WILEY InterScience
DISCOVER SOMETHING GREAT

Efficient Generation of Cycle Time-Throughput Curves through Simulation and Metamodeling

Feng Yang,[1] Bruce Ankenman,[2] Barry L. Nelson[2]

[1] *Department of Industrial and Management Systems Engineering, West Virginia University, Morgantown, West Virginia 26506-6107*

[2] *Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, Illinois 60208-3119*

Received 20 June 2005; revised 20 June 2006; accepted 29 June 2006
DOI 10.1002/nav.20188
Published online 29 August 2006 in Wiley InterScience (www.interscience.wiley.com).

Abstract: A cycle time-throughput (CT-TH) curve, which quantifies the relationship of long-run average cycle time to throughput rate, plays an important role in strategic planning for manufacturing systems. In this paper, a nonlinear regression metamodel supported by queueing theory is developed to represent the underlying CT-TH curve implied by a manufacturing simulation model. To estimate the model efficiently, simulation experiments are built up sequentially using a multistage procedure. Extensive numerical experiments are presented to demonstrate the effectiveness of the proposed procedure. © 2006 Wiley Periodicals, Inc. Naval Research Logistics 54: 78–93, 2007

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

1. INTRODUCTION

Planning for manufacturing, either at the factory or at the enterprise level, requires answering "what if" questions involving (perhaps a very large number of) different scenarios for product mix, production targets, and capital expansion. A key performance measure for evaluating these scenarios is the implied *cycle time*, a random variable representing the time required for a job or lot to traverse a given routing in a production system (e.g., [5]). A company can control cycle time by controlling the rate at which lots are started in the factory (lot-start rate or, equivalently, throughput rate). Computer simulation is a powerful tool for estimating long-run average cycle time for given operating conditions. However, this is only a snapshot of the system's performance profile; a more comprehensive picture is provided by a cycle time-throughput (CT-TH) curve over a range of throughput rates. Analytically tractable queueing models can produce such curves, but they invariably require significant simplification of the actual manufacturing system.

This paper addresses the generation of simulation-based CT-TH curves for the long-run average cycle time of systems without batch processing policies or for systems with batch policies operating in the range of throughput for which mean cycle time is monotonically increasing (at low throughput rates, cycle time can actually decrease due to the reduced delay required for a batch to form). The goal is to provide a methodology that requires nothing of the analyst beyond the simulation model, a throughput range of interest, and a measure of the required precision for the estimated curve. The result is a complete response profile like that provided by a tractable queueing model, but with the fidelity of a simulation model.

There is already a substantial literature on fitting CT-TH curves to simulation responses, including [3, 4, 9]. However, their work either assumes more knowledge of the form of the CT-TH curve than is actually available in practice or provides an experiment design to work within a given computing budget rather than being driven by a desired precision. Our experience, derived from extensive simulations of real and stylized manufacturing systems [1, 6], reveals that CT-TH curves deviate substantially from the forms assumed in the literature. Further, on many occasions, there is no natural budget. Stated differently, there is sufficient time to estimate the CT-TH curve (which is only done once), so it is more critical to insure that it is accurate and precise than it is to estimate it quickly. Both of these issues emphasize a need to be more adaptive than available procedures.

Correspondence to: B. Ankenman (ankenman@northwestern.edu)

© 2006 Wiley Periodicals, Inc.

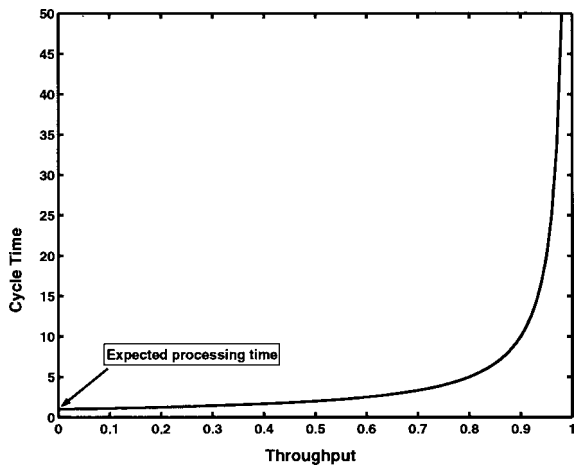

Figure 1. A generic CT-TH curve.

A highly adaptable method used by van Beers and Kleijnen [10] for estimating CT-TH curves is Kriging, which is a weighted average interpolation approach. Kriging is most appropriate for situations in which the user has little or no knowledge of the true underlying surface to be modeled. However, in our case, we have substantial justification for a particular parametric model family for the CT-TH curves, as will be shown in Section 1.1, so it is highly beneficial to exploit that information. Therefore, our goal is to make fitting this parametric model adaptive in terms of the parameters that we fit (to improve accuracy) and the experiment we design (to control precision).

1.1. Statement of the Problem

Without loss of generality, we assume that the capacity of the system is 1, so that the system throughput x is equivalent to the traffic intensity. In manufacturing systems where product batching is not implemented, CT-TH curves normally follow the shape in Figure 1. When the throughput is close to zero, cycle time is almost equal to the pure processing time; as the throughput approaches the upper limit of stability, the cycle time increases nonlinearly and tends to infinity before actually reaching full capacity. Although in theory the system throughput can take any value in $(0, 1)$, manufacturers are usually interested in a much narrower range of throughput that is likely to deliver a competitive output rate as well as an acceptable cycle time. We assume that the range of interest, say $[x_L, x_U]$, is given.

The objective of our research is to estimate a CT-TH curve via sequential experimentation. We suppose that the simulation experiment is made up of a number of independent simulation runs. The distribution of the output response (average cycle time) from a replication, Y, is dependent on the system throughput, x. This input–output relationship can be represented by the following metamodel, which is called the expected cycle-time (ECT) model,

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

where

$$\mu_t(x, \mathbf{c}, p) = \frac{\sum_{\ell=0}^{t} c_\ell x^\ell}{(1-x)^p} \tag{2}$$

is the model of the expected value of $Y(x)$. We use the subscript t to denote the degree of the polynomial factor in x. Both p and t, as well as the vector $\mathbf{c} = (c_1, c_2, \ldots, c_t)$, are unknown parameters in the model. Our focus is on long-run average, or "steady-state" expected cycle time. The form of model (2) is motivated by heavy traffic analysis of queueing systems. Specifically, Whitt [11] shows that as $x \to 1$ the expected cycle time for most queueing systems takes this form. In addition, the expected cycle time for several simple queueing systems discussed later in this section take this form for all values of x. Our focus on the steady-state expected cycle time implies that initialization bias has somehow been mitigated (perhaps by deleting some of the cycle-time observations at the beginning of each replication). We let the true expected value be denoted by $\mu(x, \mathbf{c}, p)$ and also define the following notation:

$Y_j(x)$: the output from the jth independent and identically distributed (i.i.d.) replication at throughput level x, which is $Y_j(x) = H(x)^{-1} \sum_{h=1}^{H(x)} CT_{jh}(x)$. Here $CT_{jh}(x)$ represents the individual cycle time of the hth job in the jth simulation replication, whose distribution depends on x only; and $H(x)$ is the selected number of jobs simulated in steady state for simulations at x. We assume that for a given throughput x the individual cycle times $CT_{jh}(x)$ are identically distributed, although not in general independent within a replication.

$n(x)$: number of replications placed at the input level x.

$\varepsilon_j(x)$: error term with expectation 0 and variance, $\sigma^2(x)$. The dependence of $\sigma^2(x)$ on x is represented by the so-called "variance model," which will be discussed later.

Remark. In our experiments, the simulation replications are performed independently without common random numbers because the sequential nature of the experimentation makes it nearly impossible to synchronize them effectively.

The expectation function (2) can be decomposed into two components, $f(x) = 1/(1-x)^p$, which accounts for the unbounded behavior of the CT-TH curve as the system is pushed close to capacity; and the polynomial function in the numerator. The form of model (2) is motivated by queueing results for a number of elementary stochastic models for

which (2) is precisely correct for all x and the heavy traffic analysis mentioned above. Consider, for example, the M/M/1 queue. The steady-state mean of cycle time is

$$\mathrm{E}[Y_j(x)] = \frac{1}{1-x}. \tag{3}$$

This relationship is of the form of the model we suggest. Another example is the $G/G/1$ queueing model, which can be regarded as the simplest representation for real manufacturing systems. An approximation of the cycle time for the $G/G/1$ is given by Hopp and Spearman [5] as

$$\mathrm{E}[Y_j(x)] \doteq \frac{1 + \left(\frac{C_a^2 + C_e^2}{2} - 1\right)x}{1-x}, \tag{4}$$

where C_a^2 and C_e^2 are the coefficients of variation of the interarrival time and effective processing time, respectively. The symbol $\doteq$ is used in this paper to represent "approximately equal to." This model is also of the form that we assume. Note that for both (3) and (4) we would set $p = 1$ in (2); while this is the correct value for many simple queueing models, it is not universally appropriate as demonstrated by the empirical examples provided in [1] and [6].

With the shape of a CT-TH curve defined by the ECT model (1), we propose a multistage procedure for efficiently collecting data to fit the curve. Before a detailed description of the procedure is given in Section 4, we first review Cheng and Kleijnen's [3] approach to the same problem. It should be noted that Cheng and Kleijnen did not specifically investigate the CT-TH curve; the response variable in their paper was expected waiting time, which differs from the expected cycle time by a constant, the expected pure processing time.

