INFORMS Journal on Computing

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Estimating Cycle Time Percentile Curves for Manufacturing Systems via Simulation

Feng Yang, Bruce E. Ankenman, Barry L. Nelson,

To cite this article:
Feng Yang, Bruce E. Ankenman, Barry L. Nelson, (2008) Estimating Cycle Time Percentile Curves for Manufacturing Systems via Simulation. INFORMS Journal on Computing 20(4):628-643. http://dx.doi.org/10.1287/ijoc.1080.0272

INFORMS Journal on Computing
Vol. 20, No. 4, Fall 2008, pp. 628–643
ISSN 1091-9856 | EISSN 1526-5528 | 08 | 2004 | 0628

DOI 10.1287/ijoc.1080.0272

Estimating Cycle Time Percentile Curves for Manufacturing Systems via Simulation

Feng Yang
Industrial and Management Systems Engineering Department, West Virginia University, Morgantown, West Virginia 26506, feng.yang@mail.wvu.edu

Bruce E. Ankenman, Barry L. Nelson
Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, Illinois 60208 {ankenman@northwestern.edu, nelsonb@northwestern.edu}

Cycle time-throughput (CT-TH) percentile curves quantify the relationship between percentiles of cycle time and factory throughput, and they can play an important role in strategic planning for manufacturing systems. In this paper, a highly flexible distribution, the generalized gamma, is used to represent the underlying distribution of cycle time. To obtain CT-TH percentile curves, we use a factory simulation to fit metamodels for the first three CT-TH moment curves throughout the throughput range of interest, determine the parameters of the generalized gamma by matching moments, and obtain any percentile of interest by inverting the distribution. To insure efficiency and control estimation error, simulation experiments are built up sequentially using a multistage procedure. Numerical results are presented to demonstrate the effectiveness of the approach.

Key words: discrete event simulation; response surface modeling; design of experiments; semiconductor manufacturing; queueing

History: Accepted by Marvin Nakayama, Area Editor for Simulation; received May 2007; revised August 2007, December 2007; accepted January 2008. Published online in *Articles in Advance* July 7, 2008.

1. Introduction

Planning for manufacturing, either at the factory or enterprise level, requires answering "what-if" questions involving (perhaps a very large number of) different scenarios for product mix, production targets, and capital expansion. Computer simulation is an essential tool for the design and analysis of complex manufacturing systems. Often, before a new system is deployed or changes are made to an existing system, a simulation model will be created to predict the system's performance. Even when no substantial changes are envisioned, simulation is used to allocate capacity among production facilities. In either case, simulation is faster and much more cost effective than experimenting with the physical system (when that is even possible). This is especially true in the semiconductor industry, which is the motivating application for this research (see, for instance, Schömig and Fowler 2000).

Simulation is popular because it can incorporate any details that are important, and the now-common practice of animating simulations means that they have a face validity that a system of equations can never hope to achieve. However, simulation can be a clumsy tool for planning: simulation can only evaluate one scenario at a time, and depending on the complexity of the system and the details of the simulation model, it may take several minutes or even hours to complete a simulation run for each scenario. This can lead to fewer questions being asked and fewer options being considered, especially when scenarios are discussed and debated in real time.

To make simulation an effective tool for planning, our approach is to use simulation to parameterize sophisticated response surface models (RSMs) that are easily explored or optimized with respect to the controllable decision variables. Although it might take significant simulation time to build the RSM, once it has been generated, the RSM is instantly able to answer what-if questions in real time (e.g., at a quarterly production planning meeting that only lasts an hour). Analytically tractable queueing models or approximations can also produce such surfaces, but they invariably require significant simplification of the actual manufacturing system. Our RSMs, generated by simulation, excel in the sense that they are relatively simple formulas like those provided by queueing models but are fitted to a high-fidelity simulation.

In this paper, we propose a simulation-based methodology to quantify the relationship between percentiles of steady-state cycle time (CT) and throughput (TH). Cycle time is defined as a random variable representing the time required for a job or lot to traverse a given routing in a production system

(Hopp and Spearman 2001). A planner can control cycle time by controlling the rate at which lots are started in the factory (lot start rate or equivalently, throughput rate). A CT-TH percentile curve is simply a given percentile of the cycle time distribution as a function of the throughput, and it can be used to discuss the trade-offs of lead time versus throughput. For instance, if the 95th percentile CT-TH curve is used to set the throughput level, then 95% of the time, the actual lead time of any given product will meet the promised delivery time. Hence, such CT-TH percentile curves can play an important role in strategic planning for manufacturing. They may be used to answer questions like: What throughput would this system be able to sustain if the lead times were quoted to be four weeks for a particular product? How much additional throughput could be generated if the lead time was quoted at six weeks? Do we have sufficient production capacity to satisfy customer demands, and how should we distribute the production among facilities? The curves are not designed for making detailed order-release decisions, however.

We perform a sequence of simulation experiments to characterize the cycle time distribution as a function of the throughput. The goal is to provide a methodology that requires nothing of the user beyond (1) the simulation model; (2) a throughput range of interest, say $[x_L, x_U]$ (the throughput has been rescaled so that $0 < x_L < x_U < 1$, where 1 is the factory capacity); (3) a percentile range of interest, say $[\alpha_L, \alpha_U]$ $(0 < \alpha_L < \alpha_U < 1$, where 1 corresponds to 100%); and (4) a measure of the required precision for the estimated curves. The result is a complete response profile that quantifies the relationship of percentiles of cycle time to throughput rate.

In our procedure, the precision of the percentile estimates is selected by the user and is expressed as a relative error (e.g., 5% or 10%). Here, "precision" only refers to the estimated percentiles of the simulated cycle time. The validity of the simulation model itself, although of great importance, is beyond the scope of this research. We assume that the company is satisfied that the simulation model is sufficiently detailed to provide useful information about the behavior of the manufacturing system in question. Once the CT-TH percentile curves are constructed, they allow planners to instantly see the limits imposed on throughput rate with decreasing lead time requirements.

The remainder of this paper is organized as follows. Section 2 provides an overview of our approach. Section 3 describes how we simultaneously estimate the first three moment curves of cycle time. In §4, the properties of the generalized gamma distribution are provided in detail. Section 5 discusses the estimation of percentiles and the statistical inference made on the estimators. Section 6 describes the experiment design used to carry out the sequential simulation experiments and gives a comprehensive presentation of the multistage procedure we have developed. The procedure is evaluated in §7 based on some queueing models and a full factory simulation.

2. Overview

In this section, we provide an overview of the methodology we propose to generate CT-TH percentile curves.

2.1. Distribution of Cycle Time

Our focus is on *cycle time*, in the sense used by Hopp and Spearman (2001, p. 321), "as a random variable that gives the time an individual job takes to traverse a routing." However, our objective in this paper is to go beyond the standard summary measure of average cycle time (which we addressed in Yang et al. 2007) and consider additional summary measures of the distribution of cycle time, in particular, percentiles of cycle time, as a function of throughput. Throughput is the rate (e.g., number of jobs per week) that jobs are completed, which is the same as the release rate of new jobs into the system over the long run if new jobs are released at a constant rate and the system itself is unchanging. Thus, we consider the throughput to be an independent variable that can be controlled by setting the release rate.

To be precise, let CT_h be the cycle time, as defined above, of the hth product or job completed. We assume that as $h \to \infty$, CT_h converges weakly to a random variable CT whose distribution is independent of h (see Billingsley 1999 for a definition of weak convergence) and has finite first four moments. The distribution of CT clearly depends on the throughput x, and we assume convergence of $CT_h(x)$ to $CT(x)$ for all $x \in (0, 1)$, where we have normalized throughput so that 1 corresponds to the capacity of the system. In fact, we actually require a bit more: we also assume that the sample estimate of the νth $(\nu = 1, 2, 3)$ moment $H(x)^{-1} \sum_{h=1}^{H(x)} (CT_h(x))^\nu$ (where $H(x)$ is the selected number of jobs simulated in steady state for simulations at x) is strongly consistent as $H(x) \to \infty$, which requires certain mild regularity conditions on the dependence structure of the cycle time output process to insure that it satisfies a strong law of large numbers (e.g., Glynn and Iglehart 1986 give conditions for regenerative processes; Chien et al. 1997 provide conditions based on mixing; and Meyn and Tweedie 1993, Chapter 17, provide conditions for general state space Markov chains). We are interested in percentiles of $CT(x)$ as a function of x.

Of course, such convergence never occurs in a physical manufacturing system, but for planning and analysis purposes we often approximate the finite-time behavior of a stochastic system by the limiting

behavior of a stationary stochastic model of it (e.g., Buzacott and Shanthikumar 1993). When the model is a mathematically tractable (or readily approximated) queueing network model, then the conditions that insure the existence of a "steady state" can often be verified. However, if the model is a discrete event stochastic simulation, as it is in this paper, then we can argue for the existence of a steady state by analogy to more precisely specified stochastic processes (see, for instance, Henderson 2001), but rarely can we formally prove it. At a practical level, we are assuming that if the driving stochastic processes (job arrivals, machine processing times, failure, and repair processes) are stationary, and the system logic (job priorities, queue disciplines, and workcenter capacities) is unchanging, then a conceptually infinitely long simulation run will yield cycle times that satisfy our assumptions. See, for instance, Law and Kelton (2000) for more on the "steady-state simulation problem." From here on, when we refer to "cycle time," we are referring to the random variable $CT(x)$.

2.2. Overview of the Method

Simulation is often used to provide percentile estimates, and substantial research effort has been devoted to the estimation of cycle time percentiles via simulation. However, efficiently generating cycle time percentile estimates remains a challenging topic for at least two reasons: Standard estimators based on order statistics may require excessive data storage unless all of the percentiles of interest are known in advance, and even then it is difficult to do sequential estimation until a fixed precision is reached (Chen and Kelton 1999). On the other hand, approximations based on only the first two moments of cycle time and assuming a normal distribution can be grossly inaccurate (McNeill et al. 2003). A technique based on the Cornish-Fisher expansion has been proposed by McNeill et al. (2003) to estimate percentiles of cycle time; it takes into account the first four moments of the cycle time distribution and allows accurate and precise percentile estimates to be generated for moderately nonnormal distributions. However, this method can only give percentiles at fixed, prespecified throughputs where simulation experiments have been performed. The methodology proposed in this paper aims at providing a more comprehensive profile of the system by generating CT-TH percentile curves throughout a throughput range.

