Multivariate Resource Performance Forecasting in the Network Weather Service*
UCSB Computer Science Technical Report 2002-12

Martin Swany and Rich Wolski
Department of Computer Science
University of California
Santa Barbara, CA 93106
{*swany,rich*}@*cs.ucsb.edu*

Abstract

This paper describes a new technique in the Network Weather Service for producing multi-variate forecasts. The new technique uses the NWS's univariate forecasters and emprically gathered Cumulative Distribution Functions ($CDFs$) to make predictions from correlated measurement streams. Experimental results are shown in which throughput is predicted for long TCP/IP transfers from short NWS network probes.

1 Introduction

Performance monitoring in Computational Grid settings is widely recognized as an essential capability [6, 8, 13, 24, 27] and there are a variety of systems available for taking those measurements, particularly for network performance [3, 7, 9, 12, 15, 17, 21, 24, 28]. For the Grid, this information is used, primarily, to optimize application execution. Applications are directed (either by a human user or an automatic scheduler) to use the resource (network, CPU, memory system, etc.) that exhibits the best measured performance.

Frequently, monitor data gathered from these tools is used as as a *prediction* of future performance. That is, an observation of past performance implies that future performance levels will be similar. For network bandwidth estimation, in particular, many users simply conduct lengthy data transfer between end-hosts, observe the throughput, and use that observation as a harbinger of future available bandwidth.

There are two significant problems with this methodology. First, it is not clear that the last observation, particularly of network throughput, is a good estimate of future levels since network performance can fluctuate significantly [14, 30]. Secondly, it requires that the resource be loaded "enough" to yield a significant estimate. On high-throughput networks with lengthy round-trip times (e.g. the NSF Abilene [1] network) enough data must be transferred to match the bandwidth-delay product [11] for the end-to-end route. This intrusiveness can be costly, both in terms of wasted resource and lost time while the probe takes place.

Performance monitoring tools such as the Network Weather Service (NWS) [23, 28] address the first problem by making the predictions explicitly using statistical techniques. Application-level schedulers [2, 4, 16, 20] have been able to use these predictions (and measures of prediction error) to achieve good execution performance levels in a variety of Grid settings, despite fluctuating resource performance.

However, to make these forecasts, current NWS methodologies (described in [27, 28]) share the second problem with other performance monitoring tools for the Grid. Performance responses that are expensive to generate with probing, such as steady-state TCP/IP throughput, are unfortunately best forecast from a history of such responses. Moreover, the dynamics of the network are frequently changing, making old measurements increasingly less statistically valuable as time passes. As such, instrumenting Grid applications or tools (e.g. GridFTP [2]) and using the observations from the instrumentation may yield dramatic inaccuracies if the time between application runs is significant.

In this work, we describe a new multivariate forecasting technique that enables the NWS to automatically correlate monitor data from different sources, and to exploit that observed correlation to make better, longer-ranged forecasts. In particular, this new cor-

*This research was supported in part by NSF grant ANI-0123911.

relative method allows the NWS to combine infrequent and irregularly spaced expensive measurements (possibly obtained through instrumentation) with regularly spaced, but far less intrusive probes to make predictions that would be far too expensive to generate with probes alone.

We demonstrate the effectiveness of the technique by showing how it can be used to forecast long HTTP transfers using a combination of short NWS TCP/IP bandwidth probes and previously observed HTTP transfers. Using short TCP/IP messages to predict steady-state throughput is the subject of significant research [5, 10, 26]. Our particular experimental verification is similar to the work described in [25], but unlike that work, our new method does not rely on linear regression.

Our results indicate that the combination of short NWS bandwidth probes and previous HTTP history can be used to generate more accurate HTTP transfer bandwidth forecasts than has been previously possible, particularly when the time between observable HTTP transfers is long.

More generally, we observe that it is often advantageous to combine data from two or more measurement streams which may correlated in some way. The HTTP experiment described herein is an example of how heavy-weight operations that would be too intrusive to duplicate as explicit probes can be monitored and combined with lightweight, inexpensive probes of a resource to produce a forecast of the heavy-weight operation. Similar correlation problems exist when predicting CPU availability [29] and real memory. As such, we believe that this new forecasting technique can be used to combine probe monitor data with application instrumentation data (e.g. internal code timings) gathered from a computation by a mechanism such as Autopilot [18] or Tau [19].

