Feedback Performance Control for Computer Systems: An LPV Approach

Wubi Qin and Qian Wang, *Member, IEEE*

Abstract— **There has been increasing research effort in applying control-theoretic approaches to performance management for computer systems such as web servers, database systems, and storage systems. Since today's Internet servers and applications are often operated under dynamically changing load conditions, linear control designs may not suffice to provide desired performance guarantees. This paper presents a Linear-Parameter-Varying (LPV) approach to the modeling and performance control of web servers in order to achieve performance-based Service Level Agreement (SLA). In particular, admission control for web servers is used as a case study. By applying system identification to empirical data, an LPV system is developed to approximate the dynamics of web server admission control from rejection ratio to response time, where the time-varying workload parameters are chosen as scheduling variables. An LPV robust control is then designed to meet response time SLA. The performance of the LPV control is compared with that of a linear design. By exploring the nature of dependence of server performance on time-varying load and operating conditions, the proposed general framework are applicable to a diverse spectrum of server-based applications. The utilization of scheduling parameters can be generalized to accommodate more sophisticated workload characterization and complicated server environments.**

I. Introduction

Today's web applications and services are often housed on an Internet data/hosting center. A guaranteed level of performance, which is referred to as Quality of Service (QoS) delivered to end customers, is often part of a service-level agreement (SLA) between the service provider and an end user. A data center may contain thousands of servers and host many applications across the servers. Quality of Service can be characterized by system availability and performance criteria such as service response time and service throughput, where response time is often a primary concern for performance control of computer servers. For an incoming traffic of requests, a server system (cluster) could use different mechanisms to achieve desired values of performance metrics, e.g., admission control that admits or denies a request into the system and the control of available resources (CPU, memory, and bandwidth) that a particular application can access.

The demand on automating the management of computer servers within a hosting center to adapt to dynamically changing load and operating conditions strongly motivates the application of feedback based control mechanisms. In addition, compared to traditional approaches to performance management, which heavily depend on queuing analysis of steady-state behavior and static optimization, control-theoretic approaches allow the design to take into account transient and time-varying behavior. There has been increasing research in applying control-theoretic approaches to the performance management of computer systems in the area of network systems [8], software systems for email servers, database servers, and web servers [5]-[8], [10]; the reader can be referred to the survey paper [1] and references therein.

The existing literature on control-theoretic approaches for performance management of software systems has focused on using system identification techniques to build a linear-time-invariant model and then designing a linear control law [1]. One major concern with such linear-time-invariant models is that they do not capture the system nonlinearity, in particular when the response time is used as the performance metric. In addition, though perceivably a linear model represents the linearization of the original nonlinear system at a nominal operating condition, the resulting linear design may not suffice to allow the system to meet the target response time when the server is experiencing time-varying load conditions. There is lack of rigorous robustness analysis for such linear designs with respect to large variations in incoming traffic in current literature. In order to deal with time-varying load variations, an adaptive control has been applied to the performance control for differentiated caching services [10], where a linear adaptive controller is designed based on online-identified linear approximations of the nonlinear web cache system. Fuzzy-logic control is used to optimize performance for the Apache web server in [6].

This work is supported in part by NSF under Grant 0325056.

Wubi Qin is with the Mechanical Engineering Department, Penn State University, PA 16802 (email: wubi@psu.edu).

Qian Wang is with the Mechanical Engineering Department, Penn State University, PA 16802 (email: quw6@psu.edu; phone: 814-865-8281; fax: 814-865-9693).

0-7803-9098-9/05/$25.00 ©2005 AACC

A workload is often characterized by two complementary distributions: the request arrival process and the service demand distribution, which capture the workload intensity and its variability. By recognizing the dependence of system performance on the time-varying load conditions, this paper aims to utilize the on-line measurements of request arrival rate and service demand in the modeling and control design.