Our approach to approximating percentiles of $CT(x)$ is to fit curves to the first three moments (equivalently, mean, variance, and skewness) of $CT(x)$ as a function of throughput x, match a highly flexible distribution (the generalized gamma distribution (GGD)) to these moments, and then invert the fitted distribution to obtain percentiles. More specifically, the strategy we propose for estimating $\mathscr{C}_\alpha(x)$, the 100α ($\alpha \in [\alpha_L, \alpha_U]$) percentile of cycle time at throughput rate $x \in [x_L, x_U]$, is outlined as follows:

1. Use an extended version of the methodology of Yang et al. (2007) to estimate not only the CT-TH mean (first moment) curve, but also the CT-TH second and third moment curves over the throughput range of interest. This allows for the prediction of the first three moments of cycle time at any throughput x, say $\mu_1(x)$, $\mu_2(x)$, and $\mu_3(x)$.

2. Use the method of moments to fit a GGD cumulative distribution function (cdf) $G(t; a(x), b(x), k(x))$ as an approximation for the cycle time distribution ($a(x)$, $b(x)$, and $k(x)$ are distribution parameters that depend on x). We write the resulting fitted GGD as $G(t; \hat{a}(x), \hat{b}(x), \hat{k}(x))$.

3. Estimate the percentile $\mathscr{C}_\alpha(x)$ by taking the inverse of the cdf of the cycle time: $\widehat{\mathscr{C}}_\alpha(x) = G^{-1}(\alpha; \hat{a}(x), \hat{b}(x), \hat{k}(x))$.

The functional form we have chosen as a metamodel for the moment curves (see §3) was motivated by a combination of the known moment curves of some simple, single-queue models (e.g., M/M/1) and heavy traffic results for more general models (including networks of queues, where a single bottleneck queue dominates in heavy traffic; see, for instance, Whitt 1989). Of course, a complex manufacturing system behaves neither like a single queue nor (typically) like a queueing network in extremely heavy traffic. Therefore, our metamodel has more free parameters than these simple models, providing greater flexibility. When considering the first moment, Yang et al. (2007) showed that this model worked remarkably well. In fact, there have been a number of papers in which queueing systems have been well-represented by metamodels, including Cheng and Kleijnen (1999), Fowler et al. (2001), and Park et al. (2002). However, Allen (2003) and Johnson et al. (2004) demonstrated that the models used in these papers can be inadequate for complex manufacturing systems, which motivated our more flexible formulation.

The ubiquitous use of the normal distribution in statistics might tempt one to try to get by with a two-moment approximation for the distribution of CT. However, even very simple models (such as the M/M/1 queue, where the steady-state cycle time is exponentially distributed) demonstrate that this will be woefully inadequate. On the other hand, we might consider using four or more moments as in McNeill et al. (2003), based on the premise that more moments provide a better characterization of the distribution. Our choice of three moments is a compromise between the obvious need for more than two moments and the practical difficulty of precisely estimating curves for higher moments.

As already described, we adopt an indirect method to derive cycle time percentiles from the moment estimates. Why not just run simulations at a very fine grid of x values and save the results? That requires the storage of a very large amount of data plus many hours of simulation to cover a fine grid, whereas our approach will simulate no more than five values of throughput x and still deliver any percentile at any x nearly instantly once the simulations are complete.

3. Estimation of CT-TH Moment Curves

As indicated in the previous section, providing the first three moment CT-TH curves over the throughput range of interest is the primary step in the estimation of $\mathscr{C}_\alpha(x)$.

In Yang et al. (2007), a metamodeling-based methodology was developed for estimating mean (first moment $\mu_1(x)$) CT-TH curves, which quantifies the relationship of long-run average cycle time to throughput rate. A nonlinear regression metamodel is developed to represent the underlying CT-TH curve, and simulation experiments are built up sequentially in an efficient manner to collect data for fitting the curve. In this section, we will generalize the method of metamodeling to simultaneously estimate the first three moment CT-TH curves.

In manufacturing simulations, moment CT-TH curves typically follow the shape in Figure 1 (see, for instance, Fowler et al. 2001, Park et al. 2002, Allen 2003, Johnson et al. 2004). The methodology used to fit the mean CT-TH curve (Yang et al. 2007) can be extended to the estimation of higher moment curves, so in this subsection we will restate the estimation method in Yang et al. (2007) in a general way to cover

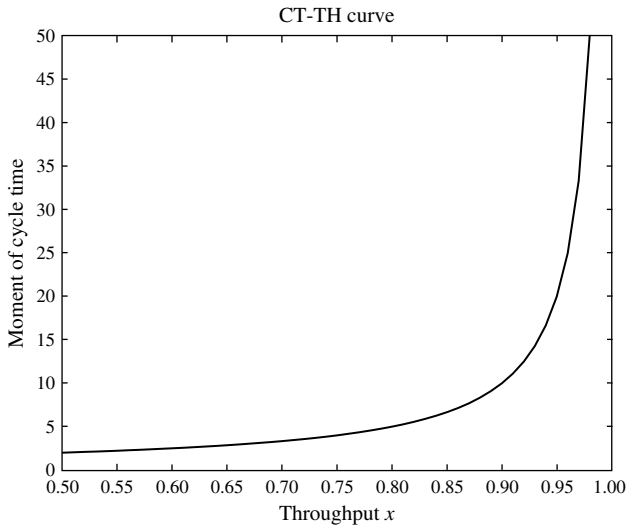

Figure 1 A Generic Moment CT-TH Curve

estimating the first three moment curves simultaneously over a given throughput range $[x_L, x_U]$ based on a single set of simulation experiments.

We suppose that the experiment is made up of a number of independent simulation replications performed at m distinct levels of throughput $\mathbf{x} = (x_1, x_2, \ldots, x_m)$ with $x_i \in [x_L, x_U]$ for $i = 1, 2, \ldots, m$. From the jth replication performed at throughput x, an output response $\{Y_j^{(\nu)}(x), \nu = 1, 2, 3\}$ can be obtained for the purpose of estimating the νth moment curve:

$$Y_j^{(\nu)}(x) = \frac{1}{H(x)} \sum_{h=1}^{H(x)} (CT_{jh}(x))^\nu \quad j = 1, 2, \ldots, n(x). \quad (1)$$

Here, $n(x)$ is the number of replications placed at throughput x, $CT_{jh}(x)$ represents the individual cycle time of the hth job completed in the jth replication at x, and $H(x)$ is the selected number of products simulated in steady state for simulations at x. A lower bound on the value of $H(x)$ could be determined following the guidelines given in Law and Kelton (2000). As explained in Yang et al. (2007), for simplicity, $H(x)$ could be set as $H(x) = H$ for all values of x (if $H(x)$ varies with x, then a simple additional step needs to be taken in the moment-curve fitting). For a given experiment consisting of a number of simulation replications carried out at m design points, the data sets

$$\mathbf{Y}^{(\nu)} = \big(Y_1^{(\nu)}(x_1), \ldots, Y_{n(x_1)}^{(\nu)}(x_1), \ldots, Y_1^{(\nu)}(x_m), \ldots, Y_{n(x_m)}^{(\nu)}(x_m)\big) \quad (2)$$

can be extracted for $\nu = 1, 2, 3$. The integer vector $\mathbf{n} = (n(x_1), n(x_2), \ldots, n(x_m))$ represents the allocation of replications to the m design points.

To these data sets $\{\mathbf{Y}^{(\nu)}, \nu = 1, 2, 3\}$, the three moment curves can be fitted. The curve fitting is based on the regression models that will be introduced below, and justification for the specific form of the models is given in Online Supplement A.1 (available at http://joc.pubs.informs.org/ecompanion.html). The metamodeling methodology applies to fitting moment curves of any order. For the sake of clarity, we omit the superscript ν representing the νth moment in the regression models that appear in the remainder of this section.

The CT-TH relationship for the νth moment curve can be represented by the following metamodel:

$$Y_j(x) = \mu(x, \mathbf{c}, p) + \varepsilon_j(x) \quad j = 1, 2, \ldots, n(x), \quad (3)$$

where

$$\mu(x, \mathbf{c}, p) = \frac{\sum_{l=0}^{t} c_l x^l}{(1-x)^p}. \quad (4)$$

Extensive experiments have shown this model to be flexible enough to model cycle time moments of realistic manufacturing simulations (Allen 2003, Johnson et al. 2004). The exponent p, the polynomial order t, and the coefficient vector $\mathbf{c} = (c_0, c_1, \ldots, c_t)$ are unknown parameters in the model.

As explained in Yang et al. (2007), the error term $\varepsilon_j(x)$ has expectation zero and variance $\sigma^2(x)$, which depends on x through a "variance model" of the form:

$$\sigma^2(x) = \frac{\sigma^2}{(1-x)^{2q}}. \tag{5}$$

Both σ^2 and q are unknown parameters. Substituting sample variance as the response in Equation (5), we can estimate the variance model. With the estimated $\hat{q}$, a simple transformation of model (3) will yield a standard nonlinear regression model with approximately constant error variance:

$$Z_j(x) = Y_j(x) \times (1-x)^q \tag{6}$$

$$= \eta(x, \mathbf{c}, r) + \delta_j = \sum_{l=0}^{t} c_l x^l (1-x)^r + \delta_j, \tag{7}$$

where $r = q - p$ is an unknown parameter and the error δ_j is assumed to have a constant variance σ^2. Because the error term in model (7) has constant variance, we estimate model (7) directly. The parameters of the original moment model (3) will be obtained indirectly by noting that the coefficients $\mathbf{c}$ in Equation (3) coincide with those in Equation (7), and p is estimated by the difference of the q and r estimates. The polynomial order t in the moment model is determined via extra sum of squares analysis in a forward selection manner.

To conclude this subsection, we summarize the method described above for estimating the νth ($\nu = 1, 2, 3$) moment CT-TH curve. First, based on the data set $\mathbf{Y}^{(\nu)}$ as defined in Equation (2), the variance model (5) is fitted and the estimated parameter $\hat{q}_\nu$ is obtained; with $\hat{q}_\nu$, the data transformation is performed on $\mathbf{Y}^{(\nu)}$ as shown in Equation (6), and the resulting transformed data set with stabilized variance can be represented by the following vector:

$$\mathbf{Z}^{(\nu)} = \big(Z_1^{(\nu)}(x_1), \ldots, Z_{n(x_1)}^{(\nu)}(x_1), \ldots, Z_1^{(\nu)}(x_m), \ldots, Z_{n(x_m)}^{(\nu)}(x_m)\big). \tag{8}$$

Finally, model (7) is fitted to $\mathbf{Z}^{(\nu)}$ and the three moment curves $\{\mu_\nu(x), \nu = 1, 2, 3\}$ are obtained over the throughput range $[x_L, x_U]$.

