integrate the forward equations, one must *close* the forward equations. For instance, if one wants to compute the mean sample path dynamics, if the forward equations are to be closed, then the equation for the time derivative of the mean cannot depend on functions that cannot be written as a function of the mean queue length. To address this problem, authors Massey and Pender (2011) showed that, by using orthogonal polynomial expansions as a surrogate or approximate distribution of the queue length process, they can approximate the dynamics of the queueing process better than the well known heavy traffic limits. One reason is that the functional forward equations are nonasymptotic and do not depend on any asymptotic scaling of the queueing process. Another reason is that the closure approximations for the mean include information from other higher moments; this is important for generating good approximations for the sample path dynamics.

Although the closure approximation methods of Massey and Pender (2011, 2013, 2016) and Pender

(2014b, c) work well for relatively simple Markovian queues with mildly nonlinear rate functions such as polynomials, the closure approximation approach with orthogonal polynomial expansions depends on the ability to derive closed form expressions for the expectations terms that appear on the right-hand side (RHS) of the functional Kolmogorov forward equations. If closed form expressions cannot be obtained by analytically computing the expectations that arise from the forward equations, then the methods of Massey and Pender (2011, 2013, 2016) and Pender (2014b, c) cannot be used since one cannot properly *close* the system of equations. Unfortunately, there are many examples of Markov processes where the expectations of the functional forward equations cannot be computed in closed form.

One example of such a stochastic jump process that is difficult to develop closed form approximations for is the nonstationary priority queue of Mandelbaum et al. (1998). A Markovian multiserver preemptive priority queue with two classes is complex because the second (low priority) class depends on the number of servers not being used by the first class (high priority) customers. In fact, the function that represents the number of servers used by the second class is a composition of maximum functions; this composition makes it extremely difficult to derive closed form expressions for approximating the moments of the priority queue. It is easy to think that choosing a better surrogate or approximate distribution could help in this situation, however, the inability to derive closed form expressions for the expectations that appear in the functional forward equations is difficult regardless of the surrogate distribution used. Thus, to compute approximations for various moments of the priority queue and even for other Markov processes, it is necessary to develop new ways to compute approximations for the expectations that arise from the functional Kolmogorov forward equations.

In this paper, we propose using Monte Carlo simulation to approximate the expectations that arise from the functional forward equations. The first step is to choose a surrogate distribution for the closure approximation. The second step is to sample the expectation terms that arise from the functional forward equations. With these sampled expectations, we then numerically integrate the differential equations associated with the functional forward equations. This finally yields our moment approximations of the original stochastic process. In this paper, we illustrate that by estimating the expectations with the sampled random variables generated from the same distribution of the closure approximation, we can approximate the mean and variance dynamics of the priority queueing process by numerically integrating only $(N^2+3N)/2$ differential equations, where N is the number of priority classes. As a result, we do not have to simulate the actual queueing process using a discrete event simulation. We will show that this produces significant time savings. We also compare our method with the closed form approximations developed in Massey and Pender (2013) and show that, although our sampling method is slower than the analytical approach by at most a factor of 2, it is not nearly as slow when compared to a discrete event simulation of the actual stochastic process.

1.1. Contributions

Our contributions in this work are the following:

- We develop a new and novel sampling algorithm that computes unknown expectations from the functional forward equations when their expectations are hard to analytically compute in closed form.
- We develop an approximation method for priority queues that is more accurate than the fluid and diffusion limit theorems of Mandelbaum et al. (1998).
- Our sampling algorithm extends the applicability of the orthogonal polynomial expansion methods of Massey and Pender (2011, 2013) and Pender (2014b, c) to a larger class of stochastic processes without computing the rate functions in closed form.
- We show that our sampling algorithm can be used to approximate functions of nonstationary stochastic jump processes in a wide variety of application settings such as healthcare, telecommunications, and service operations.

1.2. Organization of the Paper

The rest of the paper continues as follows: In Section 2, we review the Erlang-A model and motivate the sampling approach. In Section 3, we construct our sampling method for the Erlang-A model and prove the pseudocode and actual code for its implementation. In Section 4, we apply the sampling algorithm to a nonstationary priority queue and show that the sampling approach is superior to the fluid and diffusion limits of Mandelbaum et al. (1998). In Section 5, we describe how to implement our algorithm for a multidimensional birth death network. Finally, Section 6 provides concluding remarks.

2. Erlang-A Model: A Simple Motivating Example

In this section, we describe the sampling algorithm for a simple, yet sufficiently complex, queueing process, i.e., the nonstationary Erlang-A model. The nonstationary multiserver queue with abandonment or the Erlang-A model is an important stochastic process for modeling service systems where customers are impatient and leave the system. Unlike the infinite server queue for which the exact distribution of the process is known, the Erlang-A model is not as tractable. It is not a closed dynamical system because it depends on other functions other than the mean. In fact, it can be shown that the time derivative of the mean and variance dynamics of the Erlang-A model satisfy the following differential equations:

$$\frac{d}{dt}E[Q(t)] \equiv \dot{E}[Q] = \lambda - \mu \cdot E[Q \wedge c] - \beta \cdot E[(Q-c)^+] \quad (1)$$

$$\begin{aligned}\frac{d}{dt}\text{Var}[Q(t)] \\ \equiv \dot{\text{Var}}[Q] &= \lambda + \mu \cdot E[Q \wedge c] + \beta \cdot E[(Q-c)^+] \\ &\quad - 2(\mu \cdot \text{Cov}[Q, Q \wedge c] + \beta \cdot \text{Cov}[Q, (Q-c)^+]), \quad (2)\end{aligned}$$

where λ is the exogeneous arrival rate, μ is the service rate of c homogeneous servers, and β is the abandonment rate.

The notation for the Erlang-A queue can be summarized as follows:

- $\lambda(t)$ is the external arrival rate at time t
- $\beta(t)$ is the abandonment rate at time t
- $\mu(t)$ is the service rate at time t
- $c(t)$ is the number of servers at time t
- $x \wedge y = \min(x, y)$
- $(x - y)^+ = \max(0, x - y)$
- $\{x < y\}$ denotes an *indicator function* that equals one if the statement is true, i.e., if $x < y$, and zero if the statement is false
- $\varphi(x)$ is the probability density function (pdf) of the standard normal distribution
- $\Phi(x) = 1 = \bar{\Phi}(x)$ is the cumulative distribution function (cdf) of the standard normal distribution.

Although we write the parameters of the Erlang-A model (λ, μ, β, c) without time dependence, they should be thought of more generally as functions of time and not as constants even though they are allowed to be constant. More generally, the functional Kolmogorov forward equations for the Erlang-A model for an integrable function $f\colon \mathbb{Z}_+ \to \mathbb{R}$ are given by the following differential equation:

$$\begin{aligned}\dot{E}[f(Q)] = \lambda \cdot E[f(Q+1) - f(Q)] \\ &+ \mu \cdot E[(f(Q-1) - f(Q)) \cdot (Q \wedge c)] \\ &+ \beta \cdot E[(f(Q-1) - f(Q)) \cdot (Q-c)^+].\end{aligned}$$

From a computational perspective, the formulas for the time derivatives of the mean and variance summarized above are not *autonomous* differential equations and hence are not a *closed* dynamical system, unless $\mu = \beta$ or $c = \infty$. This means that the differential equations do not only depend on functions of the mean and variance of the queue length processes. In fact, Pender (2014c) show that the expectations that contain max and min terms actually depend on higher moments beyond the mean and variance of the queue length process.

Remark 1. Throughout the remainder of the paper, we refer to the expectations and covariance terms derived from the functional forward equations *rate functions*. Thus, terms such as $E[Q \wedge c]$, $E[(Q-c)^+]$, $\text{Cov}[Q, Q \wedge c]$, and $\text{Cov}[Q, (Q-c)^+]$ will be called *rate functions*. They should not be confused with the time varying functions for the arrival, service or abandonment rates λ, μ, β.

The fact that the differential equations for the moments of many important queueing processes are not autonomous differential equations motivated the work of Rothkopf and Oren (1979), Clark (1981), and Taaffe and Ong (1987). However, to our knowledge, none of these authors have provided proof of why and when the differential equations are not autonomous. Pender (2014c) was the first to give an explicit proof for multiserver queues with or without abandonment, i.e., that they are not autonomous when the number of servers is not infinite or the service rate is not equal to the abandonment rate. Here we provide without proof one of the main contributions of Pender (2014c).

Proposition 1 (Pender 2014c). *Suppose that Q is square integrable or square summable and is a non-negative random variable. Then we have the following relationship between the maximum and minimum functions, the moments, and iterated stationary excess distributions*

$$E[(Q-c)^+] = \frac{1}{2} \cdot E[Q^2] \cdot f_Q^{(2)}(c), \quad (3)$$

$$E[Q \wedge c] = E[Q] - \frac{1}{2} \cdot E[Q^2] \cdot f_Q^{(2)}(c), \quad (4)$$

where the stationary excess distribution is defined as

$$f_Q^{(1)}(y) = \frac{P(Q > y)}{E[Q]}. \quad (5)$$

Its iterates are defined as outlined in Pender (2014c).

Proposition 1 illustrates that the maximum and minimum functions depend on the second moment of the queueing distribution. This implies that to characterize the behavior of the time derivative of the mean of the queue length process, it is also necessary to understand some information about the variance of the queue length process. This carries over when trying to understand the covariance terms of the queue length process. In fact, Pender (2014c) shows that the nth order of the covariance of the queue length process (for example $\text{Cov}[Q^n, (Q-c)^+]$) depends on the $(n+2)$th moment of the queue length process.

As a result, we have demonstrated that it is an important problem and necessary to find new and simple ways of *closing* the functional forward equations without having closed form rate functions. This new method will enable us to obtain accurate approximations for computing moments of a wider class of nonstationary stochastic jump processes.

One such method, which relies on the closed form expressions of rate functions, was developed by Massey and Pender (2011). The main idea of this method was to project the queueing process onto a finite number of Hermite polynomials. It turns out that the first non-deterministic expansion is a Gaussian closure approximation for the queue length process, i.e.,

$$Q(t) \approx q(t) + \sqrt{v(t)} \cdot X, \tag{6}$$

where X is a standard Gaussian random variable. This Gaussian approximation, called the Gaussian variance approximation (GVA) in Massey and Pender (2011), provides the following closed form expressions for the rate functions that arise from the functional forward equations.