This paper presents a Linear-Parameter-Varying (LPV) approach to the modeling and design of the admission control for Internet web servers, and makes the following contributions: 1) through direct system identification using empirical data, we build a Linear-Parameter-Varying system to approximate the dynamics from the request rejection ratio of admission control to response time, where the workload's time-varying arrival rate and service demand are used as scheduling parameters; 2) based on the identified model, a Linear-Parameter-Varying robust controller is designed for admission control in achieving target response time. We show that the LPV modeling and control provides significantly improvement in stability and performance by utilizing detail load information; 3) though evaluated only for a simplistic admission control problem with limited system complexity, our framework allows the generalization to accommodate more sophisticated models for workload characterization, and is applicable to a variety of performance management problems for server clusters.

II. An LPV Formulation for Admission Control for Internet Web Servers

A. Problem Formulation

A typical web application consists of a front-end web server that services HTTP requests, a Java application server that contains the application logic, and a backend database server. While the general framework of our modeling and control are applicable to a variety of applications and server environment. We focus on the performance management of web servers that service client requests. As illustrated by Fig. 1, when a request arrives, if it is admitted by the admission control mechanism, it is directed into a queue to a server that can process the request in certain (service) time. The service time is proportional to the request's service demand (file size) and inversely proportional to the service resource that is allocated for processing the request. Consequentially, the performance metrics such as response time and throughput for these requests can be controlled through admission control and resource allocation mechanisms. In this paper, we focus on the design of admission control so that the response time SLA can be met for admitted requests.

The performance management such as admission control or resource allocation can be controlled either at request level or within a time window. For a request-level control, the decision for admission control directly determines whether to admit the next request in a queue or to simply reject it so that the response time of each request that is admitted can meet the target value. For a window-basis control, a sampling period (time window) is first chosen. Then based on the average statistics of the requests arriving in a sampling period, e.g., the mean arrival rate and mean service time (or mean service rate that is the reciprocal of service time), a control decision can be made so that the average response time within each sampling period meets the target response time $\overline{T}$ for all the sampling intervals.

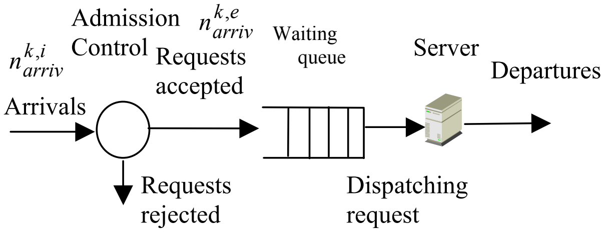


Fig. 1. Admission control for a queuing system.

Let Δt be the sampling interval. Consider the time period $\left[k\Delta t,(k+1)\Delta t\right]$, let $n^{k,i}_{arriv}$ and $n^{k,e}_{arriv}$ denote the number of incoming requests and the number of requests that are admitted to enter the queue, respectively, and let n^{k}_{srved} denote the number of requests that have been served. Assume that the *ith* request in the queue at $\left[k\Delta t,(k+1)\Delta t\right]$ has response time T_i^k. We can calculated the following statistics from measurements:

$$\lambda^i(k) = n^{k,i}_{arriv} / \Delta t\,,\ \lambda^e(k) = n^{k,e}_{arriv} / \Delta t$$

$$\mu(k) = n^{k}_{srved} / \Delta t\,,\ T(k) = \sum_{i=1}^{n^k_{srved}} T_i^k \,/\, n^{k}_{srved}$$

where $\lambda^i(k)$ denotes the incoming mean arrival rate, $\lambda^e(k)$ denotes the mean arrival rate for the requests admitted into the system, which we call the effective arrival rate, $\mu(k)$ denotes the mean service rate, and $T(k)$ denotes the average response time in the *kth* period.

We focus on window-based performance control in this paper, for which conventional control-theoretic (rather than discrete-event control) approaches can be applied. For window-based admission control design, one can choose the control variable as the maximum number of requests that are allowed to enter the system in a sampling period. In this paper, we choose the rejection ratio $\theta(k)$ as control variable for admission control. Consequentially, the admission control is performed through dynamically deciding the rejection ratio for the incoming traffic. When a rejection ratio $\theta(k)$ in the *kth* sampling period is determined, a request comes in this sampling period is denied to enter the system with probability $\theta(k)$, or is admitted to the system with probability 1- $\theta(k)$.