4. The Generalized Gamma Distribution

The distribution family chosen to fit the individual cycle times for manufacturing settings should be able to provide a good fit for a variety of cycle time distributions. As noted by Rose (1999), for complicated manufacturing systems, cycle times tend to be close to normally distributed. However, as the system is loaded with heavier traffic, even for complicated systems, cycle time distributions become more and more skewed (McNeill et al. 2003). In our method, the generalized gamma distribution is adopted because, to the best of our knowledge, it is the most flexible three-parameter distribution in terms of coverage of the skewness kurtosis plane.

The three-parameter generalized gamma distribution (GGD3), first presented in Stacy (1962), has the following pdf (probability density function):

$$g(t, a, b, k) = \frac{|k|}{\Gamma(a)} \cdot \frac{t^{ak-1}}{b^{ak}} \cdot \exp[-(t/b)^k], \quad t > 0,\ a > 0,\ b > 0,\ k \neq 0, \tag{9}$$

where a and k are the shape parameters and b is the scale parameter. As illustrated in Ashkar et al. (1988), the GGD can cover a wide range of skewness as well as kurtosis. Also, the GGD includes a variety of distributions as special cases, such as exponential ($a = k = 1$), gamma ($k = 1$), and Weibull ($a = 1$) distributions. The lognormal and normal distributions also arise as limiting cases.

In addition to its shape flexibility, the reason why we adopt the GGD (as opposed to the Cornish-Fisher expansion proposed by McNeill et al. 2003 or other flexible distributions such as the Johnson family) is because it only involves three parameters, which means only the first three moment curves $\{\mu_\nu(x), \nu = 1, 2, 3\}$ need to be estimated to provide a fit of the cycle time distribution. In our experience, precisely estimating higher moment curves can be very difficult. As explained in §3, the νth moment curve is estimated based on the data set (2). When $\nu \geq 4$, the steepness of the moment curve $\mu_\nu(x)$ and the heteroscedasticity of variance in the data (2) become so pronounced that it requires substantially more simulation data to obtain well-estimated moment curves, as illustrated through the M/M/1 example in Online Supplement A.1.

A location parameter t_0 can be added to GGD3, and the resulting 4-parameter distribution (GGD4) is obtained by shifting the lower bound of GGD3 from $t = 0$ to $t = t_0$. The properties of any variable T following a GGD4 can be derived from those of $T - t_0 \sim$ GGD3. In GGD-based fitting of a cycle time distribution, t_0 signifies the lower bound of individual cycle times, which might be known in advance. In cases where t_0 is difficult to specify, we can set $t_0 = 0$ because GGD3 is flexible enough to give an adequate fit even if the origin of the underlying distribution deviates from zero. In light of these features, we will

focus our attention on GGD3 and present some of its properties.

Noncentral moments of GGD3 are given by

$$m_\nu = \frac{b^\nu \Gamma(a+\nu/k)}{\Gamma(a)} \quad \nu = 1, 2, 3, \ldots, \tag{10}$$

where ν is the order of the moment. The moments are defined only if $a + \nu/k > 0$. In the remainder of this paper, we assume that the moments exist, i.e., $a + \nu/k > 0$ for $\nu = 1, 2, 3$. Choosing any three distinct values for ν will provide the equations required by the method of moments to obtain the three distribution parameters a, b, and k.

GGD3 can be regarded as a generalization of the two-parameter gamma distribution (GD2) by supplying a positive parameter k as an exponent for the exponential factor of GD2(a, b) whose pdf is

$$f(t) = \frac{t^{a-1}e^{-t/b}}{b^a\Gamma(a)}, \quad t > 0,\ a > 0,\ b > 0. \tag{11}$$

Suppose $T \sim$ GGD3(a, b, k). Then, $T' = (T/b)^k \sim$ GD2$(a, 1)$ (standard gamma distribution). We have implemented numerical methods to calculate the α percentile of T', $\mathscr{C}'(\alpha; a, 1)$. The corresponding percentile of variable T, say $\mathscr{C}(\alpha; a, b, k)$, can be obtained by the straightforward transformation:

$$\mathscr{C}(\alpha; a, b, k) = b(\mathscr{C}'(\alpha; a, 1))^{1/k}. \tag{12}$$

The partial derivatives of the percentile $\mathscr{C}(\alpha; a, b, k)$ with respect to the GGD parameters are as follows:

$$\begin{aligned}
\frac{\partial \mathscr{C}(\alpha; a, b, k)}{\partial a} &= \frac{b}{k}(\mathscr{C}'(\alpha; a, 1))^{1/(k-1)} \frac{\partial \mathscr{C}'(\alpha; a, 1)}{\partial a}, \\
\frac{\partial \mathscr{C}(\alpha; a, b, k)}{\partial b} &= (\mathscr{C}'(\alpha; a, 1))^{1/k}, \\
\frac{\partial \mathscr{C}(\alpha; a, b, k)}{\partial k} &= \frac{-b}{k^2}(\mathscr{C}'(\alpha; a, 1))^{1/k} \log \mathscr{C}'(\alpha; a, 1).
\end{aligned} \tag{13}$$

Note that the partial derivative $\partial \mathscr{C}'(\alpha; a, 1)/\partial a$ can be approximately calculated using the finite difference:

$$\frac{\partial \mathscr{C}'(\alpha; a, 1)}{\partial a} = \frac{\mathscr{C}'(\alpha; a + \Delta a/2, 1) - \mathscr{C}'(\alpha; a - \Delta a/2, 1)}{\Delta a}. \tag{14}$$

Therefore, the first derivatives (13) can all be obtained numerically.

5. Estimation of Percentiles

In this section, we describe in detail how we fit the generalized gamma distribution at a given throughput level $x \in [x_L, x_U]$ based on the first three moment estimates, how the percentiles are estimated once $G(t, \hat{a}(x), \hat{b}(x), \hat{k}(x))$ is obtained, and how an approximate standard error can be provided for the percentile estimators.

5.1. Point Estimation

As explained in §3, the first three moment curves can be fitted simultaneously based on a number of simulation experiments performed at different levels of throughput. Therefore, for any $x \in [x_L, x_U]$, the first three moments can be predicted by $\{\hat{\mu}_\nu(x), \nu = 1, 2, 3\}$. Substituting the moment estimates into Equation (10) results in the following equations:

$$\begin{aligned}
\hat{\mu}_1(x) &= \frac{\hat{b}(x)\Gamma(\hat{a}(x) + 1/\hat{k}(x))}{\Gamma(\hat{a}(x))}, \\
\hat{\mu}_2(x) &= \frac{\hat{b}(x)^2\Gamma(\hat{a}(x) + 2/\hat{k}(x))}{\Gamma(\hat{a}(x))}, \\
\hat{\mu}_3(x) &= \frac{\hat{b}(x)^3\Gamma(\hat{a}(x) + 3/\hat{k}(x))}{\Gamma(\hat{a}(x))}.
\end{aligned} \tag{15}$$

Solving Equation (15) numerically gives the three estimated distribution parameters $(\hat{a}(x), \hat{b}(x), \hat{k}(x))$ for the fitted GGD distribution at throughput x. With the estimated distribution of cycle time at throughput rate x, $G(t; \hat{a}(x), \hat{b}(x), \hat{k}(x))$, the percentile $\mathscr{C}_x(\alpha)$ can be estimated for any $\alpha \in [\alpha_L, \alpha_U]$ utilizing the relationship shown in Equation (12).

5.2. Statistical Inference for the Percentile Estimator

Drawing inference about a parameter obtained indirectly is difficult in general. In this paper, the delta method (Lehmann 1999) is applied to make inferences concerning the estimated percentiles.

The percentile $\mathscr{C}_\alpha(x)$ is estimated based on the fitted GGD distribution and is obviously a function of the distribution parameters a, b, and k. The first-order approximation using the delta method provides the following estimation for the variance of percentile estimators, where for convenience we suppress the dependence of a, b, and k on x:

$$\begin{aligned}
&\mathrm{Var}[\widehat{\mathscr{C}_\alpha}(x)] \\
&\doteq \left(\frac{\partial \mathscr{C}_\alpha(x)}{\partial a}\right)^2 \mathrm{Var}[\hat{a}] + \left(\frac{\partial \mathscr{C}_\alpha(x)}{\partial b}\right)^2 \mathrm{Var}[\hat{b}] \\
&\quad + \left(\frac{\partial \mathscr{C}_\alpha(x)}{\partial k}\right)^2 \mathrm{Var}[\hat{k}] + 2\left(\frac{\partial \mathscr{C}_\alpha(x)}{\partial a}\right)\left(\frac{\partial \mathscr{C}_\alpha(x)}{\partial b}\right)\mathrm{Cov}[\hat{a}, \hat{b}] \\
&\quad + 2\left(\frac{\partial \mathscr{C}_\alpha(x)}{\partial b}\right)\left(\frac{\partial \mathscr{C}_\alpha(x)}{\partial k}\right)\mathrm{Cov}[\hat{b}, \hat{k}] \\
&\quad + 2\left(\frac{\partial \mathscr{C}_\alpha(x)}{\partial k}\right)\left(\frac{\partial \mathscr{C}_\alpha(x)}{\partial a}\right)\mathrm{Cov}[\hat{k}, \hat{a}].
\end{aligned} \tag{16}$$

In Equation (16), the partial derivatives of the percentile $\mathscr{C}_\alpha(x)$ with respect to the GGD parameters can be approximately calculated from Equation (13) by substituting the estimates $\hat{a}$, $\hat{b}$, $\hat{k}$, and $\widehat{\mathscr{C}_\alpha}(x)$.