Proposition 2 (Massey and Pender 2011). *Under the first order Hermite polynomial expansion for the Erlang-A queueing model, we have the following expressions for the rate functions*:

$$\begin{aligned} E[(Q-c)^+] &= \sqrt{v} \cdot (\varphi(\chi) - \chi \cdot \bar{\Phi}(\chi)) \\ E[Q \wedge c] &= q - \sqrt{v} \cdot (\varphi(\chi) - \chi \cdot \bar{\Phi}(\chi)) \\ \mathrm{Cov}[Q, (Q-c)^+] &= v \cdot \bar{\Phi}(\chi) \\ \mathrm{Cov}[Q, Q \wedge c] &= v \cdot \Phi(\chi), \end{aligned}$$

where

$$\chi = \frac{c-q}{\sqrt{v}}. \tag{7}$$

Fortunately for the Erlang-A model, the rate functions have closed form expressions given in Proposition 2; this enables one to quickly compute the mean and variance cumulants with high accuracy, as shown in Massey and Pender (2011). However, for other functions, especially those that are multidimensional, it is not immediately clear that the rate functions or expectations that arise from the functional forward equation have closed form expressions when integrated with the closure approximation. This is even true for Gausisan closure approximations, which are relatively easy to compute using the Hermite polynomial calculus developed in Massey and Pender (2013). One such example of an expectation that is difficult to compute in closed form is the spread option from the mathematical finance literature or the number of abandoning customers from a priority queue with low priority. Thus, it is necessary to develop a new method that can approximate these expectations or rate functions without much computational effort. Although we have closed form expressions for the rate functions under the GVA, we will demonstrate our new sampling algorithm for the Erlang-A model so that the reader can understand the algortithm for a simple example. A more complex example of a multidimensional priority queue will be provided later in the paper.

3. The Sampling Algorithm

In this section, we give a description of the sampling algorithm used to compute the expectations and covariance terms for which we do not have closed form formulas. The main idea of our sampling approach is as follows. Because we cannot always explicitly compute the expectations and covariance terms from the functional forward equations, we propose that we should sample and estimate them instead with our closure approximation distribution. For the Gaussian closure approximation and the rate functions of the Erlang-A model, one can compute estimates of the rate functions by sampling from the closure approximation distribution and averaging the samples. Thus, for the Erlang-A model, if we fix the value of t, we can compute an estimate of the exact expectation by the following averaging procedure:

$$E[(Q(t)-c(t))^+] \approx \frac{1}{m} \sum_{i=1}^{m} (Y_i - c(t))^+ \tag{8}$$

and

$$E[Q(t) \wedge c(t)] \approx \frac{1}{m} \sum_{i=1}^{m} (Y_i \wedge c(t)), \tag{9}$$

where (Y_i) are i.i.d. Gaussian random variables with mean $q(t)$ and variance $v(t)$ and m is the number of samples used to estimate the expectation. For covariance terms we use the fact that they can be written in terms of expectations; therefore we have that

$$\begin{aligned} &\mathrm{Cov}[Q(t), (Q(t) \wedge c(t))] \\ &\approx \frac{1}{m} \sum_{i=1}^{m} Y_i \cdot (Y_i \wedge c(t)) - \left(\frac{1}{m} \sum_{i=1}^{m} Y_i \right) \cdot \left(\frac{1}{m} \sum_{i=1}^{m} (Y_i \wedge c(t)) \right) \end{aligned} \tag{10}$$

and

$$\begin{aligned} \mathrm{Cov}[Q(t), (Q(t)-c(t))^+] &\approx \frac{1}{m} \sum_{i=1}^{m} Y_i \cdot (Y_i - c(t))^+ \\ &\quad - \left(\frac{1}{m} \sum_{i=1}^{m} Y_i \right) \cdot \left(\frac{1}{m} \sum_{i=1}^{m} (Y_i - c(t))^+ \right). \end{aligned} \tag{11}$$

Thus, using the sampling algorithm with m samples and a Gaussian closure distribution for the Erlang-A model yields the following differential equations for estimating the mean and variance of the queue length process:

$$\dot{E}[Q] \approx \lambda - \mu \cdot \frac{1}{m} \sum_{i=1}^{m} (Y_i \wedge c(t)) - \beta \cdot \frac{1}{m} \sum_{i=1}^{m} (Y_i - c(t))^+ \tag{12}$$

$$\begin{aligned}\dot{\mathrm{V}}\mathrm{ar}[Q] \approx{}& \lambda + \mu\cdot\frac{1}{m}\sum_{i=1}^{m}(Y_i\wedge c(t)) + \beta\cdot\frac{1}{m}\sum_{i=1}^{m}(Y_i - c(t))^+ \\ & -2\cdot\mu\cdot\left(\frac{1}{m}\sum_{i=1}^{m}Y_i\cdot(Y_i\wedge c(t)) - \left(\frac{1}{m}\sum_{i=1}^{m}Y_i\right)\right. \\ & \left.\cdot\left(\frac{1}{m}\sum_{i=1}^{m}(Y_i\wedge c(t))\right)\right) \\ & -2\cdot\beta\cdot\left(\frac{1}{m}\sum_{i=1}^{m}Y_i\cdot(Y_i - c(t))^+ - \left(\frac{1}{m}\sum_{i=1}^{m}Y_i\right)\right. \\ & \left.\cdot\left(\frac{1}{m}\sum_{i=1}^{m}(Y_i - c(t))^+\right)\right). \end{aligned} \tag{13}$$

3.1. Pseudocode and MATLAB Code for Sampling Algorithm

The following is pseudocode for the sampling algorithm that we use to compute the mean and covariance matrix of an arbitrary Markovian stochastic network with functional forward equations.

Algorithm 1 (Functional Kolmogorov forward equation sampling algorithm)

```
 1: procedure FKFE($\vec{\mu}, \vec{\Sigma}$)
                ▷ Computes mean and variance of queue
 2:   $n \leftarrow$   ▷ number of samples
 3:   $d \leftarrow$   ▷ dimension of queueing network
 4:   $\Delta t \leftarrow$   ▷ time spacing for Euler scheme
 5:   $F \leftarrow$   ▷ final time
 6:   $T \leftarrow F/\Delta t$   ▷ number of time points
 7:   for $i = 1 \rightarrow T$ do   ▷ Loop for each time point
 8:       for $j = 1 \rightarrow d$ do   ▷ Loop for each queue
 9:           $samp$ ← random samples generated for each queue
                ▷ In the next two steps we take the sampled values and compute the expectations and variance terms in functional forward equations.
10:           $\mu(i,j) \leftarrow \mu(i-1,j) + \Delta t \cdot E[\text{rate functions(samp)}]$
11:           $\Sigma(i,j) \leftarrow \Sigma(i-1,j) + \Delta t \cdot \text{Var}[\text{rate functions(samp)}]$
12:       end for
13:   end for
14:   return $\mu, \Sigma$   ▷ The mean is $\vec{\mu}$ and the variance is $\vec{\Sigma}$
15: end procedure
```

In addition to providing the pseudocode for the sampling algorithm, we summarize the steps of implementing our sampling algorithm in English to give the reader a clear understanding of the sampling algorithm methodology.

- Choose a Markovian stochastic model to approximate.
- Derive the functional Kolmogorov forward equations for the chosen stochastic model.
- Sample the expectations from the functional forward equation from a closure approximation or surrogate distribution, examples include (Gaussian, Poisson, Gamma, etc.).
- Compute the moments of the stochastic model by numerically integrating the functional forward equations using the sampled expectations.

For the convenience of the reader, we also provide the MATLAB code used to construct the sampling algorithm for the Erlang-A model. It is clear from the small number of lines that it is easy to implement for this model.

```
 1 % Initializing mean and variance differential equations
 2 q = q0*ones(1,numsteps); % Mean queue length
 3 v = v0*ones(1,numsteps); % Variance of queue length
 4 % Starting Euler scheme for numerically integrating differential equations.
 5 for i = 1:numsteps-1
 6 %Computes Gaussian Random Variables for Sampling Rate Functions
 7 Q = q(i)*ones(1,numsamp) + sqrt(v(i))*randn(1,numsamp);
 8 %Computes Sample Mean and Variance of Queue Length
 9 m01 = mean(Q);
10 v01 = var(Q);
11 %Here we start the sampling of the rate functions
12 m1 = mean( min(Q, c(i)*ones(1,numsamp)) ); % E[ (Q wedge c) ]
13 m2 = m01 - m1; % E[ (Q - c)+ ]
14 m3 = mean(Q.*min(Q,c(i)*ones(1,numsamp))) - m01*m1; % Cov[ Q , Q wedge c ]
15 m4 = v01 - m3; % Cov[ Q , (Q - c)+ ]
16 % Using sampled rate functions to compute the next iteration of scheme
17 q(i + 1) = q(i) + dt*(lambda(i) - mu(i)*m1 -beta(i)*m2 );
18 v(i + 1) = v(i) + dt*(lambda(i) + mu(i)*m1+beta(i)*m2 - 2*(mu(i)*m3 + beta(i)*m4));
19 end
```

To relate the pseudocode with the actual MATLAB code, we have provided numerous comments in the MATLAB to convey what we are doing at each step. Our goal is to convey to the reader a better understanding of how to develop their own code after choosing a stochastic model to approximate. A deeper look at the pseudocode and MATLAB code shows that the main sampling part for the rate functions is given in steps 8–9 of the pseudocode. These steps in the pseudocode are implemented in lines 11–15 of the MATLAB code. Moreover, the Euler integration of the differential equations using the sampled rate functions is given in lines 16–18 of the MATLAB code and match steps 10 and 11 in the pseudocode.

3.2. How Many Random Variables to Use?

In sampling the rate functions, it is reasonable to ask how many samples are needed to estimate them with good accuracy. The next proposition gives us insight on how many samples one would need to approximate the maximum function, however, the idea also extends to other functions that are common in the queueing or stochastic processes literature by a similar variance bounding argument.