1.2. Summary of Cheng and Kleijnen's Method

Cheng and Kleijnen [3] developed a procedure to improve the design of simulation experiments for the purpose of estimating a CT-TH curve with a limited computational budget. In their paper, a linear regression model is developed to represent the CT-TH relationship by using the model form (1), but assuming p is known (or equivalently that the function $f(x) = 1/(1-x)^p$ is completely specified). The variance of the error term depends on x as

$$\mathrm{Var}[\varepsilon(x)] = [g(x)\sigma]^2, \tag{5}$$

where $g(x)$ is also assumed known from asymptotic theory or other considerations.

The design of the experiment consists of the location of the design points $\mathbf{x} = (x_1, x_2, \ldots, x_m)$ and the fraction of a total of N replications assigned to those points $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_m)$. The design is constructed to minimize

$$PM_0 = \frac{\int_{x_L}^{x_U} w(x)\mathrm{Var}[\mu_t(x, \hat{\mathbf{c}}, p)]dx}{\int_{x_L}^{x_U} w(x)dx} = \sigma^2 \sum_{i=1}^{m} \left(\frac{a_i(\mathbf{x})}{r(x_i)}\right)^2 n_i^{-1}, \tag{6}$$

where $w(x)$ is a weight function chosen by the experimenter. The performance measure PM_0 is the weighted-average variance of the estimated expected response over the throughput range of interest. In (6), $r(x_i) = f(x_i)/g(x_i)$, $a_i(\mathbf{x})$ is a function of the design vector $\mathbf{x}$ (for the specific form of a_i, see [3]), σ^2 is the constant term in the error variance (5), and $(n_1, n_2, \ldots, n_m) = N\boldsymbol{\pi}$ is the integer vector representing the number of replications assigned to the design points. Obviously, PM_0 can be expressed in units of σ^2/N, and the resulting normalized measure simplifies to

$$PM = \sum_{i=1}^{m} \left(\frac{a_i(\mathbf{x})}{r(x_i)}\right)^2 \pi_i^{-1}. \tag{7}$$

This measure will also be used as the design criterion in our procedure. To solve this optimization problem Cheng and Kleijnen relaxed the constraint that $N\boldsymbol{\pi}$ be integer, making the π_i continuous decision variables, and then rounded $N\pi_i$ to obtain the actual allocation; we will adopt a similar approach.

The CK Procedure for fitting the model (1) can be summarized as follows. Given $f(x)$, $g(x)$, a maximum value of t, and a fixed budget of N replications, find the optimal design $(\mathbf{x}, \boldsymbol{\pi})$ by minimizing PM. With the design points $\mathbf{x}$ fixed, carry out simulation experiments sequentially and adjust the allocation $\boldsymbol{\pi}$. Once the total number of runs has been exhausted, use backward selection to decide the appropriate polynomial order of model (1) and obtain the fitted curve.

Cheng and Kleijnen's method leaves open the question of how to specify $f(x)$ and $g(x)$, which affect the design of the experiment and, more importantly, the adequacy of model (1) to represent the true CT-TH curve. When these two functions are known, CK is highly effective and efficient and works within a fixed budget, which our procedure is not designed to do. However, for complicated manufacturing systems, there is not likely to be sufficient information to infer such characteristics. In other words, obtaining good choices for $f(x)$ or $g(x)$, although not impossible, is difficult in practice. Further, we have strong empirical evidence [1, 6] that the $f(x)$ and $g(x)$ used by Cheng and Kleijnen can be far from correct in realistic manufacturing simulations. We next discuss the consequences of misspecifying $f(x)$ and $g(x)$ and the improvements we have made in this regard.

The most important characteristic of a CT-TH curve is its sensitivity to throughput. Note that the rate of increase of the

curve is dominated by $f(x)$ (or more specifically the value of p) as x approaches 1. Since they consider $f(x)$ to be given, Cheng and Kleijnen depend on the polynomial numerator of (2) to adjust the fitted curve for misspecification of $f(x)$. They argue that the error in p can be corrected by adding more terms to the polynomial, and therefore the desired accuracy can always be achieved. Unfortunately, increasing the polynomial order t means increasing the number of unknown parameters in the model, and therefore more design points must be included for the purpose of estimation. This implies that significant computing effort might be required for no reason other than adjusting for the misspecification of p, which may not be the most efficient way to make use of limited resources. Also, a model incorporating a high-order polynomial may not preserve the monotonicity of the curve throughout the range of interest, which is known to be a property of the CT-TH curve.

In light of these issues, we incorporate p as an unknown parameter. Rather than correcting for misspecification using the polynomial numerator, our attention is directed toward obtaining an accurate and precise estimate of p so that the polynomial order can stay as low as possible. This also tends to produce a well-behaved curve in terms of preserving monotonicity. At the same time, it makes possible a good fit with a relatively small number of unknown parameters and therefore better uses the simulation effort at a smaller number of design points.

In the CK Procedure, steps are taken to recover, to some extent, from the effect of an incorrectly chosen $g(x)$. Thus, a misspecified $g(x)$ does not seriously hurt the efficiency of the CK Procedure aside from a non-optimal location of design points. Nevertheless, making a better choice of design points is worthwhile if it can be accomplished easily. In our proposed procedure, a parametric form is assumed for $g(x)$, based on queueing theory, and a small amount of experimental effort is expended to fit it. This leads to a better experiment design.

In summary, the CK Procedure requires more prior knowledge of the system than can usually be assumed in practice. Therefore, it is of practical interest to develop a method where little or no prior information is required for the implementation. The procedure suggested in this paper provides such an alternative, while also driving the process by a required precision rather than a fixed computing budget.

The remainder of the paper is organized as follows: Section 2 discusses the challenges introduced when we allow p in (2) to be a parameter and consider $g(x)$ in (5) to be unknown, while Section 3 describes how we determine the experiment design for this more general model. In Section 4 we assemble all of these pieces into a comprehensive description of the procedure, and we evaluate the procedure in Section 5. The paper concludes with an illustration using a realistic manufacturing simulation and a summary.

2. ISSUES RELATED TO NONLINEAR REGRESSION

The key consequence of treating p in model (1) as an unknown parameter is that fitting the model becomes a nonlinear estimation problem. In this section, we discuss the issues that arise in fitting the nonlinear regression model and how we address them.

2.1. The Error Term

Since one of our goals is to continue to collect simulated data until an accurate model, estimated to a prespecified precision, is obtained, we must be able to derive valid statistical inference about our fitted model. Nonlinear regression inference is based on specific assumptions about the error term, usually that it is normally distributed with zero mean, constant variance, and independent across replications [2]. Independence across replications can be assured by assignment of random number streams. Normality of the response Y can be justified by appealing to the Central Limit Theorem for weakly dependent random variables; indeed, the average of a large number of individual cycle times is approximately normally distributed and thus so is the error term. That leaves only the constant variance assumption.

The variability of individual cycle times increases dramatically as throughput approaches its capacity, and this drives the variance of $Y_j(x)$, the average cycle time, to infinity as well. Thus, we must stabilize the variance to use standard methods of statistical inference. Our variance model is based on queueing analysis. As the traffic intensity approaches 1, the asymptotic variance of the sample mean of some queueing systems is known to be well approximated by a model of the form [11]

$$\begin{aligned}\sigma_A^2(x) &= \lim_{H(x)\to\infty} H(x)\mathrm{Var}[Y_j(x)] \\ &= \lim_{H(x)\to\infty} H(x)\mathrm{Var}\left[H(x)^{-1}\sum_{h=1}^{H(x)} CT_h(x)\right] \\ &\doteq \frac{\sum_{k=0}^{K} b_k x^k}{(1-x)^{2q}}. \qquad (8)\end{aligned}$$

Note that the asymptotic variance $\sigma_A^2(x)$ is not the marginal variance of the individual cycle times, $\mathrm{Var}[CT_h(x)]$, but is related to the variance of the sample mean cycle time in the following way: For simplicity, set $H(x) = H$ for all values of x. Recall that the output $Y_j(x)$ is the average of $\{CT_h(x); h = 1, 2, \ldots, H\}$, the cycle times of all the products simulated in a replication at design point x. Thus, for large H, $\sigma^2(x) \doteq \sigma_A^2(x)/H$, and the variance of $Y_j(x)$ has approximately the same form as the right-hand side of (8). For x close to 1, this

suggests the model

$$\sigma^2(x) = [g(x)\sigma]^2 \doteq \frac{\sigma^2}{(1-x)^{2q}}, \tag{9}$$

which is the model that CK use, but they assume q is known.

For many simple queueing models $q = 2$, but we have seen empirically that, for more complicated queueing networks, it can be markedly different from 2 [1, 6]. To estimate the error variance, $\sigma^2(x)$, we use the obvious estimator $S^2(x)$, the sample variance estimated from the i.i.d. replications taken at x. For the purpose of estimating q, we use the variance model

$$S^2(x) = \frac{\sigma^2}{(1-x)^{2q}} \cdot \tau(n(x)), \tag{10}$$

where $\tau(n(x)) \sim \chi^2_{n(x)-1}/(n(x)-1)$ is a multiplicative error (see Appendix A.1 for more information) that depends on $n(x)$, the number of replications performed at throughput rate x. This provides a good approximation in the range of throughput we are investigating, say, $[0.5, 0.95]$. The model is appealing because (10) can be transformed into a linear regression by taking the logarithm

$$\log S^2(x) = \log \sigma^2 - 2q \log(1-x) + v(x). \tag{11}$$

This transformed variance model (11) is linear; thus, in our procedure, (11) will be the model that is fitted directly with the ordinary least square method. Note that if $H(x)$ is chosen to vary with x, then $S^2(x)$ in (11) must be replaced by $S^2(x)H(x)$ to obtain a valid fit.