Because the GGD parameters are estimated by matching the first three moments of the GGD distribution to the moment estimates $\{\hat{\mu}_\nu(x), \nu = 1, 2, 3\}$, the variances and covariances in Equation (16) are functions of the variances and covariances of $\hat{\mu}_1(x)$, $\hat{\mu}_2(x)$, and $\hat{\mu}_3(x)$. This is where the delta method (first-order approximation) is applied for a second time. Using matrix notation, we have the following relationship as derived in Ashkar et al. (1988):

$$\begin{pmatrix} \mathrm{Var}[\hat{a}] \\ \mathrm{Var}[\hat{b}] \\ \mathrm{Var}[\hat{k}] \\ \mathrm{Cov}[\hat{a},\hat{b}] \\ \mathrm{Cov}[\hat{a},\hat{k}] \\ \mathrm{Cov}[\hat{b},\hat{k}] \end{pmatrix} \doteq \begin{pmatrix} C_{11} & C_{12} & \cdots & C_{16} \\ C_{21} & C_{22} & \cdots & C_{26} \\ C_{31} & C_{32} & \cdots & C_{36} \\ C_{41} & C_{42} & \cdots & C_{46} \\ C_{51} & C_{52} & \cdots & C_{56} \\ C_{61} & C_{62} & \cdots & C_{66} \end{pmatrix}^{-1} \begin{pmatrix} \mathrm{Var}[\hat{\mu}_1(x)] \\ \mathrm{Var}[\hat{\mu}_2(x)] \\ \mathrm{Var}[\hat{\mu}_3(x)] \\ \mathrm{Cov}[\hat{\mu}_1(x),\hat{\mu}_2(x)] \\ \mathrm{Cov}[\hat{\mu}_1(x),\hat{\mu}_3(x)] \\ \mathrm{Cov}[\hat{\mu}_2(x),\hat{\mu}_3(x)] \end{pmatrix}, \tag{17}$$

where the matrix $\mathbf{C}$ is given by

$$\begin{pmatrix} D_{11}^2 & D_{12}^2 & D_{13}^2 & 2D_{11}D_{12} \\ D_{21}^2 & D_{22}^2 & D_{23}^2 & 2D_{21}D_{22} \\ D_{31}^2 & D_{32}^2 & D_{33}^2 & 2D_{31}D_{32} \\ D_{11}D_{21} & D_{12}D_{22} & D_{13}D_{23} & D_{11}D_{22}+D_{12}D_{21} \\ D_{11}D_{31} & D_{12}D_{32} & D_{13}D_{33} & D_{11}D_{32}+D_{12}D_{31} \\ D_{21}D_{31} & D_{22}D_{32} & D_{23}D_{33} & D_{21}D_{32}+D_{22}D_{31} \end{pmatrix}$$

$$\begin{matrix} 2D_{11}D_{13} & 2D_{12}D_{13} \\ 2D_{21}D_{23} & 2D_{22}D_{23} \\ 2D_{31}D_{33} & 2D_{32}D_{33} \\ D_{11}D_{23}+D_{13}D_{21} & D_{12}D_{23}+D_{13}D_{22} \\ D_{11}D_{33}+D_{13}D_{31} & D_{12}D_{33}+D_{13}D_{32} \\ D_{21}D_{33}+D_{23}D_{31} & D_{22}D_{33}+D_{23}D_{32} \end{matrix}, \tag{18}$$

with $D_{rj} = \partial\mu_r(x)/\partial\zeta_j (r, j = 1, 2, 3)$ and $\zeta_1 = a$, $\zeta_2 = b$, $\zeta_3 = k$.

From Equation (10), the partial derivatives can be approximately calculated by substituting the estimated GGD parameters into

$$D_{11} = \frac{-b}{k^2\Gamma(a)}[\Gamma(a+1/k)\Psi(a+1/k)],$$

$$D_{12} = \frac{\Gamma(a+1/k)}{\Gamma(a)},$$

$$D_{13} = b\frac{\Gamma(a+1/k)}{\Gamma(a)}[\Psi(a+1/k) - \Psi(a)],$$

$$D_{21} = \frac{-2b^2}{k^2\Gamma(a)}[\Gamma(a+2/k)\Psi(a+2/k)],$$

$$D_{22} = 2k\frac{\Gamma(a+2/k)}{\Gamma(a)},$$

$$D_{23} = b^2\frac{\Gamma(a+2/k)}{\Gamma(a)}[\Psi(a+2/k) - \Psi(a)],$$

$$D_{31} = \frac{-3b^3}{k^2\Gamma(a)}[\Gamma(a+3/k)\Psi(a+3/k)],$$

$$D_{32} = 3k^2\frac{\Gamma(a+3/k)}{\Gamma(a)},$$

$$D_{33} = b^3\frac{\Gamma(a+3/k)}{\Gamma(a)}[\Psi(a+3/k) - \Psi(a)], \tag{19}$$

where

$$\Psi(t) = \frac{1}{\Gamma(t)}\frac{d\Gamma(t)}{dt}.$$

Clearly, from the derivation above, estimating $\mathrm{Var}[\widehat{\mathscr{C}}_\alpha(x)]$ requires obtaining the moment estimators $\{\hat{\mu}_\nu(x), \nu = 1, 2, 3\}$ and their variances and covariances. The estimators $\{\hat{\mu}_\nu(x), \nu = 1, 2, 3\}$ can be obtained by following the methodology explained in §3; estimating their variances and covariances is discussed in Online Supplement A.3.

6. Procedure for Estimating Percentiles of Cycle Time

In this section, we discuss issues related to experiment design and give a description of the proposed procedure for estimating percentiles. To provide context, a high-level description of the procedure is provided in Figure 2.

In summary, simulation experiments are carried out sequentially until the prespecified stopping criterion is satisfied. The experimentation is initiated with a starting design that allocates an equal number (chosen by the user) of replications to the two end points

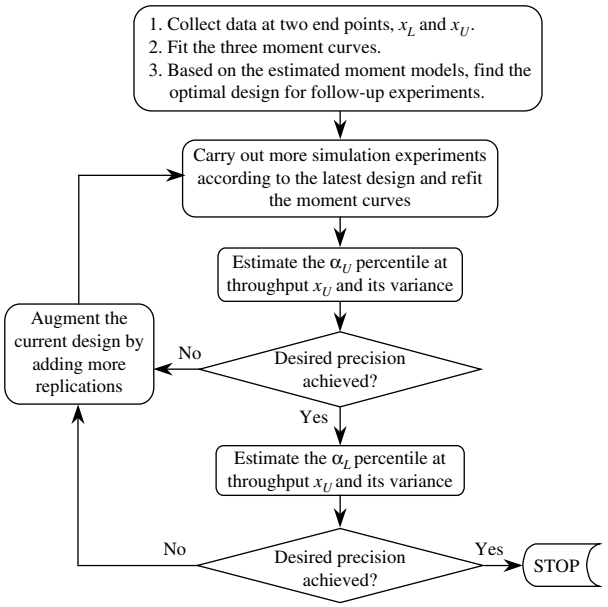

Figure 2 Flow Chart for the Multistage Procedure

of the throughput range $[x_L, x_U]$. As the procedure progresses, new design points are included and additional replications are added in batches. Each batch of replications is allocated to the design points to minimize PM (defined in Equation (22)), an experiment design criterion that is related to the variance of the percentile estimators. Because the design criterion depends on unknown parameters of the moment curves, the current best estimates of the parameters are used in the allocation of each batch of replications. As more simulation data are collected, increasingly precise estimators are obtained until the precision of the estimators matches the stopping criterion.

6.1. Experiment Design

As already noted, the experiments are started by a design that allocates an equal number of replications to the upper and lower end of the throughput range. This design will then be augmented by including more design points and more replications as the procedure progresses. To determine the follow-up design, we need to answer three questions based on the estimates obtained from the current data set: (i) How many additional replications, say ΔN, will be added? (ii) At what design points (throughput levels) will the simulations be executed? (iii) How many of the ΔN replications should be allocated to each design point? We use the vector $\mathbf{x} = (x_1, x_2, \ldots, x_m)$ to represent the set of design points included in the design, and $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_m)$ to represent the fractions for the total replications assigned to each design point. At each step, the values of $\mathbf{x}$ and $\boldsymbol{\pi}$ will be determined conditional on the fact that some replications have already been allocated to certain design points.

6.1.1. Design Criterion. Our goal is to develop a method to estimate the percentile $\mathscr{C}_\alpha(x)$ for $x \in [x_L, x_U]$ and $\alpha \in [\alpha_L, \alpha_U]$. Therefore, the experiment design will seek to minimize some measure of the variance of $\widehat{\mathscr{C}}_\alpha(x)$. Suppose that N is the number of replications available for allocation. A natural performance measure, which is inherited from Cheng and Kleijnen (1999), is the weighted average variance over the throughput range of interest normalized by N:

$$PM_0 = N \frac{\int_{x_L}^{x_U} w(x)\mathrm{Var}[\widehat{\mathscr{C}}_{\alpha_U}(x)]\,dx}{\int_{x_L}^{x_U} w(x)\,dx}, \tag{20}$$

where $w(x)$ is the weight function, which in the simplest case is one, and $N\mathrm{Var}[\widehat{\mathscr{C}}_{\alpha_U}(x)]$ is the normalized variance, which is independent of N. As explained in §5.2, from the first three fitted moment curves $\{\hat{\mu}_\nu(x), \nu = 1, 2, 3\}$, the variance $\mathrm{Var}[\widehat{\mathscr{C}}_\alpha(x)]$ can be estimated for any percentile estimator $\widehat{\mathscr{C}}_\alpha(x)$ ($x \in [x_L, x_U]$, $\alpha \in [\alpha_L, \alpha_U]$). We chose to base Equation (20) on the variance of the largest percentile α_U because $\widehat{\mathscr{C}}_{\alpha_U}(x)$ is typically much more variable than other percentile estimators. Unfortunately, it is not practical to determine $[\mathbf{x}, \boldsymbol{\pi}]$ by minimizing PM_0 because $\mathrm{Var}[\widehat{\mathscr{C}}_\alpha(x)]$ can only be numerically estimated for given values of x and α (§5.2). Hence, we use the simple finite difference approximation of Equation (20):

$$PM_0 \doteq N \sum_{\kappa \in \boldsymbol{\Omega}_x} \mathrm{Var}[\widehat{\mathscr{C}}_{\alpha_U}(\kappa)] \cdot \Delta\kappa, \tag{21}$$

where $\boldsymbol{\Omega}_x$ is a chosen set of evenly spaced grid points in the range $[x_L, x_U]$ and $\Delta\kappa$ is the interval between two neighboring points. Obviously, $\Delta\kappa$ is a constant that can be dropped from Equation (21), so we define our design criterion as

$$PM = N \sum_{\kappa \in \boldsymbol{\Omega}_x} \mathrm{Var}[\widehat{\mathscr{C}}_{\alpha_U}(\kappa)]. \tag{22}$$

Measure PM is a function of the design $[\mathbf{x}, \boldsymbol{\pi}]$ as illustrated in Online Supplement A.5. Evaluating PM for given $[\mathbf{x}, \boldsymbol{\pi}]$ requires providing an estimate for $\mathrm{Var}[\widehat{\mathscr{C}}_{\alpha_U}(\kappa)]$, which can be obtained at any later stage of experimentation, where simulation data are available for the estimation of the first three moment curves (for details, see Online Supplement A.5). Hence, at a point where further experiments are to be carried out, PM can be approximately calculated for given $[\mathbf{x}, \boldsymbol{\pi}]$, which enables us to apply a numerical search method to the problem of optimizing PM. Note that $N \times \boldsymbol{\pi}$ is not restricted to be an integer in the search for the optimal solution of $[\mathbf{x}, \boldsymbol{\pi}]$.

