Variance and Derivative Estimation of Virtual Performance

YUJING LIN, Supply Chain Optimization Technology, Amazon.com, USA
BARRY L. NELSON, Northwestern University, USA

Virtual performance is a class of time-dependent performance measures conditional on a particular event occurring at time τ_0 for a (possibly) nonstationary stochastic process; virtual waiting time of a customer arriving to a queue at time τ_0 is one example. Virtual statistics are estimators of the virtual performance. In this article, we go beyond the mean to propose estimators for the variance, and for the derivative of the mean with respect to time, of virtual performance, examining both their small-sample and asymptotic properties. We also provide a modified K-fold cross validation method for tuning the parameter k for the difference-based variance estimator, and we evaluate the performance of both variance and derivative estimators via controlled studies and a realistic illustration. The variance and derivative provide useful information that is not apparent in the mean of virtual performance.

CCS Concepts: • **Mathematics of computing** → **Bootstrapping**; **Nonparametric statistics**; • **Computing methodologies** → Simulation evaluation;

Additional Key Words and Phrases: Nearest-neighbor regression, queueing simulation, output analysis

ACM Reference format:
Yujing Lin and Barry L. Nelson. 2018. Variance and Derivative Estimation of Virtual Performance. *ACM Trans. Model. Comput. Simul.* 28, 3, Article 17 (July 2018), 20 pages.
https://doi.org/10.1145/3209959

1 INTRODUCTION

"Virtual statistics," as we define them, are estimators for performance measures that are conditional on the occurrence of an event at a particular time, say τ_0. This class of measures we call virtual performance at time τ_0, denoted by $V(\tau_0)$. Lin and Nelson (2016) and Lin et al. (2017) focus on estimating the mean of some time-dependent virtual performance, denoted by $v(\tau_0) = \mathrm{E}[V(\tau_0)]$, for a (possibly) nonstationary stochastic process using the output of computer simulation, and they propose a k-nearest-neighbors ($k$nn) estimator of it. Lin et al. (2017) show that $k$nn is a simple, easy-to-tune, robust estimator for the mean of virtual performance from time-dependent, strongly correlated simulation sample paths. In this article, we go beyond estimation of the virtual mean $v(\tau_0)$, and we develop methods to estimate the virtual variance of $V(\tau_0)$ and the derivative with respect to time of the virtual mean $dv(\tau_0)/dt$.

This research was partially supported by the National Science Foundation under Grant No. CMMI-1537060 and GOALI co-sponsor SAS Institute. A less complete version of this work previously appeared in Lin and Nelson (2017).

Authors' addresses: Y. Lin, Amazon.com, 1743 NW 63rd Street, Unit A, Seattle, WA, 98107; email: yjlin17@gmail.com; B. L. Nelson, Department of Industrial Engineering & Management Sciences, Northwestern University, Evanston, IL 60208-3119 USA; email: nelsonb@northwestern.edu.

https://doi.org/10.1145/3209959

To motivate the study of virtual statistics and the need for more than just the mean, consider an emergency department (ED). In this example, if walk-in patients find the ED is too congested upon their arrival, then they might leave immediately to seek help elsewhere. The patients who decide to stay are registered first and then wait for a nurse to walk them to an available bed. Some of these patients need radiology and treatment from a doctor. From the perspective of patients and administrators, it is of interest to know how long it might take patients to get a bed, or how long they have to stay in the ED when further radiology and treatment are required, if they arrive to the ED at, say, 7:30 AM. However, it is important to be aware of the probability that patients do not wait due to congestion so that the ED can better adjust the number of nurses and doctors. The *k*nn estimator proposed by Lin and Nelson (2016) could be used to estimate the mean of the virtual waiting time to get a bed for the patient who arrives at τ_0 = 7:30 AM, and the virtual probability that a patient leaves immediately if they arrive at τ_0 = 7:30 AM (or any other time). Of course, neither the patient nor the ED is likely to experience exactly the mean of the virtual waiting time or the mean of the virtual probability. The *variance* (or standard deviation) of the virtual waiting time or the virtual probability provides a more complete description of the distribution that the patient might experience. Furthermore, if a patient arrives to the ED slightly earlier or later than time τ_0, then the ED administrators might want to know whether or not this change would lead to a much longer expected waiting time; that is, is the mean $v(\tau_0)$ changing rapidly with time? Such a question can be answered by the *derivative* of the mean of the virtual waiting time with respect to the arrival time. Therefore, it is valuable to go beyond the mean of virtual performance to study the virtual variance and derivative.

How might we estimate the virtual variance? In regression problems when the true regression function is unknown, a typical approach for estimating the response variance is to first estimate the regression function, and then to derive the response variance from the residuals; this is called *residual variance estimation*. There exists substantial research on residual variance estimation using nearest neighbors. For example, Liitiäinen et al. (2010) describe a residual variance estimator using nearest neighbor statistics, and Liitiäinen et al. (2008) study variance estimation for a general setting that covers non-additive heteroscedastic noise under non-i.i.d. sampling. We introduce both a residual variance estimator and an alternative difference-based variance estimator adapted to our virtual performance setting.

The derivative of the mean response reveals how the system will respond to a change in the time that the trigger event occurs. Additionally, the derivative allows simulation users to obtain some idea of how many time points at which they should estimate the virtual mean $v(\tau_0)$ to understand the system performance profile. For example, if we estimate the mean $v(\tau_0)$ at a set of time points and find the derivatives at these points are large, then more time points are needed to fully characterize the mean performance profile. However, if the derivative of $v(\tau_0)$ is close to 0 at some time points, then it is not necessary to estimate the mean of virtual performance at times close to τ_0, because we know $v(t)$ changes very slowly near τ_0.

The finite difference (FD) method has been widely used for derivative estimation in simulation. Although FD is well known, we show later why it is incompatible with a nonparametric *k*nn approach. In addition to FD, there are many other types of derivative estimators. One of them is similar to the residual variance estimation scheme; that is, one should estimate the unknown regression function first by using some smooth functions such as polynomials or splines and then compute the estimator by taking the derivative of the estimated regression function with respect to time. For example, Zhou and Wolfe (2000) study the estimation of derivatives using splines. Gasser and Müller (1979) and Gasser and Müller (1984) describe kernel-based derivative estimators. A more recent derivative estimation method is based on weighted slopes of symmetric observations around the time $t = \tau_0$ of interest. De Brabanter et al. (2013) and De Brabanter and Liu (2015) study this type of estimator and show its asymptotic properties. Although all of these approaches can apply to virtual performance settings, we focus on the weighted-slopes type of derivative estimator, because it can be treated as an extension of our existing $k$nn mean estimation results.

The remainder of this article is organized as follows. We start with a summary of our work on mean estimation for virtual performance in Section 2, which includes important assumptions and results from Lin et al. (2017). In Section 3, we formally define our variance and derivative estimators for virtual performance. The asymptotic properties of the proposed estimators under specific conditions on the system of interest and the growth rate of the tuning parameter k are offered in Section 4. We introduce a modified K-fold cross validation method for tuning the parameter of the difference-based variance estimator in Section 5. To evaluate the performance of the proposed variance and derivative estimators, we apply our method to controlled studies in Section 6.1, comparing the estimators with the true variance and derivative of virtual performance. We also apply our proposed virtual statistics to a simulated ED problem in Section 6.2. Some conclusions are provided in Section 7. Portions of this article were published in the Proceedings of the 2017 Winter Simulation Conference as Lin and Nelson (2017).

2 THE kNN METHOD FOR THE MEAN

We first present the definition of virtual performance given in Lin et al. (2017). Consider a stochastic point process that begins at time $T_{\text{start}} \equiv 0$ and ends at time $T_{\text{end}} \equiv T$ where $\mathrm{E}(T^2) < \infty$. The random event times are $0 < t_1 < t_2 < \cdots < t_M \leq T$; in the simulation setting these will typically be the times that a common type of event occurs, such as "customer arrival" or "machine failure," although that is not essential. We will call all of these events "arrivals" from here on even though they may not be.

The simulation also generates an output process $Y_1, Y_2, \ldots$ that we call the performance measure. We assume there is a unique Y_i associated with each t_i; for notation, we denote this $Y(t_i)$, which simply means this is the Y associated with arrival time t_i. In the setting we have in mind, $t_i < t_{i+1}$ does not necessarily imply that $Y(t_i)$ is realized in the simulation before $Y(t_{i+1})$. For instance, if $t_1 < t_2$ are the arrival times of the first and second customers to a queue, and $Y(t_1)$ and $Y(t_2)$ their respective sojourn times, customer 2 might depart before customer 1 if overtaking can occur. The process $\{(t_i, Y(t_i));\ i = 1, 2, \ldots, M\}$ is the basic data of interest in this article

For a fixed time $0 < \tau_0 \leq T$, let $V_i(\tau_0) \overset{\mathcal{D}}{=} (Y(t_i) | t_i = \tau_0)$. This is a random variable having the distribution of the performance of the ith arrival, given that arrival occurred at time τ_0. While possibly interesting it its own right, we focus instead on $V(\tau_0)$, where

$$\Pr\{V(\tau_0) \leq y_0\} = \sum_{i=1}^{\infty} \Pr\{V_i(\tau_0) \leq y_0\}\, q_i(\tau_0) \tag{1}$$

and $q_i(\tau_0) = \Pr\{t_i = \tau_0 | \text{an arrival occurs at } \tau_0\}$. In other words, $V(\tau_0)$ is the performance Y for an arrival at τ_0, given *some* arrival occurred at time τ_0. We refer to this as the *virtual performance at* τ_0.

Remark. There are several definitions of "virtual" performance in the literature. Our definition is conditional on an arrival from the *nominal* arrival process occurring at time τ_0; other definitions are for a real or phantom customer *injected* into the nominal system at τ_0, or what is experienced by an arbitrary random arrival. Our definition is particularly appropriate for profiling performance of a nonstationary system over time.

Lin et al. (2017) propose a $k$nn method for estimating $v(\tau_0) = \mathrm{E}[V(\tau_0)]$ from n independent simulation replications and provide two approaches for measuring the error of the $k$nn mean estimator. Therefore, the simulation data are $\{(t_{ij}, Y(t_{ij}));\ i = 1, 2, \ldots, M_j\}, j = 1, 2, \ldots, n$, where the subscript j denotes the jth replication. We assume that all these data are *retained* rather than summarized, which facilitates tuning the estimator to the characteristics of the data, and estimating virtual performance for any time τ_0 specified in advance or later. We also assume $\mathrm{E}[Y^2(t_{ij})] < \infty$ for all t_{ij}, implying that the simulation-generated performance measure for each arrival has finite mean and variance.[1] In this article, we focus on the same type of stochastic ouput process but will study different virtual statistics. The development is based on some important results from Lin et al. (2017). Therefore, we restate the relevant assumptions and results in this section.