B. Identification of System Model

Linear ARX Model: The dynamic relation from the rejection ratio $\theta(k)$ for admission control to system response time $T(k)$ is essentially nonlinear. A simple modeling solution is to construct a linear time-invariant empirical model using system identification techniques. Assuming that a linearization of the original nonlinear dynamic system at a nominal operating condition can be approximated by an ARX model as follows,

$$A(q)T(k) = B(q)\theta(k) + e(k) \qquad (1)$$

with

$$A(q) = 1 + a_1 q^{-1} + \cdots + a_{na} q^{-na} \qquad (2)$$

$$B(q) = b_1 q^{-1} + \cdots + b_{nb} q^{-nb} \qquad (3)$$

Where q is the delay operator, *na* and *nb* determine the system order. The constant coefficients a_i and b_i are computed through running system identification algorithms on data $(\theta(k), T(k))$ that is collected at a nominal operating condition, which could correspond to a nominal load condition.

LPV-ARX: In order for the system to adapt to dynamically varying load conditions, we formulate a Linear-Parameter-Varying system by defining workload parameters as scheduling variables. That is, we model the system dynamics to depend on time-varying exogenous load parameters (request arrival rate and service demand) whose trajectories are unknown a priori but can be measured on-line by the controller. We assume that the coefficients a_i and b_i in (2-3) are functions of load conditions, which are characterized by request mean arrival rate and service demand. Thus the system dynamic model is specified as follows:

$$A(q,r)T(k) = B(q,r)\theta(k) + e(k) \qquad (4)$$

with

$$A(q,r) = 1 + a_1(r(k-1))q^{-1} + \cdots + a_{na}(r(k-na))q^{-na} \quad (5)$$

$$B(q,r) = b_1(r(k-1))q^{-1} + \cdots + b_{nb}(r(k-nb))q^{-nb} \qquad (6)$$

The coefficients $a_i(r), i = 1,\cdots,na$, $b_j(r), j = 1,\cdots,nb$ are unknown functions of $r(k)$ and are to be estimated from empirical data. The model (4) sometimes is referred to as the LPV-ARX model.

In general, $r(k)$ in (4-6) could be a vector that includes all parameters characterizing workload behavior (e.g., arrival rate, file size, locality). For simplification, a single scheduling variable, workload intensity that is defined as the ratio of incoming-request mean arrival rate $\lambda^i(k)$ and mean service rate $\mu(k)$, is used here. That is,

$$r(k) = \lambda^i(k) / \mu(k)$$

Given the on-line measurements of request arrival and service demand, $r(k)$ can be easily calculated in real time. Since the traffic load varies in a much slower time scale (usually in minutes for a web server application) compared to the system dynamics (where the response time for a web request is expected to be less than 5-6 seconds), the LPV system (4-6) is slow varying, which satisfies the conditions for a general LPV control design.

C. Identification Algorithms for LPV Systems

In (4-6), the function relation of coefficients $a_i(r), i = 1,\cdots,na$, and $b_j(r), j = 1,\cdots,nb$ in terms of the load parameter $r(k)$ could be nonlinear in general. We start by assuming a Linear Fractional Transformation (LFT) dependence of the system plant on the scheduling variable $r(k)$ or by assuming that $r(k)$ enters (4-6) in a polynomial manner. A straightforward approach for estimating the LPV-ARX coefficients $a_i(r), i = 1,\cdots,na$ and $b_j(r), j = 1,\cdots,nb$ can be conducted as follows: 1) identify a set of linear time-invariant ARX models (as (1)) corresponding to a sequence of values of workload intensity *r*, 2) then derive $a_i(r), i = 1,\cdots,na$, $b_j(r), j = 1,\cdots,nb$ by interpolating the corresponding coefficients of the linear time-invariant ARX models.