Proposition 3. *Suppose that S_i is a Gaussian random variable with mean q and variance v, then for the maximum function we have that*

$$E\left[\left|\frac{1}{n}\sum_{i=1}^{n}(S_i-c)^+ - E[(Q-c)^+]\right|\right] \le \frac{\tilde{\sigma}_1}{\sqrt{n}}, \tag{14}$$

where $\chi=(c-q)/\sqrt{v}$ and

$$\tilde{\sigma}_1 = \mathrm{Var}[(X-\chi)^+] \tag{15}$$

$$= -\chi\cdot\varphi(\chi)+(\chi^2+1)\cdot\bar{\Phi}(\chi)-\varphi(\chi)^2 - \chi^2\cdot\bar{\Phi}(\chi)^2+2\cdot\chi\cdot\varphi(\chi)\cdot\bar{\Phi}(\chi). \tag{16}$$

Proof. *We begin by showing that*

$$E\left[\left|\frac{1}{n}\sum_{i=1}^{n}(S_i-c)^+ - E[(Q-c)^+]\right|\right] \le \sqrt{E\left[\left(\frac{1}{n}\sum_{i=1}^{n}(S_i-c)^+ - E[(Q-c)^+]\right)^2\right]} = \sqrt{\mathrm{Var}\left[\frac{1}{n}\sum_{i=1}^{n}(S_i-c)^+\right]} = \frac{1}{\sqrt{n}}\cdot\sqrt{\mathrm{Var}[(S_i-c)^+]}.$$

Now it remains to show that the following identity holds:

$$\mathrm{Var}[(X-\chi)^+] = -\chi\cdot\varphi(\chi)+(\chi^2+1)\cdot\bar{\Phi}(\chi)-\varphi(\chi)^2 - \chi^2\cdot\bar{\Phi}(\chi)^2+2\cdot\chi\cdot\varphi(\chi)\cdot\bar{\Phi}(\chi),$$

where X is a standard Gaussian random variable.

Because the variance can be split into the second moment minus the first moment squared, it only remains to compute the first and second moment of the function $(X-\chi)^+$, which can be shown using Stein's lemma of Stein (1986).

$$E[((X-\chi)^+)]^2 = E[(X-\chi)\cdot\{X\ge\chi\}]^2 = (\varphi(\chi)-\chi\cdot\bar{\Phi}(\chi))^2$$

$$\begin{aligned} E[((X-\chi)^+)^2] &= E[(X-\chi)^2\cdot\{X\ge\chi\}] \\ &= E[(X^2-2\cdot\chi\cdot X+\chi^2)\cdot\{X\ge\chi\}] \\ &= E[(X^2-1)\cdot\{X\ge\chi\}]-2\cdot\chi \\ &\quad \cdot E[X\cdot\{X\ge\chi\}]+(\chi^2+1)\cdot E[\{X\ge\chi\}] \\ &= \chi\cdot\varphi(\chi)-2\cdot\chi\cdot\varphi(\chi)+(\chi^2+1)\cdot\bar{\Phi}(\chi) \end{aligned}$$

Combining these two expressions yields our result. □

This proposition implies that if we want our sampled expectation to differ by at most ϵ, then we need at most σ/ϵ^2 samples to achieve our desired accuracy. The number of samples needed for other rate function is also on the order of σ/ϵ^2, however, the value of σ changes for each different function.

Proposition 4. *Let S_i be a Gaussian random variables with mean q and variance v. For the terms that arise in the computation of the variance of the queue length we have that*

$$E\left[\left|\frac{1}{n}\sum_{i=1}^{n}S_i\cdot(S_i-c)^+ - E[Q\cdot(Q-c)^+]\right|\right] \le \frac{\tilde{\sigma}_2}{\sqrt{n}}, \tag{17}$$

where

$$\tilde{\sigma}_2 = q\cdot\sqrt{v}\cdot\mathrm{Var}[(X-\chi)^+]+v\cdot\mathrm{Var}[X\cdot(X-\chi)^+] \tag{18}$$

$$= v\cdot((\chi^2+3)\cdot\bar{\Phi}(\chi)-\chi\cdot\varphi(\chi)+\bar{\Phi}(\chi)^2) \tag{19}$$

$$+q\cdot\sqrt{v}\cdot((\chi^2+1)\cdot\bar{\Phi}(\chi)-\varphi(\chi)^2 - \chi^2\cdot\bar{\Phi}(\chi)^2+\chi\cdot\varphi(\chi)\cdot\bar{\Phi}(\chi)). \tag{20}$$

Proof. We begin by showing that

$$E\left[\left|\frac{1}{n}\sum_{i=1}^{n}S_i\cdot(S_i-c)^+ - E[Q\cdot(Q-c)^+]\right|\right] \le \sqrt{E\left[\left(\frac{1}{n}\sum_{i=1}^{n}S_i\cdot(S_i-c)^+ - E[Q\cdot(Q-c)^+]\right)^2\right]} = \sqrt{\mathrm{Var}\left[\frac{1}{n}\sum_{i=1}^{n}S_i\cdot(S_i-c)^+\right]} = \frac{1}{\sqrt{n}}\cdot\sqrt{\mathrm{Var}[S_i\cdot(S_i-c)^+]}.$$

Moreover, we know that

$$\mathrm{Var}[S_i\cdot(S_i-c)^+] = q\cdot\sqrt{v}\cdot\mathrm{Var}[(X-\chi)^+] + v\cdot\mathrm{Var}[X\cdot(X-\chi)^+],$$

where X is a standard Gaussian random variable.

Because the first term has been computed in Proposition 3, it now remains to show that the following identity holds:

$$\mathrm{Var}[X\cdot(X-\chi)^+] = \bar{\Phi}(\chi)^2+(\chi^2+3)\cdot\bar{\Phi}(\chi)-\chi\cdot\varphi(\chi),$$

where X is a standard Gaussian random variable.

Because the variance can be split into the second moment minus the first moment squared, it only remains to compute the first and second moment of the function $X\cdot(X-\chi)^+$, which can also be derived using Stein's (1986) lemma. The first moment squared

Table 1 **GVA Exact vs. FKFE Sampling for Erlang-A Model**

Samples (m)	$\lambda(t)$	$c(t)$	$\beta(t)$	$\mu(t)$	Mean – Error $\cdot 10^{-3}$	Var – Error $\cdot 10^{-2}$	Time (seconds)
Simulation	$10+2\cdot\sin(t)$	10	0.5	1	0.8745	0.77	173.712
GVA – Exact	$10+2\cdot\sin(t)$	10	0.5	1	0	0	2.3938
10	$10+2\cdot\sin(t)$	10	0.5	1	1.900 ± 0.14	2.96 ± 0.08	4.7330
40	$10+2\cdot\sin(t)$	10	0.5	1	0.873 ± 0.053	0.78 ± 0.035	4.7432
100	$10+2\cdot\sin(t)$	10	0.5	1	0.526 ± 0.027	0.39 ± 0.019	4.8295
400	$10+2\cdot\sin(t)$	10	0.5	1	0.258 ± 0.014	0.16 ± 0.0065	5.1568
Simulation	$40+8\cdot\sin(t)$	40	0.5	1	0.65	0.71	417.025
GVA – Exact	$40+8\cdot\sin(t)$	40	0.5	1	0	0	2.3557
10	$40+8\cdot\sin(t)$	40	0.5	1	0.994 ± 0.061	0.82 ± 0.07	4.6994
40	$40+8\cdot\sin(t)$	40	0.5	1	0.455 ± 0.022	0.34 ± 0.04	4.7456
100	$40+8\cdot\sin(t)$	40	0.5	1	0.272 ± 0.012	0.25 ± 0.02	4.8237
400	$40+8\cdot\sin(t)$	40	0.5	1	0.136 ± 0.007	0.12 ± 0.006	5.1010
Simulation	$100+20\cdot\sin(t)$	100	0.5	1	0.47	0.57	906.126
GVA – Exact	$100+20\cdot\sin(t)$	100	0.5	1	0	0	2.3452
10	$100+20\cdot\sin(t)$	100	0.5	1	0.584 ± 0.034	0.78 ± 0.06	4.6859
40	$100+20\cdot\sin(t)$	100	0.5	1	0.274 ± 0.018	0.35 ± 0.03	4.7852
100	$100+20\cdot\sin(t)$	100	0.5	1	0.167 ± 0.007	0.23 ± 0.015	4.8257
400	$100+20\cdot\sin(t)$	100	0.5	1	0.085 ± 0.005	0.11 ± 0.005	5.0837

has the following expression in terms of the Gaussian tail cdf:

$$\begin{aligned}E[(X\cdot(X-\chi)^+)]^2 &= E[(X^2-\chi\cdot X)\cdot\{X\geq\chi\}]^2\\ &= E[(X^2-1-\chi\cdot X+1)\cdot\{X\geq\chi\}]^2\\ &= (\chi\cdot\varphi(\chi)-\chi\cdot\varphi(\chi)+\bar{\Phi}(\chi))^2\\ &= \bar{\Phi}(\chi)^2.\end{aligned}$$

Last, the second moment has the following expression:

$$\begin{aligned}&E[(X\cdot(X-\chi)^+)^2]\\ &\quad= E[X^2\cdot(X-\chi)^2\cdot\{X\geq\chi\}]\\ &\quad= E[(X^4-2\cdot\chi\cdot X^3+\chi^2\cdot X^2)\cdot\{X\geq\chi\}]\\ &\quad= E[((X^4-6\cdot X^2+3)-2\cdot\chi\cdot(X^3-3\cdot X)\\ &\qquad\quad+(\chi^2+6)\cdot X^2-6\cdot\chi\cdot X-3)\cdot\{X\geq\chi\}]\\ &\quad= (\chi^3-3\cdot\chi)\cdot\varphi(\chi)-2\cdot(\chi^3-\chi)\cdot\varphi(\chi)+(\chi^3+6\cdot\chi)\\ &\qquad\cdot\varphi(\chi)-6\cdot\chi\cdot\varphi(\chi)+(\chi^2+6-3)\cdot\bar{\Phi}(\chi)\\ &\quad= (\chi^2+3)\cdot\bar{\Phi}(\chi)-\chi\cdot\varphi(\chi).\end{aligned}$$

Once again combining these two expressions yields our desired result. □

In Table 1 we provide the relative errors made between the sampling method and the exact GVA equations. We see that the sampling method is only a factor of 2 slower than the analytical expressions for the Erlang-A model, so we do not lose much computational time when using the sampling method. Moreover, we see that the error is quite small and decreases as we increase the number of samples to compute the expectations present in the rate functions at each iteration. It is also apparent from our confidence intervals that we are reasonably confident about our sampling method even when the number of samples is small. Furthermore, we see in Table 1 that by simulating the exact stochastic queueing model, we need 72 times more computational effort. This only increases as we add more dimensions to the problem and increase the rates of the stochastic process, i.e., η is large. However, we see that our sampling method does not need as much computational effort to compute the mean and variance of the Erlang-A model.

Remark 2. Although we present the pseudocode for the mean and variance of a stochastic network, it can be extended to other moments as well. We should also mention that more terms may sometimes be required in the polynomial expansion to accurately describe the dynamics of higher moments. This can be seen in Massey and Pender (2013) where they use an additional polynomial to approximate the skewness. Next, we apply this pseudocode to the priority queue with abandonment and upgrades and comment on its performance in estimating the mean and variance of the queueing process.