Denote the superposed process of all of the observed arrival times by $\mathcal{T}_n = \{t_{ij} : i = 1, 2, \ldots, M_j, j = 1, 2, \ldots, n\}$. The $k$nn estimator of $v(\tau_0)$, $\bar{V}(\tau_0)$, proposed by Lin et al. (2017) is

$$\bar{V}(\tau_0) = \frac{1}{k}\sum_{\ell=1}^{k} Y(\tau_0^{(\ell,n)}), \tag{2}$$

where $\tau_0^{(1,n)} < \tau_0^{(2,n)} < \cdots < \tau_0^{(k,n)}$ are the *sorted* k nearest neighbors to τ_0 from the superposed process $\mathcal{T}_n$, and $Y(\tau_0^{(\ell,n)})$ is the corresponding observed output for $\ell = 1, 2, \ldots, k$. Notice that the "closeness" here is based on $|\tau_0^{(\ell,n)} - \tau_0|$ regardless of replication and ties are broken arbitrarily. Thus, $\bar{V}(\tau_0)$ is computed from a mix of independent and dependent output data that are not identically distributed in general.

The system of interest analyzed in this article satisfies the same properties assumed in Lin et al. (2017). Let the arrival-counting process associated with t_{ij} from a generic replication of the dynamic system to be denoted by $\{N(t) : t \geq 0\}$. For any time interval $(t - w/2, t + w/2]$ with $w > 0$, let the number of arrivals within $(t - w/2, t + w/2]$ to be denoted by $N^w(t) = N(t + w/2) - N(t - w/2)$. If τ_0 is very close to the endpoint 0, then $t - w/2$ might be negative so that $N(t - w/2)$ is not defined. A similar issue occurs for τ_0 that is close to T. Thus, we further define $N(t) = N(0)$ for $t \leq 0$, and $N(t) = N(T)$ for $t \geq T$. For each replication, suppose $\{N(t) : t \geq 0\}$ satisfies the following properties for all $t \in (0, T]$:

$$\Pr\{N^w(t) \geq 1\} = \lambda_t w + o(w) \quad \text{and} \quad \Pr\{N^w(t) \geq 2\} = o(w), \tag{3}$$

where $\lambda_t > 0$ is the arrival process intensity at time t and $o(w)$ indicates a term for which $\lim_{w\to 0} o(w)/w = 0$. Note that Equation (3) is weaker than the conditions for a Poisson arrival process, because the latter also requires independent increments.

Lin et al. (2017) show that if $k/n \to 0$ as $n \to \infty$, then the *smallest symmetric* interval that contains the k nearest neighbors of τ_0, denoted by $W_n^k(\tau_0)$, converges to 0 in L^2 norm and almost surely; and the k nearest neighbors are asymptotically from distinct replications, implying that they are asymptotically independent. These results are used to prove consistency of $\bar{V}(\tau_0)$ for $v(\tau_0)$ in Lin et al. (2017), and they will also be important here.

3 VIRTUAL VARIANCE AND DERIVATIVE ESTIMATION

In this section, we define the variance and derivative of the mean for the virtual performance of our stochastic process, and propose our variance and derivative estimators.

3.1 Variance Estimation

The variance of the virtual performance $V(\tau_0)$ is $\sigma^2(\tau_0) = \mathrm{Var}(V(\tau_0))$. We define a class of $k$nn variance estimator to be

$$\widehat{\sigma}^2(\tau_0) = \sum_{(\ell,m)\in\mathcal{V}(\tau_0)} \phi_{\ell m}\left[Y(\tau_0^{(\ell,n)}) - Y(\tau_0^{(m,n)})\right]^2, \tag{4}$$

[1] For notational simplicity, we refer to this assumption as $\mathrm{E}[Y^2(t)] < \infty$ from here on.

where the set $\mathcal{V}(\tau_0)$ contains the indices of the pairs $(\tau_0^{(\ell,n)}, \tau_0^{(m,n)})$ used for computing $\widehat{\sigma}^2(\tau_0)$ and $\phi_{\ell m}$ is the associated set of weights; we will consider two different sets of indices and weights.

If $\mathcal{V}(\tau_0) = \{(\ell, m) \mid \ell \neq m \in \{1, 2, \ldots, k\}\}$ (i.e., all the pairs of observations are used), and we use $\phi_{\ell m} = 1/(2k(k-1))$ for all (ℓ, m), then $\widehat{\sigma}^2(\tau_0)$ coincides with the *sample variance* of the k nearest neighbors; it is also called a *residual-based* variance estimator, which we denote by $\widehat{\sigma}^2_{RB}(\tau_0)$ with $\mathcal{V}(\tau_0) = \mathcal{V}_{RB}(\tau_0)$; i.e.,

$$\widehat{\sigma}^2_{RB}(\tau_0) = \sum_{(\ell,m)\in\mathcal{V}_{RB}(\tau_0)} \frac{1}{2k(k-1)} \left[Y(\tau_0^{(\ell,n)}) - Y(\tau_0^{(m,n)})\right]^2 = \frac{1}{k-1} \sum_{\ell=1}^{k} \left[Y(\tau_0^{(\ell,n)}) - \bar{Y}(\tau_0)\right]^2. \quad (5)$$

Our residual-based variance estimator $\widehat{\sigma}^2_{RB}(\tau_0)$ is different from a typical sample variance, which is computed from k *independent* measurements at $t = \tau_0$. Since it is very unlikely we will obtain any, much less multiple, observations at τ_0 due to the nature of virtual performance, our proposed residual-based variance estimator is constructed based on the k nearest neighbors around τ_0 and these k observations are usually dependent.

The residual-based variance estimator in Equation (5) involves the pairs $(\tau_0^{(\ell,n)}, \tau_0^{(m,n)})$ from the k nearest neighbors. By contrast, Rice (1984) proposes a first-order *difference-based* variance estimator, denoted by $\widehat{\sigma}^2_{DB}(\tau_0)$, that only contains the pairs of any two *successive* observations; thus the corresponding index set $\mathcal{V}(\tau_0)$ becomes $\mathcal{V}_{DB}(\tau_0) = \{(\ell, m) \mid m = \ell - 1, \ell \in \{2, \ldots, k\}\}$ and the associated weights are $\phi_{\ell m} = 1/(2(k-1))$, so

$$\widehat{\sigma}^2_{DB}(\tau_0) = \sum_{(\ell,m)\in\mathcal{V}_{DB}(\tau_0)} \frac{1}{2(k-1)} \left[Y(\tau_0^{(\ell,n)}) - Y(\tau_0^{(m,n)})\right]^2 = \frac{1}{2(k-1)} \sum_{\ell=2}^{k} \left[Y(\tau_0^{(\ell,n)}) - Y(\tau_0^{(\ell-1,n)})\right]^2. \quad (6)$$

Compared with the residual-based variance estimator defined in Equation (5), a difference-based variance estimator like Equation (6) removes the trend in the mean. There exist other variations of difference-based variance estimators. For example, Gasser et al. (1986) introduce pseudo-residuals to construct their difference-based variance estimator, which assigns each squared difference its own weight based on their distances to the point of interest. Typically, equally weighted difference-based variance estimators are applied for problems with equi-spaced design points, and many relevant papers like Rice (1984) assume independence among the observations. However, the superposed arrivals in $\mathcal{T}_n$ could be very dense if either the arrival intensity or the number of replications n is large, so all observations within the superposed sample path $\mathcal{T}_n$ are close to each other and the impact of the differing distances will be less significant. As for the independence assumption, we will establish the asymptotic independence for the k nearest neighbors around τ_0 under certain conditions on the system and the growth rate of k. Therefore, we suggest the equally-weighted difference-based variance estimator defined in Equation (6).

To further compare these two $k$nn variance estimators, $\widehat{\sigma}^2_{RB}(\tau_0)$ and $\widehat{\sigma}^2_{DB}(\tau_0)$, we establish their asymptotic properties in Section 4, and propose a parameter-tuning approach for $\widehat{\sigma}^2_{DB}(\tau_0)$ in Section 5.

3.2 Derivative Estimation

The derivative of $v(t)$ evaluated at $t = \tau_0$ is $v'(\tau_0) = dv(t)/dt|_{t=\tau_0}$. As mentioned in Section 1, the traditional FD method cannot be effectively used in a virtual statistics problem. If the FD δ is small, as it should be for low bias, then the arrival times t_{ij} in the interval $[\tau_0, \tau_0 + \delta]$ or $[\tau_0 - \delta, \tau_0 + \delta]$ may be nearly the *same*, and therefore cancel in a FD estimator.

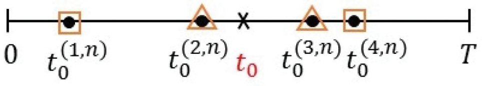

Fig. 1. An example of constructing $\widehat{\beta}_{SWD}(\tau_0)$ based on four nearest neighbors. The △ and □ indicate the neighbors that are paired.

A näive derivative estimator for $v'(t)$ at $t = \tau_0$ is $(Y(\tau_0^{(\ell,n)}) - Y(\tau_0^{(m,n)}))/(\tau_0^{(\ell,n)} - \tau_0^{(m,n)})$, where $(\tau_0^{(\ell,n)}, Y(\tau_0^{(\ell,n)}))$ and $(\tau_0^{(m,n)}, Y(\tau_0^{(m,n)}))$ are two observations near τ_0. Motivated by this insight, we define a class of derivative estimators for $v'(\tau_0)$:

$$\widehat{\beta}(\tau_0) = \sum_{(\ell,m)\in\mathcal{D}(\tau_0)} \omega_{\ell m} \left[\frac{Y(\tau_0^{(\ell,n)}) - Y(\tau_0^{(m,n)})}{\tau_0^{(\ell,n)} - \tau_0^{(m,n)}} \right], \quad \text{where} \quad \omega_{\ell m} = \frac{(\tau_0^{(\ell,n)} - \tau_0^{(m,n)})^2}{\sum_{(r,s)\in\mathcal{D}(\tau_0)} (\tau_0^{(r,n)} - \tau_0^{(s,n)})^2}. \tag{7}$$

The derivative estimator defined in Equation (7) is the weighted average of the slopes of two neighbors within $\mathcal{D}(\tau_0)$, and the weight ω_{lm} is proportional to the difference between $\tau_0^{(\ell,n)}$ and $\tau_0^{(m,n)}$. Similar to the index set $\mathcal{V}(\tau_0)$ in the variance estimator, $\mathcal{D}(\tau_0)$ contains the indices of all pairs $(\tau_0^{(\ell,n)}, \tau_0^{(m,n)})$ used for computing $\widehat{\beta}(\tau_0)$.