Most of current literature on LPV system identification is based on the assumption that the scheduling parameters enter the system in a LFT manner [4], [9], [11], and [12]. The system identification algorithm used in this paper is based on the Least Mean Square algorithms from [4], where the LPV plant has polynomial dependence on the scheduling parameters.

```
Least Mean Square Algorithm:
1)    Initialize the estimated $\hat{\Theta}_0$,
2)    $\varepsilon_k \leftarrow y(k) - trace(\hat{\Theta}_k^* \Psi_k)$
3)    $\hat{\Theta}_{k+1} \leftarrow \hat{\Theta}_k + \alpha \varepsilon_k \Psi_k$
```

Fig. 2. Least Mean Square algorithm for identification of a polynomial parameter-dependent LPV system.

Assume that functions $a_i(r)$, $i = 1,\cdots,n_a$ and $b_j(r)$, $j = 1,\cdots,n_b$ in (5-6) are polynomials in the load intensity *r* of order *N-1*, i.e.,

$$\begin{aligned} a_i(r) &= a_i^1 + a_i^2 r + \cdots + a_i^N r^{N-1} \\ b_j(r) &= b_j^1 + b_j^2 r + \cdots + b_j^N r^{N-1} \end{aligned} \qquad (7)$$

Define an $n \times N$ matrix Θ containing all the coefficients to be identified and define the extended regression operator Ψ containing the input/output data and the parameter trajectories,

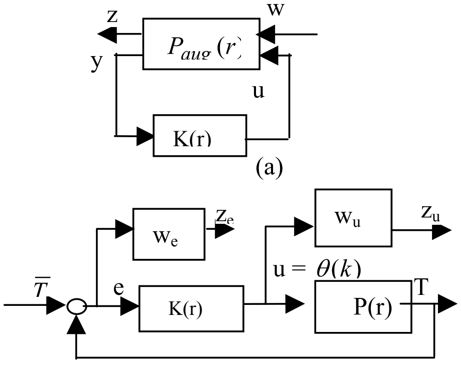
$$\Theta = \begin{bmatrix} a_1^1 & \cdots & a_1^N \\ \vdots & \vdots & \vdots \\ a_{na}^1 & \cdots & a_{na}^N \\ b_0^1 & \cdots & b_0^N \\ \vdots & \vdots & \vdots \\ b_{nb}^1 & \cdots & b_{nb}^N \end{bmatrix} \qquad (8)$$

$$\Psi_k = \begin{bmatrix} -T(k-1) \\ \vdots \\ -T(k-n_a) \\ \theta(k) \\ \vdots \\ \theta(k-n_b) \end{bmatrix} \begin{bmatrix} 1 & r(k) & \cdots & r^{N-1}(k) \end{bmatrix} \qquad (9)$$

The Least Mean Square algorithm in Fig. 2 is used to compute the estimate $\hat{\Theta}$ iteratively. This algorithm does not require the scheduling variable to be slow varying, but requires the persistency of excitation for the inputs and scheduling parameters.

D. LPV Control Design

Based on the LPV-ARX model (4-6), we dynamically control the admission control rejection ratio $\theta(k)$ by design an LPV robust control so that the response time $T(k)$ will meet the target value $\overline{T}$. The LPV control can be classified as a generalized gain-scheduling control. As illustrated by Fig. 3(a), it designs a parameter-dependent controller $K(r)$ (possibly depends on the rate of change $\dot{r}$ as well) to stabilize the closed-loop system for all admissible parameter trajectories $r(k)$, minimizing the effect of the exogenous input w on the controlled variable z in certain norm (e.g., H_∞ norm for an LPV-H_∞ control formulation). The augmented plant $P_{aug}(r)$ includes the actual LPV plant $P(r)$ to be controlled as well as auxiliary weighting functions that are specified for closed-loop performance criteria.



(a)

(b)

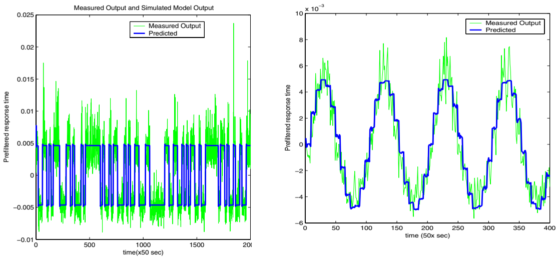
Fig. 3. Block diagram for LPV control structure.