4. Application to a Priority Queue Model

In this section, we introduce the priority queueing model that we will use to demonstrate the effectiveness our new sampling method. An important reason we consider a priority queueing model is that, unlike the Erlang-A model, the priority queueing model we present does not have closed form expressions for all of its rate functions. Thus, we will show that our sampling method can overcome this difficulty and still approximate the mean and variance with good accuracy and substantially less computational effort than a discrete event simulation of the queueing process.

4.1. Motivating Applications

In considering a priority queue model, we are motivated by two important applications. The first is the application to patient flow dynamics in healthcare systems. In a typical hospital, patients are triaged into different customer classes based on the severity of their health condition, see for example García et al. (1995) and Siddharthan et al. (1996). Patients needing urgent care have priority over patients whose condition is observed to be more stable and does not depend too heavily on receiving immediate care. Because the triage procedure is not perfect and patients might be triaged into the wrong class we allow the low priority class patients to be upgraded to the high priority class if needed. This also accounts for patients whose condition deteriorates while waiting to be seen and therefore need immediate care.

Another motivating application setting is that of call centers with multiple classes of customers with different priority or service levels. Customers who are deemed *important* receive service over customers who have a lower level of priority. This is quite common in call centers for bank services where customers with more money in the bank or special accounts might have priority over customers with regular accounts and less money. As customers wait for service, the low priority customers can transition to the higher priority class or be *upgraded* if they wait too long while in the low priority class without receiving service from an agent. This upgrading procedure facilitates some type of fairness; customers in the lower class receive upgraded service and eventually talk to an agent quicker.

4.2. The Nonstationary Priority Queue with Abandonment

Priority queues are well studied in the queueing literature. However, many priority queueing models assume that the arrival and service rates are constant functions of time, see for example Green (2006) and references therein. In this work, we study the priority queue with nonstationary arrival rates and with the additional features of customer abandonment and upgrades to a higher priority class. A queueing model with upgrades in the single server context was studied in Down and Lewis (2010) and nonstationary dynamics were studied in Mandelbaum et al. (1998). In this paper, we apply our sampling algorithm to a priority queue with two customer classes. We assume that the priority queue is preemptive. Although we consider a two-class model here, our sampling algorithm can handle any number of classes; the two dimensional example is only used to effectively illustrate the main idea for a low dimensional model. To construct the priority queueing model, we use time changed Poisson processes such as those in the paper by Mandelbaum et al. (1998). Thus, our priority queue with abandonment and upgrades $\{Q_1(t), Q_2(t) \mid t \geq 0\}$ can be represented by the following stochastic time changed integral equations:

$$\begin{aligned} Q_1(t) &= Q_1(0) + \Pi_1\left(\int_0^t \lambda_1(s)\,ds\right) - \Pi_2\left(\int_0^t \mu_1 \cdot (Q_1(s) \wedge c(s))\,ds\right) \\ &\quad - \Pi_3\left(\int_0^t \beta_1 \cdot (Q_1(s) - c(s))^+\,ds\right) \\ &\quad + \Pi_7\left(\int_0^t p \cdot \beta_2 \cdot (Q_2(s) - (c(s) - Q_1(s))^+)^+\,ds\right) \end{aligned} \tag{21}$$

$$\begin{aligned} Q_2(t) = Q_2(0) &+ \Pi_4\left(\int_0^t \lambda_2(s)\,ds\right) \\ &- \Pi_5\left(\int_0^t \mu_2 \cdot (Q_2(s) \wedge (c(s) - Q_1(s))^+)\,ds\right) \\ &- \Pi_6\left(\int_0^t \beta_2 \cdot (Q_2(s) - (c(s) - Q_1(s))^+)^+\,ds\right), \end{aligned} \tag{22}$$

where $\Pi_i \equiv \{\Pi_i(t) \mid t \geq 0\}$ for $i = 1, 2, 3, 4, 5, 6$ are i.i.d. standard (rate 1) Poisson processes. The deterministic time change for Π_1 transforms it into a nonhomogenous Poisson arrival process with rate $\lambda_1(t)$. Subjecting Π_2 to a random time change causes it to count the number of service departures from c servers and exponentially distributed service times function with mean $1/\mu_1$. A random time change of Π_3 counts the number of abandonments from c identical and homogeneous servers in the first queue and exponentially distributed abandonment times of mean $1/\beta_1$. A deterministic time change of Π_4 counts the number of arrivals to the second queue. A random time change of Π_5 counts the number of service departures from the second queue and $(c - Q_1)^+$ available servers and exponentially distributed service times with mean $1/\mu_2$. A random time change of Π_6 counts the number of abandonments from the second queue to get upgraded or to leave the system altogether with exponentially distributed abandonment times with mean $1/\beta_2$. Last, a random time change of Π_7 counts the number of customers who are upgraded from the second queue to the first queue where p is the fraction of the customers who do not leave the system.

The notation for the two-class priority queue can be summarized as follows:

- $\lambda_i(t)$ is the external arrival rate of class i at time t
- $\beta_i(t)$ is the abandonment rate of class i at time t
- $\mu_i(t)$ is the service rate of class i at time t
- p is the probability that a customer or patient from the low priority class, who has waited sufficiently, is upgraded to the high priority class
- $c(t)$ is the number of servers available for service at time t.

It can be shown using the theory of Mandelbaum et al. (1998) that our priority queueing model with abandonment and upgrades fits into a class of stochastic processes called *Markovian service networks*. With this knowledge, we can construct an associated, scaled or *uniformly accelerated* queueing process where the new arrival rate function is $\eta \cdot \lambda$ and the new number of servers is $\eta \cdot c$ for some positive scale factor $\eta > 0$. A healthcare interpretation of the asymptotic scaling would be to simultaneously scale up the patient demand (arrival rate) and the patient supply (beds or nurses). This is natural in large hospitals with a large number of patients and nurses or beds. Likewise, a call center interpretation would be to simultaneously scale up the customer demand (arrival rate) and the number of call center agents. Taking the following pointwise limits gives us the *fluid* and *diffusion* models of Mandelbaum et al. (1998), i.e.,

$$\lim_{\eta\to\infty} \frac{1}{\eta} Q^{\eta} = q \text{ a.s. and } \lim_{\eta\to\infty} \sqrt{\eta}\cdot\left(\frac{1}{\eta}Q^{\eta} - q\right) \overset{d}{=} \hat{Q}, \tag{23}$$

where the deterministic process q, the *fluid mean*, is governed by the following two dimensional dynamical system:

$$\begin{aligned} \overset{\bullet}{q}_1 &= \lambda_1 - \mu_1 \cdot (q_1 \wedge c) - \beta_1 \cdot (q_1 - c)^+ \\ &\quad + p \cdot \beta_2 \cdot (q_2 - (c - q_1)^+)^+ \\ \overset{\bullet}{q}_2 &= \lambda_2 - \mu_2 \cdot (q_2 \wedge (c - q_1)^+) \\ &\quad - \beta_2 \cdot (q_2 - (c - q_1)^+)^+. \end{aligned} \tag{24}$$

Moreover, as pointed out in Mandelbaum et al. (1998), if the set of time points $\mathscr{A}$

$$\mathscr{A} = \{t \mid q_1(t) = c(t)\} \cup \{t \mid q_2(t) = (c(t) - q_1(t))^+\} \tag{25}$$

has measure zero, then $\hat{Q}$ is a Gaussian diffusion process whose variance combines with the fluid mean to form a five-dimensional dynamical system given by Equation (24) and

$$\begin{aligned} \overset{\bullet}{\mathrm{Var}}[\hat{Q}_1] &= \dot{v}_1 \\ &= \lambda_1 + \mu_1 \cdot (q_1 \wedge c) + \beta_1 \cdot (q_1 - c)^+ \\ &\quad + p \cdot \beta_2 \cdot (q_2 - (c - q_1)^+)^+ - 2 \cdot v_1 \cdot \mu_1 \cdot \zeta_{11} \\ &\quad - 2 \cdot v_1 \cdot \beta_1 \cdot \zeta_{31} + 2 \cdot p \cdot \beta_2 \cdot (v_1 \cdot \zeta_{41} + k \cdot \zeta_{42}) \end{aligned}$$

$$\begin{aligned} \overset{\bullet}{\mathrm{Var}}[\hat{Q}_2] &= \dot{v}_2 \\ &= \lambda_2 + \mu_2 \cdot (q_2 \wedge (c - q_1)^+) + (1-p) \cdot \beta_2 \\ &\quad \cdot (q_2 - (c - q_1)^+)^+ - 2 \cdot \mu_2 \cdot (k \cdot \zeta_{21} + v_2 \cdot \zeta_{22}) \\ &\quad - 2 \cdot (1-p) \cdot \beta_2 \cdot (k \cdot \zeta_{41} + v_2 \cdot \zeta_{42}) \\ &\quad - 2 \cdot p \cdot \beta_2 \cdot (v_2 \cdot \zeta_{42} + k \cdot \zeta_{41}) \end{aligned}$$

$$\begin{aligned} \overset{\bullet}{\mathrm{Cov}}[\hat{Q}_1, \hat{Q}_2] &= \dot{k} \\ &= -p \cdot \beta_2 \cdot (q_2 - (c - q_1)^+)^+ - \mu_1 \cdot k \cdot \zeta_{11} \\ &\quad - \mu_2 \cdot (v_1 \cdot \zeta_{21} + v_2 \cdot \zeta_{22}) \\ &\quad - \beta_1 \cdot k \cdot \zeta_{31} - (1-p) \cdot \beta_2 \cdot (v_1 \cdot \zeta_{41} + v_2 \cdot \zeta_{42}) \\ &\quad - p \cdot \beta_2 \cdot (v_1 \cdot \zeta_{41} + k \cdot \zeta_{42}) \\ &\quad + p \cdot \beta_2 \cdot (v_2 \cdot \zeta_{42} + k \cdot \zeta_{41}), \end{aligned}$$

where we have that $v_1 = \mathrm{Var}[Q_1], v_2 = \mathrm{Var}[Q_2], k = \mathrm{Cov}[Q_1, Q_2]$,

$$\alpha_1(q_1, q_2) = (q_1 \wedge c) \tag{26}$$

$$\alpha_2(q_1, q_2) = (q_2 \wedge (c - q_1)^+) \tag{27}$$

$$\alpha_3(q_1, q_2) = (q_1 - c)^+ \tag{28}$$

$$\alpha_4(q_1, q_2) = (q_2 - (c - q_1)^+)^+, \tag{29}$$

and where we define the following expressions for the derivatives of the previous functions:

$$\zeta_{ij} \equiv \frac{\partial}{\partial q_j} \alpha_i(q_1, q_2). \tag{30}$$

Remark 3. Note that the variance and covariance differential equations have indicator functions, which cause the equation to depend discontinuously on the mean queue length. These discontinuities occur exactly where we assume the set of time points $\mathscr{A}$ has measure zero. The measure zero assumption is known as the *lingering condition* and it is well known that the fluid and diffusion limits are not as accurate when mean queue length processes linger around the set $\mathscr{A}$. In fact, the diffusion limits are also no longer Gaussian at those points.

