Obtaining Quantitative Predictions From Monotone Relationships

Joseph Hellerstein

IBM T. J. Watson Research Center
P.O. Box 704
Yorktown Heights, New York 10598
jlh@ibm.com

Abstract

Quantitative predictions are typically obtained by characterizing a system in terms of algebraic relationships and then using these relationships to compute quantitative predictions from numerical data. For real-life systems, such as mainframe operating systems, an algebraic characterization is often difficult, if not intractable. This paper proposes a statistical approach to obtaining quantitative predictions from monotone relationships -- non-parametric interpolative-prediction for monotone functions (NIMF). NIMF uses monotone relationships to search historical data for bounds that provide a desired level of statistical confidence. We evaluate NIMF by comparing its predictions to those of linear least-squares regression (a widely-used statistical technique that requires specifying algebraic relationships) for memory contention in an IBM computer system. Our results suggest that using an accurate monotone relationship can produce better quantitative predictions than using an approximate algebraic relationship.

1. Introduction

Numerical or quantitative predictions of system behavior are frequently required in tasks such as forecasting, diagnosis, and planning. Typically, quantitative predictions are obtained by characterizing a system in terms of algebraic relationships and then using these relationships to compute quantitative predictions from numerical data. Unfortunately, for real-life systems an algebraic characterization is often difficult, if not intractable. This paper describes an approach to obtaining quantitative predictions from monotone relationships, and applies this approach to predicting memory contention in an IBM computer system.

Why is it often so difficult to obtain accurate algebraic characterizations of real-life systems? Our experience with analyzing measurements of computer systems, in particular the IBM operating system Virtual Machine/System Product (VM/SP), suggests that the major impediment to an algebraic characterization is the absence of sufficiently detailed information about the system's operation. For example, the performance of VM/SP systems is often constrained by contention for the first sixteen megabytes of main memory (referred to as **low memory**), even though there may be sixty-four megabytes or more of main memory. Low-memory contention is a consequence of the operating system using twenty-four bit addressing and requiring that many system services use memory that is directly addressable by the operating system. A key indicator of low-memory contention is the rate at which pages below sixteen megabytes are taken from users in the multi-programming set. Constructing an algebraic relationship between this measure and parameters such as the virtual machine input/output rate and the number of logged-on users requires using these parameters to quantify the frequency and execution times of operating-system service-requests (e.g., spool operations, messages exchanged through the inter-user communication vehicle, and file opens) as well as the low-memory demands of each service requested (e.g., bytes required, page and/or cache alignments, and algorithm used when fixed-sized pools are empty). Unfortunately, such detailed information is rarely available.

When we are unable to construct algebraic relationships, we often have qualitative knowledge in the form of monotone relationships. For example, in VM/SP intuition and experience strongly suggest that low-memory contention increases with the virtual machine input/output rate and the number of logged-on users. Another example in CPU-bound VM/SP systems is the relationship between response time and a workload characterized by CPU utilization and the rate of small transactions. Again, an algebraic characterization appears to be intractable; however, we expect response time to decrease with the rate of

small transactions (since more small transactions means fewer large ones, in a resource-constrained system) and to increase with CPU utilization. Still other examples where monotone relationships apply but algebraic relationships are difficult to construct include the following: relating lock contention to user activity, relating disk operations to the virtual machine input/output rate, and relating working set size to the virtual machine input/output rate and CPU demands.

If an accurate algebraic characterization of the system is unavailable, how can we obtain quantitative predictions? One approach is to approximate the unknown algebraic relationship by a simple function, such as a polynomial. Herein, we present an alternative approach in which quantitative predictions are computed directly from monotone relationships. Our experience with this approach, as shown in section 3, suggests that using an accurate monotone relationship frequently results in better predictions than using an approximate algebraic relationship.