In model (11), σ^2 is just a nuisance parameter, while q is the parameter of interest since it plays a crucial role in stabilizing the variance for the ECT model (1). If the variance model is correct, then transforming the response Y_j by multiplying by $(1-x)^q$ will yield a constant variance and result in a standard nonlinear regression model:

$$\begin{aligned} Z_j = Y_j \times (1-x)^q &= \eta_t(x, \mathbf{c}, r) + \delta_j \\ &= \sum_{\ell=0}^{t} c_\ell x^\ell (1-x)^r + \delta_j, \end{aligned} \tag{12}$$

where $r = q - p$ is an unknown parameter and we assume that $\delta_j = \varepsilon_j(x) \times (1-x)^q \sim \mathrm{Norm}(0, \sigma^2)$. Therefore, we will estimate model (12) directly and then obtain the parameter estimators of the ECT model (1) indirectly by noting that the coefficients $\mathbf{c}$ in model (1) coincide with those in (12), and p is estimated by the difference between the q and r estimates.

2.2. Other Issues

Specifying the Polynomial Order of the ECT Model

The appropriate order of the polynomial in the ECT model is determined in a forward-selection manner. More specifically, based on a data set, transformed models $\eta_t(x, \mathbf{c}, r)$ for $t = 0, 1, \ldots$ are successively fitted. At each advanced stage ($t > 0$), after fitting $\eta_t(x, \mathbf{c}, r)$, an extra sum of squares analysis (see [2], p. 103, for details) is performed and we keep increasing the value of t until the variability explained by the highest order term in the model is found to be insignificant at a confidence level of 95%. This approach tends to keep t small and therefore preserve the monotonicity of the curve.

Starting Values of Nonlinear Parameters

Obtaining good starting values for the unknown parameters is important in nonlinear regression. We first consider the starting value for the parameter r. The limiting form of model (12), as $x \to 1$, is $\eta_0(x, \mathbf{c}, r) = c_0(1-x)^r$. Thus, $\hat{r}$ can be obtained by fitting model $\eta_0(x, \mathbf{c}, r)$, for which we have shown that the convergence of nonlinear least-squares estimators to the global optimum is guaranteed even without good starting values. For the throughput range of interest, say $[0.5, 0.95]$, the estimate $\hat{r}$ from fitting $\eta_0(x, \mathbf{c}, r)$ provides a good initial value of r for fitting the higher-order model $\eta_1(x, \mathbf{c}, r)$. More generally, we obtain an initial value of r for fitting model $\eta_t(x, \mathbf{c}, r)$ from the estimator $\hat{r}$ from fitting model $\eta_{t-1}(x, \mathbf{c}, r), t = 1, 2 \ldots$. Given any fixed value of r, we can perform a simple linear regression to obtain the starting values of the coefficients $\mathbf{c}$. Thus, our forward-selection procedure provides a natural way of determining starting values for the parameters of the nonlinear regression models.

3. PROCEDURE FOR DETERMINATION OF SIMULATION INPUTS

This section is devoted to construction of the experiment design and issues related to computational efficiency. To provide context, a high-level description of the procedure is provided in Figure 2 along with the sections of the paper in which the key components are described. The detailed procedure is given in Section 4.

The experiment design consists of the design points $\mathbf{x}$, the throughput levels at which simulations will be executed, and the allocation $\boldsymbol{\pi}$, the fraction of the available simulation replications assigned to each design point. The best choice of $(\mathbf{x}, \boldsymbol{\pi})$ depends on the true ECT and variance curves. In our procedure, models of the ECT and variance curves are estimated ever more precisely as simulation data are obtained, and the choice of what design points to add or at which points to make additional replications is guided by the current best estimate of the model.

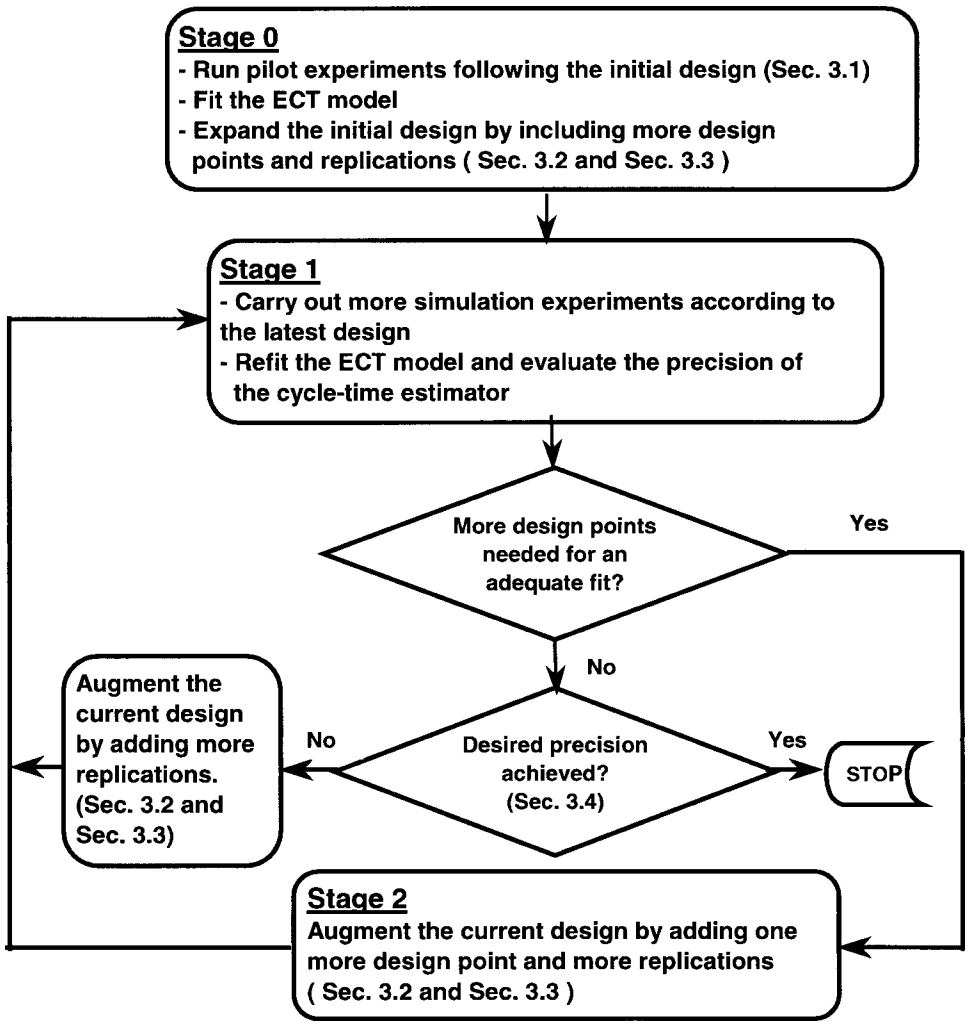

Figure 2. Flow chart for the multistage procedure.

3.1. Design for Estimating the Variance Model

As explained in Section 2.1, the variance of the response $Y(x)$ as a function of x is far from constant, so to obtain valid statistical inference the first step is to stabilize the variance. Therefore, we fit the transformed variance model (11) before performing any other analysis. Since the variance-stabilizing transformation involves only the parameter q, the experiment design for fitting model (11) should emphasize precise estimation of q.

Kiefer and Wolfowitz [7] show that the variance-minimizing number of design points for fitting a linear model such as (11) is equal to the number of unknowns, 2 in our case. In Appendix A.1 we show that the optimal allocation of simulation effort to minimize $\mathrm{Var}[\hat{q}]$ is to assign the same number of replications to the lower and upper bounds, x_L and x_U. To achieve this, the number of initial replications, selected by the user, should be chosen to be an even number.

3.2. Design for Estimating the ECT Model

Design Criterion

We use the *PM* measure (6) employed by Cheng and Kleijnen [3] as the design criterion for fitting the ECT model (1). The variance $\mathrm{Var}[\mu_t(x, \hat{\mathbf{c}}, \hat{p})]$, and hence the *PM* measure, depend on $\mathbf{x}$ and $\boldsymbol{\pi}$, and the optimal design is determined by minimizing *PM* with respect to these decision variables.

Recall that CK treat both p and q as known parameters, which leads to the simplified form of *PM* in (7), for which the optimal design is relatively easy to obtain. On the other hand, we treat p and q as unknown parameters to be estimated. Actually, $\hat{p}$ is obtained indirectly through the relationship $\hat{p} = \hat{q} - \hat{r}$, where $\hat{q}$ and $\hat{r}$ are estimated from fitting models (11) and (12), respectively. Two difficulties are encountered in evaluating *PM* as a function of $\mathbf{x}$ and

$\boldsymbol{\pi}$: (1) Although the estimators of $\mathrm{Var}[\hat{q}]$ and $\mathrm{Var}[\hat{r}]$ can be obtained from model fitting, there is no direct way to estimate the covariance $\mathrm{Cov}[\hat{q}, \hat{r}]$, and hence $\mathrm{Var}[\hat{p}] = \mathrm{Var}[\hat{q} - \hat{r}]$ is difficult to estimate. (2) The ECT is a nonlinear regression model, and the variance of its expected response estimator, $\mathrm{Var}[\mu_t(x, \hat{\mathbf{c}}, \hat{p})]$, depends on the unknown parameters. Good estimates of these unknowns are not usually available when the experiment design is constructed.

Our approach is to use a small preliminary experiment to estimate q and then p using the experiment design outlined in Section 3.1. We then treat these estimates $\hat{q}$ and $\hat{p}$ as known values so that we can harness the formula for PM in (7). As the sequential estimation procedure continues, we use updated estimates of both the ECT and the variance models to refine the design. However, this sequential updating introduces another complication that we describe next.