Next, we will give the details on how the optimization problem is constructed and solved to augment the current experiments at each stage.

6.1.2. Optimal Experiment Design. We propose solving the following constrained nonlinear optimization problem to guide further simulation experiments given that some replications have already been allocated:

$$\min_{\mathbf{x}, \boldsymbol{\pi}} \quad PM(\mathbf{x}, \boldsymbol{\pi}), \tag{23}$$

$$\text{subject to} \quad \{x_1, x_2, \ldots, x_m\} \supseteq \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{m_c}\} \tag{24}$$

$$x_L \le x_1 < x_2 < \cdots < x_m \le x_U$$

$$\sum_{i=1}^{m} \pi(x_i) = 1$$

$$\pi(x_i) \ge lb(x_i) \quad \text{for } i = 1, 2, \ldots, m.$$

The input parameters, decision variables, and constraints of Equation (23) are as follows:

Input Parameters

- The range of throughput $[x_L, x_U]$.
- m_c and m ($m \ge m_c$), the number of design points before and after augmenting the design, respectively.

• The old design points $\{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{m_c}\}$ and the allocation of simulation replications already made at those points $\{n_c(\hat{x}_1), n_c(\hat{x}_2), \ldots, n_c(\hat{x}_{m_c})\}$. Note that $n_c(x) = 0$ for $x \notin \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{m_c}\}$.

• The total number of replications already allocated N_c and the increment of replications to be added to the current design ΔN. Therefore, we have $N = N_c + \Delta N$. For the same reason as explained in Yang et al. (2007), guiding the choice of ΔN at each stage is important and the way to determine ΔN is detailed in Online Supplement A.6. Both N_c and ΔN are used to calculate the lower bounds $lb(x_i) = \max\{n_c(x_i), 2\}/(N_c + \Delta N)$ for $i = 1, 2, \ldots, m$. We set $lb_i \geq 2/(N_c + \Delta N)$ to ensure that at least two replications are assigned to any point x_i included in the design.

Decision Variables

• The new set of design points $\mathbf{x} = \{x_1, x_2, \ldots, x_m\}$, whose values are forced to be increasing in the subscript.

• The updated allocation of simulation effort $\boldsymbol{\pi} = \{\pi_1, \pi_2, \ldots, \pi_m\}$, which has to be mapped into integer numbers of replications, say $\{n(x_i), i = 1, 2, \ldots, m\}$, assigned to the design points. We use a simple rounding in our method by setting $n(x_i) = \lceil N\pi_i \rceil$. By rounding up, we insure that each design point gets at least as many replications as called for by the optimal solution, and because our goal is to achieve a fixed precision (rather than optimize a fixed budget), this can do no harm. The resulting integer solution could be suboptimal but seems to work well in our numerical experiments.

Constraints

• The constraint (24) forces the new set of design points to include the old points.

• The meanings of the other constraints are obvious.

To solve the optimization problem, we need to choose starting values for each of the decision variables. In all the experiments considered in this paper, the starting values of x are chosen to be evenly spaced throughout the interval of throughput, and the fraction of replications at each design point x_i initiates from $N^{-1}(n_c(x_i) + \Delta N/m)$ $(i = 1, 2, \ldots, m)$.

In the procedure, Equation (23) is solved to augment the current design when an assessment of the chosen percentile estimates shows that subsequent experimental effort is necessary. The design may be augmented in two different ways: (1) adding design points and replications, and (2) adding replications only. Augmentation of type 1 only occurs once in the procedure, when we expand the starting design which only consists of experiments performed at the two end points x_L and x_U to a m-point design. (Guidelines for determining the number of design points m is provided in §6.3.) Afterward, the location of design points are fixed, and only the allocation of simulation effort can be modified by assigning more replications to the current design points. In our experiments, the optimization problem (23) is coded in Matlab, and the Matlab optimization function "fmincon" is used to solve the nonlinear constrained problem (with $m = 5$ as will be explained in §6.3). This takes about 150 seconds on a computer with a processor speed of 3 GHz.

6.2. Stopping Rule

The proposed procedure collects simulation data to allow for estimation of $\mathscr{C}_\alpha(x)$ for $x \in [x_L, x_U]$ and $\alpha \in [\alpha_L, \alpha_U]$ with both ranges of interest being specified by the user. Moreover, our method provides an error estimate for any percentile estimator $\widehat{\mathscr{C}}_\alpha(x)$ $(x \in [x_L, x_U], \alpha \in [\alpha_L, \alpha_U])$ (§5.2). Obviously, the upper end of throughput is where the variability of cycle time is most pronounced, and it is known that estimators of larger percentiles are more variable than their lower counterparts. Consequently, $\widehat{\mathscr{C}}_{\alpha_U}(x_U)$ is considered to possess the highest variability among all the estimable percentiles, which motivates us to use the relative error of $\widehat{\mathscr{C}}_{\alpha_U}(x_U)$ as the stopping criterion for our procedure. By controlling the precision of the most variable estimator $\widehat{\mathscr{C}}_{\alpha_U}(x_U)$, we hope that other percentiles will also be well estimated.

Specifically, we let the user specify a precision level, say $100\gamma\%$, and the procedure terminates only when the condition

$$\frac{2\mathrm{SE}[\widehat{\mathscr{C}}_{\alpha_U}(x_U)]}{\widehat{\mathscr{C}}_{\alpha_U}(x_U)} \leq 100\gamma\%$$

is satisfied. We define $\mathrm{SE}[\cdot] = \sqrt{\mathrm{Var}[\cdot]}$. Moreover, a safe fallback strategy is adopted. As illustrated in Figure 2, a check is also performed on the precision of $\widehat{\mathscr{C}}_{\alpha_L}(x_U)$, and simulation data will be collected until

$$\frac{2\mathrm{SE}[\widehat{\mathscr{C}}_{\alpha_L}(x_U)]}{\widehat{\mathscr{C}}_{\alpha_L}(x_U)} \leq 100\gamma\%.$$

Constraining the precision of any percentile estimator $\mathscr{C}_\alpha(x)$ within a certain prespecified level is difficult, but by controlling the relative precision of the two estimators $\widehat{\mathscr{C}}_{\alpha_U}(x_U)$ and $\widehat{\mathscr{C}}_{\alpha_L}(x_U)$, we hope to impose precision control on percentile estimators throughout the range we consider.

6.3. The Multistage Procedure

This subsection is devoted to an overall description of the multistage procedure, which is diagrammed in Figure 2.

The procedure is divided into two stages: In the initial stage, pilot simulation runs are performed at the two end points of the throughput range to provide the

preliminary data for model estimation; in the second stage, the experiments are augmented to include, say, m design points, and simulation runs are added in an efficient manner until the desired precision level on the chosen percentile estimators is achieved.

The number of design points m is a user-specified parameter, which has to be set to allow for the good estimation of the first three moment curves; that is, m should be sufficiently large to allow the moment model (4) to include enough polynomial terms to generate a good fit for the νth ($\nu = 1, 2, 3$) moment curve. As pointed out by Yang et al. (2007), the value of m must be determined through consideration of the system being investigated. In our extensive experiments with both simple queueing models and realistic manufacturing systems, we have never encountered a situation where five design points provided an inadequate fit for the moment curves over $[x_L, x_U] = [0.5, 0.95]$ (a throughput range much wider than the range within which a real manufacturing system is typically operated). Hence, we recommend setting $m = 5$ if no reliable information is available to suggest the use of fewer points.

Inputs. Simulation model of the system being investigated; precision level $100\gamma\%$, which is defined as the relative error on the chosen percentiles; throughput range $[x_L, x_U]$; percentage range $[\alpha_L, \alpha_U]$; number of design points m; and initial number of replications N_0.

Outputs. Fitted moment curves $\{\hat{\mu}_\nu(x), \nu = 1, 2, 3\}$ and the inferred variance-covariance information, from which the percentile estimate $\widehat{\mathscr{C}}_\alpha(x)$ and an approximate standard error $\widehat{\mathrm{SE}}[\widehat{\mathscr{C}}_\alpha(x)] = \sqrt{\widehat{\mathrm{Var}}[\widehat{\mathscr{C}}_\alpha(x)]}$ can be provided for $x \in [x_L, x_U]$ and $\alpha \in [\alpha_L, \alpha_U]$.

Stage 0. Initially, N_0 replications are allocated evenly to the two end points x_L and x_U. The three moment curves $\{\hat{\mu}_\nu(x), \nu = 1, 2, 3\}$ are then estimated by fitting models (5) and (7) as described in §3. At this point, the polynomial order of $\{\hat{\mu}_\nu(x), \nu = 1, 2, 3\}$ is equal to zero because of the constraint imposed by the number of design points (two). With the estimated moment models and the inferred variance information, we:

1. determine ΔN, the number of replications to be added to the initial design (see Online Supplement A.6 for the determination of the value of ΔN);

2. find the optimal design $(\mathbf{x}, \boldsymbol{\pi})$ consisting of m points by solving the nonlinear optimization problem (23).

Stage 1. In this stage, we fix the m design points and keep allocating more replications to those points until the desired precision is achieved. Three tasks are to be completed in the following steps:

Step 1. *Run more simulation experiments*. Assign ΔN additional runs to the design points found in the previous stage according to the latest updated loadings $\boldsymbol{\pi}$. Refit the three moment curves and search for an appropriate polynomial order for each fitted model; we follow the forward selection method suggested in Yang et al. (2007). Obtain the estimate of the distribution of cycle time at x_U, $G(t; \hat{a}(x_U), \hat{b}(x_U), \hat{k}(x_U))$ and then estimate the percentile $\mathscr{C}_{\alpha_U}(x_U)$ by inverting the cdf.

Step 2. *Evaluate the precision of* $\widehat{\mathscr{C}}_{\alpha_U}(x_U)$. Estimate the standard error of $\widehat{\mathscr{C}}_{\alpha_U}(x_U)$. If the desired precision is achieved ($2\widehat{\mathrm{SE}}[\widehat{\mathscr{C}}_{\alpha_U}(x_U)]$ is less than $\gamma\%$ of $\widehat{\mathscr{C}}_{\alpha_U}(x_U)$), then move to Step 3. Otherwise, conditional on the current design points, find the value of ΔN at the current point and solve Equation (23) to adjust the loadings $\boldsymbol{\pi}$ of the design according to the latest estimated moment curves. Go back to Step 1.