Our approach to prediction is statistical. Referred to as *non-parametric interpolative-prediction for monotone functions (NIMF)*, our approach assumes the existence of historical data, which is appropriate for domains such as computer performance, financial analysis, and demographic studies. Often, the historical data is highly variable; indeed, providing a point estimate (e.g., an expected value) may be meaningless. For this reason, NIMF produces *prediction intervals* at a user-specified *confidence level* (e.g., 75%). A prediction interval consists of a lower bound (y_L) and an upper bound (y_H), with the following interpretation: The probability that the predicted value lies between y_L and y_H is at least as large as the confidence level. NIMF uses monotone relationships to search the historical data for y_L and y_H.

Monotone relationships have been used in many contexts, such as predicting changes in qualitative state (e.g., (Forbus, 1984), (Kuipers, 1986), and (DeKleer84)), monitoring dynamic systems (Dvorak and Kuipers, 1989), and explaining quantitative predictions produced by algebraic relationships (e.g., (Apte and Hong, 1986)). More recently, there has been interest in the probabilistic semantics of qualitative influences (Wellman, 1987) and probabilistic considerations in qualitative simulation ((Dvorak and Sacks, 1989)). Our work further extends the application of monotone relationships by demonstrating their use in quantitative, statistical prediction for situations in which numerical data are available but an algebraic characterization is intractable.

The remainder of this paper is organized as follows. Section 2 describes the NIMF technique. Section 3 evaluates NIMF by comparing its predictions to those of least-squares regression, a widely-used statistical technique that requires specifying algebraic relationships. Our conclusions are contained in section 4.

2. Approach

Our approach to obtaining quantitative predictions from monotone relationships was motivated by observations of performance analyst who tune VM/SP computer systems. One aspect of tuning is workload assignment, in which users of computing services are assigned to one of several computer systems in a manner so that computing resources (e.g., CPU, input/output bandwidth, and memory) are utilized within prescribed guidelines. Clearly, this task requires an ability to predict the resource utilizations of an assignment. The most common approach to predicting the performance of computer systems is based on queueing theory (Kleinrock, 1975). Queueing theory characterizes computer systems in terms of stochastic processes, which permits deriving algebraic relationships between measurement variables. While queueing theory has proven effective for modeling "active" resources (e.g., CPU, input/output operations), it has not been particularly effective for modeling "passive" resources, such as memory. In large VM/SP computer systems, contention for low memory is often the primary performance bottleneck.

How then do performance analysts predict low-memory contention? Lacking a formal approach to the problem, analysts often use an informal approach. We illustrate this by predicting **LOSTEALRAT** (the rate at which pages in low memory are taken from users in the multi-programming set) from **LOGGED** (the number of logged-on users) and **VIO** (virtual machine input/output rate). (All three variables can be obtained from the Virtual Machine Monitor Analysis Program (VMMAP) (IBM, 1985).) Suppose that a workload assignment would result in a computer system having an average of 500 logged-on users with an average aggregate VIO rate of 500. Although we know of no algebraic equation that relates LOSTEALRAT to LOGGED and VIO, we do have an excellent understanding in terms monotone relationships. Specifically, for each logged-on user, data structures are allocated in low memory to describe the virtual address space; so we expect LOSTEALRAT to increase with LOGGED. Further, each VIO re-