Constrained Nonlinear Optimization

Conditional on $\hat{q}$ and $\hat{p}$, the optimal design $(\mathbf{x}, \boldsymbol{\pi})$ can be obtained by minimizing PM. However, since data are collected sequentially, the ECT and variance models are continually refined. Each time we reoptimize the design for the refined models, the throughput levels that have already been chosen and the replications that have already been allocated are constraints on the optimization. CK ignore the data that have already been allocated and simply solve the unconstrained problem for the new allocation $\boldsymbol{\pi}$; therefore, their new allocation may not be achievable due to a fixed budget. We propose solving the following constrained optimization problem each time we reoptimize the design:

$$\min_{\mathbf{x},\pi} PM(\mathbf{x}, \boldsymbol{\pi}) \tag{13}$$

$$\begin{aligned} s.t. \quad & \{x_1, x_2, \ldots, x_m\} \supseteq \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{m_c}\} \qquad (14) \\ & x_L \le x_1 < x_2 < \cdots < x_m \le x_U \\ & \sum_{i=1}^{m} \pi_i = 1 \\ & \pi_i \ge lb_i \quad \text{for } i = 1, 2, \ldots, m. \end{aligned}$$

The input parameters, decision variables, and constraints of (13) are given 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. The choice of ΔN at each stage will be discussed in the next subsection. Both N_c and ΔN are used to calculate the lower bounds $lb_i = \max\{n_c(x_i), 2\}/(N_c + \Delta N)$ for $i = 1, 2, \ldots, m$. We set $lb_i \ge 2/(N_c + \Delta N)$ 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. In practice, the design points are also required to be a certain minimum distance from each other.
- The updated allocations of simulation effort $\boldsymbol{\pi} = \{\pi_1, \pi_2, \ldots, \pi_m\}$ in (13). Following the strategy of Cheng and Kleijnen [3], the $\boldsymbol{\pi}$ are treated as continuous decision variables in the optimization by relaxing the constraint that $N\boldsymbol{\pi}$ be an integer. The allocation we actually use is then obtained by rounding up; that is, we set $n(x_i) = \lceil N\pi_i \rceil$, which adds at most m additional replications relative to any other rounding of this solution.

  Note that the equivalent representations of PM_0 in Eq. (6) depend on the allocations n_i being an integer, while in the relaxed, and scale-free, PM of (7) the corresponding π_i are continuous valued. Since we jointly optimize $\mathbf{x}$ and $\boldsymbol{\pi}$ in (13), we cannot claim that rounding up reduces the objective function's value relative to the true integer optimal solution of (6), because the integer solution probably places the design points $\mathbf{x}$ differently and we cannot evaluate (6) for non-integral solutions. However, we can claim that, given the design point $\mathbf{x}$ obtained from the relaxed problem, rounding all $N\boldsymbol{\pi}$ up reduces the objective function (6) more than any other rounding scheme since (6) is a decreasing function of n_i.

Constraints.

- The constraint (14) forces the new set of design points to include the old points.
- The meanings of the other constraints are obvious.

In the procedure, (13) is solved to augment the current design when an assessment of the ECT model fit 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, which is not part of CK, gives our procedure more latitude in terms of obtaining an adequate fit.

We use coordinate descent methods [8] to solve this constrained nonlinear optimization problem. More specifically, in each iteration, we fix $\mathbf{x}$, find the value of $\boldsymbol{\pi}$ that minimizes PM, and then update $\boldsymbol{\pi}$; we next search for the optimal $\mathbf{x}$ conditional on the fixed $\boldsymbol{\pi}$. This iterative process is repeated until convergence is obtained. The proof of convergence, along with the algorithm, is available at `users.iems.northwestern.edu/~nelsonb/Publications/YANOLC.pdf`.

Starting Values of the Design Points

Since (13) is decomposed into two nonlinear subproblems as described in Section 3.2, it is only necessary to provide initial values for the design points $\mathbf{x}$ to start the search for the optimal design. In all the experiments considered in this paper, we chose starting values of the design points to be evenly spaced throughout the interval of throughput when expanding the 2-point design to an m-point design. More extreme starting values, such as those clustering at the lower or upper end, have also been used in a few trial cases and no difficulties were encountered in locating the optimum.

3.3. Choice of the Number of Additional Replications

In our procedure, experimentation is performed sequentially. Except for the initial experiment, simulations are carried out according to the design obtained by solving the constrained nonlinear optimization problem (13). In (13), ΔN, the number of additional runs to add to the current m_c-point design, is a parameter that can be chosen by the experimenter. However, we provide a systematic way to determine ΔN in Appendix A.3.

Guiding the choice of ΔN at different stages of the experiment is important. If the increment is too small, then computational effort may be wasted due to the refitting, reevaluation, and determination of further designs after each increment. On the other hand, if ΔN is too large, then more replications may be run than are actually required.

3.4. Stopping Rule for the Procedure

CT-TH curves help manufacturers decide at what throughput they should run the system so that they can deliver the products on time as promised. The desired throughput that can suit customers' demands, say x_0, is usually of great interest, and it is also where high precision must be achieved on the CT-TH curve. Our goal is to estimate the expected cycle time at throughput x_0 with a specified precision, while still estimating the CT-TH curve for all $x_L \le x \le x_U$ well. Therefore, the stopping criterion is relative error at x_0, while the design criterion is the integrated variance of the estimated curve over the entire range $[x_L, x_U]$. Later, in Section 5, when we evaluate this overall approach we check the relative error at a number of points in $[x_L, x_U]$, not just at x_0.

The procedure verifies that the relative error stopping criterion has been achieved when the half length of the confidence interval for $\mu(x_0, \mathbf{c}, p)$ is less than $\gamma\%$ of the estimated expected response $\mu_t(x_0, \hat{\mathbf{c}}, \hat{p})$, where $\gamma\%$ is specified by the user. We are not in favor of using absolute error because cycle time varies so greatly over the range of throughput, and unless the user already has a good idea of the throughput at the upper end there is no good way to specify an absolute error criterion.

As explained in Section 3.2, estimating $\mathrm{Var}[\hat{p}]$, and hence $\mathrm{Var}[\mu_t(x_0, \hat{\mathbf{c}}, \hat{p})]$, is difficult. Once again, approximations are adopted with regard to the computation of $\mathrm{Var}[\mu_t(x_0, \hat{\mathbf{c}}, \hat{p})]$. In our experiments we found that $\hat{q}$ and $\hat{r}$ are positively correlated with an estimated correlation as large as 0.996; thus, $\hat{p} = \hat{q} - \hat{r}$ is much less variable than $\hat{r}$. Considering this, a conservative confidence interval for $\mathrm{Var}[\mu_t(x_0, \hat{\mathbf{c}}, \hat{p})]$ can be estimated by substituting $\widehat{\mathrm{Var}}[\hat{r}]$ for $\widehat{\mathrm{Var}}[\hat{p}]$. See Appendix A.2 for details.

4. THE MULTISTAGE PROCEDURE

In this section, we first give a detailed description of the multistage procedure that was diagrammed in Figure 2. This procedure will be referred to as YAN. We then present a brief summary of the differences between CK and YAN.

4.1. Description of the Procedure

Initially, the number of design points m must be determined through consideration of the system being investigated. There should be a sufficient number of design points to allow for a good fit of the ECT model. Once the locations of the m design points are found by solving (13) for the first time, we fix them and then sequentially update the sampling allocations with increasingly more precisely estimated models. This is based on the argument made by Cheng and Kleijnen, which is also confirmed in our experiments, that identifying the location of optimal design points is of secondary importance compared with having the optimal number of runs at each point. The YAN procedure is divided into three stages.

Stage 0

In this stage, N_0 replications are allocated evenly to the two end points x_L and x_U. We then fit the variance model (11) to the data, stabilize the variance for the dataset with $\hat{q}$, and fit the transformed model (12) with $t = 0$. Note that t cannot be greater than 0 because the dataset has only two design points. With the estimated models (11) and (12), we seek to expand the initial design:

1. Determine ΔN, the number of replications to be added to the initial design, by following the method described in Appendix A.3.
2. Solve (13) to find the optimal design $(\mathbf{x}, \boldsymbol{\pi})$ consisting of m points given that $N_0/2$ replications have already been assigned to each of the two end points and ΔN replications will be added.

The number of points m is chosen to be equal to $t_{max} + 2$, where t_{max} is the polynomial degree that is expected to be high enough to provide a good fit for the ECT model. Both N_0 and t_{max} can be user-specified parameters, but we recommend $N_0 = 16$ and $t_{max} = 2$.

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 or the procedure is directed to Stage 2. Three tasks are to be completed in the following steps.

Step 1: Run more simulation experiments. Assign ΔN additional runs to the m design points found in the previous stage according to the latest updated loadings $\boldsymbol{\pi}$. Reestimate the variance model (11) using all of the available data at x_L and x_U, and stabilize the variance for the expanded data set.

Step 2: Estimate the ECT model. Search for the appropriate polynomial order of the ECT model by fitting (12) using the forward selection method. Starting with $t = 0$, keep increasing the value of t until the best fit $\eta_{t_c}(x, \hat{\mathbf{c}}, \hat{r})$ (t_c is the polynomial order of the best fitted model obtained so far) is identified or the highest order constrained by m is reached. The latter is regarded as "a rare event" in practice, since m is chosen to be "sufficient" for the estimation of the system being investigated. If, however, m is inadequate, then we move to Stage 2, where an extra design point is added, followed by additional runs that allow estimation of the higher-order term; otherwise, we continue with the next step.