In summary, the contributions that this new work makes are:

- the description of a new correlation technique that greatly improves observed prediction error, particularly for longer-ranged predictions than have been previously studied, and
- an empirical evaluation of the technique to the problem of long-message bandwidth prediction using short-message bandwidth probes.

Our results show better prediction accuracy than has been previously reported in either the networking or Grid computing literature and indicate a general methodology for combing historical instrumentation data with periodic explicit measurements.

2 Forecasting using Correlation

The NWS generalized forecasting system can be thought of as having access to a collection of measurement series. These series are characterized by units and frequency. In order to make multivariate predictions, a system needs to operate on some subset of these measurement series with a combination of *correlation* and *forecasting*. The correlator serves to map X values onto Y values where the X values are plentiful and "cheap" and the Y values are "rare" and expensive. The univariate forecaster produces forecasts, which represent the current and future performance of a single dataset. At present, the NWS provides us with a rich set of tools for the latter, and our goal is to augment those tools by addressing the former.

Given two correlated variables, knowledge of one at a given point in time provides us some information as to the value of the other. Stated more formally, being able to fix the value of one of the variables allows us to make reasonable assumptions about the probability of various values of the other variable, based on their history of correlation. The question is how to exploit that empirically to make a prediction.

2.1 Correlator Methodology

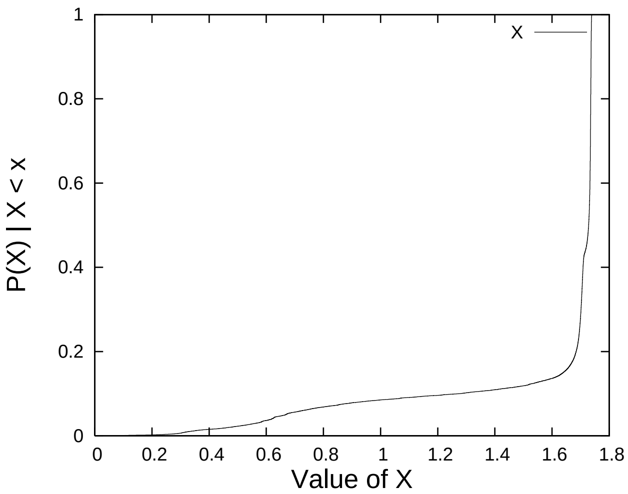

Figure 1. CDF **of NWS data (**X**)**

The correlator can determine the mapping between the X and Y values in a number of ways. Methods such as linear regression and traditional correlation operate on datasets of equal sizes and assume that the variables

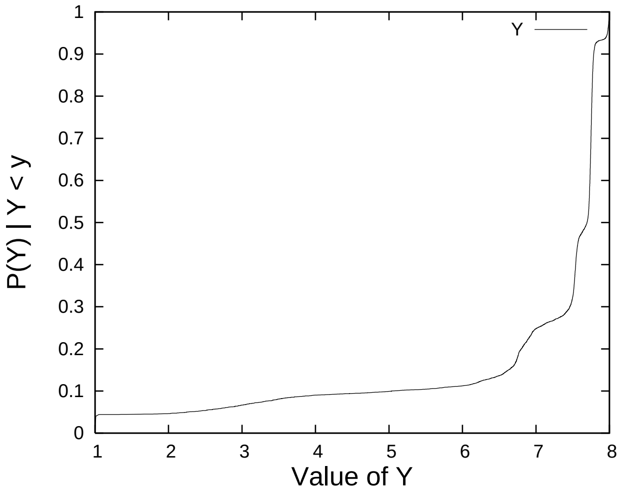

Figure 2. CDF **of HTTP data (**Y**)**

in question are related linearly. However, in the case of measurement data gathered from a variety of sources:

- data sets may be of different sizes
- data items within each set may be gathered with different frequencies and/or with different regularities.
- the units of measure may be different between data sets.

The mechanism that our correlator uses is similar to rank correlation techniques. A rank correlation measure sorts datasets does linear correlation based on their position in the sorted list, (their "rank"). For our purposes, rank correlation is attractive because it is non-parametric. A non-parametric technique is appropriate since we cannot be certain of the distribution of the data. However, if there are different amounts of data in the respective datasets, something must be done to make the set sizes equal.