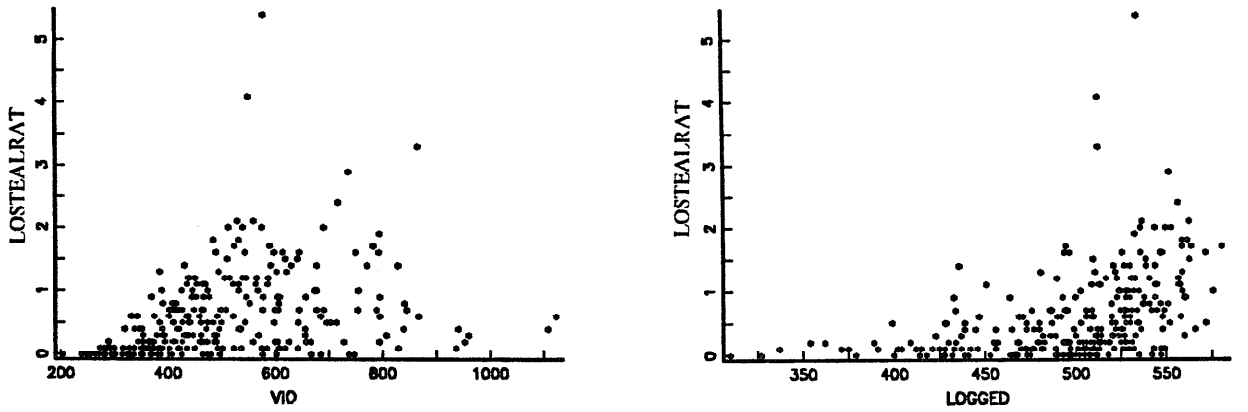

Figure 1. Scatter plots

quires that transient data structures be allocated in low memory, and so LOSTEALRAT should increase with VIO as well. That is,

MR_1: LOSTEALRAT increases with VIO and LOGGED.

MR_1 provides analysts with an approach to searching historical data for potential bounds. For example, to find a lower bound for the point VIO = 500 and LOGGED = 500, the analyst considers data for which VIO $\leq$ 500 and LOGGED $\leq$ 500. Similarly, finding an upper bound involves examining data for which VIO $\geq$ 500 and LOGGED $\geq$ 500.

Once the sets of potential bounds are identified, analysts often resort to heuristics, such as setting y_L to the largest element in the set of potential lower bounds and y_H to the smallest element in the set of potential upper bounds. Unfortunately, such heuristics do not indicate the confidence level of the resulting prediction interval, and they certainly do not permit choosing bounds so that a particular confidence level is achieved.

Translating the above approach into a formal statistical technique requires that we address randomness in the measurement data. For example, Figure 1 displays scatter plots of LOSTEALRAT vs. LOGGED and LOSTEALRAT vs. VIO for measurements taken from a VM/SP computer system; these plots suggest a high degree of randomness. We say that a monotone relationship exists between the **response variable** y (e.g., LOSTEALRAT) and the **explanatory variables** $x_1, \ldots, x_J$ (e.g., x_1 = VIO and x_2 = LOGGED) if, and only if, there is a monotone function g such that

$$y_i = g(\mathbf{x}_i) + \varepsilon_i, \quad (1)$$

where y_i is the i-th measurement of the response variable, $\mathbf{x}_i = (x_{i,1}, \ldots, x_{i,J})$ is the i-th measurement of the explanatory variables, and ε_i is the i-th error term. Randomness is handled by the ε_i, which are assumed to be realizations of continuous, independent, and identically distributed random variables. We make no assumption about g's functional form. However, we do assume that g's **directional effects** are known; that is, for the j-th explanatory variable (x_j), we know if g is non-increasing or non-decreasing. (If g is differentiable, this is equivalent to knowing the sign of g's first derivatives.) Since we do not assume that error terms are drawn from a specific distribution and we make no assumption about g's functional form, our approach is *non-parametric*. Further, the approach that we will describe is appropriate only if there are existing measurements within the region in which a prediction is desired; that is, our approach provides interpolation, not extrapolation. These characteristics of our approach as well as its being applicable only to monotone functions motivate the name *non-parametric interpolative-prediction for monotone functions (NIMF)*.

The NIMF procedure takes as input

- $\mathbf{x}^*$ - values of explanatory variables at which a prediction is desired
- a description of g in terms of its directional effects
- $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_I, y_I)$ - historical data containing values of explanatory variables paired with the corresponding value of the response variable

NIMF computes prediction intervals by finding a lower bound (y_L) and an upper bound (y_H) for the unknown response (Y^*) such that

$$P(y_L \le Y^* \le y_H) \ge 1 - \alpha, \tag{2}$$

where $1 - \alpha$ is the desired confidence level. Typical values for $1 - \alpha$ are 75%, 90%, or 95%.