For an affine parameter-dependent plant $P_{aug}(r)$, the LPV control that seeks an affine parameter-dependent controller $K(r)$ (scheduled by the measurements of $r(k)$) is often reduced to solving a set of parameter-dependent Linear Matrix Inequalities (LMIs) [2], [3]. For LPV system with polynomial parameter dependence, a Sum-of-Squares based approach was presented by [13] for control synthesis.

III. Simulation Results and Evaluation

A. Workload Description and Model Validation

We first identify the LPV-ARX model (4) and study how well the model (4) would fit the empirical data. The system identification is based on a set of synthetic workloads running on computer simulation. The inter-arrival time of incoming requests follows an exponential distribution with mean arrival rate $\lambda^i(k)$ in the *kth* sampling period. The request service time of the synthetic workload also follows an exponential distribution with mean service rate $\mu(k)$ in the *kth* sampling interval. Without loss of generality, we fix the mean service rate $\mu(k)$ to be a constant 100 requests/sec, and manipulate workload intensity $r(k)$ by varying the incoming arrival rate $\lambda^i(k)$. Note that though exponential distributions are used here, the underlying approach does not preclude using any other distributions. After requests are admitted to the system, they are served in a first-come first-serve (FCFS) manner. We assume that there is no caching effect for the current system.



(a)

(b)

Fig. 4. Construct an ARX model at workload intensity r = 0.5 using system identification; the input/output data shown in the figure are prefiltered/detrended data. (a) Predicted vs. measured data with pseudo-random binary rejection ratio. (b) Validation against a different set of data obtained using a multiple-step input as rejection ratio.

Fig. 4(a) plots the predicted response time by ARX model (10) versus the measured response time for the pseudo-random binary rejection ratio that is used in the system identification. Fig. 4(b) shows the validation of model (10) against data with a multiple-step rejection ratio (which is not used as t(a)ıing data in system identification).

From the results in Fig. 4, we can see that the linear approximation has captured the major system dynamics at a nominal load condition.

Next we apply the LPV system identification algorithm in Fig. 2 to estimate the coefficients $a_i(r)$, $i = 1, \cdots, n_a$ and $b_j(r)$, $j = 1, \cdots, n_b$ for the LPV model (4-6). A pseudo-random binary signal is used to generate rejection ratio $\theta(k)$ and a random signal is used to generate the scheduling parameter, workload intensity $r(k)$, for the LPV system identification. They are plotted in Fig. 5(a) and 5(b). By running the Least Mean Square algorithm in Fig. 2 on the empirical data $(\theta(k), r(k))$ and the resulting $T(k)$, we derive the following LPV-ARX (1,1) model with affine dependence on workload intensity,

$$\begin{aligned} T(k+2) &= [0.3464 + 0.1313 r(k+1)] * T(k+1) \\ &+ [0.2527 + 0.1187 r(k)] * T(k) \\ &+ [-0.0007 + 0.0443 r(k+1)] \theta(k+1) + e(k+2) \end{aligned} \tag{11}$$

To validate the identified model (11), we use a different set of rejection ratio input and workload intensity parameter trajectories as shown in Fig. 5(c) and Fig. 5(d). Fig. 6 plots the LPV predicted response time versus the measured response time, which demonstrates the validity of the identified LPV model.

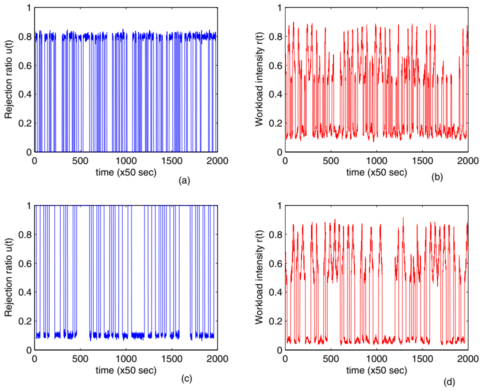


Fig. 5. Input and scheduling parameter trajectories used in LPV system identification and model validation. (a) A pseudo-random binary signal used to generate rejection ratio in system identification; (b) A random signal used to generate workload intensity r(k) as scheduling parameter in system identification; (c) A pseudo-random binary signal used for rejection ratio in model validation; (d) A random signal used for workload intensity r(t) in model validation.