A natural choice of $\mathcal{D}(\tau_0)$ is to employ the same k nearest neighbors used in estimating the mean, $\bar{V}(\tau_0)$. Then $\mathcal{D}(\tau_0) = \{(\ell, m) \mid \ell \neq m \in \{1, 2, \ldots, k\}\}$. In this case, we can express $\widehat{\beta}(\tau_0)$ as

$$\widehat{\beta}(\tau_0) = \sum_{\ell\neq m}^{k} \frac{(\tau_0^{(\ell,n)} - \tau_0^{(m,n)})^2}{\sum_{r\neq s}^{k} (\tau_0^{(r,n)} - \tau_0^{(s,n)})^2} \cdot \frac{Y(\tau_0^{(\ell,n)}) - Y(\tau_0^{(m,n)})}{\tau_0^{(\ell,n)} - \tau_0^{(m,n)}} = \frac{\sum_{\ell=1}^{k} (\tau_0^{(\ell,n)} - \bar{t})(Y(\tau_0^{(\ell,n)}) - \bar{V}(\tau_0))}{\sum_{\ell=1}^{k} (\tau_0^{(\ell,n)} - \bar{t})^2},$$

where $\bar{t} = \sum_{\ell=1}^{k} \tau_0^{(\ell,n)}/k$. For this choice of $\mathcal{D}(\tau_0)$, the derivative estimator defined in Equation (7) coincides with the ordinary least squares (OLS) estimator. Such a derivative estimator, denoted by $\widehat{\beta}_{OLS}(\tau_0)$ associated with $\mathcal{D}_{OLS}(\tau_0)$, can also be viewed as the estimated slope coefficient for a linear regression model of the k nearest neighbors to τ_0.

De Brabanter et al. (2013) and De Brabanter and Liu (2015) propose a different choice of $\mathcal{D}(\tau_0)$ for constructing $\widehat{\beta}(\tau_0)$. Instead of using all $(\tau_0^{(\ell,n)}, \tau_0^{(m,n)})$, they only choose the pairs where $\tau_0^{(\ell,n)}$ and $\tau_0^{(m,n)}$ are symmetric around τ_0. We call such a derivative estimator the *symmetric weighted difference* (SWD) estimator, and the corresponding index set becomes $\mathcal{D}_{SWD}(\tau_0) = \{(\ell, m) \mid \ell + m = 2\tilde{k} + 1, \ell > m \in \{1, 2, \ldots, \tilde{k}\}\}$, where $\tilde{k}$ is the number of involved pairs (i.e., slopes).

A simple illustration for constructing $\widehat{\beta}_{SWD}(\tau_0)$ is shown in Figure 1. Suppose we use the four nearest neighbors around τ_0 to construct $\widehat{\beta}(\tau_0)$, then $\widehat{\beta}_{OLS}(\tau_0)$ will involve all $4 \times (4 - 1) = 12$ pairs of $(\tau_0^{(\ell,n)}, \tau_0^{(m,n)})$ while $\widehat{\beta}_{SWD}(\tau_0)$ will only include two pairs: $(\tau_0^{(4,n)}, \tau_0^{(1,n)})$ and $(\tau_0^{(3,n)}, \tau_0^{(2,n)})$.

The number of involved slopes $\tilde{k}$ must satisfy $\tilde{k} \leq k/2$. In the simple example shown above, $\tilde{k} = 2$ when four nearest neighbors are chosen, which is the best situation. If $\tau_0^{(2,n)}$ also locates on the same side of τ_0 as $\tau_0^{(3,n)}$ and $\tau_0^{(4,n)}$, then $\widehat{\beta}_{SWD}(\tau_0)$ will only contain one slope computed from $(\tau_0^{(2,n)}, \tau_0^{(1,n)})$. The worst case is that all these four nearest neighbors are on one side of τ_0, so that we cannot construct $\widehat{\beta}_{SWD}(\tau_0)$ according to its definition. Therefore, to construct a SWD estimator $\widehat{\beta}_{SWD}(\tau_0)$, we do not use the original k nearest neighbors. Instead, we choose the k nearest neighbors to τ_0 from $(0, \tau_0]$ and another k nearest neighbors to τ_0 from $(\tau_0, T]$, and sort these $2k$ neighbors as $\tau_{0,SWD}^{(1,n)} < \tau_{0,SWD}^{(2,n)} < \cdots < \tau_{0,SWD}^{(2k,n)}$. Then the index set $\mathcal{D}(\tau_0)$ for $\widehat{\beta}_{SWD}(\tau_0)$ is

$\mathcal{D}_{SWD}(\tau_0) = \{(\ell, m) \mid \ell + m = 2k+1, \ell > m \in \{1, 2, \ldots, k\}\}$ so that $\widehat{\beta}_{SWD}(\tau_0)$ is constructed on k pairs of symmetric observations around τ_0. Note that these $2k$ neighbors might not be the $2k$ *nearest* neighbors to τ_0. The asymptotic properties of $\widehat{\beta}_{OLS}(\tau_0)$ and $\widehat{\beta}_{SWD}(\tau_0)$ are established in Section 4.

Remark. Typically OLS and SWD estimators are used when data are independent and homoscedastic, but the retained simulation data of interest to us is very unlikely to satisfy these two conditions, which is the main difference between our setting and the existing literature. However, we can establish similar asymptotic properties under certain conditions on the system and the growth rate of k. More details are provided in Section 4.

4 ASYMPTOTIC PROPERTIES OF VARIANCE AND DERIVATIVE ESTIMATORS

In this section, we establish the asymptotic properties of the proposed variance and derivative estimators. The proofs for all the asymptotic results are provided in Appendices A–E.

Theorem 4.1. *Suppose that the system of interest satisfies* $\mathrm{E}[Y^2(t)] < \infty$ *and its arrival-counting process satisfies Equation (3), and that the true response surface* $v(t)$ *and the marginal variance* $\sigma^2(t)$ *are Lipschitz continuous with finite Lipschitz constants* $L_1, L_2 > 0$ *for any* $t_1, t_2 \in [0, T]$. *If* $k/n \to 0$ *as* $k, n \to \infty$, *then*

(i) the residual-based variance estimator $\widehat{\sigma}^2_{RB}(\tau_0)$ *is asymptotically unbiased and consistent for* $\sigma^2(\tau_0)$;

(ii) the difference-based variance estimator $\widehat{\sigma}^2_{DB}(\tau_0)$ *is asymptotically unbiased for* $\sigma^2(\tau_0)$;

(iii) if in addition, $\mathrm{E}[T^4] < \infty$ *and* $\mathrm{E}[V^4(\tau_0)]$ *is also Lipschitz continuous with finite Lipschitz constant* $L_3 > 0$ *for any* $t_1, t_2 \in [0, T]$, *then* $\widehat{\sigma}^2_{DB}(\tau_0)$ *is asymptotically consistent for* $\sigma^2(\tau_0)$.

De Brabanter and Liu (2015) show the asymptotic unbiasedness and consistency for the derivative estimator $\widehat{\beta}_{SWD}(\tau_0)$, but they only consider cases where all observations are independent and homoscedastic. We employ the key part of their proof and then extend it to our problem in which the observations might be dependent and heteroscedastic. Before establishing the asymptotic properties for $\widehat{\beta}_{SWD}(\tau_0)$, we need to establish the following lemma.

Lemma 4.2. *Suppose that the system of interest satisfies* $\mathrm{E}[Y^2(t)] < \infty$ *and its arrival-counting process satisfies (3). Let* $\tau^{(1,n)}_{0,SWD} < \tau^{(2,n)}_{0,SWD} < \cdots < \tau^{(2k,n)}_{0,SWD}$ *be the sorted* $2k$ *observations used for computing* $\widehat{\beta}_{SWD}(\tau_0)$. *Define* $W^{2k}_{SWD}(\tau_0) = \tau^{(2k,n)}_{0,SWD} - \tau^{(1,n)}_{0,SWD}$ *as the smallest interval that contains these* $2k$ *observations, and*

$$I^{2k}_{SWD}(\tau_0) = \begin{cases} 1, & \text{if } \tau^{(1,n)}_{0,SWD}, \tau^{(2,n)}_{0,SWD}, \ldots, \tau^{(2k,n)}_{0,SWD} \text{ are from distinct replications} \\ 0, & \text{otherwise.} \end{cases}.$$

If $k/n \to 0$ *as* $n \to \infty$, *then*

(i) $W^{2k}_{SWD}(\tau_0) \xrightarrow{L^2} 0$, *implying that* $\lim_{\substack{n \to \infty \\ k/n \to 0}} \mathrm{E}[(W^{2k}_{SWD}(\tau_0))^2] = 0$; *and* $W^{2k}_{SWD}(\tau_0) \xrightarrow{a.s.} 0$; *further*

(ii) $\Pr\{I^{2k}_{SWD}(\tau_0) = 1\} \to 1$; *that is,* $\{Y(\tau^{(1,n)}_{0,SWD}), Y(\tau^{(2,n)}_{0,SWD}), \ldots, Y(\tau^{(2k,n)}_{0,SWD})\}$ *are asymptotically independent.*

Theorem 4.3. *Suppose that the system of interest satisfies* $\mathrm{E}[Y^2(t)] < \infty$ *and its arrival-counting process satisfies Equation (3), and that* $v(t)$ *is twice continuously differentiable with* $v''(t) < \infty$ *and* $\sup_{t \in [0,T]} \sigma^2(t) = \sigma^2_{sup} < \infty$. *If* $k/n \to 0$ *as* $k, n \to \infty$, *then*

(i) *$\widehat{\beta}_{SWD}(\tau_0)$ is asymptotically unbiased for $v'(\tau_0)$;*
(ii) *if in addition, $k^{3/2}/n \to \infty$ as $k, n \to \infty$, then $\widehat{\beta}_{SWD}(\tau_0)$ is asymptotically consistent for $v'(\tau_0)$.*

We can use the same proof of the asymptotic unbiasedness of $\widehat{\beta}_{SWD}(\tau_0)$ from De Brabanter and Liu (2015) for proving part (i) in Theorem 4.3, since neither the independence nor homoscedasticity assumption is required for showing asymptotic unbiasedness. The proof for part (ii) is also based on De Brabanter and Liu (2015), but we need to transform our problem into their situation where both the independence and homoscedasticity assumption are required; see Appendix E. The proof for Theorem 4.4 is similar to the one for Theorem 4.3.