Step 3: Evaluate the precision of the estimator. Estimate the confidence interval for $\mu(x_0, \mathbf{c}, p)$, and two cases are considered:

- If the desired precision is achieved (the half width of the confidence interval for $\mu(x_0, \mathbf{c}, p)$ is less than $\gamma\%$ of $\mu(x_0, \hat{\mathbf{c}}, \hat{p})$), then stop and report the results.
- Otherwise, based on the best estimated models (11) and (12) obtained so far, determine the value of ΔN at the current point and then solve (13) for the optimal loadings $\boldsymbol{\pi}$ of the next design given that the m design points are fixed. Go back to Step 1.

Stage 2

We augment the experiment design by including an $(m+1)$st design point and ΔN additional replications allocated to the $m + 1$ points. The updated design $(\mathbf{x}, \boldsymbol{\pi})$ is found by solving (13). Then we move to Stage 1.

Depending on what is learned from Stage 1, the design may be augmented with one or more additional design points to provide the support necessary to estimate an adequate model. The procedure moves to Stage 2 only when the value of m, which is selected before any experiment is carried out, turns out not to be a good guess for the current situation. In other words, the highest order term that is estimable in the polynomial is significant, indicating that increasing the polynomial order, which is not allowed in Stage 1, might result in a better fit.

4.2. Summary of Key Differences between Two Procedures

Regression Metamodels

CK assume p is known and use the linear model (1) to represent the CT-TH curve. We model the response surface with the full nonlinear model (1) with p unknown, which offers more flexibility and better properties of the fitted model.

Error Term

Compared to the YAN procedure, a stronger assumption regarding the variance model is required in the CK procedure. We assume that $[\sigma g(x)]^2$ has the form (9) with unknown parameters, while CK assume that $g(x)$ is known and only σ^2 is not.

Augmenting the Design

In the CK procedure, the number of design points m is fixed throughout the process. Therefore, m must be chosen large enough to support good estimation of the curve, and no remedies for violation of this assumption are provided. In contrast, YAN offers the potential to augment the current design by incorporating more design points, which provides the support necessary to estimate an adequate model with unexpected higher-order terms. Moreover, as explained in Section 3.3, YAN provides guidance regarding the increment of replications, ΔN, while CK nearly always takes the smallest increment possible.

Stopping Criterion

The CK procedure terminates once the computing budget is exhausted, while YAN terminates when the desired precision is achieved.

5. EMPIRICAL EVALUATION

In this section, we discuss the numerical results of simulation experiments to illustrate the efficiency of the YAN procedure, as well as to compare it to the CK procedure.

5.1. Summary of Evaluation Methodology

Comparison of YAN to CK was with respect to several queueing systems and one response surface model. In all of the cases considered, the true expected cycle time throughout the experimental region was known, and hence the quality of model estimation could be evaluated. Rather than use only systems simulation examples, we also chose several response surface models for which we could control all the model parameters, including p and the polynomial coefficients in the ECT model (1) and q in variance model (10). The response surface models were selected mainly for illustrating the applicability of the procedure in a wide range of cases without intending to represent any specific real systems.

Two summary measures were used to assess how well each procedure estimates a CT-TH curve. A worst-case measure is the maximum relative deviation of the mean cycle time predicted by the fitted model from the true value over the range $[x_L, x_U]$, defined as

$$D^w = \max_{x \in [x_L, x_U]} |\hat{\mu}(x) - \mu(x)|/\mu(x). \tag{15}$$

The measure D^w checks the accuracy of the fitted curve at locations where the lack of fit is most pronounced. In addition to D^w, the overall accuracy of the fitted model across the range of interest was measured by

$$D^a = \frac{\int_{x_L}^{x_U} |\hat{\mu}(x) - \mu(x)|/\mu(x)\, dx}{x_U - x_L}, \tag{16}$$

the average deviation of the estimated curve from the true curve. We chose D^w and D^a to be the performance measures because they are both relative deviations and dimensionless. For each model considered, the entire experiment was repeated a number of times and the measures D^w and D^a were averaged across these "macro-replications."

Cheng and Kleijnen's method is designed to work with a computing budget constraint, while our procedure aims to achieve a user-specified relative precision (specified by $\gamma\%$). To compare these two procedures, we needed to ensure that they were implemented with the same computational effort. In our experiments, the YAN procedure was first applied for, say, K macro-replications, with $N_1, N_2, \ldots, N_K$ corresponding to the number of simulation runs required in each. The average $\overline{N} = K^{-1} \sum_{k=1}^{K} N_k$ was then used as the computing budget for each of the K macro-replications of CK.

The performance of Cheng and Kleijnen's method depends on prior knowledge of the system being investigated, or more specifically, the forms of $f(x)$ and $g(x)$. If the value of p is incorrectly specified, then Cheng and Kleijnen's method might fall short. In all the experiments where the CK procedure was implemented we took $f(x) = 1/(1-x)$ and $g(x) = 1/(1-x)^2$, which are the forms assumed by Cheng and Kleijnen in their numerical examples.

For the YAN procedure, the value of the initial sample size N_0 was set at 16 replications. This choice is of limited importance, since the variance model is updated as the experimentation proceeds.

5.2. Queueing Systems

We compared the two procedures, CK and YAN, through the queueing systems, M/M/1/FIFO, M/M/1/SPT (nonpreemptive shortest processing time first), and M/M/1/LPT (nonpreemptive longest processing time first), which were also examined by Cheng and Kleijnen [3]. For each system, 100 macro-replications were performed and the measures D^w and D^a evaluated for each fitted curve. From these 100 macro-replications where a common procedure was implemented, the sample mean and its standard error of these two measures were obtained. This allowed a detailed examination of the accuracy of the estimated CT-TH curves resulting from each particular procedure.

For YAN, the throughput range of interest is given as $[0.5, 0.95]$, and the procedure was driven by a desired relative error of $\gamma\% = 5\%$ at $x_0 = x_U = 0.95$. Table 1 displays the performance measures, D^w and D^a, associated with the two procedures. Each case is discussed in detail below.

5.2.1. M/M/1/FIFO

For M/M/1/FIFO system, the user-specified design parameter m (number of design points) was chosen to be 4 for both procedures. Recall that the true underlying CT-TH curve and variance-TH relationship are represented by (3) and (9), respectively, with $p = 1$ and $q = 2$, which coincide with the assumed forms of $f(x)$ and $g(x)$ for CK. Not surprisingly,

Table 1. Sample mean and its standard error of D^w and D^a from 100 macro-replications of the M/M/1/FIFO, M/M/1/SPT, and M/M/1/LPT.

	CK procedure		YAN procedure	
System	$\overline{D}^w$ $SE(\overline{D}^w)$	$\overline{D}^a$ $SE(\overline{D}^a)$	$\overline{D}^w$ $SE(\overline{D}^w)$	$\overline{D}^a$ $SE(\overline{D}^a)$
FIFO	1.004%	0.688%	2.475%	1.038%
	0.144%	0.061%	0.174%	0.060%
SPT	5.125%	1.439%	3.630%	0.471%
	0.251%	0.043%	0.226%	0.023%
LPT	5.896%	2.388%	2.914%	1.436%
	0.161%	0.066%	0.129%	0.048%

Table 2. Empirical frequency distribution of the number of polynomial terms in the fitted models (100 macro-replications) for M/M/1/FIFO.

	Number of polynomial terms			
Procedure	1	2	3	4
CK	87	3	6	4
YAN	95	5	0	0

CK performs better than YAN. However, the results in Table 1 show that YAN provides a good fit in the sense of achieving the desired accuracy and precision. Further, YAN tends to yield a simpler form of the fitted model in terms of the number of parameters in the polynomial, although the mode for both procedures is 1 term. Table 2 shows the empirical frequency distribution of number of parameters included in the polynomial in the final regression model from each procedure. (For this case, the true number of parameters in the polynomial is 1.)

5.2.2. *M/M/1/SPT and M/M/1/LPT*

For these two systems, m was set to be 5 for the CK procedure as Cheng and Kleijnen suggested and 4 for the YAN procedure. Straightforward functional relationships like (3) cannot be obtained for SPT and LPT from queueing analysis, although the underlying true cycle time at any throughput rate can be computed numerically. As pointed out by Cheng and Kleijnen, both queues behave markedly different from FIFO.

Table 1 clearly shows that in these two cases, YAN has superior performance to CK in which the incorrect $f(x)$ and $g(x)$ are used, i.e., significant improvement of the fit was achieved by the YAN procedure in terms of the D^w and D^a measures. Suppose that the comparison between the two procedures is relative to the CK procedure. Then for the SPT system, $\overline{D}^w$ decreased about 29% and $\overline{D}^a$ decreased 67%; for the LPT system, the decreases were 50 and 40%, respectively. Note that although the CK procedure is likely to be highly accurate at the chosen design points $\mathbf{x}$, large deviations between the fitted and true curves at other non-design points can occur. The use of D^w and D^a allows us to check for departures of the fitted model from the true surface throughout the experimental region.

Table 3. Empirical frequency distribution of the number of polynomial terms in the fitted models (100 macro-replications) for SPT and LPT.

		Number of polynomial terms				
Procedure		1	2	3	4	5
SPT	CK	0	0	48	41	11
	YAN	83	16	0	1	0
LPT	CK	0	0	0	1	99
	YAN	96	3	1	0	0

Table 4. Sample mean and variance of D^w and D^a from 100 macro-replications of the selected RSM.

CK procedure		YAN procedure	
$\overline{D}^w$ $SE(\overline{D}^w)$	$\overline{D}^a$ $SE(\overline{D}^a)$	$\overline{D}^w$ $SE(\overline{D}^w)$	$\overline{D}^a$ $SE(\overline{D}^a)$
2.607%	0.849%	1.709%	0.695%
0.110%	0.033%	0.077%	0.032%

As in the FIFO case, YAN tended to yield a simpler fitted model than the CK procedure, as shown in Table 3, where the mode for YAN is 1 term but 3 or 5 for CK. Fewer design points were generally required in YAN compared to the fixed number, 5, used in the CK procedure. For both SPT and LPT cases, of 100 macro-replications to which YAN was applied, more than 90% of them led to a design incorporating only 4 design points, the starting value of m, and the maximum number of design points used by YAN is 6.