4.3. Functional Forward Equations for Priority Queue

Using the functional Kolmogorov forward equations for multidimensional birth-death processes, we derive the following functional forward equations for the mean and variance of the priority queue with abandonment and upgrades. We provide explicit formulas for the two dimensional case, but all of our methods can also be applied to any finite number of dimensions. Thus, the mean, variance, and covariance of the priority queueing process satisfy the following differential equations:

$$\begin{aligned} \overset{\bullet}{E}[Q_1] &= \lambda_1(t) - \mu_1 \cdot E[(Q_1(t) \wedge c(t))] \\ &\quad - \beta_1 \cdot E[(Q_1(t) - c(t))^+] \\ &\quad + p \cdot \beta_2 \cdot E[(Q_2(t) - (c(t) - Q_1(t))^+)^+] \\ \overset{\bullet}{E}[Q_2] &= \lambda_2(t) - \mu_2 \cdot E[(Q_2(t) \wedge (c(t) - Q_1(t))^+)] \\ &\quad - \beta_2 \cdot E[(Q_2(t) - (c(t) - Q_1(t))^+)^+] \end{aligned}$$

$$\begin{aligned}
\overset{\bullet}{\mathrm{Var}}[Q_1] &= \lambda_1(t)+\mu_1\cdot E[(Q_1(t)\wedge c(t))]\\
&\quad +\beta_1\cdot E[(Q_1(t)-c(t))^+]\\
&\quad +p\cdot\beta_2\cdot E[(Q_2(t)-(c(t)-Q_1(t))^+)^+]\\
&\quad -2\cdot\mu_1\cdot\mathrm{Cov}[Q_1(t),(Q_1(t)\wedge c(t))]\\
&\quad -2\cdot\beta_1\cdot\mathrm{Cov}[Q_1(t),(Q_1(t)-c(t))^+]+2\cdot p\\
&\quad \cdot\beta_2\cdot\mathrm{Cov}[Q_1(t),(Q_2(t)-(c(t)-Q_1(t))^+)^+]\\
\overset{\bullet}{\mathrm{Var}}[Q_2] &= \lambda_2(t)+\mu_2\cdot E[(Q_2(t)\wedge(c(t)-Q_1(t))^+)]\\
&\quad +\beta_2\cdot E[(Q_2(t)-(c(t)-Q_1(t))^+)^+]-2\cdot\mu_2\\
&\quad \cdot\mathrm{Cov}[Q_2(t),(Q_2(t)\wedge(c(t)-Q_1(t))^+)]-2\\
&\quad \cdot\beta_2\cdot\mathrm{Cov}[Q_2(t),(Q_2(t)-(c(t)-Q_1(t))^+)^+]\\
\overset{\bullet}{\mathrm{Cov}}[Q_1,Q_2] &= -\mu_1\cdot\mathrm{Cov}[Q_2(t),(Q_1(t)\wedge c(t))]-\mu_2\\
&\quad \cdot\mathrm{Cov}[Q_1(t),(Q_2(t)\wedge(c(t)-Q_1(t))^+)]\\
&\quad -\beta_1\cdot\mathrm{Cov}[Q_2(t),(Q_1(t)-c(t))^+]-\beta_2\\
&\quad \cdot\mathrm{Cov}[Q_1(t),(Q_2(t)-(c(t)-Q_1(t))^+)^+]\\
&\quad +p\cdot\beta_2\\
&\quad \cdot\mathrm{Cov}[Q_2(t),(Q_2(t)-(c(t)-Q_1(t))^+)^+]\\
&\quad -p\cdot\beta_2\cdot E[(Q_2(t)-(c(t)-Q_1(t))^+)^+].
\end{aligned}$$

Following the approach of Massey and Pender (2011), a logical next step is to use a two dimensional version of the GVA to close the system of equations. A Gaussian distribution is natural in a priority queue setting since the fluid and diffusion limits for this process are also Gaussian under the mild conditions given in Mandelbaum et al. (1998). However, a complication arises when we attempt to close the forward equations with the GVA. We cannot compute, in closed form, some of the expectations or the covariance terms in the functional forward equations with respect to the Gaussian distribution. For example, the expectation that represents the number of customers who receive upgraded service, i.e.,

$$E[(Q_2(t)-(c(t)-Q_1(t))^+)^+] \tag{31}$$

is a *composition* of maximum functions. Thus, it is extremely difficult to compute in closed form an analytical expression for the expectation of Equation (31). This issue is not unique to the Gaussian distribution. The Laguerre approach in Pender (2014a), the Poisson-Charlier approach of Engblom and Pender (2014), and even the Poisson approach of Pender (2014c) all have the same difficulties. They are all difficult since it is extremely hard, if not intractable, to compute these types of expectations and covariance terms with analytical or closed form expressions. Moreover, the expectation in Equation (31) is not the only expectation that cannot be calculated in closed form. In the two-dimensional setting, it can be shown that four terms cannot be computed in closed form with respect to the Gaussian distribution. Thus, it is imperative to find ways of computing or approximating these expectations without losing the computational advantage of simply integrating the forward equations. Thus, in the sequel, we develop a new sampling method to circumvent the fact that the analytical expressions cannot be computed in closed form for the functional forward equations.

Although we present the pseudocode for the mean and variance of a stochastic network, it can be extended to other moments. Note also that sometimes more terms may be required in the polynomial expansion to accurately describe the dynamics of higher moments. This can be seen in Massey and Pender (2013) where they use an additional Hermite polynomial to approximate the skewedness. Next, we apply this pseudocode to the priority queue with abandonment and upgrades and comment on its performance in estimating the mean and variance of the queueing process.

4.4. Priority Queue Numerics

In this section, we apply our sampling algorithm to the priority queue with abandonment and upgrades. As mentioned, it is an important model for healthcare triage systems and where customers need to receive differentiated service. In the following examples, we compare our sampling algorithm to the fluid and diffusion limit theorems, which are the state of the art for priority queue approximations for the mean and variance.

In Figure 1, we plot the mean of the two dimensional priority queue using discrete event simulation with the fluid limit and the mean approximation using our algorithm with a Gaussian surrogate distribution. To compute the mean and variance of the discrete event simulation of the actual queueing process, we averaged over 10,000 simulations. With 10,000 individual realizations we are nearly confident that our sample of the mean and variance is within 1% of the true mean and variance. Moreover, when implementing our algorithm, we used 100 independent random variables to construct a good estimate of the expectations. We see that for the first and second queue, the sampled version of GVA or our algorithm using a Gaussian surrogate distribution is approximating the mean of the discrete event simulation much better than the fluid limit given by the limit theorems. Improvement in the approximation of the mean ranges from 2%–4%. One reason that we see an improvement over the fluid limit is that the fluid limit is the best deterministic approximation of the queue length and does not incorporate any variation of the

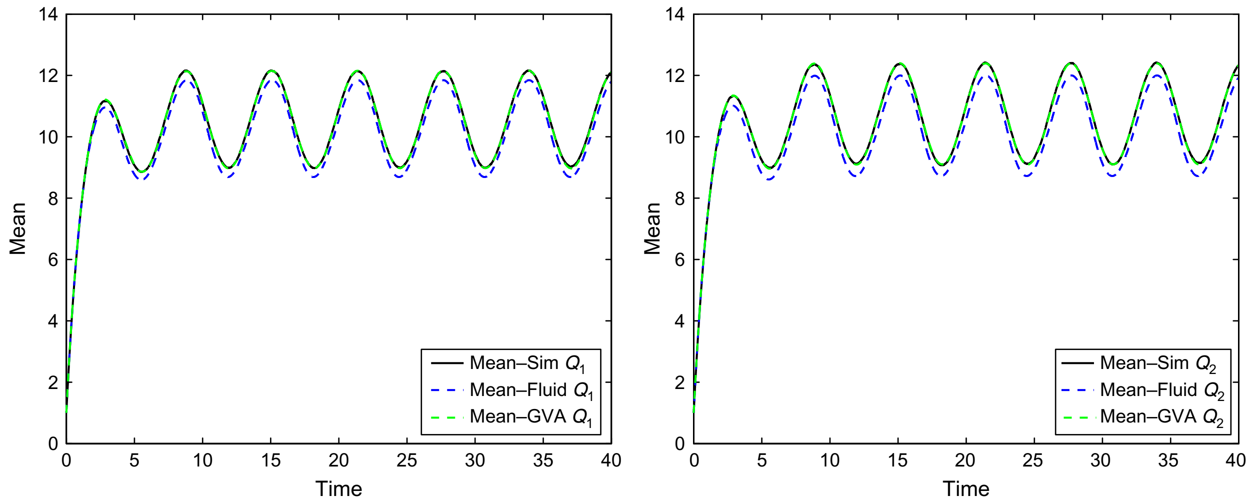

Figure 1 (Color online) Mean of Q_1 (Left). Mean of Q_2 (Right)

Note. $\lambda_1(t) = \lambda_2(t) = 10 + 2 \cdot \sin(t)$, $\mu_1 = \mu_2 = 1$, $\beta_1 = \beta_2 = 0.5$, $q_1(0) = q_2(0) = 1$, $c = 20$, $p = 0.25$.