THEOREM 4.4. *Suppose that the system of interest satisfies* $\mathrm{E}[Y^2(t)] < \infty$ *and its arrival-counting process satisfies Equation (3), and that $v(t)$ is twice continuously differentiable with $v''(t) < \infty$ and* $\sup_{t\in[0,T]} \sigma^2(t) = \sigma^2_{sup} < \infty$. *If $k/n \to 0$ as $k, n \to \infty$, then*

(i) *$\widehat{\beta}_{OLS}(\tau_0)$ is asymptotically unbiased for $v'(\tau_0)$;*
(ii) *if in addition, $k^2/n \to \infty$ as $k, n \to \infty$, then $\widehat{\beta}_{OLS}(\tau_0)$ is asymptotically consistent for $v'(\tau_0)$.*

From Theorems 4.3–4.4, we see that k should not increase faster than n but should not increase too slowly either. The growth rate of k affects the width of the interval $W^{2k}_{SWD}(\tau_0)$. If k grows too slowly, then $W^{2k}_{SWD}(\tau_0)$ might be too narrow such that the observations are too close to each other, which is harmful in derivative estimation. Specifically, the number of nearest neighbors k for $\widehat{\beta}_{SWD}(\tau_0)$ should increase faster than the k for $\widehat{\beta}_{OLS}(\tau_0)$. This is because $\widehat{\beta}_{OLS}(\tau_0)$ uses many more weighted slopes so its variance can be better controlled.

5 PRACTICAL APPROACH

In practice, we need to determine the tuning parameter k to construct good variance and derivative estimators based on finite sample paths. We discuss how to tune the parameter k in this section.

We know $\widehat{\sigma}^2_{RB}(\tau_0)$ is the sample variance of the k nearest neighbors, so it is natural to use the same optimal $k^\star$, denoted by $k^\star_{mean}$, tuned from the mean estimation procedure. Lin et al. (2017) introduce a leave-one-replication-out cross validation (LORO CV) method to obtain $k^\star_{mean}$. For the difference-based variance estimator $\widehat{\sigma}^2_{DB}(\tau_0)$, we suggest two k values: one is simply $k^\star_{mean}$, which we recommend if estimating $v(\tau_0)$ is also of interest; the other, denoted by $k^\star_{db}$, is obtained by tuning k directly without the mean estimation, as described in Algorithm 1. Both $k^\star_{mean}$ and $k^\star_{db}$ are tuned *once*, using all of the output data, and then applied for any $\tau_0 \in [0, T]$.

A simple example to illustrate how Algorithm 1 works is provided in Appendix F. Since the variance is not directly observable, Algorithm 1 does cross validation at a set of user-chosen *test points* by computing the sample variance of observations close to these test points, but all coming from different replications so they are independent. These sample variances then stand in for the usual observed response in typical CV.

Tuning the parameter for $\widehat{\sigma}^2_{DB}(\tau_0)$ via Algorithm 1 is computationally cheaper than what we do for $\widehat{\sigma}^2_{RB}(\tau_0)$. For instance, for a single trial value of k, say k_0, the computational effort required for computing the EMSE(k_0) of $\widehat{\sigma}^2_{DB}(\tau_0)$ is $O(KM_{\text{test}} \log(\sum_{j=1}^n M_j))$, where M_{test} is the number of test points chosen in Algorithm 1. However, the computational effort required by $\widehat{\sigma}^2_{RB}(\tau_0)$ to compute EMSE(k_0) is the same as for virtual mean estimation, which is $O((\sum_{j=1}^n M_j)\log(\sum_{j=1}^n M_j))$. Typically, we have $\sum_{j=1}^n M_j \gg KM_{\text{test}}$, because $\sum_{j=1}^n M_j$ increases rapidly as we increase the number of

ALGORITHM 1: $k$nn method via K-fold cross validation for $\widehat{\sigma}^2_{DB}(\tau_0)$

```
Input fixed test vector $\mathbf{t}_{\text{test}} = \{t_1, t_2, \ldots, t_{M_{\text{test}}}\}$ and search range $k_L < k_U$, NN = "nearest neighbors."
Randomly divide the $n$ replications into $K$ folds of approximately equal size.
for $\ell = 1, 2, \ldots, K$ do
    $S_{\text{test}} \leftarrow \{\mathbf{Y}_j, \mathbf{t}_j; j = 1, 2, \ldots, n_\ell\}$, where $\mathbf{t}_j = \{t_{1j}, t_{2j}, \ldots, t_{M_j j}\}$, $\mathbf{Y}_j = \{Y(t_{1j}), Y(t_{2j}), \ldots, Y(t_{M_j j})\}$, and $n_\ell$ is the number of replications in the $\ell^{th}$ fold.
    $S_{\text{train}} \leftarrow$ all data except $S_{\text{test}}$.
    Find the one nearest neighbor from each $\mathbf{t}_j \in S_{\text{test}}$ for each $t_m \in \mathbf{t}_{\text{test}}$.
    Compute the sample variance $S^2_\ell(t_m)$ using these independent $n_\ell$ observations for each $t_m \in \mathbf{t}_{\text{test}}$.
    Find $k_U$ NN in $S_{\text{train}}$ to each $t_m \in \mathbf{t}_{\text{test}}$.
    Store the indices of the $k_U$ NN to each $t_m \in \mathbf{t}_{\text{test}}$ into an index matrix $\mathbf{M}_{\text{ind}} \in \Re^{M_{\text{test}} \times k_U}$, where the $i$th row in $\mathbf{M}_{\text{ind}}$ contains the indices of the $k_U$ NN to $t_m \in \mathbf{t}_{\text{test}}$.
    for $k \in [k_L, k_U]$ do
        Extract the first $k$ columns from $\mathbf{M}_{\text{ind}}$.
        Find the $k$ NN to each $t_m \in \mathbf{t}_{\text{test}}$ and compute the difference-based estimator $\widehat{\sigma}^2_{DB,\ell}(t_m, k)$.
    end for
end for
for $k \in [k_L, k_U]$ do
    Compute EMSE$(k) = \left(\sum_{\ell=1}^{K} \sum_{m=1}^{M_{\text{test}}} [S^2_\ell(t_m) - \widehat{\sigma}^2_{DB,\ell}(t_m, k)]^2\right) / (M_{\text{test}} \times K)$.
end for
Choose $k^\star_{db}$ that results in the minimum EMSE$(k)$.
```

replications n, or if we have a very dense arrival counting process, while the number of folds K for CV is usually 10 and the number of test points M_{test} is often chosen to be much smaller than the number of arrivals in any repliation M_j.

Taking one queueing system we are going to analyze in Section 6.1 as an example, $\sum_{j=1}^{100} M_j =$ 30,852 while $KM_{\text{test}} = 10 \times 15 = 150$. Hence, if one is only interested in the variance of $V(\tau_0)$, then obtaining a difference-based variance estimator $\widehat{\sigma}^2_{DB}(\tau_0)$ from Algorithm 1 is much cheaper.

As for the two derivative estimators, we propose to use the same optimal $k^\star_{mean}$ value as we use for virtual mean estimation. That is, we use the same $k^\star_{mean}$ nearest neighbors to fit a linear regression model and the estimated slope coefficient is $\widehat{\beta}_{OLS}(\tau_0)$. For $\widehat{\beta}_{SWD}(\tau_0)$, we choose $k^\star_{mean}$ nearest neighbors to τ_0 from each side of τ_0 and then use these $2k^\star_{mean}$ neighbors to compute $\widehat{\beta}_{SWD}(\tau_0)$. Note that there might be fewer than $k^\star_{mean}$ observations (e.g., only $\tilde{k} < k^\star_{mean}$ observations) on one side of τ_0 if τ_0 is close to the end points 0 or T. If that happens, then we only choose $\tilde{k}$ nearest neighbors from each side of τ_0 such that $\widehat{\beta}_{SWD}(\tau_0)$ will be constructed from $\tilde{k}$ slopes.

6 EXPERIMENTS

In this section, we first study a series of phase-type queueing systems for which we can compute the virtual performance measures as controlled studies, and then we apply our methods to the simulated ED problem introduced in Section 1.

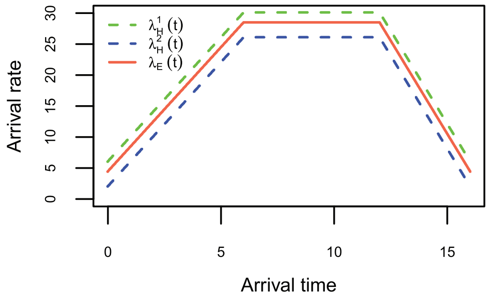

Fig. 2. Arrival rates for the hyperexponential (dashed) and Erlang (solid) arrival processes.

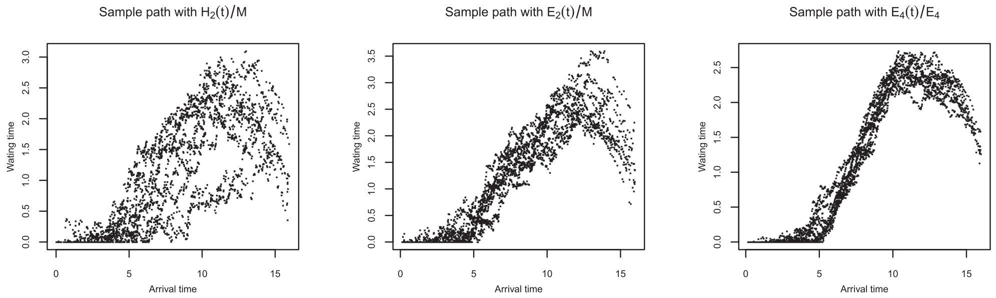

Fig. 3. Sample paths of 10 replications for the three queueing systems.