5.3. Comparison Based on a Response Surface Model

In the previous section we showed that YAN provides a more accurate estimation of the CT-TH curve when behavior of the system differs from M/M/1/FIFO. When modeling CT-TH response curves via Eq. (1) for real manufacturing systems, we have found that the best value of p can be significantly different from 1, resulting in a much more dramatic departure from M/M/1/FIFO behavior than the priority queueing systems considered above. Here, we will further investigate the performance of the two procedures based on the estimation of a response surface model (RSM) which, although somewhat extreme, does represent what might be encountered in practice [6].

For illustration we take

$$\mathrm{E}[Y(x)] = \mu_1(x) = \frac{3 + 10x}{(1-x)^{0.1}}, \tag{17}$$

with the true variance model being $[g(x)\sigma]^2 = [0.56/(1-x)^{0.6}]^2$ (implying $\sigma = 0.56$ and $q = 0.6$). We make the error normally distributed. The number of design points was fixed at 6 for the CK procedure; a starting value of 4 design points was used in YAN. The numerical results are given in Table 4. As expected, applying the YAN procedure leads to large improvements in fit, with a relative decrease of 34% in maximum deviation and 18% in average deviation.

Moreover, applying the CK procedure, when the assumed p is far from its true value, sometimes results in loss of important properties of the CT-TH curve. As shown in Figure 3,

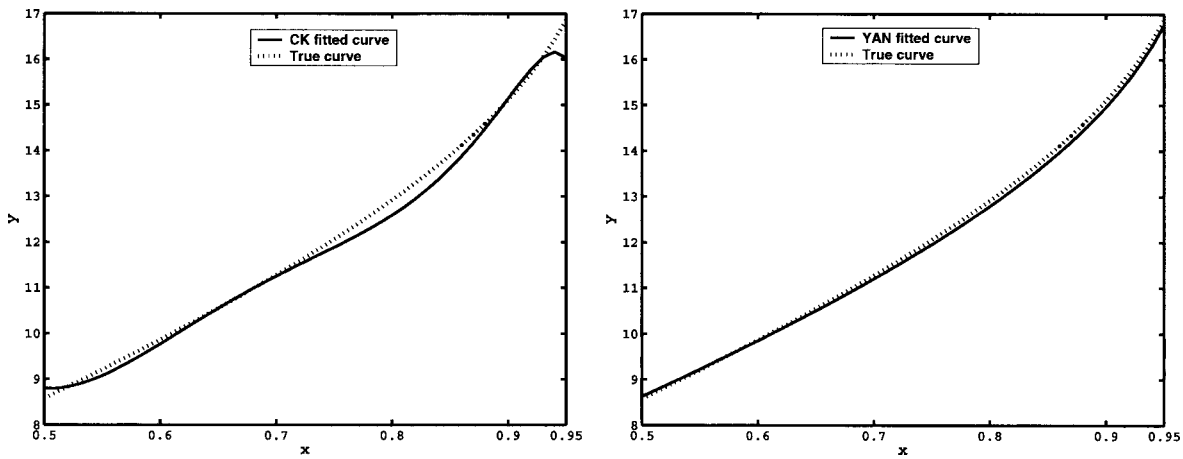

Figure 3. Fitted curve from CK (left) and YAN (right).

the fitted curve resulting from CK displays a tiny downward turn at the high end, while cycle time is known to increase with respect to throughput. The selected CK curve plotted in Figure 3 was an extreme one, and nonmonotonic behavior was observed in only 5 of the 100 fitted curves. However, 10% of the CK fitted curves had a negative second derivative somewhere in the experimental range, whereas the true CT-TH curve is convex throughout. The YAN procedure, which aims at obtaining a good estimate of the exponent term p, reduces the order of the polynomial and is largely free of these problems.

The effectiveness of the YAN procedure was further evaluated based on a number of response surface models. Due to space limitation, these results are reported at `users.iems.northwestern.edu/~nelsonb/Publications/YANOLC.pdf`.

6. AN EXAMPLE OF MANUFACTURING SYSTEMS

In this section, we apply the YAN procedure to a semiconductor wafer fab simulation with behavior quite different from an $M/M/1$ queue. We consider a model describing a real wafer fab, provided by the Modeling and Analysis for Semiconductor Manufacturing Lab at Arizona State University (`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 rate and mix. Jobs of different types follow different process steps and thus have different expected cycle times. In this example, we assume that the product mix is fixed and the two products are considered separately. For the estimated CT-TH curve of Prod1, the response is the mean cycle time of Prod1, and the independent variable is the joint throughput of both products. To estimate Prod1's CT-TH curve, we design to minimize the variance of the estimated expected cycle time for Prod1. We independently apply YAN a second time to estimate the CT-TH curve for Prod2.

For the implementation of the YAN procedure, the range of throughput was chosen to be [0.5, 0.95], the precision level was set at 3% for the upper-end point $x_0 = x_U = 0.95$, and the initial value of m was chosen to be 4.

Since the true underlying curve is unknown, points evenly distributed in the range of throughput were selected to check lack of fit in the fitted model at those locations. Substantial additional data were collected at the check points to obtain the "nearly true" estimates for expected cycle times (about seven times as much computational effort as used in the YAN procedure was invested in these check points, since at each check point, simulation experiments were performed until the standard error of the expected cycle time estimate was essentially zero). A comparison between these highly precise estimates and those predicted by the fitted model obtained from YAN is given in Table 5. Column μ represents the nearly true cycle times and column $\hat{\mu}$ the estimates from applying YAN procedure once for each product independently. The departures of the fitted model from the true surface at the selected points are within 2% relative error. The

Table 5. Comparison of the estimated expected cycle time to the "true" values.

Check points	Prod1			Prod2		
	μ	$\hat{\mu}$	Error	μ	$\hat{\mu}$	Error
0.52	471.9	467.9	−0.8%	617.0	606.1	−1.8%
0.58	479.7	471.7	−1.7%	618.0	608.0	−1.6%
0.64	481.1	478.2	−0.6%	626.7	615.7	−1.8%
0.70	493.4	488.8	−0.9%	638.0	630.7	−1.1%
0.76	511.4	505.8	−1.1%	661.4	656.0	−0.8%
0.82	540.8	534.1	−1.2%	698.7	697.1	−0.2%
0.88	595.9	586.1	−1.6%	767.3	767.2	−0.01%
0.94	703.0	708.6	0.8%	896.2	912.9	1.9%

Table 6. Experiment designs resulting from the YAN procedure.

	Prod1	Prod2
Design points	0.50, 0.62, 0.80, 0.95	0.50, 0.61, 0.70, 0.81, 0.95
Allocation of replications	8, 6, 13, 25	8, 9, 13, 17, 24

fitted CT-TH curves for Prod1 and Prod2 are given in (18) and (19), respectively:

$$\text{Prod1} \quad \widehat{Y}(x) = \frac{480.43 - 225.12x}{(1-x)^{0.34}} \tag{18}$$

$$\text{Prod2} \quad \widehat{Y}(x) = \frac{736.80 - 624.29x + 337.72x^2}{(1-x)^{0.25}}. \tag{19}$$

Apparently, the values of p differ markedly from 1 for both models. The resulting experiment designs are given in Table 6.

From the "Error" columns in Table 5, we conjecture that the fitted curve intersects with the true curve at some point close to the upper end of throughput. At throughput levels lower than the intersection point, we underestimate the cycle times (negative error), and at throughput levels high than the intersection point, we overestimate the cycle times (positive error). This consistent pattern of errors is what we would like to have: it indicates that the fitted curve does not oscillate around the true curve while trying to maintain a certain statistical precision. We believe that this is the result of making p an active parameter in the ECT model.

7. SUMMARY

A nonlinear regression model has been developed for the estimation of CT-TH curves in manufacturing systems. For the purpose of efficiently estimating such a curve, a multistage procedure has been proposed to collect data via simulation experiments until a prespecified precision is achieved for the fitted curve.

It is important to note that our proposed model, which is motivated by queueing theory, is different from the linear model suggested by Cheng and Kleijnen [3] in that an additional unknown parameter p is introduced to capture the curvature of CT-TH curves. The necessity of including p as an unknown parameter is suggested by the fact that the actual value of p for a system is usually hard to obtain based on prior knowledge alone and has been shown to vary over a wide range. Possible negative effects of misspecifying p in the CK procedure have been explained in the paper. In addition, Cheng and Kleijnen's prior assumption on the variance of cycle time is also dropped in YAN. Numerical experiments show that our method can be more efficient than the CK procedure in the sense of achieving higher precision at the same computational expense, but the key contribution is driving the design by a prespecified precision.

ACKNOWLEDGMENTS

This research was supported by National Science Grant DMI-0140385. Additional thanks go to Professors John Fowler and Gerald Mackulak from Arizona State University, the Associate Editor, and two anonymous referees.