Step 3. *Evaluate the precision of* $\widehat{\mathscr{C}}_{\alpha_L}(x_U)$. Estimate the standard error of $\widehat{\mathscr{C}}_{\alpha_L}(x_U)$. If the desired precision is achieved ($2\widehat{\mathrm{SE}}[\widehat{\mathscr{C}}_{\alpha_L}(x_U)]$ is less than $\gamma\%$ of $\widehat{\mathscr{C}}_{\alpha_L}(x_U)$), then stop. Otherwise, conditional on the current design points, solve Equation (23) to adjust the loadings $\boldsymbol{\pi}$ of the design according to the latest estimated moment curves. Go back to Step 1.

7. Numerical Evaluation

In this section, we evaluate the performance of the proposed procedure based on queueing models. In our experiments, we have considered the following G/G/1 queueing systems: M/M/1, M/E_2/1, D/E_2/1, and D/M/1. These models cover deterministic, Erlang, and exponential (representing no, moderate, and high variability) distributions for the interarrival and processing times, and they represent a range of cycle time distributions while still being analytically tractable. We use these simple models to allow control of factors that might affect procedure performance; a realistic full factory simulation is studied in the next section.

Not surprisingly, our procedure performs best on M/M/1, where the assumptions concerning the form of moment models and the distribution of cycle times are known to be true. Among these four systems, our procedure has the worst performance on the D/M/1 system. Due to space constraints, we only present the results for M/M/1 and D/M/1.

7.1. Results for Queueing Systems

For both M/M/1 and D/M/1, the true percentiles of cycle time at different throughputs can be analytically computed, and hence the quality of percentile estimation can be easily evaluated. For each model, the proposed procedure was applied 100 times. Then, from each of the 100 macroreplications, selected percentile estimates were compared with their true values.

In our experiments, the throughput range of interest was chosen to be $[x_L, x_U] = [0.7, 0.95]$ and the percentile range $[\alpha_L, \alpha_U] = [0.85, 0.95]$, where we have normalized the throughput so that the maximum system capacity is one. The precision level of the relative error used as the stopping criterion was set at $100\gamma\% = 5\%$ (see §6.2). For all the queueing models considered, the location parameter t_0 (see §4) was set at 0 throughout the throughput range. As already noted, our procedure is able to give percentile estimates $\mathscr{C}_\alpha(x)$ for any point in the two-dimensional region defined by the percentile $\alpha \in [\alpha_L, \alpha_U]$ and throughput $x \in [x_L, x_U]$. We call this region the feasible region. To evaluate the accuracy and precision of the percentile estimation, checkpoints were selected inside this feasible region, as shown in Figure 3. At each of these points, the estimates were compared with the true percentiles of the queueing system.

7.1.1. Point Estimators. All the point estimators for percentiles performed similarly well in terms of deviation from the true value for both M/M/1 and D/M/1. Two types of plots were made to display graphically the 100 realizations of each percentile estimator made at the checkpoints: (i) relative error plots, where the y-axis is defined as

$$\frac{\text{Percentile Estimate} - \text{True Percentile}}{\text{True Percentile}} \times 100\%, \quad (25)$$

and (ii) absolute error plots, in which percentile estimates are plotted around their true values.

Figure 4 shows the percentile estimation results for M/M/1. Figures 4(a)–(c) are relative error plots with the percentile α being 85%, 90%, and 95%, respectively. For these graphs, the x-axis represents throughput rate x, and every point in the graph represents the relative deviation at corresponding checkpoint (α, x), calculated by Equation (25) from one of the 100 macroreplications. Notice that a very high proportion of the relative deviations of the percentile estimates at the selected checkpoints are within 5% (the precision level $100\gamma\%$ imposed prior to experimentation). Figures 4(a′)–(c′) are the absolute error plots, in which the solid curve represents a piecewise linear version of the true percentile curve across the throughput range and the percentile estimates are plotted in absolute units. From these plots, it is evident that the variability of the percentile estimators at the highest throughput $x_U = 0.95$ is the most pronounced and, as explained in §6.2, it has been well controlled in our procedure.

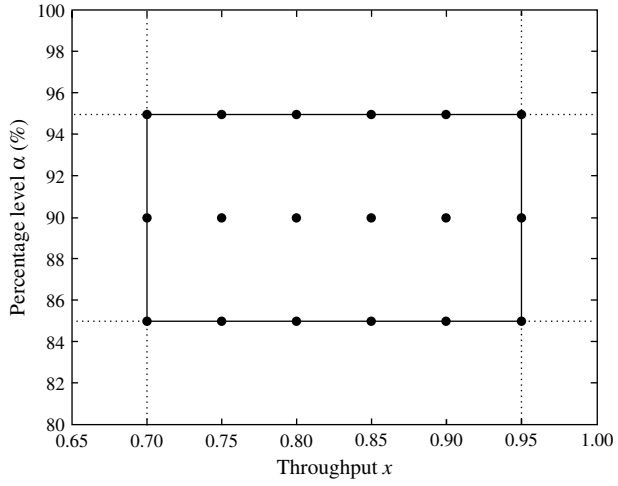

Figure 3 Checkpoints Selected in the Feasible Region

Figure 5 shows an analogous plot for the D/M/1 system and similar conclusions can be drawn, although the performance is not as good as the M/M/1, especially when the throughput is at $x = 0.95$.

7.1.2. Standard Error (SE). An estimator of the standard error $\mathrm{SE}[\widehat{\mathscr{C}}_\alpha(x)] = \sqrt{\mathrm{Var}[\widehat{\mathscr{C}}_\alpha(x)]}$ is provided for each percentile estimator $\widehat{\mathscr{C}}_\alpha(x)$ by the procedure described in §6.3. Our goal in this section is to evaluate the quality of the SE estimator. Tables 1 and 2 show the results for M/M/1 and D/M/1, respectively. The column labeled "Sample stdev" is the sample standard deviation of the percentile point estimators calculated from the 100 realizations of the percentile estimator; therefore, it is an unbiased estimator of the true standard error. The "Average SE" column is the average of the 100 standard error estimators $\widehat{\mathrm{SE}}[\widehat{\mathscr{C}}_\alpha(x)]$, each one of which is estimated from within a single macroreplication.

Table 1 shows that for M/M/1, the mean of the standard error estimate in the "Average SE" column is close to, but consistently less than, the unbiased external estimate of the standard deviation found in the "Sample stdev" column. The underestimation trend is more apparent for the D/M/1. Nevertheless, the estimated standard error $\widehat{\mathrm{SE}}[\widehat{\mathscr{C}}_\alpha(x)]$ provided by the procedure can still give the user a rough idea about how variable the percentile estimator is.

In the absence of any knowledge about the distribution of the percentile estimators, it would be natural to attempt to form a 95% confidence interval for the percentile by using

$$\widehat{\mathscr{C}}_\alpha(x) \pm 1.96 \times \widehat{\mathrm{SE}}[\widehat{\mathscr{C}}_\alpha(x)]. \quad (26)$$

For M/M/1, Equation (26) works well in terms of coverage and gives a conservative confidence interval. However, for D/M/1, the coverage probability was lower than the nominal level. This can be explained by underestimation of the standard error and nonnormality of the percentile of cycle time estimator. In the

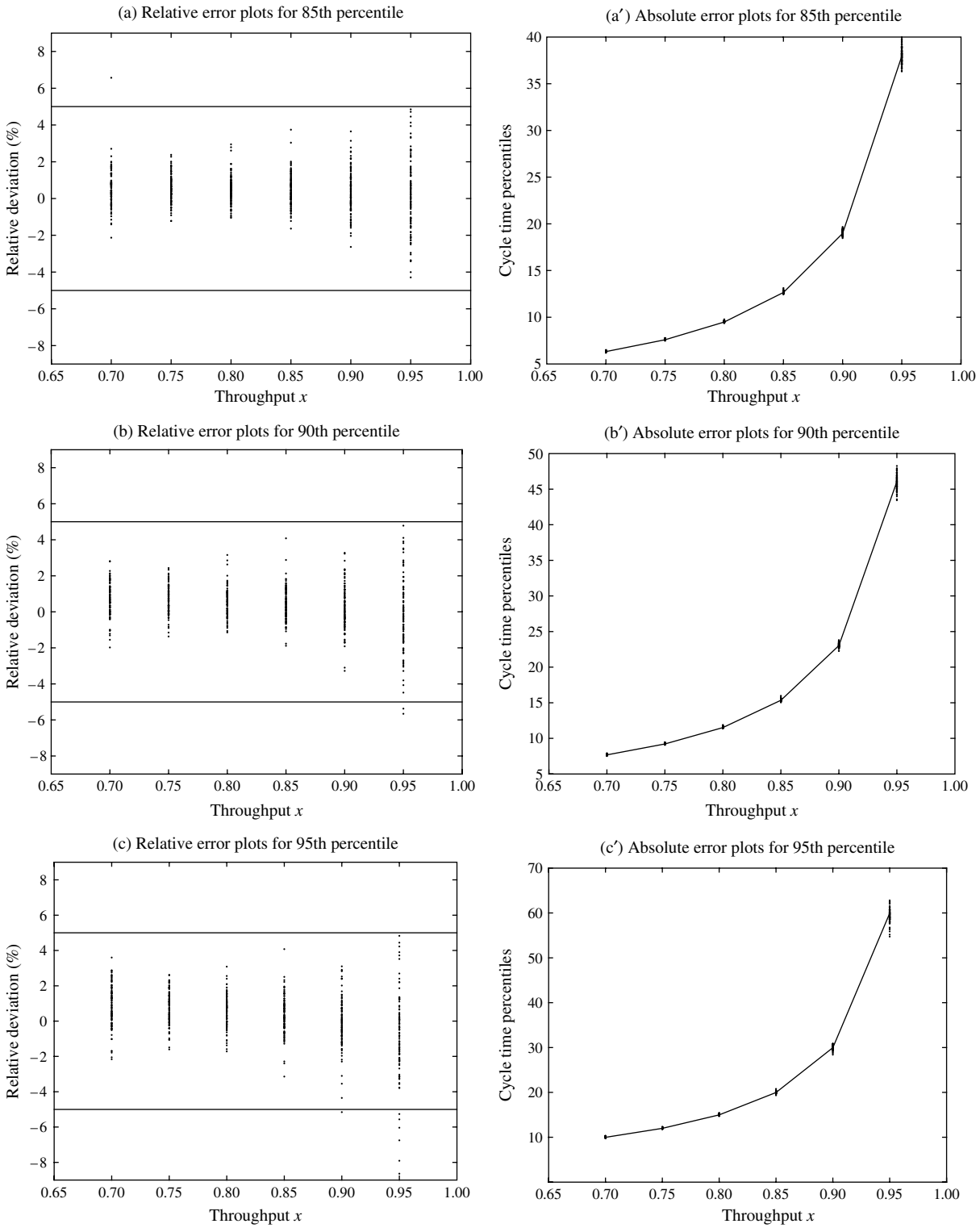

Figure 4 Plots of the Percentile Estimates for M/M/1 (100 Macroreplications)

case with D/M/1, it appears that nonnormality is the dominant factor.