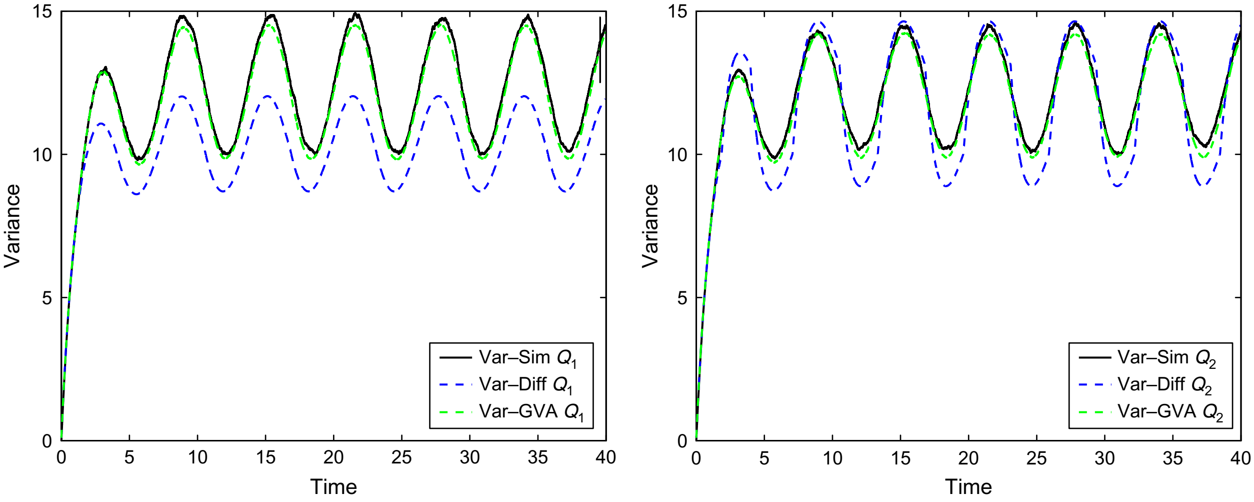

Figure 2 (Color online) Mean of Q_1 (Left). Mean of Q_2 (Right)

Note. $\lambda_1(t) = \lambda_2(t) = 10 + 2 \cdot \sin(t)$, $\mu_1 = \mu_2 = 1$, $\beta_1 = \beta_2 = 0.5$, $q_1(0) = q_2(0) = 1$, $c = 20$, $p = 0.25$.

stochastic model into its approximation of the sample path behavior. However, the sampled version of GVA and even the unsampled version allow for variability to be incorporated into the mean estimation, which improves the estimation since it has more information about the stochastic process.

Not only is our algorithm better at approximating the moments of the priority queue, but also it is much faster than a discrete event simulation is on par with the speed of the fluid and diffusion limits. In fact, for the previous example the fluid and diffusion limits were computed in two seconds, our sampling algorithm was computed in six seconds, and the discrete event simulation with 10,000 independent realizations was computed in 164 seconds. Thus, our algorithm allows us to obtain very accurate approximations in significantly less computational time than a discrete event simulation. This time savings is even more important when the dimension is larger and the parameters are larger.

In Figure 2, we plot the variances of the two dimensional priority queue with abandonment and upgrades with the diffusion limits and the variance approximation from the sampled version of the GVA. We see that for both the first and second queue, the GVA sampled version is of the GVA is estimating the simulated variance significantly better than the diffusion limit. In fact, the improvement ranges from 10%–15%, which is much larger than the improvement of the mean estimation. One possible reason that the variance is not approximated as well via the diffusion limit is the error in approximating the mean.

For example, if that mean is incorrect by a difference of ε, then the error made in the variance using the wrong mean is ε^2 as seen by the following:

$$\text{Var}_{\varepsilon}[X] = E[(X - E[X] - \varepsilon)^2] \tag{32}$$

$$= \text{Var}[X] + \varepsilon^2. \tag{33}$$

In fact, it resembles the variance and bias decomposition of the mean squared error, which is a well known result in statistics.

4.5. Additional Numerical Examples

Next, we provide additional numerical examples to illustrate our sampling algorithm in the priority queue setting. In the first additional example given in Figure 3, we scale up the earlier example by factor 10. We see again that the sampling method is approximating the mean, variance, and covariance with better accuracy than the fluid and diffusion limits. It is clear from the plots that the most improvement is given by the variance and covariance. The mean does not need as much improvement since the limit theorem for the mean is stronger than the diffusion limit.

In the second additional example given in Figure 4, we present an example where the queue is overloaded and there are no upgrades. We see that the sampling method is approximating the mean, variance, and covariance with accuracy about equal to the fluid and diffusion limits.

In the third and final additional example given in Figure 5, the queue is underloaded relative to the average arrival rate of both queues and the number of servers. We see that the sampling method is approximating the mean, variance, and covariance with significantly better accuracy than the fluid and diffusion limits. This is especially true for the variance of the first queue and the covariance of both queues.

5. General Multidimensional Birth-Death Network

In this section, we derive the functional Kolmogorov forward equations for multidimensional birth-death networks. These equations will be useful for deriving the time dependent behavior of arbitrary functionals of many different queueing networks that model interactions between different stations. More specifically, we use these forward equations to derive non-asymptotic approximations for the queueing network using our sampling algorithm. For our multidimensional birth-death network, we let $Q = (Q_1, Q_2, \ldots, Q_N)$, on $\mathbb{Z}_+^N$ with state dependent birth, death, and transition rates (respectively $\lambda(x)$, $\delta(x), D_{ij}(x), \tilde{D}_{ij}(x)$ where $x \in \mathbb{Z}_+^N$.

Proposition 5 (Engblom 2014, Theorem 4.5). *Suppose the birth, death, and transition rates satisfy for each queue $x \in \mathbf{Z}_+$ and for each i and j,*

$$D_{ij}(x) + \tilde{D}_{ij}(x) + \lambda_i(x) + \delta_i(x) \leq C(1+x), \tag{34}$$

and suppose further that $f\colon \mathbf{Z}_+^N \to \mathbf{R}^N$ is bounded by some finite pth order moment, $|f(x)| \leq C_p(1+x^p)$. Then we have the following set of functional Kolmogorov forward equations for a general multidimensional birth-death process:

$$\begin{aligned}\dot{E}[f(Q)] &= \sum_{i=1}^{N} E[\alpha_i(Q)\cdot(f(Q+e_i)-f(Q))] \\ &\quad + \sum_{i=1}^{N} E[\delta_i(Q)\cdot(f(Q-e_i)-f(Q))] \\ &\quad + \sum_{i=1}^{N}\sum_{j=1}^{N} E[D_{ij}(Q)\cdot(f(Q_i-e_i+e_j)-f(Q))] \\ &\quad + \sum_{i=1}^{N}\sum_{j=1}^{N} E[\tilde{D}_{ij}(Q)\cdot(f(Q+e_i-e_j)-f(Q))],\end{aligned} \tag{35}$$

where we have the following interpretations for the queueing network rate functions

$$\alpha_i(Q) = \text{external arrivals to } i\text{th queue} \tag{36}$$

$$\delta_i(Q) = \text{departures from } i\text{th queue out of system,} \tag{37}$$

$$D_{ij}(Q) = \text{internal arrivals to } j\text{th queue from } i\text{th queue,} \tag{38}$$

$$\tilde{D}_{ij}(Q) = \text{internal arrivals to } i\text{th queue from } j\text{th queue} \tag{39}$$

and e_i is an N-dimensional vector of all zeroes, except the ith entry, which is one. If one specializes to functions such as $\{Q_i,\, Q_i\cdot Q_j - E[Q_i]\cdot E[Q_j],\, (Q_i - E[Q_i])^2\}$, one gets the following expressions for the mean, covariance, and variance functions:

$$\dot{E}[Q_i] = E[\alpha_i(Q)] - E[\delta_i(Q)] - \sum_{j=1}^{N} E[D_{ij}(Q)] + \sum_{j=1}^{N} E[\tilde{D}_{ij}(Q)], \tag{40}$$

$$\begin{aligned}\dot{\text{Var}}[Q_i] &= E[\alpha_i(Q)] + E[\delta_i(Q)] \\ &\quad + \sum_{j=1}^{N} E[D_{ij}(Q)] + \sum_{j=1}^{N} E[\tilde{D}_{ij}(Q)] \\ &\quad + 2\cdot\text{Cov}[Q_i, \alpha_i(Q)] - 2\cdot\text{Cov}[Q_i, \delta_i(Q)] \\ &\quad - 2\cdot\sum_{j=1}^{N}\text{Cov}[Q_i, \tilde{D}_{ij}(Q)] \\ &\quad + 2\cdot\sum_{j=1}^{N}\text{Cov}[Q_i, D_{ij}(Q)],\end{aligned} \tag{41}$$

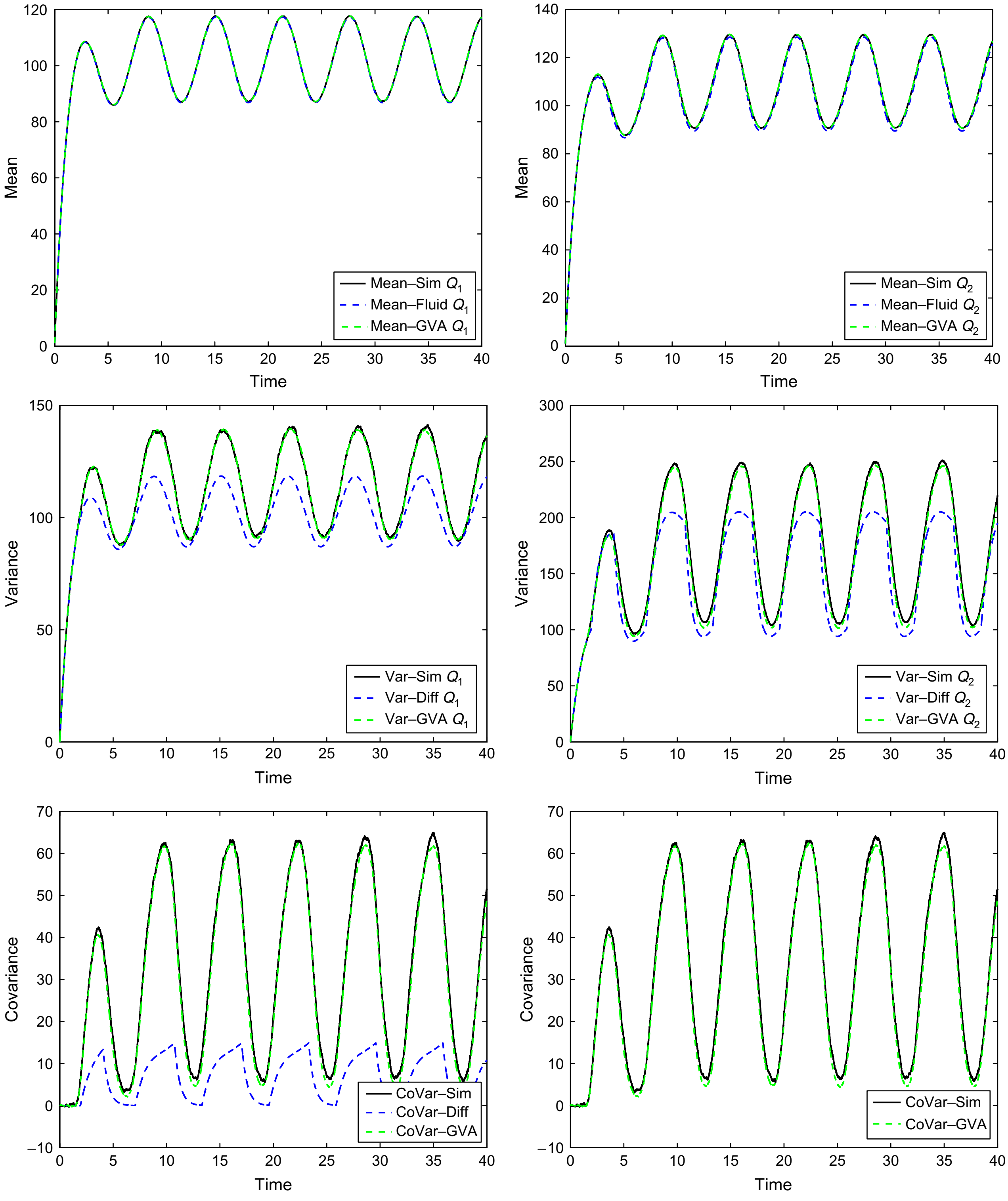

Figure 3 (Color online) Discrete Event Sim = 167 secs, Fluid-Diffusion = 2 secs, Sampling Algorithm = 7 secs

Note. $\lambda_1(t) = 100 + 20 \cdot \sin(t)$, $\lambda_2(t) = 100 + 20 \cdot \sin(t)$, $\mu_1 = 1$, $\mu_2 = 1$, $\beta_1 = 0.5$, $\beta_2 = 1$, $c = 22$, $q_1(0) = q_2(0) = 1$, $p = 0.25$.