APPENDIX

A.1. Optimal Design for Estimating the Variance Model

In the variance model (10), $S^2(x)$ is the sample variance estimated from the replications simulated at the design point x:

$$S^2(x) = \frac{1}{n(x)-1}\sum_{j=1}^{n(x)}(Y_j(x) - \overline{Y}(x))^2. \tag{20}$$

The sample variance $S^2(x)$ is an unbiased estimator of the true variance $\sigma^2(x) = \sigma^2/(1-x)^{2q}$, where σ^2 and q are unknown parameters. Based on the assumed normality of the output response $Y_j(x)$, we have

$$S^2(x) = \frac{\sigma^2}{(1-x)^{2q}} \times \tau(n(x)) = \frac{\sigma^2}{(1-x)^{2q}} \times \frac{U(n(x))}{n(x)-1}, \tag{21}$$

where $U(n(x)) \sim \chi^2(n(x)-1)$. Thus, in the log-variance model (11), the error term $v(n(x))$ is distributed as

$$\log \frac{U(n(x))}{n(x)-1}. \tag{22}$$

For convenience, we rewrite model (11) as follows

$$\log S^2(x) = \log \sigma^2 - 2q \log(1-x) + v(n(x)). \tag{23}$$

Clearly, (23) is a linear model with two unknown parameters, σ^2 and q. The error depends only on $n(x)$, the number of replications taken at x, and not directly on the value of x.

The goal is to design an experiment that minimizes the variance of $\hat{q}$ given the total number of replications, say N_0, to be allocated. Stated mathematically, we want to determine the vector $\mathbf{x} = (x_1, x_2, \ldots, x_m)$ and the corresponding allocation $\mathbf{n} = (n_1, n_2, \ldots, n_m)$ that solves

$$\min_{\mathbf{x},\mathbf{n}} \operatorname{Var}[\hat{q}] \tag{24}$$

$$s.t. \sum_{i=1}^{m} n_i = N_0$$

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

The difficulty stems from the complicated form of the error term.

Properties of the Error

For any design point x_i with $n(x_i)$ replications, define $\nu_i = n(x_i) - 1$, so the error term can be written as $v(\nu_i) \stackrel{\mathcal{D}}{=} \log(U_i/\nu_i)$ where $U_i \sim \chi^2(\nu_i)$.

Let $\psi(z) = \Gamma'(z)/\Gamma(z)$ be the digamma function. We can show that

$$\mathrm{E}[v(\nu)] = \psi(\nu/2) + \log\frac{2}{\nu} \tag{25}$$

$$\mathrm{Var}[v(\nu)] = w(\nu) = \psi'(\nu/2) = \int_0^\infty \frac{te^{-\nu t/2}}{1-e^{-t}}dt. \tag{26}$$

The expectation of the error $\mathrm{E}[v(\nu)]$ depends on ν. Although $\mathrm{E}[v(\nu)] \to 0$ as $\nu \to \infty$, $\mathrm{E}[v(\nu)] \neq 0$ when ν is finite. For $\nu \in [5, 50]$, the expectation ranges from -0.2 to -0.02. If $\mathrm{E}[v(\nu)]$ were a constant, then according to Kiefer and Wolfowitz ([7], Theorem 2), the optimal number of design points m should be 2, the number of unknowns in the model. We adopt this as an approximation.

With m chosen to be 2, the variance of $\hat{q}$ can be expressed as

$$\begin{aligned}\mathrm{Var}[\hat{q}] &= \frac{1/4}{\sum_{i=1}^2 w(\nu_i)^{-1}(x_i - \bar{x}_w)^2}\\ &= \frac{1}{4}(x_1 - x_2)^{-2}(w(\nu_1) + w(\nu_2)),\end{aligned}$$

where $\bar{x}_w = \left(\sum_{i=1}^2 w(\nu_i)^{-1}x_i\right)\Big/\left(\sum_{i=1}^2 w(\nu_i)^{-1}\right)$. Upon examination, we see that the optimal locations are $x_1 = x_L$ and $x_2 = x_U$.

To determine the allocation of N_0 replications between these two design points, we need to minimize $w(\nu_1) + w(\nu_2)$ with respect to ν_1 and ν_2, which is equivalent to solving

$$\min_{\nu_1} h(\nu_1), \tag{27}$$

where $h(\nu_1) = w(\nu_1) + w(d - \nu_1)$, and $d = N_0 - 2$ is the total degrees of freedom.

We first prove that the discrete function $w(\nu_1)$ is convex with respect to ν in the sense of having decreasing successive differences. It is easy to show that for any value of ν_1,

$$\begin{aligned}&(w(\nu_1) - w(\nu_1 + 1)) - (w(\nu_1 - 1) - w(\nu_1))\\ &= \int_0^\infty \frac{t}{1-e^{-t}}e^{-\nu t/2}\left(-\sqrt{e} - 1/\sqrt{e} + 2\right)dt\\ &< 0.\end{aligned}$$

The last inequality follows because the integrand is negative throughout the range of integration. Thus, $h(\nu_1)$ is also convex with respect to ν_1. Since $h'(d/2) = 0$, we conclude that $d/2$ is the unique minimum point of $h(\nu_1)$ provided $d/2$ is an integer, and $\nu_1 = \nu_2 = d/2$. If $d/2$ is fractional, set $\nu_1 = \lfloor d/2 \rfloor$ and $\nu_2 = d - \nu_1$.

In summary, the optimal design for estimating the transformed variance model (11) employs two design points, x_L and x_U, with an equal number of replications assigned to each of them. In our experiments, with the replications performed at the two end points, two sample variances, $S^2(x_L)$ and $S^2(x_U)$, can be obtained. Based on these two sample variances, the two-parameter model (23) can be fitted. Note that throughout the YAN procedure, we only use the data collected at the two end points for estimating model (23) even with simulation runs available at other design points. This is because the number of replications assigned by the optimal design to some middle point may not be large enough to provide a good estimate of the variance at that point, which in our experience can hurt the accuracy of the estimated model (23). On the other hand, the two end points always having at least $N_0/2$ replications assigned to each of them are reliable sources for estimating sample variances.

A.2. Confidence Interval for Expected Cycle Time

Suppose we want to achieve a relative precision level of $\gamma\%$ at x_0. Recall that the precision level γ means that the $100(1-\alpha)\%$ CI has a relative half width of $\gamma\%$. Therefore, we need a confidence interval for the expected response at x_0. Assuming that $q = \hat{q}$ is a given constant, let $\boldsymbol{\theta} = (r, c_0, c_1, \ldots, c_{t_c})$ be the parameters for the transformed model (12) with t_c being the best polynomial order obtained so far. Let $T_c = t_c + 2$ denote the total number of parameters. The fitted transformed model is $\eta_{t_c}(x, \hat{\boldsymbol{\theta}}_c) = \left(\sum_{\ell=0}^{t_c} \hat{c}_\ell x^\ell\right)(1-x)^{\hat{r}}$, and the estimated ECT model can be expressed as

$$\mu_{t_c}(x, \hat{\boldsymbol{\theta}}_c) = \frac{\sum_{\ell=0}^{t_c} \hat{c}_\ell x^\ell}{(1-x)^{q-\hat{r}}}. \tag{28}$$

According to Bates and Watts [2], an approximate $100(1-\alpha)\%$ CI for the expected response at x_0 is then

$$\mu_{t_c}(x_0, \hat{\boldsymbol{\theta}}_c) \pm t(N_c - T_c, \alpha/2) \times s\sqrt{\mathbf{v}_c{}'(\widehat{\mathbf{V}}_c'\widehat{\mathbf{V}}_c)^{-1}\mathbf{v}_c} \tag{29}$$

with notation defined as follows:

s^2: the residual mean square error resulting from fitting model (12) based on $N_c - T_c$ degrees of freedom, where N_c is the total number of replications collected so far;

$t(N_c - T_c, \alpha/2)$: the $1-\alpha/2$ quantile of t distribution with $N_c - T_c$ degrees of freedom;

$\widehat{\mathbf{V}}_c$: the $N_c \times T_c$ derivative matrix with elements $\{v_{kj}\}$ defined as

$$v_{kj} = v_j(x_k) = \left.\frac{\partial \eta_{t_c}(x_k; \boldsymbol{\theta})}{\partial \theta_j}\right|_{\hat{\boldsymbol{\theta}}_c} \qquad k = 1, 2, \ldots, N_c;\ j = 1, 2, \ldots, T_c; \tag{30}$$

$\mathbf{v}_c = [v_j(x_0)]_{T_c \times 1} = [\partial\mu_{t_c}(x_0, \hat{\boldsymbol{\theta}}_c)/\partial\theta_1, \partial\mu_{t_c}(x_0, \hat{\boldsymbol{\theta}}_c)/\partial\theta_2, \ldots, \partial\mu_{t_c}(x_0, \hat{\boldsymbol{\theta}}_c)/\partial\theta_{T_c}]'$.

A.3. Incrementing the Number of Runs

At different stages during the experiment we must augment the current design by adding more simulation experiments. The question is, how many more replications, say ΔN, do we need to run? Recall that the target level for the relative error is set at $\gamma\%$, and adding ΔN replications should help to achieve that goal. More specifically, suppose the achieved precision level obtained so far is $\gamma_c\%$. Then the ΔN additional replications should drive it down to, say, $\gamma_1\%$ ($\gamma \le \gamma_1 < \gamma_c$).

We consider two different situations. If $\gamma_c\% \le 5 \times \gamma\%$, then we would like ΔN to be large enough to achieve $\gamma\%$, and thus we set $\gamma_1\% = \gamma\%$. If $\gamma_c > 5 \times \gamma\%$, which means the current estimate is very poor, then a relatively conservative step ΔN should be taken to avoid overshooting based on imprecise parameter estimates. In the latter case, we set $\gamma_1\% = \gamma_c\%/5 > \gamma\%$ in our experiments. The constant 5 was found to be efficient in our experiments, but can be changed by the user.

Next, we illustrate how the number ΔN is computed for achieving the desired precision level $\gamma_1\%$. The goal is to obtain a relatively quick approximation for ΔN, not a highly refined estimate. Therefore, we solve a relaxation of the allocation problem to obtain a proportion of the replications to be allocated to the design points for any value of the total number of replications N. We then solve for the (continuous) value of N required to reduce the relative error to the desired level and round it up.

