Dynamic Trade-off Analysis of QoS and Energy Saving in Admission Control for Web Service Systems

C. Poussot Vassal, M. Tanelli, M. Lovera


MAHMOUD AWAD

CS 895 - Autonomic Computing - Fall 2010
Dr. Daniel A. Menasce
George Mason University
Presentation Outline

- Objective

- Introduction
  - Control Theory Techniques Used in Autonomic Computing Systems
    - Dynamic Voltage Scaling (DVS) Mechanism
    - Admission Control
  - Linear Parametrically Varying (LPV) framework
    - Affine parameter dependence (LPV-A)
    - Input-affine parameter dependence (LPV-IA)
  - Model Predictive Control (MPC) framework

- Approach used for QoS/Energy Trade off

- Experimental results
Objective

- Find an optimal self-management policy to manage the trade-off between QoS and energy saving objectives in Web service systems.

- (Autonomic/QoS/Save Energy/Dynamic user behavior/variable workload)

- Propose a **general approach** that can accommodate different QoS measures together with other energy-related constraints within the same framework.
• Some utility-based approaches used in autonomic computing rely on the assumption that the system is in **steady state**.

• In Web-based systems, the variability of the user behavior and the difference in system workload are important when trying to find an optimal trade-off management policy.

• The Web service system is assumed to use **Dynamic Voltage Scaling (DVS) Mechanism** for its CPUs and **Admission Control** for controlling incoming requests.

• The dynamic user behavior (and therefore the system workload conditions) is modeled using **the Linear Parameter-Varying (LPV) framework**.

• The LPV model is used in the **Model Predictive Control (MPC) framework** to solve the optimization problem.
Control Theory Techniques:
Dynamic Voltage Scaling (DVS) Mechanism

- A power management technique in computer architecture.
- Voltage used in a component (E.g., CPU) is increased or decreased, depending upon circumstances; overvolting, undervolting.
- Undervolting is done in order to conserve power.
- Overvolting is done in order to increase computer performance.
Control Theory Techniques: Admission Control

- A mechanism by which requests (e.g., web service requests) are rejected in order to maintain QoS for the running application.

- More accepted requests leads to better QoS, more rejected requests lead to worse QoS.

- Trade-off is needed between DVS and Admission Control in order to conserve energy while maintaining an acceptable level of QoS.

- Paper focuses on DVS and Admission Control. Not concerned with resource allocation in virtualized environments.
A "Linear Parameter Varying" (LPV) system is defined as a linear system whose coefficients depend on an exogenous time-varying parameter, e.g.,

\[ x_{k+1} = A(p_k)x_k + B(p_k)u_k \]
\[ y_k = C(p_k)x_k + D(p_k)u_k \]

The exogenous parameter \( p_k \) is unknown a priori, however, it can be measured or estimated upon operation of the system. Some bounds on \( p_k \), such as its magnitude and rate of change, may still be known a priori.

The use of the word "Parameter" was meant to distinguish LPV systems from time-varying systems for which the time variations are known beforehand (as in periodic systems). [time-varying parameter vs. the time being the parameter?]
Introduction

Linear Parametrically Varying (LPV) framework

- Both LPV-A and LPV-IA models are concerned with the way in which $p_k$ enters the system.

Affine parameter dependence (LPV-A) Model

- $A(p_k) = A_0 + A_1 p_{1,k} + \ldots + A_s p_{s,k}$ (Same for B, C and D) where $p_{i,k}$, $i = 1, \ldots, s$ denote the i-th component of vector $p_k$.

Input-affine parameter dependence (LPV-IA) Model

- Only the B and D matrices are considered as parametrically varying, while A and C are assumed to be constant, i.e., $A = A_0$, $C = C_0$. 
In order to perform linear system identification in LPV systems, errors between the real output and the predicted output of the model are minimized using a cost function (to be minimized).

\[ V_N(\theta) := \sum_{k=1}^{N} \| y_k - \hat{y}_k(\theta) \|_2^2 = E_N^T(\theta)E_N(\theta), \]

with respect to \( \theta \), where

\[ E_N^T(\theta) = \begin{bmatrix} (y_1 - \hat{y}_1(\theta))^T & \cdots & (y_N - \hat{y}_N(\theta))^T \end{bmatrix} \]
Model Predictive Control (MPC) framework

• Uses a process model to determine future outputs for a determined horizon N (called the prediction horizon).