The Cumulative Distribution Function (CDF) is defined as the probability for some set of sample points that a value is less than or equal to some real valued x. If the probability density function (PDF) is known, then it is defined as follows:

$$CDF_X(x) = \int_{-\infty}^{x} PDF_X \tag{1}$$

Empirically, we can approximate the CDF by:

$$CDF_X(x) = \sum_{i=1}^{position_x} \frac{1}{count_x} \tag{2}$$

Where $count_x$ is the total elements in the sample set. Which is equivalent to:

$$CDF_X(x_{target}) = \frac{count_{x'} \mid x' \leq x_{target}}{count_x} \tag{3}$$

So, we are able to compute the empirical CDF for both the X and Y variables (CDF_X and CDF_Y, respectively.) Now we have a way to rank measurements in datasets of different sizes. By computing the CDF for multiple datasets, we are able to compare them in a way similar to the rank correlation technique.

However, since our goal is forecasting, we are not interested in using this to quantify the correlation between the X and Y datasets, but rather in using them as a mapping function from X to Y[1] We can use the position of value in X (x_{target}) to produce a value of Y ($y_{forecast}$). This allows us to inexpensively compute a relationship between the two sets.

Specifically:

$$y_{forecast} = CDF_Y^{-1}(CDF_X(x_{target})) \tag{4}$$

This gives is the value in Y (in CDF_Y) associated with the position x_{target}. If the CDF position of x_{target} ($CDF_X(x_{target})$) is between measured Y values, linear interpolation is used to determine ($y_{forecast}$).

The x_{target} can be chosen in a variety of ways. Tests have shown that the most effective way to choose x_{target} is by using the NWS forecasters to produce the value. This is because the univariate forecast is less sensitive to transient outliers (i.e. for all the same reasons that univariate forecasting is effective at all.) This is simply:

$$x_{target} = Forecast(X) \tag{5}$$

[1]Indeed, no correlation coefficient is defined for this novel correlation technique, although we are investigating this for selection among datasets.

As an illustration of the CDF forecaster, consider the plots of the CDFs computed from the data described in Section 4. The NWS (X) values are shown in Figure 1 and the HTTP (Y) values are shown in Figure 2. To compute the forecast for Y we use the position in the CDF of x_{target}, which yields some real number between 0 and 1. The forecast is the value of y at that point in the CDF of Y.

The CDF is a useful measure for a number of reasons. One reason deals with the practicality of delivering this information to clients of this system – Grid programs and schedulers. A CDF dataset may be compressed with varying degrees of loss. In simple cases, a handful of points can describe the CDF with linear interpolation between specified points.

This predictive method then forms both the X and Y CDFs. At each prediction cycle, a univariate forecast of X is taken and that X value's position is determined. That value is used to determine the value in Y at the corresponding position, which is the prediction.

We have implemented an adaptive version of this correlator that uses varying amounts of history used to form the CDF of each dataset. These different versions are selected against based on their accuracy in the same way that the univariate forecasters do [28]. We have also implemented a prototype that alters the computation of the CDF and weights current values more heavily when creating the mapping function.

The final paper will deal with the issues of computation and compression of CDFs in greater detail.

2.2 Error

The NWS uses prediction error internally to choose optimal forecasts and provides that information to clients as a measure of the forecast accuracy.

Some notion of normalized error is necessary so that over-prediction and under-prediction (and excessively high or low measurement values) are represented fairly. For instance, 1Mbit prediction error is quite different for transfers that observe 100Mbits/second than for those that see 10Mbits/second. So, to determine the accuracy of various datasets which might not have the same units and to evaluate the overall performance of the system, we compute the Mean Normalized Error Percentage (MNEP) as the last prediction's error over the current average value.

$$MNEP = \sum |value_t - prediction_t| / mean_t \quad (6)$$

Again, this is required due to the forecast-oriented application of the system. The mean used ($mean_t$) is the current mean and is reevaluated at each timestep.