6.1 Controlled Studies

Lin et al. (2017) study the virtual waiting times for a series of phase-type queueing models to evaluate the performance of $\bar{V}(\tau_0)$. In this article, we use the same phase-type queueing models to evaluate the performance of the proposed variance and derivative estimators.

We study three phase-type FIFO queueing models: $H_2(t)/M/s/c$, $E_2(t)/M/s/c$, and $E_4(t)/E_4/s/c$, where H_2 stands for two-phase hyperexponential distribution, E_2 (or E_4) stands for two-phase (or four-phase) Erlang distribution, and M stands for exponential distribution. The nonstationary arrival rate functions are piecewise linear (see Figure 2), the service rate is $\mu = 20$, the number of servers is $s = 1$, the system capacity is $c = 50$, and the mixing probability p within the $H_2(t)$ distribution is 0.4. Sample paths from ten replications, which illustrate the trend and variability for these three systems, are shown in Figure 3. We see that the actual waiting time for each arrival varies over time for all three systems, which indicates the nonstationarity of the systems. Taking $E_4(t)/E_4/1/c$ as an example, the waiting time for many arrivals occurring from $t = 0$ to $t = 5$ is 0 while the waiting time could be as long as 2–2.5min for the arrivals that occur from $t = 10$ to $t = 15$. However, the long-run average waiting time is about 1.3min. Therefore, it would be misleading to use the long-run average waiting time as an estimator for the mean virtual waiting time at a specific $t = \tau_0$, which is why we need virtual statistics.

The reason we choose these phase-type queueing models for the empirical study is that we can compute the virtual performance measures of interest. Lin et al. (2017) describe how to compute the expected value of virtual waiting time using Kolmogorov forward equations (KFEs), and we can compute the variance and derivative based on the same technique. Refer to Appendix G for details. *Overall, the proposed variance and derivative estimators turn out to estimate the true values very well for all three systems.*

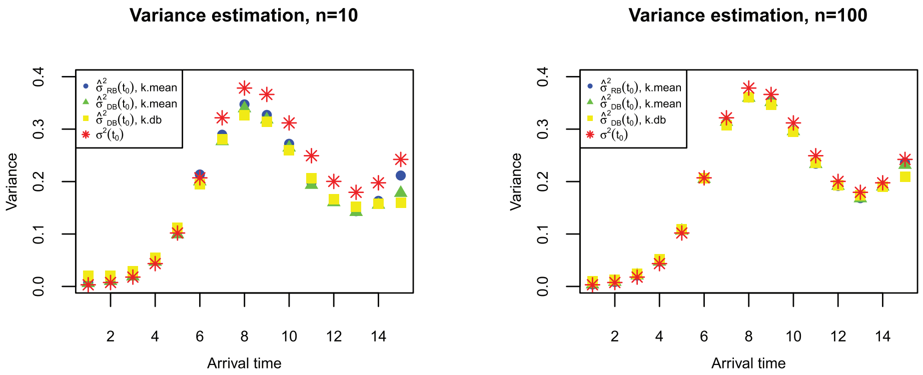

Fig. 4. Performance of $\widehat{\sigma}^2_{RB}(\tau_0)$ vs. $\widehat{\sigma}^2_{DB}(\tau_0)$ for $H_2(t)/M/1/c$ system.

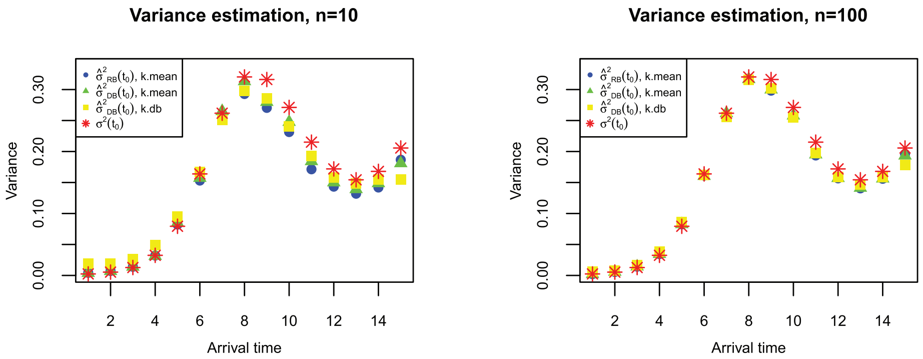

Fig. 5. Performance of $\widehat{\sigma}^2_{RB}(\tau_0)$ vs. $\widehat{\sigma}^2_{DB}(\tau_0)$ for $E_2(t)/M/1/c$ system.

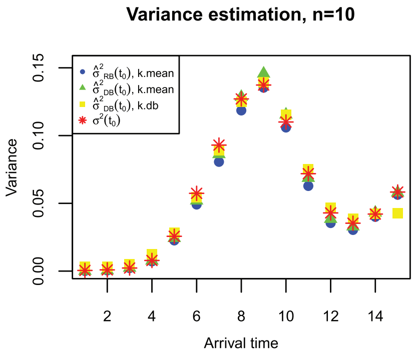

Fig. 6. Performance of $\widehat{\sigma}^2_{RB}(\tau_0)$ vs. $\widehat{\sigma}^2_{DB}(\tau_0)$ for $E_4(t)/E_4/1/c$ system.

We first present the simulation results for the variance estimators. In Section 5, we have discussed how to choose appropriate k values for the $k$nn variance estimators. We use $k^\star_{mean}$ tuned from LORO CV for $\widehat{\sigma}^2_{RB}(\tau_0)$; and we try two k values for $\widehat{\sigma}^2_{DB}(\tau_0)$: one is $k^\star_{mean}$ and the other is $k^\star_{db}$ tuned directly from Algorithm 1. The performance of the variance estimators averaged across 100 macro-replications for the three systems are presented in Figures 4–6, where n indicates the number of replications within each macro-replication. In the figures, ● corresponds to $\widehat{\sigma}^2_{RB}(\tau_0)$

using $k^\star_{mean}$; ▲ corresponds to $\widehat{\sigma}^2_{DB}(\tau_0)$ using $k^\star_{mean}$; ■ corresponds to $\widehat{\sigma}^2_{DB}(\tau_0)$ using $k^\star_{db}$; and $*$ corresponds to the true value of $\sigma^2(\tau_0)$.

From the figures we can see that the performance of the variance estimators is good overall in the sense that they track the true variances; in fact, the difference-based variance estimators tend to be very close to the true variance when the system has low variability, as in Figure 6. However, all of the variance estimators are a little bit biased when the system is highly variable and n is small, which manifests itself by under estimating the true variance. For example, both the $H_2(t)/M/1/c$ and $E_2(t)/M/1/c$ queues are more variable than the $E_4(t)/E_4/1/c$, as shown in Figure 3, especially during the time period of $t = 7$ to $t = 10$. When only 10 replications are used we find that all of the variance estimators tend to be lower than the true variance in this time period. However, the bias largely disappears as n increases from 10 to 100; see Figures 4 and 5. Some bias is inevitable in virtual statistics, since observing an arrival at exactly τ_0 is almost certain not to happen; the goal of cross validation is to balance bias and variance.

In these examples, we find both $k^\star_{mean}$ and $k^\star_{db}$ are larger than the number of replications n so there always exist dependence among the k nearest neighbors. Thus, the variance estimators tend to underestimate the variance due to positive correlation, especially when the system has high variability; in other words, the more variable the system is, the more biased the variance estimators could be if the number of replications n is too small. This is because the optimal k value tuned from either LORO CV or K-fold CV is larger when the system has higher variability (e.g., $k^\star_{mean} \approx 200$ for $H_2(t)/M/1/c$ and $k^\star_{mean} \approx 50$ for $E_4(t)/E_4/1/c$ when $n = 10$ for both of these two systems). Thus, the dependence issue is more severe for the more variable system. Nevertheless, we still provide good estimates—as shown in the figures—even in the presence of this dependence, which means CV is effectively accounting for it.

Remark. Notice that even though it is possible for $k^\star_{mean}$ and $k^\star_{db}$ to be greater than the number of replications, n, they will never be greater than the total number of elementary observations $Y(t_{ij})$, because we fit them via CV; i.e., we cannot test a value of k larger than the number of elementary observations we have. Eventually, as n becomes large enough, we will find $k^\star < n$, because bias will dominate variance.

To assess the variability of these variance estimators, we ran $R = 100$ macro-replications for all scenarios. Take $\widehat{\sigma}^2_{DB}(\tau_0)$ as an example: We estimate its variance by $\sum_{r=1}^{R}[\widehat{\sigma}^2_{DB,r}(\tau_0) - \overline{\widehat{\sigma}^2_{DB}}(\tau_0)]^2/(R-1)$, where $\widehat{\sigma}^2_{DB,r}(\tau_0)$ is the difference-based variance estimator computed from the rth macro-replication and $\overline{\widehat{\sigma}^2_{DB}}(\tau_0) = \sum_{r=1}^{R} \widehat{\sigma}^2_{DB,r}(\tau_0)/R$. We find that the variance of $\widehat{\sigma}^2_{RB}(\tau_0)$ is very close to the variance of $\widehat{\sigma}^2_{DB}(\tau_0)$ with $k^\star_{mean}$. Even though $\widehat{\sigma}^2_{RB}(\tau_0)$ includes many more pairs of observations, $\widehat{\sigma}^2_{DB}(\tau_0)$ removes the trend in the mean response function such that the variance caused by the regression function can be effectively reduced. As for the other $k$nn difference-based variance estimator, $\widehat{\sigma}^2_{DB}(\tau_0)$ with $k^\star_{db}$, the optimal $k^\star_{db}$ tuned from Algorithm 1 is much larger than $k^\star_{mean}$. Hence, the variance of $\widehat{\sigma}^2_{DB}(\tau_0)$ with $k^\star_{db}$ can be further reduced and it is smaller than the variance of the other two estimators.