NIMF consists of three steps. The first step selects sets of potential bounds by using the monotone relationship that describes g. This is accomplished by observing that a monotone relationship imposes a partial order on values of explanatory variables. Specifically, given $\mathbf{x}_1$ and $\mathbf{x}_2$, the partial order $\mathbf{x}_1 \prec \mathbf{x}_2$ holds if, and only if, the following relationship is present for all explanatory variables (x_j):

- $x_{1j} \le x_{2j}$, if g is non-decreasing in x_j
- $x_{1j} \ge x_{2j}$, if g is non-increasing in x_j

The **set of potential lower bounds**, S_L, is the subset of $\{y_i \mid \mathbf{x}_i \prec \mathbf{x}^*\}$ that consists of the M y_i whose $\mathbf{x}_i$ are closest to $\mathbf{x}^*$. ("Closest to" is defined as the Euclidean distance measure normalized by standard deviation.) By picking $\mathbf{x}_i$ close to $\mathbf{x}^*$, we hope to reduce $|g(\mathbf{x}_i) - g(\mathbf{x}^*)|$ and hence reduce the width of prediction intervals. The **set of potential upper bounds**, S_H, is a subset of $\{y_i \mid \mathbf{x}^* \prec \mathbf{x}_i\}$, and is chosen in the same manner as S_L.

NIMF's second and third steps select y_L from S_L and y_H from S_H in a manner so that at least a $1 - \alpha$ confidence level is obtained. Our approach is similar to that taken by Bradley (1968) to obtain confidence intervals for distribution percentiles. Assuming that $y_L \le y_H$, it suffices to pick y_L and y_H such that

$$P(y_L \le Y^*) \ge 1 - \frac{\alpha}{2}$$
$$P(y_H \ge Y^*) \ge 1 - \frac{\alpha}{2}$$

To find y_L, we proceed by considering its components. Let $y_i \in S_L$, with $y_i = g(\mathbf{x}_i) + \varepsilon_i$. If g is monotone and we know the directional effect of each x_j, then $g(\mathbf{x}_i) \le g(\mathbf{x}^*)$ (by construction). So

$$\begin{aligned} P(y_i \le Y^*) &= P(g(\mathbf{x}_i) + \varepsilon_i \le g(\mathbf{x}^*) + \varepsilon^*) \\ &\ge P(\varepsilon_i \le \varepsilon^*) \\ &= .5. \end{aligned}$$

(The last step is a result of the error terms being continuous, independent, and identically distributed.) Let $N_L = size(S_L)$, and let $E_{L,k}$ be the event that at least k elements in S_L are smaller than Y^*. Since the ε_i are realizations of independent and identically distributed random variables, the binomial distribution applies:

$$P(E_{L,k}) \ge \sum_{k=n}^{N_L} \binom{N_L}{n} .5^{N_L}$$

Let $y_{L,k}$ be the k-th smallest element in S_L. A key observation is that the event $E_{L,k}$ is equivalent to the event that $y_{L,k} \le Y^*$(Waite, 1987) . Hence,

$$P(y_{L,k} \le Y^*) \ge \sum_{k=n}^{N_L} \binom{N_L}{n} .5^{N_L}.$$

NIMF's second step finds k_L such that

$$P(y_{L,k_L} \le Y^*) \ge 1 - \frac{\alpha}{2},$$

and k_H such that

$$P(y_{H,k_H} \ge Y^*) \ge 1 - \frac{\alpha}{2}.$$

To minimize the width of prediction intervals, NIMF finds the largest k_L and the smallest k_H that satisfy the above inequalities. In its third step, NIMF selects the prediction interval bounds; $y_L = y_{L,k_L}$ and $y_H = y_{H,k_H}$.

We illustrate the second and third steps of the NIMF procedure by computing a prediction interval for LOSTEALRAT when VIO = 500 and LOGGED = 500. Let $1 - \alpha = 75\%$, and assume that the following potential-bounds sets have been constructed:

$$\begin{aligned} S_L &= \{0, .1, .2, .3, .6, 1.2\} \\ S_H &= \{.5, .7, 1.1., 1.4, 1.5, 1.6\} \end{aligned} \tag{3}$$

From Eq. (3), we see that both sets have six elements, so $N_L = N_H = 6$. Let $\phi(N, \alpha)$ be defined as follows

$$\phi(N, \alpha) = \max \{k \mid \sum_{n=k}^{N} \binom{N}{n} .5^N \ge 1 - \frac{\alpha}{2}\}. \tag{4}$$