3 Experimental Methodology

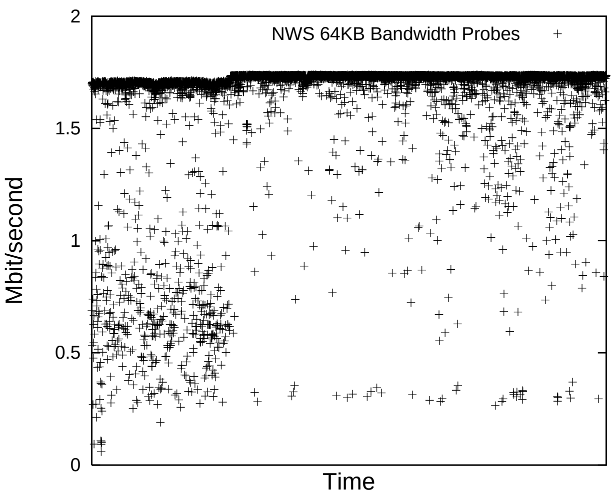

Figure 3. 64KB NWS Probes.

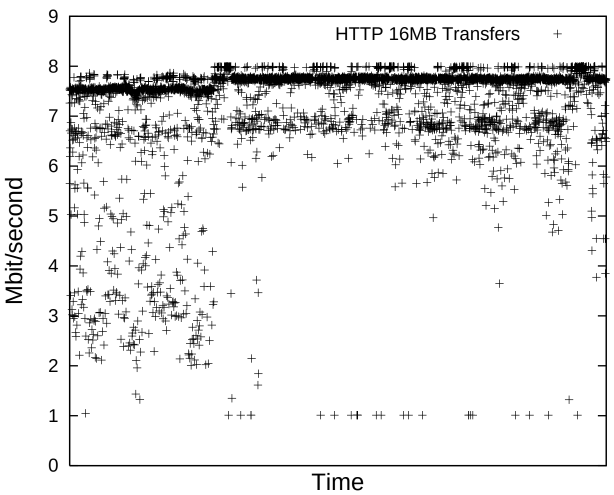

Figure 4. 16MB HTTP Transfers.

To investigate the effectiveness of our new technique, we used the following experimental procedure. We generated default-sized NWS bandwidth measurements (64K bytes), using the TCP/IP end-to-end sensor between a pair of machines every 10 seconds. Every minute, we initiated a 16MB HTTP transfer and the recorded the observed transfer bandwidth. The NWS data is depicted in Figure 3 and the HTTP data is shown in Figure 4. While the exact correlative relationship is not clear visually, the shapes of each data set appear to share similar characteristics, particularly with respect to the presence of modes.

Our goal was to use this data to investigate the accuracy with which the short 64K NWS transfers could be used to predict the long 16MB HTTP transfers given successively longer periods between 16MB transfers. This experimental configuration emulates the case where instrumentation data gathered from an application (e.g. HTTP/web client) is ranging from plentiful (frequent 16MB transfers) to scarce. In particular, we wish to study how well NWS 64K messages can be used to predict large messages that are separated by several hours as the network conditions typically change over this time scale.

We chose to use HTTP transfers as a prediction target for several reasons. HTTP is, of course, widely used in the Internet. Further, HTTP is a key component of SOAP and Web services and we are likely to see this used more and more in high-performance computing. This data represents two of the cases mentioned previously. That is, it is representative both of the relationship between short and long network experiments, and also of instrumentation data (which it indeed is) being fed into the forecasting service.

Our basis of comparison is prediction error measures as either the mean absolute difference between prediction and corresponding measurement, or the square root of the mean square prediction error. We compare three prediction methods: the *last value* method which uses the last observed value as a prediction, the univariate NWS forecast applied solely to the 16MB probe history, and the new correlative method.

We chose 16MB as our test case to limit intrusiveness. Also, there is less variability, in general, in the observed bandwidth as the amount of data transferred gets larger (evidenced one of many places by the better predictive accuracy of longer transfers seen in [25]). This can be intuitively explained by the notion that any transient periods of high congestion are amortized over the longer transfer time. So, 16MB transfers are not so long as to make things easy, but long enough to represent real applications and real data.

4 Results

The following data represents 10 days of measurement data between Argonne National Laboratory (ANL) and UC Santa Barbara (UCSB). These results are based on a version of the forecaster we have implemented that automatically computes the empirical CDF for each forecast incrementally, based on all previous measurement data. That is, the forecaster "learns" as it goes and acquires more data. In these results we train the CDF with fifteen 16MB measurements before we begin tabulating forecast error. The X-axis of these graphs represents the time between 16MB transfers. That is, we gradually decimate the Y dataset until there are measurements approximately every 500 minutes.