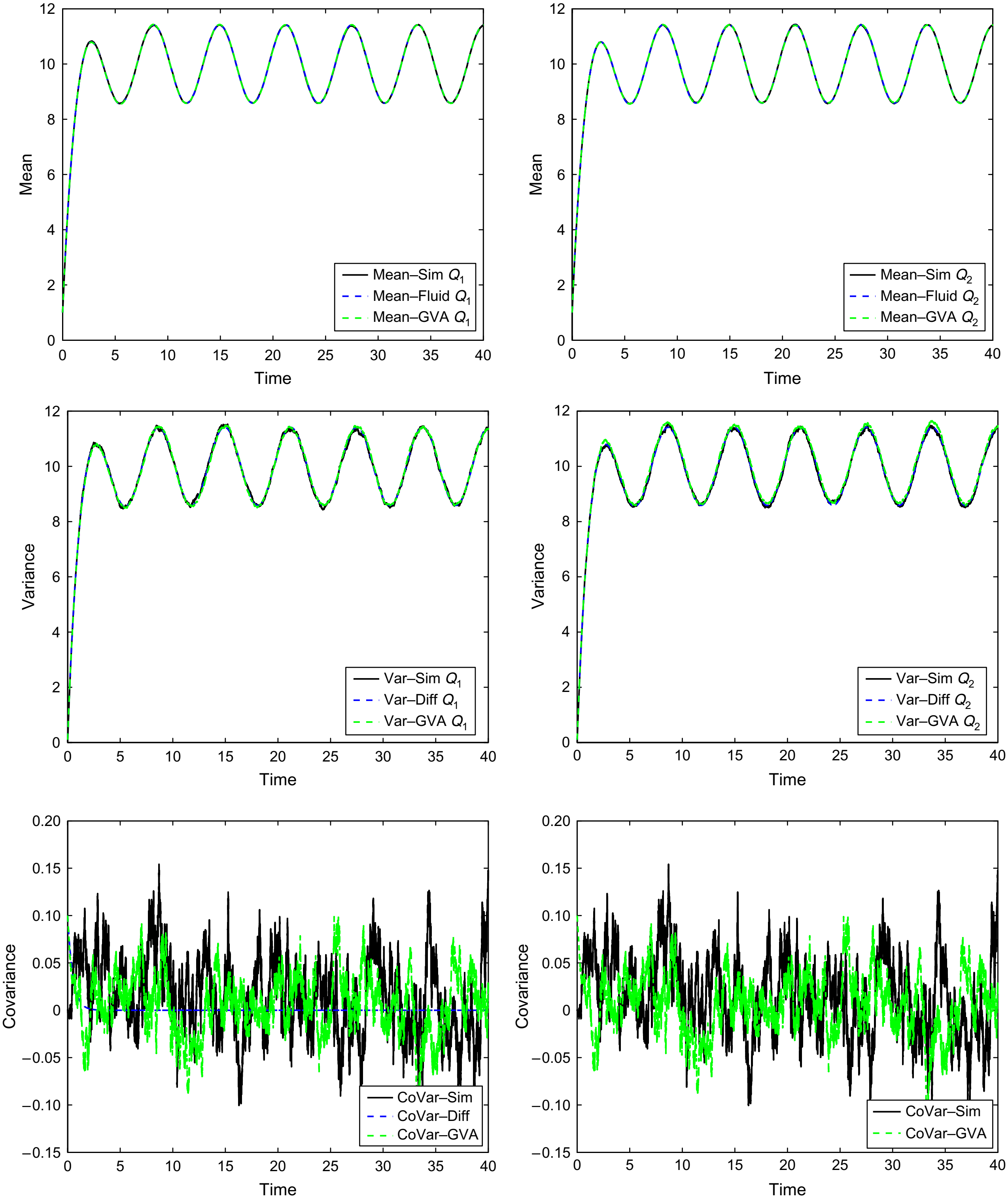

Figure 4 (Color online) Discrete Event Sim = 165 secs, Fluid-Diffusion = 2 secs, Sampling Algorithm = 7 secs

Note. $\lambda_1(t) = 10 + 2 \cdot \sin(t)$, $\lambda_2(t) = 10 + 2 \cdot \sin(t)$, $\mu_1 = 1$, $\mu_2 = 1$, $\beta_1 = 1$, $\beta_2 = 1$, $c = 10$, $q_1(0) = q_2(0) = 1$, $p = 0$.

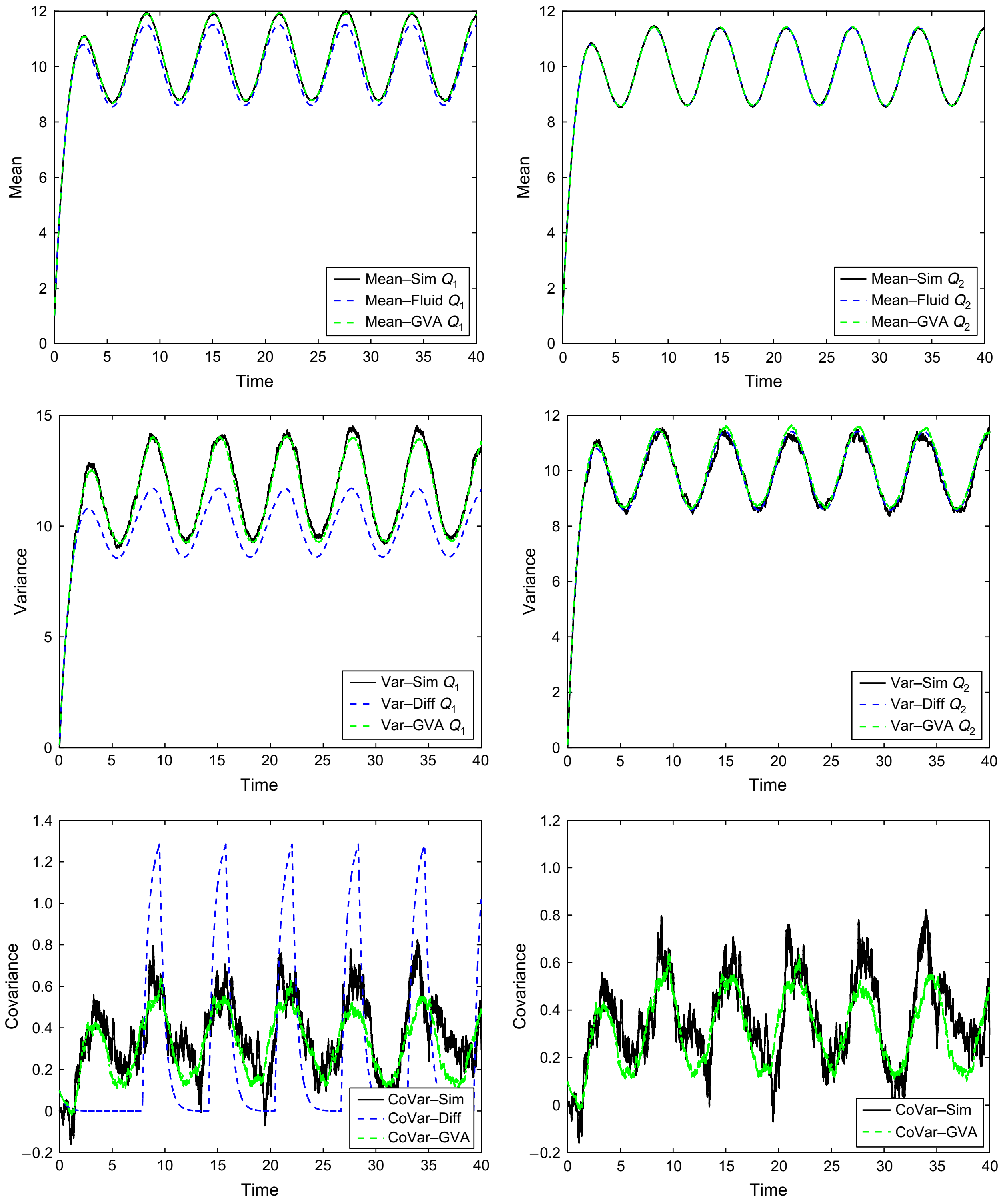

Figure 5 (Color online) Discrete Event Sim = 200 secs, Fluid-Diffusion = 2 secs, Sampling Algorithm = 7 secs

Note. $\lambda_1(t) = 10 + 2 \cdot \sin(t)$, $\lambda_2(t) = 10 + 2 \cdot \sin(t)$, $\mu_1 = 1$, $\mu_2 = 1$, $\beta_1 = 0.5$, $\beta_2 = 1$, $c = 22$, $q_1(0) = q_2(0) = 1$, $p = 0.25$.

$$\begin{aligned}\overset{\bullet}{\text{Cov}}[Q_i, Q_l] = &-\sum_{j=1}^{N} E[D_{ij}(Q)] - \sum_{j=1}^{N} E[\tilde{D}_{ij}(Q)] \\ &- \text{Cov}[Q_i, \delta_l(Q)] - \text{Cov}[Q_l, \delta_i(Q)] \\ &+ \sum_{j=1}^{N} \text{Cov}[Q_i, D_{ij}(Q) - \tilde{D}_{ij}(Q)] \\ &- \sum_{j=1}^{N} \text{Cov}[Q_l, D_{ij}(Q) - \tilde{D}_{ij}(Q)]. \qquad (42)\end{aligned}$$

5.1. Erlang Loss Queueing Network

One important example of a birth-death network is the $(M_t/M_t/c_t/k_t + M_t)^N$ queue, also known as an Erlang-loss network. In this model, we assume that each station's arrival process is a nonstationary Poisson process with a deterministic arrival rate function $\lambda_i(t)$, $t \geq 0$ when there is space available for the customer to join the queue. The service times are independent random variables with service rate $\mu_i(t)$ and customers at station i receive service from $c_i(t)$ parallel and homogenous servers. We also assume that there are $k_i(t)$ waiting spaces at each station. However, since customers are impatient, they are allowed to abandon the waiting spaces at rate $\beta_i(t)$ if they do not initiate service quickly enough. Moreover, if a customer receives service at station i and they do not leave the network, then they are randomly routed to station j with probability τ_{ij} if the station is not full. However, if a customer abandons station i and they do not leave the network permanently, then they are randomly routed to station j with probability γ_{ij} if the station is not full.