Then,

$$\begin{aligned} k_L &= \phi(N_L, \alpha) \\ &= \phi(6, .25), \end{aligned}$$

Or,

$$\phi(6, .25) = \max \{k \mid \sum_{n=k}^{6} \binom{6}{n} .5^6 \ge .875\}.$$

Performing the necessary computations, we determine that $k_L = 2$. For k_H, we have

$$\begin{aligned} k_H &= N_H - \phi(N_H, \alpha) + 1 \\ &= 6 - 2 + 1 \\ &= 5. \end{aligned}$$

We use these indexes to find the prediction interval bounds; y_L is the second smallest element in S_L, and

y_H is the fifth smallest element in S_H. That is, $y_L = .1$, and $y_H = 1.5$.

NIMF's ability to produce prediction intervals depends on the historical data provided and the monotone relationships used. When NIMF cannot compute a lower bound, $y_L = -\infty$; when an upper bound cannot be computed, $y_H = \infty$. One situation in which NIMF cannot produce a bound is when there is insufficient historical data; that is, N_L (N_H) is so small that $k_L = 0$ ($k_H = N_H + 1$) at the $1 - \alpha$ confidence level. In most computer installations, data are cheap to collect and plentiful; so a missing bound can often be obtained by simply including more data. Alternatively, the analyst can reduce the confidence level.

There is a second situation in which NIMF can produce prediction intervals, but the results are inconsistent with the monotone relationship. This situation occurs when $y_L > y_H$. A statistically valid prediction interval can be produced by taking y_L to be the smaller bound and y_H to be the larger bound. However, our feeling is that this situation suggests an error in the underlying model, which should be surfaced to the user.

3. Case Study

This section presents a case study in which NIMF's predictions are compared to those of linear least-squares regression (hereafter, just *regression*), a widely-used statistical technique that requires an algebraic specification of variable relationships (Draper and Smith, 1968). We compare NIMF and regression by using the data in Figure 1 as the historical data from which NIMF potential-bounds sets are obtained and regression constants are estimated. Prediction intervals are then constructed at values of VIO and LOGGED (the $\mathbf{x}^*$ variables) contained in separately acquired test data; the test data also include measurements of LOSTEALRAT at each $\mathbf{x}^*$, which we use to evaluate the prediction intervals.

Prediction intervals are typically evaluated based on two criteria:

- coverage (percent of LOSTEALRAT values in the test data that lie in their prediction interval)
- prediction interval width

Since confidence level is a user-specified parameter, coverage is viewed as a constraint rather than an optimization criteria. So, the preferred technique is the one that minimizes prediction interval width subject to the constraint that coverage is at least as large as the specified confidence level.

First, we briefly describe the regression procedure. A regression model takes the same form as Eq. (1), but stronger assumptions are made: g's functional form must be known, and (to obtain prediction intervals) ε_i must be normally distributed. A functional form is an algebraic relationship with unknown constants. For example,

$$\hat{y}_i = b_0 + b_1 L_i + b_2 V_i \tag{5}$$

Where:

$\hat{y}_i$ = i-th estimated LOSTEALRAT
L_i = i-th measured LOGGED
V_i = i-th measured VIO

Here, the unknown constants are the b_j. In essence, regression is a curve-fitting technique: Unknown constants are estimated by using the historical data to find values that minimize the total squared error, where $\varepsilon_i = y_i - \hat{y}_i$. The quality of a regression model can be evaluated by R^2, which is the fraction of the response variability that is accounted for by the regression model.

To compare NIMF and regression, we need to construct models using both approaches. A NIMF model is a monotone relationship; we use MR_1. For regression, the choice of model is more difficult since we must specify an algebraic relationship for an unknown g. Our approach is to approximate g by an n-degree polynomial. We choose n by considering polynomials of increasing degree until there is no improvement in R^2. Equation 5 is a first degree polynomial. Below are second and third degree polynomials.

$$\hat{y}_i = b_0' + b_1' L_i + b_2' V_i + b_3' L_i^2 + b_4' L_i V_i + b_5' V_i^2 \tag{6}$$

$$\hat{y}_i = b_0'' + b_1'' L_i + b_2'' V_i + b_3'' L_i^2 + b_4'' L_i V_i + b_5'' V_i^2 + b_6'' L_i^3 + b_7'' L_i^2 V_i + b_8'' L_i V_i^2 + b_9'' V_i^3 \tag{7}$$

For the data in Figure 1, the R^2 for Eq. (5) is .26; for Eq. (6), .34; and for Eq. (7), .37. A fourth degree polynomial showed no increase in R^2; so we use Eq. (7).