Most of the notation is as in Appendix A.2. In addition, we define

m_c: the number of design points included in the current design;

$\boldsymbol{\pi}^* = (\pi_1^*, \pi_2^*, \ldots, \pi_{m_c}^*)$: the optimal allocation vector, which is obtained by solving

$$\min_{\boldsymbol{\pi}} PM(\mathbf{x}, \boldsymbol{\pi}) \tag{31}$$
$$s.t. \sum_{i=1}^{m_c} \pi_i = 1$$
$$\pi_i \geq 0 \quad i = 1, 2, \ldots, m_c$$

conditional on $\hat{q}$ and $\hat{p}$. Note that the lower-bound constraints (15) are ignored in (31) to simplify the calculation;

N: the total number of replications required to achieve the precision of $\gamma_1\%$ in the ideal situation (using the allocation $\boldsymbol{\pi} = \boldsymbol{\pi}^*$);

$n_i = N\pi_i^*$: the number of replications needed at the design point x_i ($i = 1, 2, \ldots, m_c$). The integer constraint on n_i is relaxed in this derivation;

$\widehat{\mathbf{V}}$: the $N \times T_c$ derivative matrix defined as (30).

GIVEN: Current data set that incorporates a total of N_c replications distributed to m_c distinct design points $\mathbf{x} = (x_1, x_2, \ldots, x_{m_c})$ and the best fitted model (28).

FIND: Total number of replications N required at the m_c points $\mathbf{x}$ to achieve the desired precision $\gamma_1\%$, assuming T_c parameters and that the allocation $\boldsymbol{\pi} = \boldsymbol{\pi}^*$ is ideal.

Stated mathematically, we want to determine the smallest N that satisfies

$$\frac{z_{\alpha/2} \times s\sqrt{\mathbf{v}_c{}'(\widehat{\mathbf{V}}'\widehat{\mathbf{V}})^{-1}\mathbf{v}_c}}{\mu_{t_c}(x_0, \hat{\boldsymbol{\theta}}_c)} \leq \gamma_1\%. \tag{32}$$

The numerator of the left-hand side is approximately the half-length of (29). In (32), $(\widehat{\mathbf{V}}'\widehat{\mathbf{V}})^{-1}$ is the only term that depends on the variable N. We will show that $(\widehat{\mathbf{V}}'\widehat{\mathbf{V}})^{-1} = N^{-1}\mathbf{B}(\mathbf{x}, \boldsymbol{\pi}^*)$, where $\mathbf{B}$ does not depend on N, and therefore

$$N = \left\lceil \frac{(z_{\alpha/2}s)^2\mathbf{v}_c{}'\mathbf{B}(\mathbf{x}, \boldsymbol{\pi}^*)\mathbf{v}_c}{(\mu_{t_c}(x_0, \hat{\boldsymbol{\theta}}_c)\gamma_1\%)^2} \right\rceil. \tag{33}$$

The derivation of matrix $\mathbf{B}(\mathbf{x}, \boldsymbol{\pi})$ is as follows.

Define the matrix $\mathbf{A} = \widehat{\mathbf{V}}'\widehat{\mathbf{V}}$ with element $a_{st} = \sum_{i=1}^{m_c} n_i v_s(x_i) v_t(x_i)$. We can write the determinant of $\mathbf{A}$ as

$$|\mathbf{A}| = \sum_{i_1=1}^{m_c} n_{i_1} v_1(x_{i_1}) \sum_{i_2=1}^{m_c} n_{i_2} v_2(x_{i_2}) \cdots \sum_{i_{T_c}=1}^{m_c} n_{i_{T_c}} v_{T_c}(x_{i_{T_c}}) \times \begin{vmatrix} v_1(x_{i_1}) & v_1(x_{i_2}) & \cdots & v_1(x_{i_{T_c}}) \\ v_2(x_{i_1}) & v_2(x_{i_2}) & \cdots & v_2(x_{i_{T_c}}) \\ \vdots & \vdots & \vdots & \vdots \\ v_{T_c}(x_{i_1}) & v_{T_c}(x_{i_2}) & \cdots & v_{T_c}(x_{i_{T_c}}) \end{vmatrix}.$$

Then $|\mathbf{A}|$, the determinant of $\mathbf{A}$, and $|\mathbf{A}_{st}|$, the cofactor of $\mathbf{A}$, can be written as

$$|\mathbf{A}| = \sum_{\substack{i_s \neq i_t \\ 1 \leq i_1, \ldots, i_{T_c} \leq m_c}} n_{i_1} n_{i_2} \cdots n_{i_T} v_1(x_{i_1}) v_2(x_{i_2}) \cdots v_{T_c}(x_{i_{T_c}}) \times \begin{vmatrix} v_1(x_{i_1}) & v_1(x_{i_2}) & \cdots & v_1(x_{i_{T_c}}) \\ v_2(x_{i_1}) & v_2(x_{i_2}) & \cdots & v_2(x_{i_{T_c}}) \\ \vdots & \vdots & \vdots & \vdots \\ v_{T_c}(x_{i_1}) & v_{T_c}(x_{i_2}) & \cdots & v_T(x_{i_{T_c}}) \end{vmatrix} \tag{34}$$

$$= N^{T_c} u(\mathbf{x}, \boldsymbol{\pi}^*) \tag{35}$$

$$|\mathbf{A}_{st}| = \sum_{\substack{i_a \neq i_b \\ i_a \neq i_t \\ 1 \leq i_1, \ldots, i_{T_c} \leq m_c}} n_{i_1} \cdots n_{i_{t-1}} n_{i_{t+1}} \cdots n_{i_{T_c}} \times v_1(x_{i_1}) \cdots v_{t-1}(x_{i_{t-1}}) v_{t+1}(x_{i_{t+1}}) \cdots v_{T_c}(x_{i_{T_c}})$$

$$\times \begin{vmatrix} v_1(x_{i_1}) & \cdots & v_1(x_{i_{t-1}}) & v_1(x_{i_{t+1}}) & \cdots & v_1(x_{i_{T_c}}) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ v_{s-1}(x_{i_{s-1}}) & \cdots & v_{s-1}(x_{i_{t-1}}) & v_{s-1}(x_{i_{t+1}}) & \cdots & v_{s-1}(x_{i_{T_c}}) \\ v_{s+1}(x_{i_{s+1}}) & \cdots & v_{s+1}(x_{i_{t-1}}) & v_{s+1}(x_{i_{t+1}}) & \cdots & v_{s+1}(x_{i_{T_c}}) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ v_{T_c}(x_{i_1}) & \cdots & v_{T_c}(x_{i_{t-1}}) & v_{T_c}(x_{i_{t+1}}) & \cdots & v_{T_c}(x_{i_{T_c}}) \end{vmatrix}$$

$$= N^{T_c - 1} u_{st}(\mathbf{x}, \boldsymbol{\pi}^*). \tag{36}$$

Steps (35) and (36) follow because $n_i = N\pi_i^*$. The functions $u(\mathbf{x}, \boldsymbol{\pi}^*)$ and $u_{st}(\mathbf{x}, \boldsymbol{\pi}^*)$ can be evaluated given $\mathbf{x}$ and $\boldsymbol{\pi}^*$. Then the inverse of $\mathbf{A}$ can be obtained as

$$(\mathbf{A}^{-1})_{st} = \frac{|\mathbf{A}_{st}|}{|\mathbf{A}|} = N^{-1}\frac{u_{st}(\mathbf{x}, \boldsymbol{\pi}^*)}{u(\mathbf{x}, \boldsymbol{\pi}^*)}. \tag{37}$$

Thus, $\mathbf{A}^{-1} = N^{-1}\mathbf{B}(\mathbf{x}, \boldsymbol{\pi}^*)$, where $\mathbf{B} = [u_{st}(\mathbf{x}, \boldsymbol{\pi}^*)/u(\mathbf{x}, \boldsymbol{\pi}^*)]_{(T_c \times T_c)}$, and Eq. (33) follows. The additional number of replications to be added is therefore $\Delta N = N - N_c$.

REFERENCES

[1] C. Allen, The impact of network topology on rational-function models of the cycle time-throughput curve. Honors Thesis, Department of Industrial Engineering & Management Sciences, Northwestern University, 2003. Available online via <users.iems.northwestern.edu/~nelsonb/Publications/CarlAllenThesis.pdf>

[2] D.M. Bates and D.G. Watts, Nonlinear regression analysis and its applications, John Wiley & Sons, New York, 1988.

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

[4] J.W. Fowler, S. Park, G.T. Mackulak, and D.L. Shunk, Efficient cycle time-throughput curve generation using a fixed sample size procedure, Int J Prod Res 39 (2001), 2595–2613.

[5] W.J. Hopp and M.L. Spearman, Factory physics: Foundations of manufacturing management, Irwin, Chicago, 1996.

[6] R. Johnson, F. Yang, B.E. Ankenman, and B.L. Nelson, Nonlinear regression fits for simulated cycle time vs. throughput curves for semiconductor manufacturing, Proceedings of the 2004 Winter Simulation Conference, 1951–1955, 2004. Available online via <http://www.informs-cs.org/wsc04papers/260.pdf>

[7] J. Kiefer and J. Wolfowitz, Optimum designs in regression problems, Ann Math Statist 30 (1959), 271–294.

[8] J. Nocedal and S.J. Wright, Numerical optimization, Springer-Verlag, 1999.

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

[10] W.C.M. van Beers and J.P.C. Kleijnen, Kriging for interpolation in random simulation, J Oper Res Soc 54(3) (2003), 255–262.

[11] W. Whitt, Planning queueing simulations, Manage Sci 35 (1989), 1341–1366.