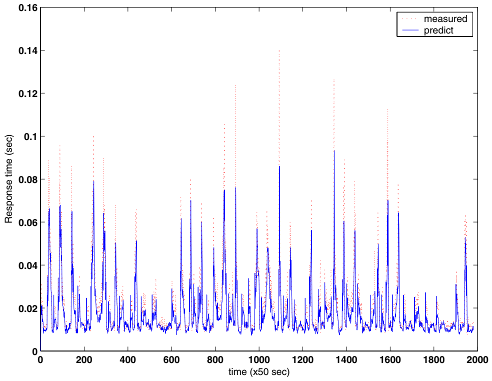


Fig. 6. Model validation on the identified LPV system model. (a) Predicted vs. measured output for the input and scheduling parameter trajectories in Fig. 5 (c)&(d).

B. Control Design Results

The admission control is performed through dynamically deciding the rejection ratio $\theta(k)$ for the requests that are admitted to achieve target response time $\overline{T}$. The admission control has to balance between achieving the target response time and maintaining certain system throughput. Rejecting all requests would definitely put response time to zero, but the service provider would not make any money by serving requests either.

For the workloads used in this paper, we specify the target response time $\overline{T}$ as *0.02 sec*. We first design a Linear-Quadratic (LQ) regulator based on the ARX model (10) that is identified at the nominal workload with intensity *r = 0.5*. In order to minimize the steady-state error in meeting target response time, an integrator is appended at the control input. Fig. 7 shows the performance of this LQ design operates at the design point (workload intensity *r = 0.5*) and at the off-design load condition (*r = 0.8*). It is noted that the LQ design is able to achieve the 0.02 sec target response time at *r = 0.5*, but when the traffic load increases to *r = 0.8*, it does not meet the target response time.

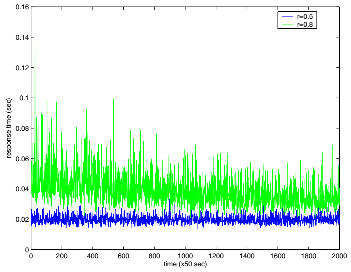
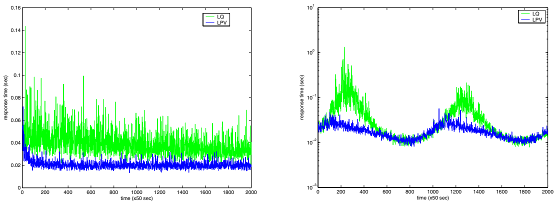


Fig. 7. Simulation results for a LQ controller (designed based on the nominal ARX model (4)) to operate under the nominal load r = 0.5 and the off-design load r = 0.8.

For the LPV-H_∞ formulation in Fig. 3 and LPV plant (11), in order to apply the LPV-H_∞ control algorithms from [2]-[3], a low-pass filter is appended to the input channel of the LPV plant (11). The bandwidth of the low-pass filter should be much higher than the feedback sampling frequency so that the system performance would not be affected. The weighting functions W_e and W_u in the LPV-H_∞ formulation in Fig. 3(b) are chosen to reduce tracking error and peak control action. A suitable set of weighting functions in s-domain is chosen as follows:

$$W_e = \frac{0.1429s + 0.4}{s + 0.02}\ ,\ W_u = \frac{0.1s^2 + s + 2.5}{0.2s^2 + 44.72s + 2500}$$

The LPV controller is then designed to satisfy $\| T_{zw} \| \le \gamma$ with $\gamma < 1$.



(a) (b)

Fig. 8. Simulation results for an LQ controller versus an LPV controller. (a) Operate under the load r = 0.8; (b) operate for a workload with dynamically changing load conditions.