Figure 2 plots 75% prediction intervals for the test data, both for regression and for NIMF[1]. The plots show the measured value of LOSTEALRAT for each

[1] We use $M = 20$.

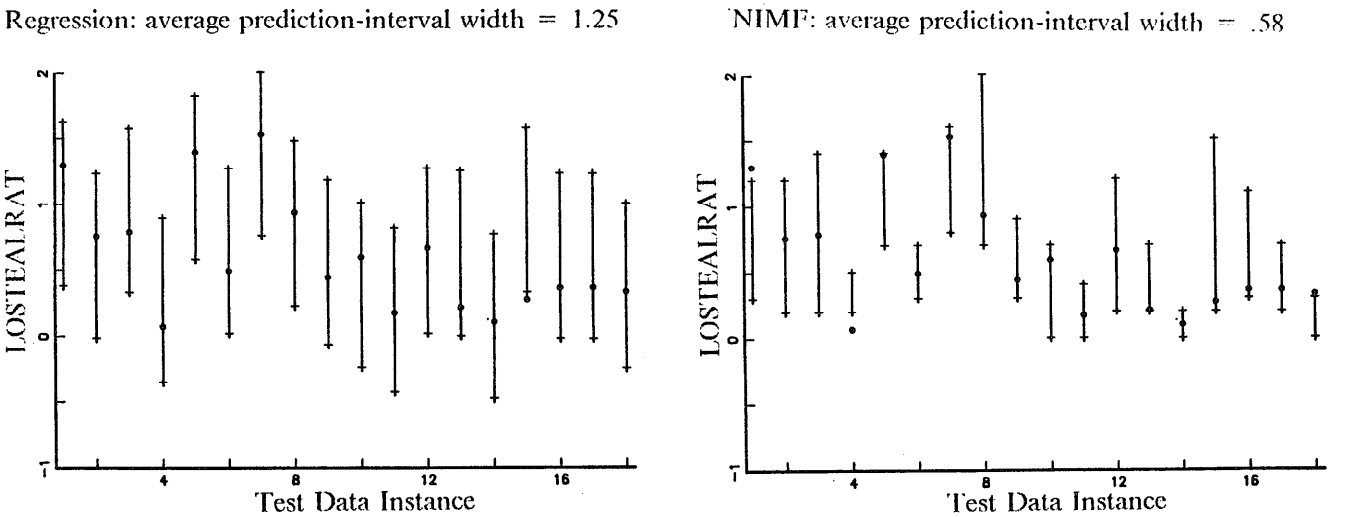

Figure 2. Prediction Intervals

test-data instance (depicted by a dot) and the associated prediction interval (indicated by a vertical line with a horizontal bar at each end). Both techniques achieve adequate coverage: 94% for regression and 83% for NIMF. However, the average width of NIMF prediction intervals (.58) is less than half that of the regression prediction intervals (1.25). Also, in several instances the regression prediction interval includes negative values, which is impossible for LOSTEALRAT (a rate). In contrast, NIMF prediction intervals are constrained to lie within the measured data; so NIMF predicts only non-negative values for LOSTEALRAT.

The foregoing is one of eighteen case studies in which we compared NIMF to regression using measurements of VM/SP computer systems (Hellerstein, 1987). The results of the other studies parallel those contained in Figure 2: In all cases adequate coverage is provided by both techniques, but NIMF consistently (17 out of 18 case studies) produces smaller prediction intervals.