The performance of the derivative estimators is provided in Figures 7–9. In the figures, ● corresponds to $\widehat{\beta}_{OLS}(\tau_0)$; ▲ corresponds to $\widehat{\beta}_{SWD}(\tau_0)$; and $*$ corresponds to the true value of the derivative. Overall, both $\widehat{\beta}_{SWD}(\tau_0)$ and $\widehat{\beta}_{OLS}(\tau_0)$ estimate the true derivative well, but $\widehat{\beta}_{SWD}(\tau_0)$ performs better than $\widehat{\beta}_{OLS}(\tau_0)$ in terms of both bias and variance. Specifically, we find that $\widehat{\beta}_{OLS}(\tau_0)$ is more biased when the variability dominates the trend in the system, e.g., during the time period of $t = 7$ to $t = 10$. This is because the true regression function is not necessarily a linear function and $\widehat{\beta}_{OLS}(\tau_0)$ assigns non-zero weight to every single pair of $(\tau_0^{(\ell,n)}, \tau_0^{(m,n)})$ for $(\ell, m) \in \mathcal{D}_{OLS}(\tau_0)$ so

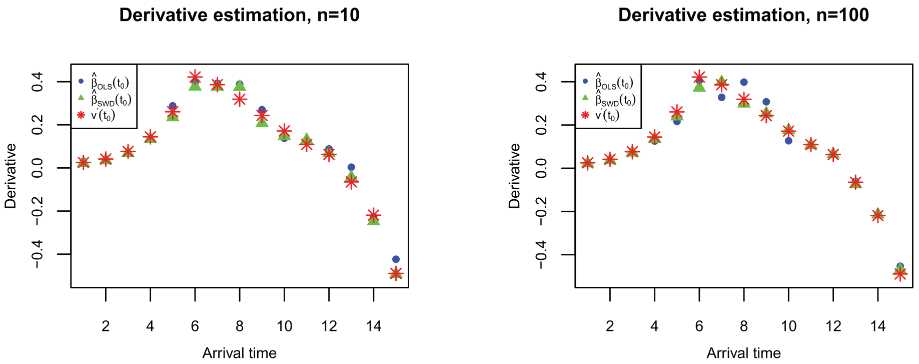

Fig. 7. Performance of $\widehat{\beta}_{OLS}(\tau_0)$ vs. $\widehat{\beta}_{SWD}(\tau_0)$ for $H_2(t)/M/1/c$ system.

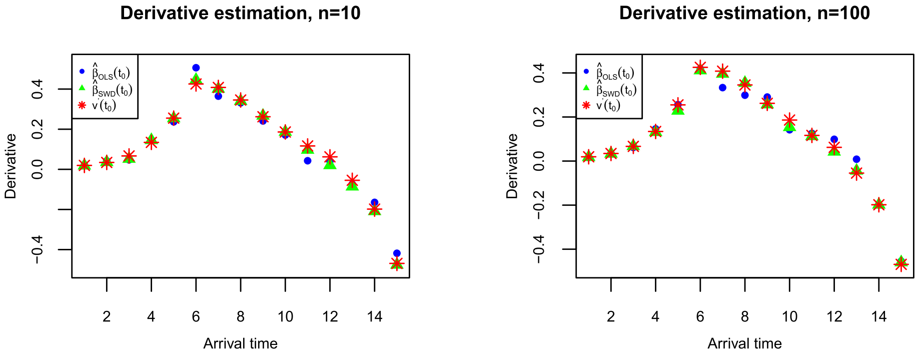

Fig. 8. Performance of $\widehat{\beta}_{OLS}(\tau_0)$ vs. $\widehat{\beta}_{SWD}(\tau_0)$ for $E_2(t)/M/1/c$ system.

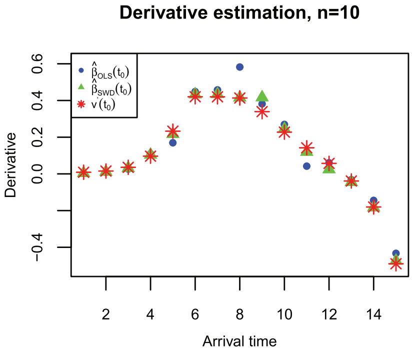

Fig. 9. Performance of $\widehat{\beta}_{OLS}(\tau_0)$ vs. $\widehat{\beta}_{SWD}(\tau_0)$ for $E_4(t)/E_4/1/c$ system.

that the bias is very likely to be increased due to lack of symmetry. We also estimated the variance of these estimators from $R = 100$ macro-replications of the experiment: even though $\widehat{\beta}_{OLS}(\tau_0)$ includes many more slopes, the slopes used in $\widehat{\beta}_{SWD}(\tau_0)$ are less variable and less biased, because the pairs $(\tau_{0,SWD}^{(\ell,n)}, \tau_{0,SWD}^{(m,n)})$ are well spread and symmetric around τ_0, so $\widehat{\beta}_{SWD}(\tau_0)$ has smaller variance.

In contrast to variance estimation, positive correlation is not that harmful for derivative estimation. Think about an extreme case where the true mean waiting time is a linear function of

Table 1. MSE of Variance and Derivative Estimators for $E_2(t)/M/1/c$ with $n = 100$

	True	MSE			True	MSE	
τ_0	$\sigma^2(\tau_0)$	$\widehat{\sigma}^2_{RB}(\tau_0)$	$\widehat{\sigma}^2_{DB}(\tau_0)$ with $k^\star_{mean}$	$\widehat{\sigma}^2_{DB}(\tau_0)$ with $k^\star_{db}$	$v'(\tau_0)$	$\widehat{\beta}_{OLS}(\tau_0)$	$\widehat{\beta}_{SWD}(\tau_0)$
1	0.0024	9.60×10^{-7}	9.35×10^{-7}	1.37×10^{-5}	0.0195	3.76×10^{-4}	8.03×10^{-5}
2	0.0054	4.09×10^{-6}	4.44×10^{-6}	9.02×10^{-6}	0.0344	1.77×10^{-3}	3.60×10^{-4}
3	0.0128	1.92×10^{-5}	1.95×10^{-5}	3.18×10^{-5}	0.0668	6.01×10^{-3}	1.49×10^{-3}
4	0.0325	6.70×10^{-5}	6.97×10^{-5}	8.84×10^{-5}	0.1343	2.76×10^{-2}	4.33×10^{-3}
5	0.0794	2.66×10^{-4}	2.59×10^{-4}	2.64×10^{-4}	0.2551	5.93×10^{-2}	1.09×10^{-2}
6	0.1639	8.51×10^{-4}	8.99×10^{-4}	6.11×10^{-4}	0.4254	1.34×10^{-1}	2.51×10^{-2}
7	0.2618	1.65×10^{-3}	1.86×10^{-3}	1.13×10^{-3}	0.4081	3.32×10^{-1}	3.87×10^{-2}
8	0.3204	1.60×10^{-3}	1.87×10^{-3}	1.40×10^{-3}	0.3455	4.10×10^{-1}	3.59×10^{-2}
9	0.3162	2.15×10^{-3}	2.34×10^{-3}	1.58×10^{-3}	0.2621	2.64×10^{-1}	4.30×10^{-2}
10	0.2712	1.81×10^{-3}	2.15×10^{-3}	1.51×10^{-3}	0.1862	1.67×10^{-1}	2.95×10^{-2}
11	0.2151	1.75×10^{-3}	1.72×10^{-3}	1.32×10^{-3}	0.1166	1.98×10^{-1}	2.51×10^{-2}
12	0.1720	1.14×10^{-3}	1.22×10^{-3}	8.67×10^{-4}	0.0623	1.36×10^{-1}	1.56×10^{-2}
13	0.1544	7.27×10^{-4}	7.11×10^{-4}	5.02×10^{-4}	−0.0546	6.03×10^{-2}	1.18×10^{-2}
14	0.1680	8.38×10^{-4}	8.85×10^{-4}	6.64×10^{-4}	−0.1982	5.07×10^{-2}	5.99×10^{-3}
15	0.2057	1.24×10^{-3}	1.32×10^{-3}	1.55×10^{-3}	−0.4689	2.48×10^{-2}	4.28×10^{-3}

time and all the k nearest neighbors are from a single replication. If these k nearest neighbors are perfectly correlated (i.e., $\rho = 1$), then the derivative estimator is unbiased and has zero variance but the variance estimator will be very poor. Thus, the positive correlation actually improves the performance of the derivative estimators in this situation.

In addition to the graphical presentation, we also display the *mean squared error* (MSE) for one case, the $E_2(t)/M/1/c$ system; see Table 1. Overall, the MSE of all the variance estimators and the $\widehat{\beta}_{SWD}(\tau_0)$ derivative estimator are at least an order of magnitude smaller than the quantity being estimated. Notice that $\widehat{\beta}_{SWD}(\tau_0)$ has substantially smaller MSE than $\widehat{\beta}_{OLS}(\tau_0)$ for some τ_0, which is what we expect due to the symmetry of the observations involved in $\widehat{\beta}_{SWD}(\tau_0)$.

To better interpret the simulation results, we choose $\tau_0 = 6$ for $E_2(t)/M/1/c$ system as an illustration. If a customer arrives to this system at $\tau_0 = 6$, then the mean estimate of the waiting time in the queue for this customer is 0.71min (obtained from Lin et al. (2017)), and the variance estimate for the waiting time is 0.16, i.e., the standard deviation is 0.4min (Figure 5). The SWD estimate is $\widehat{\beta}_{SWD}(6) \approx 0.45$, meaning that the rate of change in the waiting time at $\tau_0 = 6$ is 0.45min per time unit, so very rapidly changing relative to the mean of 0.71min.

6.2 Emergency Department Example

Section 6.1 presents a series of controlled studies in which we can evaluate the performance of virtual statistics by comparing them with the true values of virtual performances. In this section, we describe a small ED model as a more realistic illustration and apply our proposed virtual statistics.[2]

This ED model consists of a waiting area, a registration desk, a triage room, a radiology station, a billing area, six beds, and four rooms that are used for patients that are admitted into the hospital. Patients enter the ED through the front door and go directly to the registration desk. If the arriving patient finds that there are more than six patients waiting in the triage room or the current average

[2] This model is adapted from `HospitalEmergencyDepartment.spfx`, a standard example that is contained in the Simio simulation software (`www.simio.com`).