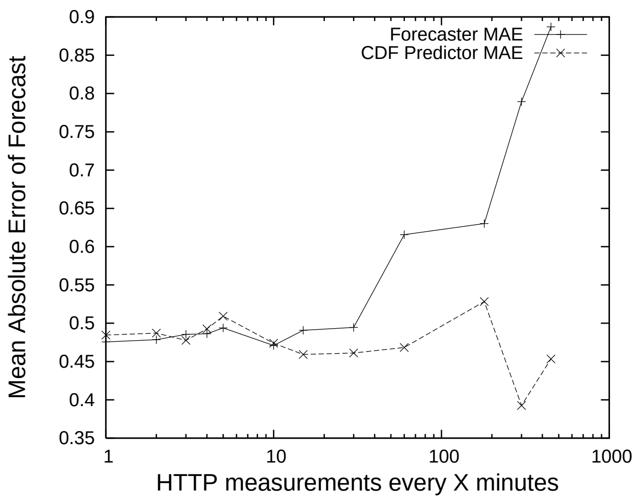

Figure 5. Comparison of Mean Absolute Error (MAE) between univariate and multivariate forecasts for different frequencies of HTTP measurements.

Figure 5 shows the Mean Absolute Error (MAE) of the NWS univariate forecasters compared to the MAE of the correlating forecasters. The NWS uses the MAE as it is a unit-preserving metric of success. We can see that the correlating forecaster enjoys slightly better, but comparable, performance as the 16MB Y values are being measured frequently, but that is looses effectiveness as the Y values become less frequent. Notice that the mean absolute prediction error is only 0.47 megabits/second at 500 minutes. That is, using the new forecasting technique, the NWS can predict 16MB transfer bandwidth using 64K measurements with a mean error of 0.47 megabits/second.

To give some notion of magnitude, we normalize the MAE by the average transfer bandwidth resulting in the MNEP. Figure 6 shows the MNEP of the MAE. The MNEP shows us the percentage of the current average value that an error represents. This is not unit-preserving, but depicts the error values in relation to the moving average of the data. Again, at 500 minutes, the 64K transfers, when used in our forecaster, can predict 16MB transfer bandwidth to with 7% on the average.

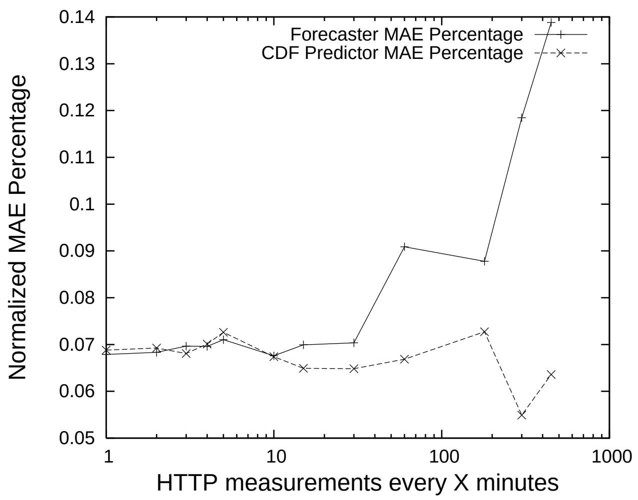

Figure 6. Comparison of Moving Normalized Error Percent (MNEP) of the Mean Absolute Error (MAE) between univariate and multivariate forecasts for different frequencies of HTTP measurements.

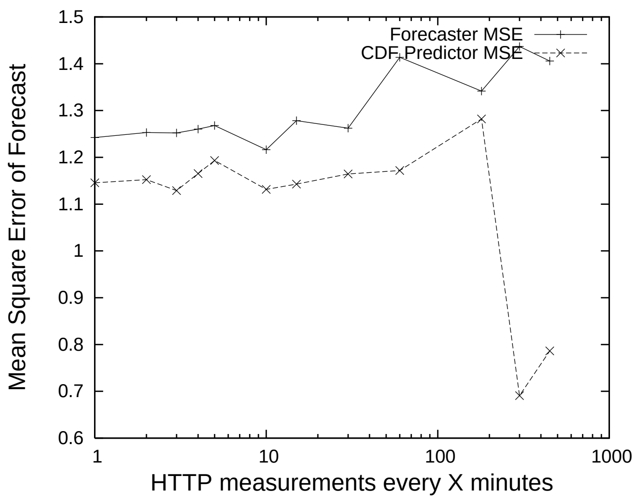

Figure 7. Comparison of the square root of the Mean Square Error (MSE) between univariate and multivariate forecasts for different frequencies of HTTP measurements.