In Fig. 8, the performance for the LPV controller to operate at workload intensity *r = 0.8* is compared with that of the LQ controller running under the same load condition. The LPV controller is able to achieve the *0.02-sec* target response time at the off-nominal load condition. Fig. 10 compares the performance of the LPV controller against that of the LQ design for a time-varying load conditions. It is noted that the LQ design only provides response time guarantee for the nominal load or less intensive traffic; the response time increases dramatically for the heavy traffic. In comparison, the LPV design adapts to the change of workload intensity very well; it provides the response time guarantee despite of the dynamically changing load conditions.

IV. Conclusion

This paper presents an identification based LPV design framework to the modeling and control for the performance management of web server systems. In particular, the admission control is studied, where the rejection ratio for incoming requests is dynamically determined so that the response time for admitted requests can meet the target value with maximal system throughput. It is the first effort to apply (nonlinear) LPV system identification and control design to explicitly model the dependence of system performance on dynamically varying load conditions. Workload parameters that characterize request arrival rate and service demand are used as scheduling parameters in the LPV modeling and control, which allows system's fast adaptation to traffic changes.

Though the design example on web server admission control in this paper ignores certain system complexity and is evaluated based on simulation of synthetic workloads, it well serves the purpose on illustrating how the LPV system identification and control design is applied to performance management. The preliminary results show the strength and effectiveness of this nonlinear modeling and control design.

Acknowledgment

The authors would like to thank Dr. A. Sivasubramiam and Dr. N. Gautam for helpful discussions. We also would like to thank Mr. Amitayu Das for setting up part of the web server simulation.

References

[1] T. F. Abdelzaher, J. A. Stankovic, C. Lu, R. Zhang, and Y. Lu, "Feedback performance control in software services," *IEEE Control Systems Magazine*, Vol. 23, No. 3, 2003, pp. 74-90.

[2] P. Apkarian and P. Gahinet, "A convex characterization of gain-scheduled H_∞ controllers," *IEEE Transactions on Automatic Control*, Vol. 40, 1995, pp. 853-864.

[3] P. Apkarian, P. Gahinet, and G. Becker, "Self-scheduled H-infinity Control of Linear Parameter-varying Systems: a Design Example," *Automatica*, Vol. 31, No. 9, 1995, pp. 1251-1261.

[4] B. Bamieh and L. Giarre, "Identification of linear parameter varying models," *Proceedings of IEEE Conference on Decision and Control*, 1999.

[5] Y. Diao, N. Gandhi, J. L. Hellerstein, S. Parekh, and D. M. Tilbury, "Using MIMO feedback control to enforce policies for interrelated metrics with applications to the Apache web servers," *Proceedings of Network Operations and Management*, 2002.

[6] Y. Diao, J. L. Hellerstein, and S. Parakh, "Optimizing quality of service using fuzzy control," *Proceedings of the 13th IFIP/IEEE International Workshop on Distributed Systems Operations and Management*, 2002.

[7] J. L. Hellerstein, Y. Diao, and S. Parekh, "A first-principles approach to constructing transfer functions for admission control in computing systems," *Proceedings of IEEE Conference on Decision and Control*, LasVegas, 2002.

[8] C. V. Hollot, V. Mishra, D. Towsley, and W. Gong, "A control theoretic analysis of RED," *Proceedings of INFOCOM*, 2001, pp. 1510-1519.

[9] L. H. Lee and K. Poola, "Identification of linear parameter varying systems via LFTs," *Proceedings of the IEEE Conference on Decision and Control*, 1996.

[10] Y. Lu, T. F. Abdelzaher, C. Lu, and G. Tao, "An adaptive control framework for QoS guarantees and its application to differentiated caching services," *Proceedings of Tenth International Workshop on Quality of Service*, 2002.

[11] C. Mazzaro, B. Movsichoff, and R. Sanchez Pena, "Robust identification of linear parameter varying systems," *Proceedings of American Control Conference*, 1999.

[12] M. Sznaier, C. Mazzaro, and T. Inanc, "An LMI approach to control oriented identification of LPV systems," *Proceedings of American Control Conference*, 2000.

[13] F. Wu and S. Prajna, "A new solution approach to polynomial LPV system analysis and synthesis," *Proceedings of American Control Conference,* 2004.