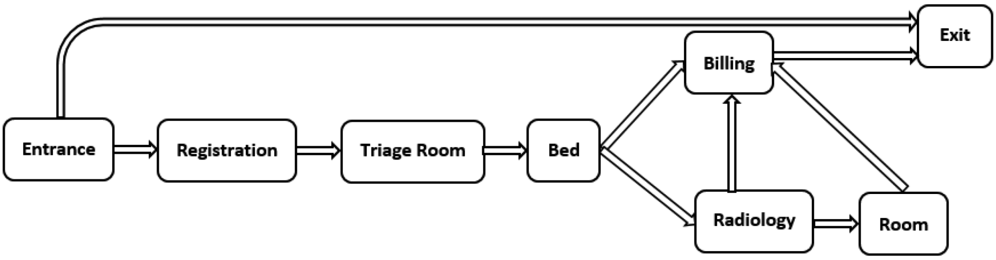

Fig. 10. Emergency department model.

waiting time to get a bed is longer than 0.6h, then they will leave this ED department immediately; otherwise, they will stay and get registered. After they are registered, they wait in the waiting area to be sent to the triage room. After they are seen in the triage room, they wait in the waiting area for an available bed. When a bed and a nurse are both available, the nurse greets the patient and walks them to the bed. The patient waits in the bed for an available doctor, who is accompanied by a nurse. They finish treating the patient and the patient either visits billing before exiting the ED or is sent to the radiology room. If a patient is seen at radiology, then they are either sent to billing and then sent home, or admitted into a room for a longer stay. Once admitted into a room, the patient is again visited by a doctor and a nurse and then released to billing and then home. Figure 10 illustrates this entire process.

The patient arrivals follow a nonstationary Poisson process with piecewise constant arrival rates. In our simulation, the arrivals are generated from $t = 0$ to $t = 24$ and the arrival rate changes hourly. The service times at different stations follow triangular distributions with different parameters, and this ED has six nurses and four doctors. This is a complicated stochastic process with many working stations and nonstationarity, and we are interested in three particular virtual performance measures:

(i) the virtual waiting time to get a bed for a patient who arrives at τ_0 and does not leave, denoted by $V_{wait}(\tau_0)$;
(ii) the virtual time in system for a patient who arrives at τ_0 and is admitted to a room, denoted by $V_{TIS}(\tau_0)$; and
(iii) the virtual probability that a patient will not leave immediately if arriving at τ_0, denoted by $V_{prob}(\tau_0)$.

The first two virtual performance measures, $V_{wait}(\tau_0)$ and $V_{TIS}(\tau_0)$, target different types of patients. Specifically, $V_{wait}(\tau_0)$ applies to all the patients who stay in the system while $V_{TIS}(\tau_0)$ only applies to patients who are admitted to a room for further treatment after radiology. The last virtual performance $V_{prob}(\tau_0)$ is different from all the virtual performances we have studied before, because it is a probability. This information is quite useful, because it indicates the fraction of patients who fail to receive treatment due to the limited resources at the ED. For all these virtual performance measures, we will present the $k$nn mean estimator, the two variance estimators, and the two derivative estimators.

We start with the first two virtual performance measures, $V_{wait}(\tau_0)$ and $V_{TIS}(\tau_0)$, since they are both time-related performance measures. We choose 24 time points at $t = $ 0:30, 1:30, . . . , 23:30 to test. Figure 11 shows the 10-replication sample paths of these two performance measures, which displays the nonstationarity of the system. The $k$nn mean estimators, denoted by $\bar{V}_{wait}(\tau_0)$ and $\bar{V}_{TIS}(\tau_0)$, are obtained from 100 replications of data, as shown in Figure 12. The corresponding

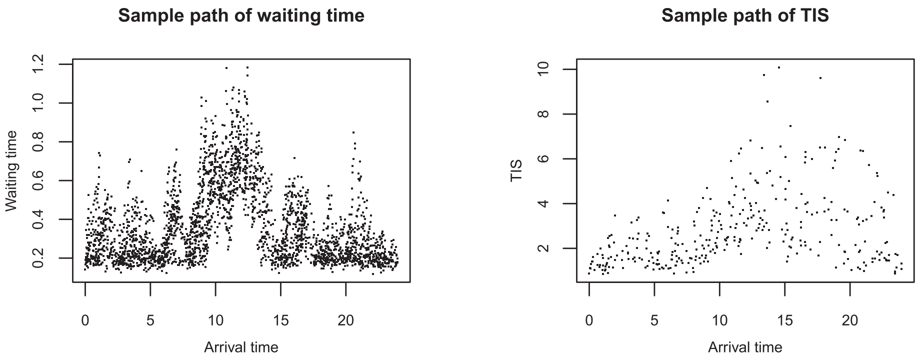

Fig. 11. Ten-replication sample paths of waiting time and TIS.

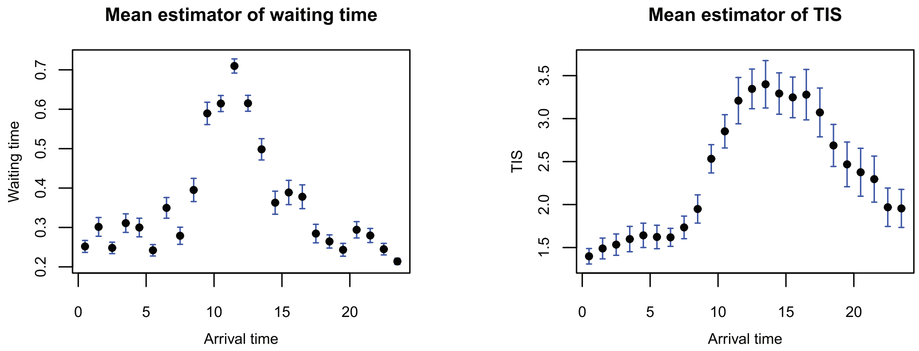

Fig. 12. $k$nn mean estimators of waiting time and TIS.

error bars are constructed on $\pm 2\hat{\tau}^{\star}$, where $\hat{\tau}^{\star^2}$ is the bootstrap variance estimator for the $k$nn mean estimator; refer to Lin et al. (2017) for more details on the bootstrap variance estimator. Although we do not have the true values of $v_{wait}(\tau_0)$ and $v_{TIS}(\tau_0)$, we can see that the $k$nn estimators capture the trend of the system well. Notice that we should tune the parameter k separately for these two virtual statistics. The optimal k tuned via LORO CV for $\bar{V}_{wait}(\tau_0)$ is $k^{\star}_{wait} = 651$ and the one for $\bar{V}_{TIS}(\tau_0)$ is $k^{\star}_{TIS} = 232$. We find $k^{\star}_{wait} > k^{\star}_{TIS}$, because the patients who are admitted to a room are simply a subset of all the patients who do not leave immediately upon their arrivals. Because the arrivals of the patients admitted to a room are less dense, the corresponding $k$nn mean estimators are more variable.

In addition to the mean, we compute both the residual-based and difference-based variance estimators to estimate $\text{Var}[V_{wait}(\tau_0)]$ and $\text{Var}[V_{TIS}(\tau_0)]$. For the difference-based variance estimator $\hat{\sigma}^2_{DB}(\tau_0)$, we use the optimal $k^{\star}$ obtained via LORO CV directly, because we find that $\hat{\sigma}^2_{DB}(\tau_0)$ does not change much with different k values in the controlled studies. Figure 13 shows the variance estimators for these two virtual performance measures, from which we see that $\hat{\sigma}^2_{RB}(\tau_0)$ and $\hat{\sigma}^2_{DB}(\tau_0)$ have very similar results. In the figures, the ▲ corresponds to $\widehat{\sigma}_{RB}(\tau_0)$; and the ● corresponds to $\widehat{\sigma}^2_{DB}(\tau_0)$.

The derivative estimation is more complicated as we notice that the derivative estimators, especially the ordinary least squares estimator $\widehat{\beta}_{OLS}(\tau_0)$, are highly variable such that the two derivative estimators might have very different values at some time points. Therefore, we report two cases obtained from two different data sets (i.e., macro replications): one is a good case where $\widehat{\beta}_{OLS}(\tau_0)$

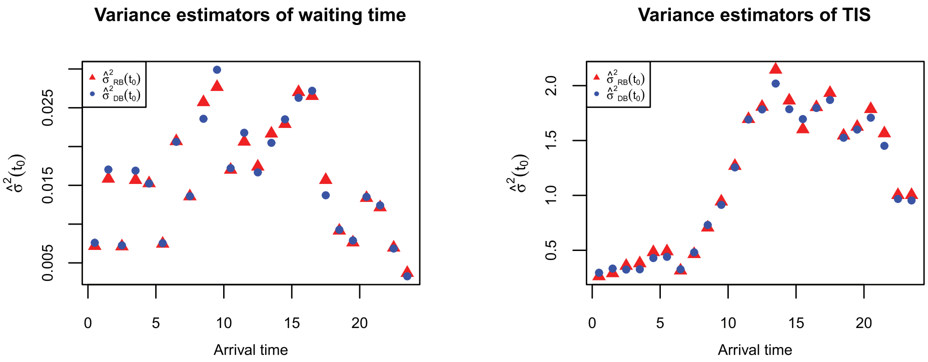

Fig. 13. Variance estimators of waiting time and TIS.

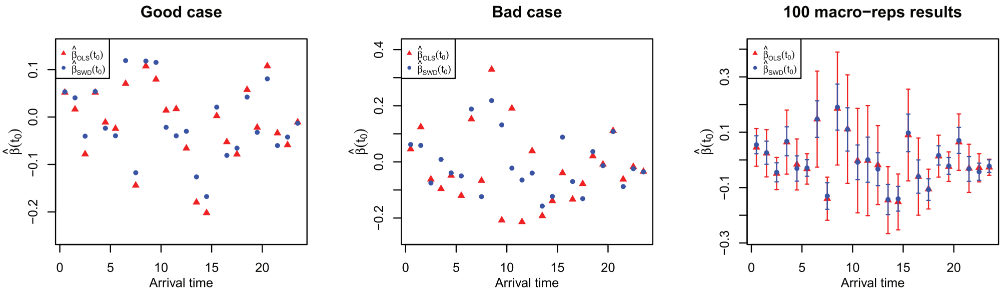

Fig. 14. Two cases of derivative estimators of waiting time and the 100-macro-replications result.

and $\widehat{\beta}_{SWD}(\tau_0)$ are close to each other, and the other is a bad case where $\widehat{\beta}_{OLS}(\tau_0)$ and $\widehat{\beta}_{SWD}(\tau_0)$ differ a lot at some time points. The simulation results for the virtual waiting time derivatives are offered in Figure 14. In the figures, the ▲ corresponds to $\widehat{\beta}_{OLS}(\tau_0)$ and the ● corresponds to $\widehat{\beta}_{SWD}(\tau_0)$.