Figure 7 shows the square root of the Mean Square Error (MSE) of both types of forecasts. This value bears resemblance to the traditional notion of standard deviation and tends to indicate the variance of the error. Using a combination of the univariate NWS forecasting technique and incrementally constructed empirical CDFs, our new technique can predict actual throughput with high-levels of accuracy.

For comparison purposes, we also include a "forecast" based on the last value. This is the "accuracy" of a prediction that assumes the the performance of a given transfer will be the same as it was the last time it was performed – actually, a common way monitor data is used in the "real" world. Figure 8 show the MNEP of the MAE for the CDF forecaster and the "Last Value" predictor. This shows that using the last value as a predictor for this dataset yields "forecasts" with over 100 percent error. Clearly, forecasting yields predictive accuracy that significantly higher than the base case "last-value" approach.

In a final paper, should this extended abstract be accepted, we will include two different classes of results. The first will be a verification of these results using the data multiple datasets, including those used in [25]. Also, we will discuss the issue of error introduced as the CDF is populated at forecaster start-up and the choice of the 15 value start-up set.

These results show that by combining instrumentation data with NWS forecasting, our technique is able to use relatively short transfers to predict long-message throughput. Moreover, the predictive accuracy does not degrade substantially when forecasts of events occurring hours apart are used as inputs. No other methodology of which we are aware from either the Grid or networking research communities has demonstrated the level of accuracy, particularly over long prediction periods. Our work indicates that the statistical techniques used by the NWS can extract and exploit the inherent performance relationship that must exist between message transfers of different sizes, and does so automatically.

5 Future Work

The implications of the predictive efficacy of CDFs are a key component of the NWS Topology Service (described in [22].) This system stores topology information and assists the predictive mechanisms of the NWS in identifying relevant data sources. Since the CDF of a given time-series of data can be be compressed significantly, it will be possible to deliver a CDF to clients and allow them to compute their own forecasts.

6 Conclusion

We have described a general Grid performance prediction architecture as well as our implementation of predictive techniques. We have shown a novel predictive technique based on the Cumulative Distribution

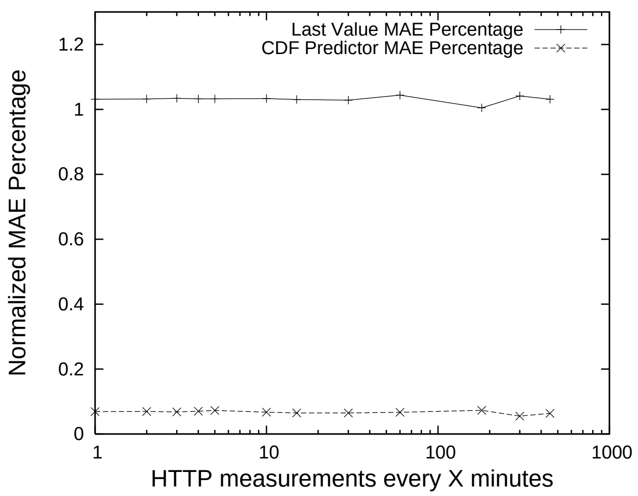

Figure 8. Comparison of Moving Normalized Error Percent (MNEP) of the Mean Absolute Error (MAE) between "Last Value" and multi-variate forecasts.

Functions (CDFs) of two time-series and observed that its performance is promising.

References

[1] Abilene. http://www.ucaid.edu/abilene/.

[2] B. Allcock, I. Foster, V. Nefedova, A. Chervenak, E. Deelman, C. Kesselman, J. Lee, A. Sim, A. Shoshani, B. Drach, and D. Williams. High-performance remote access to climate simulation data. In Proc. SC2001, Denver, Colorado, N ovember 2001.

[3] Active measurement project (AMP). `http://amp.nlanr.net`.