If we did not consider the possibility of customers being lost when there is not enough capacity to accommodate them, then this model would be a nonstationary Jackson network with abandonment or the $(M_t/M_t/c_t + M_t)^N$ queue. Jackson networks with abandonment fall into the class of queueing models known as Markovian service networks, which were analyzed extensively by Mandelbaum et al. (1998). It is also shown in Mandelbaum et al. (1998) that Jackson networks with abandonment have fluid and diffusion limits and can be approximated by Gaussian diffusions under mild technical conditions. However, the indicator function for the loss network, which prevents customers from joining a queue if there are not enough waiting spaces, precludes the same techniques being applied to the loss systems, since they rely on the rate functions to be Lipschitz continuous functions of the queue length. Thus, we develop a new approach to estimate the performance of these types of models using the functional Kolmogorov forward equations of the queue length process. The functional Kolmogorov forward equations for the $(M_t/M_t/c_t/k_t + M_t)^N$ queue are identical to the above equations for a multidimensional birth-death network, however, they have the following network rate functions:

$$\alpha_i(Q) = \lambda_i \cdot \{Q_i < c_i + k_i\}, \qquad (43)$$

$$\delta_i(Q) = \mu_i \cdot (Q_i \wedge c_i) + \beta_i \cdot (Q_i - c_i)^+, \qquad (44)$$

$$\begin{aligned}\tilde{D}_{ij}(Q) = &\mu_i \cdot \tau_{ij} \cdot (Q_i \wedge c_i) \cdot \{Q_j < c_j + k_j\} \\ &+ \beta_i \cdot \gamma_{ij} \cdot (Q_i - c_i)^+ \cdot \{Q_j < c_j + k_j\}, \qquad (45)\end{aligned}$$

$$\begin{aligned}D_{ij}(Q) = &\mu_j \cdot \tau_{ji} \cdot (Q_j \wedge c_j) \cdot \{Q_i < c_i + k_i\} \\ &+ \beta_j \cdot \gamma_{ji} \cdot (Q_j - c_j)^+ \cdot \{Q_i < c_i + k_i\}. \qquad (46)\end{aligned}$$

The main difficulty of Erlang-loss networks when compared to their Jackson network counterparts is that if a customer is routed to another station, there must be enough capacity at the station, otherwise the customer cannot join the next station. This creates dependence between stations that is not observed in traditional Jackson networks where customers can always join the next station since each station has an infinite capacity. Moreover, this dependence causes problems when trying to close the forward equations with a Gaussian distribution since each rate function includes an additional indicator function to preserve the loss feature of the network.

If one were to use a Gaussian distribution for approximating the mean and variance in the forward equations, one would have to make an additional assumption to generate closed form approximations for the network. To consider the full Gaussian distribution and to correctly characterize the true covariance under the Gaussian assumption, we must expand the rate functions in terms of an infinite series of Hermite polynomials. See Pender (2014a) for more details on this expansion in the bivariate case.

To circumvent the dependence issue and the infinite series of Hermite polynomials seen in Pender (2015), Pender (2013) assumes that the pairwise stations are asymptotically independent. This assumption yields simple closed form expressions for the rate functions of the functional forward equations. The independence assumption is also supported by the recent work of Gurvich and Perry (2012), which analyzes overflow networks in heavy traffic. One of their main results shows that in heavy traffic, the overflow stations are asymptotically independent. However, in the nonstationary setting with finite rates, it is clear that dependence contributes to the dynamics. Thus, using our sampling method, we can overcome this dependence and generate approximations for the queueing network that include the dependent structure of the network. Although we do not include numerical examples for this queueing network in this paper, we have performed several numerical experiments and our algorithm also performs well at estimating the network mean and variance.

6. Conclusion and Extensions

In this paper, we develop a new sampling method to approximate nonstationary stochastic jump processes using simulation and ordinary differential equations. The simulation that we use does not actually simulate the stochastic process itself, which can be quite computationally expensive; rather, it simulates the functional forward equations. The functional forward equations describe the functional dynamics of the stochastic process. Some important functions include the mean and covariance matrix of the stochastic process. We show that our sampling method accurately approximates the mean and covariance of a variety of examples while saving significant computational effort. In our examples where the exact approximation is known, our sampling method is slower by a factor of 2, while simulating the process is slower by a factor of 85. Where the exact solution is not known, our sampling method is slower by a factor of 3, whereas a discrete event simulation of the priority queue is slower by a factor of 24. However, in all of the examples, our sampling algorithm is very accurate at reproducing the simulated dynamics.

We have demonstrated that our sampling method works for nonstationary queues using Hermite polynomial expansions, however, one can extend this to other orthogonal polynomial sequences such as the Laguerre polynomials, Poisson-Charlier polynomials, and Meixner polynomials, which are orthogonal to the gamma, Poisson, and negative-binomial distributions, respectively. Pender (2014a, b, c) shows that these orthogonal sequences are all accurate at approximating the dynamic behavior of nonstationary and state dependent stochastic jump processes. Thus, one can not only apply our methodology to various birth-death processes in the queueing literature such as Erlang-loss networks and priority queues with finite capacity, but we can also use our methodology in other applications other than queueing theory such as epidemic and branching processes where the Poisson, negative binomial, and gamma distributions might arise. Finally, it would be great to apply our new algorithms for non-Markovian queues such as those in the work of Ko and Pender (2016), Pender and Ko (2016). This extension would produce even more accurate approximations for general queueing processes.

References

Clark GM (1981) Use of Polya distributions in approximate solutions to nonstationary $M/M/s$ queues. *Comm. ACM* 24(4): 206–217.

Down DG, Lewis ME (2010) The n-network model with upgrades. *Probab. Engrg. Informational Sci.* 24(02):171–200.

Engblom S (2014) On the stability of stochastic jump kinetics. *Appl. Math.* 5(19):3217–3239.

Engblom S, Pender J (2014) Approximations for the moments of nonstationary and state dependent birth-death queues. Accessed June 22, 2016, http://arxiv.org/abs/1406.6164.

García ML, Centeno MA, Rivera C, DeCario N (1995) Reducing time in an emergency room via a fast-track. *Simulation Conf. Proc. Winter* (IEEE, Arlington, VA), 1048–1053.

Green L (2006) Queueing analysis in healthcare. *Patient Flow: Reducing Delay in Healthcare Delivery* (Springer, New York), 281–307.

Gurvich I, Perry O (2012) Overflow networks: Approximations and implications to call center outsourcing. *Oper. Res.* 60(4): 996–1009.

Ko YM, Pender J (2016) Strong approximations for time varying infinite-server queues with non-renewal arrival and service processes. Accessed June 22, 2016, http://people.orie.cornell.edu/jpender/.

Liu Y, Whitt W (2014) Many-server heavy-traffic limit for queues with time-varying parameters. *Ann. Appl. Probab.* 24(1): 378–421.

Mandelbaum A, Massey WA (1995) Strong approximations for time-dependent queues. *Math. Oper. Res.* 20(1):33–64.

Mandelbaum A, Massey WA, Reiman MI (1998) Strong approximations for Markovian service networks. *Queueing Systems* 30(1–2):149–201.

Massey WA (1985) Asymptotic analysis of the time dependent $M/M/1$ queue. *Math. Oper. Res.* 10(2):305–327.

Massey WA, Pender J (2011) Skewness variance approximation for dynamic rate multiserver queues with abandonment. *Performance Evaluation Rev.* 39(2):74.

Massey WA, Pender J (2013) Gaussian skewness approximation for dynamic rate multi-server queues with abandonment. *Queueing Systems* 75(2):243–277.

Massey WA, Pender J (2016) Approximating and stabilizing Jackson networks with abandonment. *Probab. Engrg. Informational Sci.* Forthcoming.

Massey WA, Whitt W (1998) Uniform acceleration expansions for markov chains with time-varying rates. *Ann. Appl. Probab.* 8(4):1130–1155.

Pender J (2013) Gaussian approximations for nonstationary loss networks with abandonment. Accessed June 22, 2016, https://people.orie.cornell.edu/jpender/Loss_Network.pdf.

Pender J (2014a) Laguerre polynomial expansions for time varying multiserver queues with abandonment. Accessed June 22, 2016, http://people.orie.cornell.edu/jpender/LSA.pdf.

Pender J (2014b) Gram Charlier expansions for time varying multi-server queues with abandonment. *SIAM. J. Appl. Math.* 74(4):1238–1265.

Pender J (2014c) A Poisson–Charlier approximation for nonstationary queues. *Oper. Res. Lett.* 42(4):293–298.

Pender J (2015) An analysis of nonstationary coupled queues. *Telecomm. Systems* 61(4):1–16.

Pender J, Ko YM (2016) Approximations for the queue length distributions of time-varying many-server queues. Accessed June 22, 2016, http://people.orie.cornell.edu/jpender/.

Rothkopf MH, Oren SS (1979) A closure approximation for the nonstationary $M/M/s$ queue. *Management Sci.* 25(6):522–534.

Siddharthan K, Jones WJ, Johnson JA (1996) A priority queuing model to reduce waiting times in emergency care. *Internat. J. Health Care Quality Assurance* 9(5):10–16.

Stein C (1986) Approximate computation of expectations. *Lecture Notes-Monograph Ser.* 7:i–164.

Taaffe MR, Ong KL (1987) Approximating nonstationary $Ph(t)/M(t)/s/c$ queueing systems. *Ann. Oper. Res.* 8(1):103–116.