7.2. Summary of Results

Through experimentation with queueing models, it has been shown that the proposed procedure has the potential to be effective in providing accurate and precise percentile estimators. By controlling the relative standard error of the percentile estimators at the upper end of the throughput range, high precision has been achieved for estimators of percentiles throughout the feasible region.

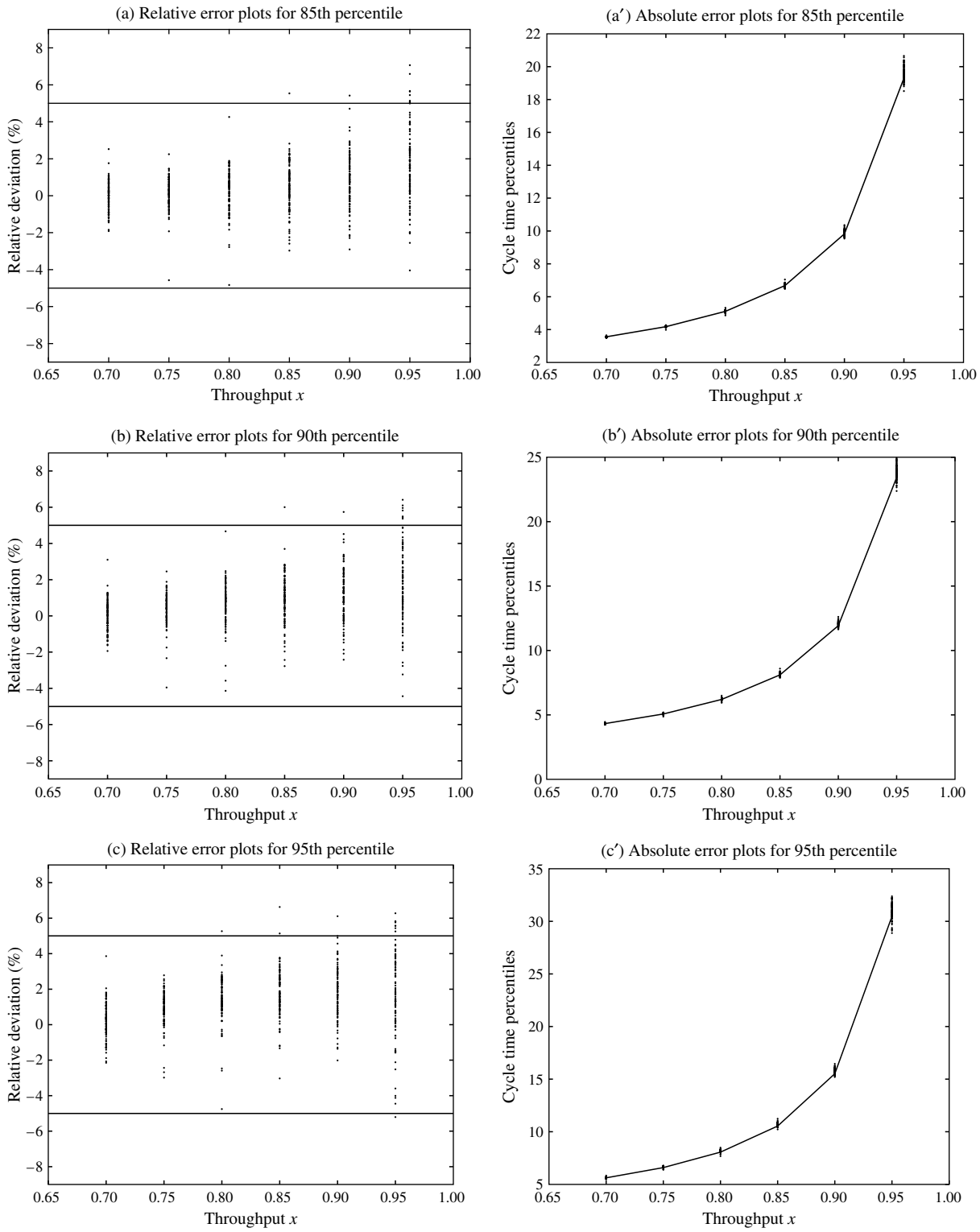

Figure 5 Plots of the Percentile Estimates for D/M/1 (100 Macroreplications)

For each percentile estimator, an estimate of the standard error is also provided that gives the user a sense of its variability. However, in the scope of our work, there is not sufficient information to draw any conclusion regarding the distribution (or limiting distribution) of the percentile estimators. Thus, no reliable confidence interval can be created based on the standard error estimation.

Of course, real manufacturing systems are networks of queues. To stress our procedure in a realistic setting, we next consider a semiconductor fabrication model.

Table 1 Estimated Standard Errors of Percentile Estimates for M/M/1

	85th percentile		90th percentile		95th percentile	
TH x	Sample stdev	Average SE	Sample stdev	Average SE	Sample stdev	Average SE
0.70	0.055	0.053	0.072	0.067	0.117	0.105
0.75	0.056	0.053	0.071	0.067	0.112	0.106
0.80	0.070	0.064	0.089	0.080	0.137	0.125
0.85	0.110	0.100	0.139	0.126	0.213	0.190
0.90	0.225	0.213	0.282	0.270	0.430	0.399
0.95	0.759	0.734	0.950	0.926	1.439	1.361

8. An Example of Manufacturing Systems

In this section, we apply the proposed procedure to a semiconductor wafer fab simulation model representing a full manufacturing factory. The model (testbed data set #1 created in Factory Explorer) was taken from the website of the Modeling and Analysis for Semiconductor Manufacturing Lab at Arizona State University (http://www.eas.asu.edu/~masmlab/). The model is designed to process two types of jobs, Prod1 and Prod2, with each type being released into the system at a constant (deterministic) rate. Jobs of different types follow different process steps and thus have different cycle time distributions. The primary sources of variability are machine failures and repairs.

In our experiments, the product mix (expressed as a percentage of production dedicated to each product type) is set as 66.7% Prod1 and 33.3% Prod2. We investigate the CT-TH relationships for the two types of products separately. For the percentile of cycle time curves to be generated, the independent variable, throughput, was defined as the overall production rate (as a percentage of the capacity) of both types of jobs that are mixed with a constant ratio. Note that the cycle time distribution for a particular type of product also depends on the product mix. In this paper, we restrict ourselves to situations where the jobs are released with fixed product mix. The construction of CT-TH-PM (product mix) surfaces is the subject of ongoing research.

Table 2 Estimated Standard Errors of Percentile Estimates for D/M/1

	85th percentile		90th percentile		95th percentile	
TH x	Sample stdev	Average SE	Sample stdev	Average SE	Sample stdev	Average SE
0.70	0.028	0.024	0.035	0.029	0.056	0.046
0.75	0.035	0.026	0.044	0.032	0.065	0.048
0.80	0.060	0.042	0.074	0.052	0.103	0.079
0.85	0.090	0.065	0.106	0.082	0.142	0.125
0.90	0.144	0.105	0.171	0.132	0.224	0.195
0.95	0.399	0.373	0.503	0.471	0.733	1.696

As already indicated, our objective is to estimate the CT-TH percentile curves for both Prod1 and Prod2 based on a single set of simulation runs. In our experiments, we chose to drive the simulation by the precision of Prod2. After accumulating sufficient data for the estimation of Prod2, we estimate the percentile curves for both products. For the implementation of our procedure, the range of throughput was chosen to be $[0.7, 0.95]$, where "1" corresponds to system capacity, and the percentile range was chosen to be $[85\%, 95\%]$. The precision level was set at $100\gamma\% = 1\%$ and the number of design points was set at $m = 5$. In the remainder of this section, we will discuss the results for Prod2 in detail; similar conclusions can be drawn for Prod1.

As explained in §4, we allow the user to introduce a fourth parameter t_0, which represents the lower bound of the GGD representing cycle time. There are at least three different ways to set the location parameter $t_0(x)$ for the GGD distribution fitted at throughput rate x: In the absence of any knowledge about the lower bound of cycle time, zero can always be used as the default value of $t_0(x)$ for any x. Next, the pure minimum processing time of the product being considered, which is usually available to the user, can be safely used for the location parameter throughout the range of throughput. These two simple settings can provide good percentile estimates, as will be explained later. However, to achieve better precision, we recommend using a third method that imposes a much tighter lower bound on the cycle time. As already noted, the distribution of steady-state cycle time varies with the range of throughput, as illustrated in Figure 6, which gives histograms of 50,000 individual cycle times for Prod2 at two throughput levels, 0.7 and 0.95. Although the theoretical pure processing time is not a function of throughput, the impact of queueing is to make the effective minimum cycle time much larger, from about 450 hours at $x = 0.7$ to about 680 hours at $x = 0.95$. As can be seen from the graphs, at high throughput levels, the steady-state cycle times are bounded well away from their pure processing time (223 hours), the theoretical minimum. Our experiment results have shown that using the empirical minimum brings significant improvement to the percentile estimates. To obtain the empirical minimum of cycle time at any $x \in [x_L, x_U]$, we propose the following method: at the five design points where simulation experiments are performed, the empirical minimal cycle time can be easily obtained. For other points, we use linear interpolation based on the five minimums at those design points.