Even in the bad situations, the two derivative estimators are close to each other at most of the time points, and the big discrepancy occurs at $t = 9{:}30, 10{:}30, 11{:}30$. From the sample paths in Figure 11, we see that the variability of the system dominates the trend from $t = 9{:}00$ to $t = 12{:}00$. The high variability in this time period is also reflected in the variance estimation results shown in Figure 13. Since the derivative estimators are more variable, we ran a side-experiment with 100 macro-replications and report the average derivative estimators with ±2 standard errors in Figure 14. We see that the two types of derivative estimators averaged across 100 macro-replications become much closer to each other, and the SWD estimator is less variable than the OLS estimator, as we discussed in Section 6.1. Similar to the results we presented for waiting time, we present two cases of single-macro-replication results and a 100-macro-replications result for the derivative estimators of true TIS in Figure 15. We see that the big discrepancy of the two derivative estimators occurs at $t = 15{:}30$ in the bad case, which makes sense, because the sample paths of TIS in Figure 11 indicate that TIS is more variable in the afternoon. Furthermore, we notice that $\widehat{\beta}_{SWD}(\tau_0)$ is more variable than $\widehat{\beta}_{OLS}(\tau_0)$ at the two endpoints $t = 0{:}30, 23{:}30$ for TIS from the third plot of Figure 15. This is because the arrivals of admitted patients are not dense enough to generate at least $k^{\star}_{TIS}$ pairs of symmetric observations around these two endpoints. Thus, the SWD estimators at the endpoints are constructed on fewer pairs of observations so that the corresponding variability becomes higher.

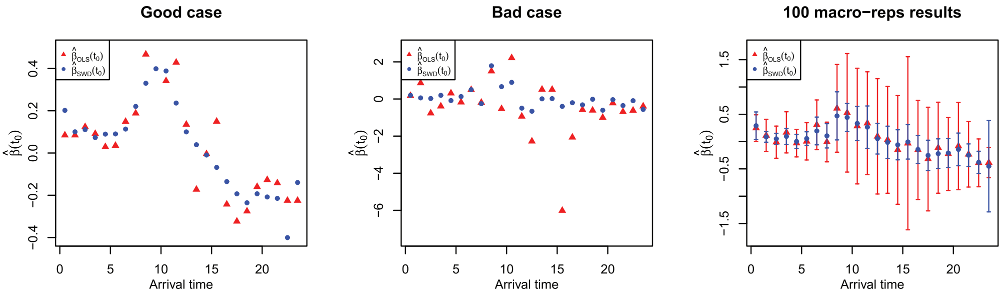

Fig. 15. Two cases of derivative estimators of TIS and the 100-macro-replications result.

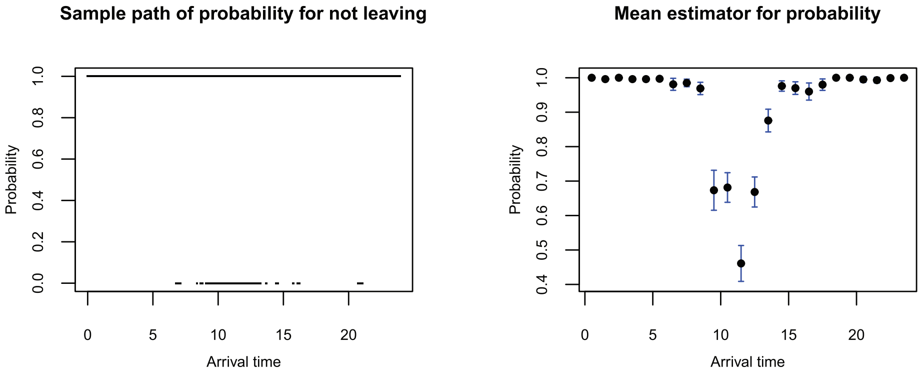

Fig. 16. 10-replication sample path and $k$nn mean estimator of the probability that patients do not leave immediately.

The 10-replication sample path for indicating whether or not the patients stays in the system upon their arrivals is given in Figure 16, where 1 means a patient stays in the system and 0 means a patient leaves immediately. We can see that most of the patients who come early or late in the day stay in the system while many patients arriving around noon have to leave, because the ED is much more congested during that time. Based on 100 replications of data, we apply LORO CV to obtain the optimal $k^{\star}_{prob} = 1{,}060$, and then construct the $k$nn estimators and the associated bootstrap variance estimators. The reason $k^{\star}_{prob} > k^{\star}_{wait}$ is that the patients who wait for a bed are a subset of all the patients generated from the arrival process. From Figure 16, we find that the probability for a patient to stay in the system is quite high in the morning and evening time and the variance of the mean estimator is very close to 0, but the patient is more likely to leave immediately if arriving around noon. For example, the probability that a patient arriving at 11:30 leaves the ED immediately is 0.55, and the $k$nn probability estimator also becomes more variable due to the variability of the system during this time.

We use the same optimal $k^{\star}_{prob} = 1{,}060$ to compute the two variance estimators, and they have very similar performance, as shown in Figure 17. As for the derivative estimation, we still report two specific cases as well as the 100-macro-replications result. Similar to what we have found previously for the two virtual performance measures, we see that the big discrepancy between these two derivative estimators of the probability still only occurs at a couple of time points in the bad case, as shown in Figure 18. Additionally, both these two derivative estimators become more variable from 9:30 to 12:30, especially the OLS estimator.

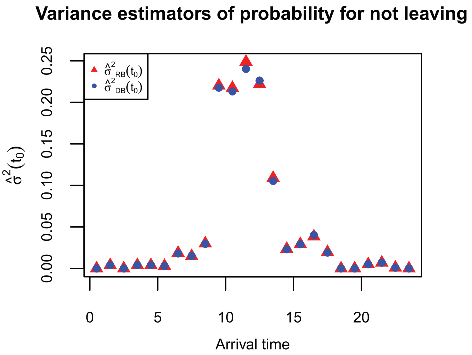

Fig. 17. Variance estimators of the probability that patients do not leave immediately.

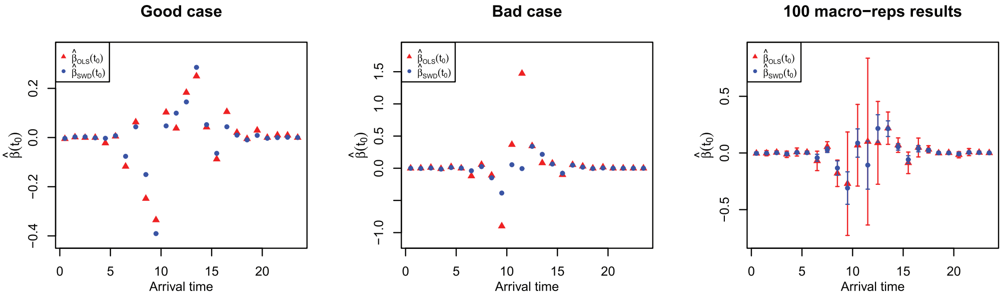

Fig. 18. Two cases of derivative estimators of probability that patients do not leave immediately and the 100-macro-replications result.

From this example, we see that our virtual statistics can be applied easily to a complicated real-world simulation, and we can also examine various virtual performance measures at the same time without running additional simulation experiments.

7 CONCLUSIONS

Virtual performance measures add insight into system performance that goes beyond the usual suite of long-run-average performance measures generated by stochastic simulations: They facilitate a time-indexed profile of system performance, which is particularly relevant for non-stationary or finite-horizon situations.

In this article, we propose two variance estimators and two derivative estimators for virtual performance based on retained sample paths from simulation experiments, as an addition to the virtual mean estimator of Lin and Nelson (2016) and Lin et al. (2017). We show the asymptotic properties of these new virtual statistics and propose a parameter tuning algorithm for the $k$nn difference-based variance estimator. The controlled studies show that employing a single, globally optimal $k^{\star}_{mean}$ obtained via cross validation for mean estimation works for both virtual variance and derivative estimation as well. Thus, estimating the mean, derivative of the mean with respect to time, and the variance of virtual performance can be done efficiently from the retained output data. However, allowing $k^{\star}$ to be a function of time—say larger during time periods when the simulation output is more variable and smaller when the mean is changing more rapidly—could lead to further improvement and is a topic of ongoing research.

REFERENCES

Kris De Brabanter, Jos De Brabanter, Bart De Moor, and Irène Gijbels. 2013. Derivative estimation with local polynomial fitting. *J. Mach. Learn. Res.* 14, 1 (2013), 281–301.

Kris De Brabanter and Yu Liu. 2015. Smoothed nonparametric derivative estimation based on weighted difference sequences. In *Stochastic Models, Statistics and Their Applications*. Springer, 31–38.

Theo Gasser and Hans-Georg Müller. 1979. *Nonparametric Estimation of Regression Functions and their Derivatives*. Sonderforschungsbereich University, 123.

Theo Gasser and Hans-Georg Müller. 1984. Estimating regression functions and their derivatives by the kernel method. *Scand. J. Stat.* 11, 3 (1984), 171–185.

Theo Gasser, Lothar Sroka, and Christine Jennen-Steinmetz. 1986. Residual variance and residual pattern in nonlinear regression. *Biometrika* 73, 3 (1986), 625–633.

Elia Liitiäinen, Francesco Corona, and Amaury Lendasse. 2008. On nonparametric residual variance estimation. *Neural Process. Lett.* 28, 3 (2008), 155–167.

Elia Liitiäinen, Francesco Corona, and Amaury Lendasse. 2010. Residual variance estimation using a nearest neighbor statistic. *J. Multivar. Anal.* 101, 4 (2010), 811–823.

Yujing Lin and Barry L. Nelson. 2016. Simulation analytics for virtual statistics via k nearest neighbors. In *Proceedings of the 2016 Winter Simulation Conference*. IEEE Press, 448–459.

Yujing Lin and Barry L. Nelson. 2017. Variance and derivative estimation for virtual performance in simulation analytics. In *Proceedings of the 2017 Winter Simulation Conference*. IEEE Press, 1856–1867.

Yujing Lin, Barry L. Nelson, and Linda Pei. 2017. *Virtual Statistics in Simulation via k Nearest Neighbors*. Technical Report. Northwestern University.

John Rice. 1984. Bandwidth choice for nonparametric regression. *Ann. Stat.* 12, 4 (1984), 1215–1230.

Shanggang Zhou and Douglas A. Wolfe. 2000. On derivative estimation in spline regression. *Statistica Sinica* 10, 1 (2000), 93–108.

Received September 2017; revised February 2018; accepted April 2018