Why does NIMF produce smaller prediction intervals? One reason is that regression assumes a specific algebraic relationship between the response and explanatory variables. If the wrong equation is chosen, then the fit is poor and prediction intervals are large. This shortcoming can, in part, be avoided by using other curve fitting techniques (e.g., cubic splines), which consider families of curves. However, these techniques still implicitly assume algebraic relationships, and are complex to apply to multivariate data. *NIMF avoids these problems by not making any assumption about the form of the unknown monotone function.* Instead, NIMF uses the function's monotonicity to construct sets of potential bounds from which prediction-interval end-points are selected.

4. Conclusions

Frequently, we require quantitative predictions for systems in which numerical data are available but the following situation exists:

- There is no known algebraic characterization for the system.
- The system can be characterized easily in terms of monotone relationships.

One could obtain quantitative predictions by approximating the unknown algebraic relationship by a simple function, such as a polynomial. This paper presents an alternative approach: generating quantitative predictions directly from monotone relationships.

Our approach, non-parametric interpolative-prediction for monotone functions (NIMF), is statistical, and hence assumes the presence of historical data (which is reasonable for domains such as computer performance, financial analysis, and demographic studies). NIMF uses the partial order imposed by a monotone relationship to construct sets of potential bounds; S_L is the set of potential lower

bounds, and S_H is the set of potential upper bounds. A simple technique based on non-parametric statistics is then employed to select the lower bound of the prediction interval (y_L) from S_L, and the upper bound (y_H) from S_H.

Do we obtain better predictions by using an accurate monotone relationship instead of an approximate algebraic relationship? Although the answer depends on many factors (e.g., the system being studied and the approximation used), our experience with predicting low-memory contention in VM/SP suggests that using an accurate monotone relationship with the NIMF procedure can produce significantly better predictions than using a polynomial approximation of the unknown algebraic relationship and employing least-squares regression. Admittedly, NIMF's superior results are not solely a consequence of using monotone relationships instead of algebraic relationships, since NIMF also makes weaker assumptions about the distribution of error terms. However, avoiding unnecessary assumptions about algebraic relationships is clearly an advantage in terms of predictive accuracy. Also, using monotone relationships simplifies model building and greatly facilitates explaining predictions.

NIMF has been implemented in APL and Prolog; the results presented here are from the Prolog implementation. Prolog is a particularly good implementation language for NIMF since monotone relationships are easily expressed as facts, and simple predicates can be used to find the sets of potential bounds.

Acknowledgements

We wish to thank Robert Berry and William W. White for their helpful comments.

References

Chidanand Apte & Se June Hong. (1986) Using Qualitative Reasoning to Understand Financial Arithmetic. *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 942-948.

James Bradley. (1968) *Distribution-Free Statistical Tests*. Prentice-Hall.

Johan De Kleer and John Seely Brown. (1984) A Qualitative Physics Based on Confluences. *Artificial Intelligence*, **24**, 7-83.

N. R. Draper & H. Smith. (1968) *Applied Regression Analysis*. John Wiley & Sons.

Daniel Dvorak & Benjamin Kuipers. (1989) Model-Based Monitoring of Dynamic Systems. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 1238-1243.

Daniel Dvorak & Elisha P. Sacks. (1989) Stochastic Analysis of Qualitative Dynamics. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 1187-1192.

Kenneth D. Forbus. (1984) Qualitative Process Theory. *Artificial Intelligence*, **24**, 85-168.

Joseph Hellerstein. (1987) An Intuitive Approach to Performance Prediction with Application to Workload Management in VS SP/HPO. *Proceedings of the Computer Measurement Group*, **December**, 53-63.

IBM. (1985) *Virtual Machine Monitor Analysis Program: User's Guide and Reference*. IBM Corporation, (SC34-2166).

Leonard Kleinrock. (1975) *Queueing Systems, Volume 1*. John Wiley.

Benjamin Kuipers. (1986) Qualitative Simulation. *Artificial Intelligence*, **29**, 289-338.

Norman Waite. (1987) Two Non-Parametric Confidence Intervals. Yorktown Heights, New York: IBM Corp. RC 13268.

Michael P. Wellman. (1987) Probabilistic Semantics for Qualitative Influences. *Proceedings of the Sixth National Conference on Artificial Intelligence*, pages 660-664.