To evaluate the percentile estimates, the checkpoints as shown in Figure 3 are used again. Because the true percentiles at those points are unknown, substantial additional data were collected at the checkpoints to obtain the "nearly true" estimates for

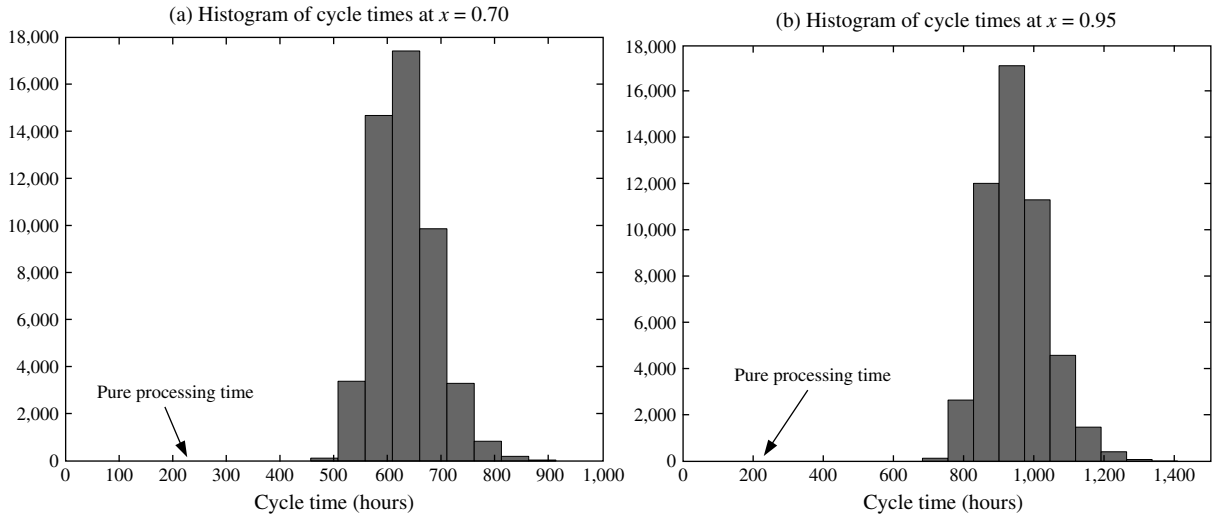

Figure 6 Histograms of Cycle Times (50,000 Cycle Times in Each Graph)

percentiles of cycle times. We present in Table 3 the numerical results for Prod2 with the throughput-sensitive location parameter being the empirical minimum at each throughput level. In Table 3, absolute deviations (defined as percentile estimates minus "true percentiles") and relative deviations (defined by Equation (25)) of the estimated percentiles are given as well as an estimate of the standard error of the percentile estimator.

The point estimates of percentiles are good in terms of the relative error; with the exception of two checkpoints at which the absolute value of the relative error is slightly above 1%, the deviations are well within the desired precision. From the sign of the deviations, it is obvious that at the lowest throughput rate the percentiles are overestimated, and that at the higher throughput levels the percentiles are underestimated. We conjecture that the bias in percentile estimates was inherited from the moment estimates. We compared the estimated moments obtained from the procedure with (very precisely estimated) "true" moments and detected the same pattern: the first three moments are all slightly overestimated at the lower end of the throughput range while being slightly underestimated at the other throughput levels. As explained in Yang et al. (2007), this consistent pattern in the estimation on moment estimates is what we expected. Due to the form of the moment model (4), the fitted moment curve is likely to increase smoothly and intersect with the underlying true moment curve at some point within the throughput range. In this case, for all three moment curves, the intersection point is somewhere close to the lower end. In other words, at throughput levels lower than the intersection, we overestimate the moments, and at throughput levels higher than the intersection, we underestimate the moments.

Based on the same data set, percentile estimates were also obtained using different settings of the location parameter: (i) $t_0(x) = 0$ and (ii) $t_0(x)$ equal to the pure processing time. The percentile estimates obtained from these two settings are still fairly good in terms of relative error. For case (i), the relative error at all the checkpoints was within 3.5%. The precision achieved in case (ii) is slightly better.

9. Summary

Estimating percentiles of cycle time via simulation is difficult due to the high variability of percentile estimators and the diversity of cycle timedistributions.

Table 3 Results for the Semiconductor Manufacturing Model

	85th percentile			90th percentile			95th percentile		
TH x	Abs. dev.	Rel. dev. (%)	Est. SE	Abs. dev.	Rel. dev. (%)	Est. SE	Abs. dev.	Rel. dev. (%)	Est. SE
0.70	4.62	0.66	0.84	5.49	0.76	1.21	5.67	0.76	1.96
0.75	−2.53	−0.35	0.63	−3.27	−0.44	0.87	−1.98	−0.26	1.39
0.80	−6.88	−0.91	0.47	−7.77	−1.00	0.66	−7.34	−0.91	1.07
0.85	−6.93	−0.86	0.63	−7.12	−0.86	0.90	−6.87	−0.80	1.48
0.90	−3.98	−0.45	0.92	−3.38	−0.37	1.17	−1.95	−0.21	1.71
0.95	−9.60	−0.91	1.64	−10.04	−0.94	2.31	−11.78	−1.06	3.73

This paper proposes a new methodology for estimating multiple cycle time percentiles throughout the throughput range of interest based on a single set of simulation runs. It has been shown through experiments on queueing models such as M/M/1 and D/M/1 and a real semiconductor manufacturing simulation that the multistage procedure provides good point estimators for percentiles of cycle time.

As a by-product of our research, our moment curves can also be used to obtain other summary statistics such as the standard deviation of cycle time. Our fitting techniques could also be employed to estimate simulation-generated "clearing functions" (a type of throughput versus work-in-process inventory curve; see Asmundsson et al. 2006). Clearing functions are used in production planning optimization models to allow the model to assess the impact on work in process and throughput of altering the production plan. This is the subject of ongoing research.

Acknowledgments

This research was supported by the National Science Foundation (Grant DMI-0140385) and the Semiconductor Research Corporation (Grant 2004-0J-1225). Additional thanks go to Professors John Fowler and Gerald Mackulak from Arizona State University. The authors thank the area editor, associate editor, and referees for help in clarifying the presentation.

References

Allen, C. 2003. The impact of network topology on rational-function models of the cycle time-throughput curve. Honors thesis, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL. http://users.iems.northwestern.edu/nelsonb/Publications/CarlAllenThesis.pdf.

Ashkar, F., B. Bobée, D. Leroux, D. Morisette. 1988. The generalized method of moments as applied to the generalized gamma distribution. *Stochastic Hydrology Hydraulics* **2** 161–174.

Asmundsson, J., R. L. Rardin, R. Uzsoy. 2006. Tractable nonlinear production planning models for semiconductor wafer fabrication facilities. *IEEE Trans. Semiconductor Manufacturing* **19** 95–111.

Billingsley, P. 1999. *Convergence of Probability Measures*, 2nd ed. John Wiley & Sons, New York.

Buzacott, J. A., J. G. Shanthikumar. 1993. *Stochastic Models of Manufacturing Systems*. Prentice Hall, Upper Saddle River, NJ.

Chen, E. J., W. D. Kelton. 1999. Simulation-based estimation of quantiles. P. A. Farrington, H. B. Nembhard, D. T. Sturrock, G. W. Evans, eds. *Proc. 1999 Winter Simulation Conf.*, Institute of Electrical and Electronics Engineers, Piscataway, NJ, 428–434, http://www.informs-cs.org/wsc99papers/059.PDF.

Cheng, R. C. H., J. P. C. Kleijnen. 1999. Improved design of queueing simulation experiments with highly heteroscedastic responses. *Oper. Res.* **47**(5) 762–777.

Chien, C., D. Goldsman, B. Melamed. 1997. Large-sample results for batch means. *Management Sci.* **43**(9) 1288–1295.

Fowler, J. W., S. Park, G. T. Mackulak, D. L. Shunk. 2001. Efficient cycle time-throughput curve generation using a fixed sample size procedure. *Internat. J. Production Res.* **39** 2595–2613.

Glynn, P. W., D. L. Iglehart. 1986. Estimation of steady-state central moments by the regenerative method. *Oper. Res. Lett.* **5** 271–276.

Henderson, S. G. 2001. Mathematics for simulation. B. A. Peters, J. S. Smith, D. J. Medeiros, M. W. Rohrer, eds. *Proc. 2001 Winter Simulation Conf.*, Institute of Electrical and Electronics Engineers, Piscataway, NJ, 83–94.

Hopp, W. J., M. L. Spearman. 2001. *Factory Physics*: *Foundations of Manufacturing Management*, 2nd ed. Irwin, Chicago.

Johnson, R., F. Yang, B. E. Ankenman, B. L. Nelson. 2004. Nonlinear regression fits for simulated cycle time vs. throughput curves for semiconductor manufacturing. *Proc. 2004 Winter Simulation Conf.*, Institute of Electrical and Electronics Enigneers, Piscataway, NJ, 1951–1955, http://www.informs-cs.org/wsc04papers/260.pdf.

Law, A. M., W. D. Kelton. 2000. *Simulation Modeling and Analysis*, 3rd ed. McGraw-Hill, New York.

Lehmann, E. L. 1999. *Elements of Large-Sample Theory*. Springer-Verlag, New York.

McNeill, J. E., G. T. Mackulak, J. W. Fowler. 2003. Indirect estimation of cycle time quantiles from discrete event simulation models using the Cornish-Fisher expansion. S. Chick, P. J. Sánchez, D. Ferrin, D. J. Morrice, eds. *Proc. 2003 Winter Simulation Conf.*, Institute of Electrical and Electronics Engineers, Piscataway, NJ, 1377–1382, http://www.informs-cs.org/wsc03papers/173.pdf.

Meyn, S. P., R. L. Tweedie. 1993. *Markov Chains and Stochastic Stability*. Springer, New York.

Park, S., J. W. Fowler, G. T. Mackulak, J. B. Keats, W. M. Carlyle. 2002. D-optimal sequential experiments for generating a simulation-based cycle time-throughput curve. *Oper. Res.* **50** 981–990.

Rose, O. 1999. Estimation of the cycle time distribution of a wafer fab by a simple simulation model. *Proc. 1999 Internat. Conf. Semiconductor Manufacturing Oper. Model. Simulation, San Francisco,* 133–138.

Schömig, A., J. W. Fowler. 2000. Modelling semiconductor manufacturing operations. *Proc. 9th ASIM Simulation Production Logist. Conf.*, Fraunhofer Institut for Production Systems and Design Technology (IPK), Berlin, 55–64.

Stacy, E. W. 1962. A generalization of the gamma distribution. *Ann. Math. Statist.* **33** 1187–1192.

Whitt, W. 1989. Planning queueing simulations. *Management Sci.* **35** 1341–1366.

Yang, F., B. E. Ankenman, B. L. Nelson. 2007. Efficient generation of cycle time-throughput curves through simulation and metamodeling. *Naval Res. Logist.* **54** 78–93.