[4] F. Berman, R. Wolski, S. Figueira, J. Schopf, and G. Shao. Application level scheduling on distributed heterogeneous networks. In *Proceedings of Supercomputing 1996*, 1996.

[5] C. Dovrolis, P. Ramanathan, and D. Moore. What do packet dispersion techniques measure? IEEE Infocom, April 2001.

[6] I. Foster and C. Kesselman. *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers, Inc., 1998.

[7] IPerf. `dast.nlanr.net/Projects/Iperf/`.

[8] The nasa information power grid. `http://science.nas.nasa.gov/Pubs/NASnews/97/09/ipg.html`.

[9] V. Jacobson. A tool to infer characteristics of internet paths. available from `ftp://ftp.ee.lbl.gov/pathchar`.

[10] M. Mathis and J. Madhavi. Diagnosing internet congestion with a transport layer performance tool. In *Proceedings of INET '96*, 1996.

[11] M. Mathis, J. Semke, J. Mahdavi, and T. Ott. The macroscopic behavior of the TCP congestion avoidance algorithm. Computer Communications Review, 27(3), July 1997., 1997.

[12] Net100. `www.net100.org`.

[13] Netlogger. `http://www-didc.lbl.gov/NetLogger`.

[14] V. Paxson. End-to-end internet packet dynamics. *IEEE/ACM Transactions on Networking*, 7(3):277–292, 1999.

[15] V. Paxson, J. Mahdavi, A. Adams, and M. Mathis. An architecture for large-scale internet measurement. *IEEE Communications*, 1988.

[16] A. Petitet, S. Blackford, J. Dongarra, B. Ellis, G. Fagg, K. Roche, and S. Vadhiyar. Numerical libraries and the grid. In *(to appear) Proc. of SC01*, November 2001.

[17] PingER. `http://www-iepm.slac.stanford.edu/pinger/`.

[18] R. L. Ribler, J. S. Vetter, H. Simitci, and D. A. Reed. Autopilot: Adaptive control of distributed applications. In *Proc. 7th IEEE Symp. on High Performance Distributed Computing*, Aug 1998.

[19] S. Shende, A. Malony, J. Cuny, K. Lindlan, P. Beckman, and S. Karmesin. Portable profiling and tracing for parallel scientific applications using c++, 1998.

[20] N. Spring and R. Wolski. Application level scheduling: Gene sequence library comparison. In *Proceedings of ACM International Conference on Supercomputing 1998*, July 1998.

[21] Surveyor. `http://www.advanced.org/surveyor/`.

[22] M. Swany and R. Wolski. Building performance topologies for computational grids. In Submission, 2002.

[23] M. Swany and R. Wolski. Representing dynamic performance information in grid environments with the network weather service. In *2nd IEEE International Symposium on Cluster Computing and the Grid (to appear)*, May 2002.

[24] B. Tierney, R. Aydt, D. Gunter, W. Smith, V. Taylor, R. Wolski, and M. Swany. A Grid Monitoring Architecture. Grid forum working group document, Grid Forum, Feburary 2001. http://www.gridforum.org.

[25] S. Vazhkudai and J. Schopf. Predicting sporadic grid data transfers. In *Proceedings 11th IEEE Symposium on High Performance Distributed Computing (to appear)*, July 2002.

[26] S. Vazhkudai, J. Schopf, and I. Foster. Predicting the performance of wide area data transfers. In *Proceedings of the 16th Int'l Parallel and Distributed Processing Symposium (IPDPS 2002)*, April 2002.

[27] R. Wolski. Dynamically forecasting network performance using the network weather service. *Cluster Computing*, 1998. also available from http://www.cs.utk.edu/ rich/publications/nws-tr.ps.gz.

[28] R. Wolski, N. Spring, and J. Hayes. The network weather service: A distributed resource performance forecasting service for metacomputing. *Future Generation Computer Systems*, 1999.

[29] R. Wolski, N. Spring, and J. Hayes. Predicting the cpu availability of time-shared unix systems on the computational grid. In *Proc. 8th IEEE Symp. on High Performance Distributed Computing*, 1999.

[30] Y. Zhang, V. Paxson, and S. Shenker. The stationarity of internet path properties: Routing, loss, and throughput. *ACIRI Technical Report*, 2000.