• How MPC works:
  1. The control problem is formulated as an optimization one based on:
     a) A mathematical model for the system.
     b) A cost function expressing the desired performance of the system over a future time horizon.
     c) Constraints on the input, state and output variables.
  2. The control action over the future horizon is computed by repeatedly solving the optimization problem online.
  3. The implementation of the computed control action is based on the so-called receding horizon principle, i.e., at each time step only the first sample of the computed control sequence is actually applied and the control problem is re-solved at the subsequent time step.
Model Predictive Control (MPC) framework

Model Predictive Control (MPC) framework

• Proof that the Web service application used (with DVS capabilities, admission control technique, processes requests using FIFO and runs on a single CPU) satisfies the formula:

\[ s_{f,k} = s_k / f_k \]  [observed results with and without DVS ]

\( s_{f,k} \): Effective service time.

\( s_k \): The average request service time, i.e., the overall CPU time needed to process a request for the considered application in the k-th time interval when the CPU runs at the maximum frequency and the CPU is fully dedicated to the execution of the application.

\( f_k \): The ratio between the frequency adopted by the server CPU in the time interval k with respect to the maximum CPU Frequency.
Figure 1: Experimental verification of $s_{f,k} = s_k/f_k$ on a Web server with DVS functionalities.
• Admission control affects request throughput:

• Request throughput $X_k$ is related to the request arrival rate $\lambda_k$ and the probability of admission $P_k$:

$$X_k = P_k \lambda_k$$

• In steady-state models (e.g., queuing model), a system is considered stable if $X_k < \mu_k$, where $\mu_k$ is the maximum service rate.

• Also, in steady-state models:

  (Average) $T = \frac{(\text{Average}) \, N}{(\text{Average}) \, X}$

  $T$: Response time,
  $N$: Variable number of requests,
  $X$: Request throughput
• When taking into account the dynamic behavior of the users and the dynamic load of the server:

• The response time $T_k$ (i.e., the system output) is given by

$$T_k = s_{f,k} + E_k \quad (E_k = T_k - s_{f,k})$$

i.e., queuing time is = response time (overall time a request stays in the system) minus actual effective service time.

• LPV models will be used to determine $E_k$, and the final response time will then be retrieved using the equation above.
Approach Used For QoS/Energy Trade-off

• Perform linear system identification to prove that the system used for experimentation is an LPV system (errors between the real output and the predicted output of the model are minimized).

• Web service application (Java servlet hosted in Apache Tomcat 6.0 application server).
  • Instrumented to simulate FIFO request processing.
  • Instrumented to consume only a portion of CPU (to simulate DVS).
  • Instrumented to accurately determine the service time of each request.

• Workload generator is based on a custom extension of the Apache JMeter 2.3.1 workload injector (generates traffic according to a Poisson distribution to simulate Internet traffic).
Approach Used For QoS/Energy Trade-off

Follows Normal Distribution
Figure 3: Time history of the server utilization $\rho_{ac,k} = \lambda_k s_k \mathcal{P}_k$ used in the identification experiments for admission control.
Approach Used For QoS/Energy Trade-off

Figure 4: Detail of the measured (solid line) and simulated (dashed line) response time obtained with an LPV-IA model for the admission control dynamics on validation data.

From slide 9: In order to perform linear system identification in LPV systems, errors between the real output and the predicted output of the model are minimized using a cost function (to be minimized).
• Next, perform the optimization using Model Predictive Control (MPC) where LPV is used as the mathematical model in the MPV framework. (LPV-MPC approach)

• The idea is to minimize an appropriate performance index representing either the QoS or the Energy saving objective while guaranteeing that the constraints on the system dynamics, control variables and system performance are fulfilled.
Cost function to be minimized, representing the trade-off between the QoS and Energy saving objectives.

**Approach Used For QoS/Energy Trade-off**

![Diagram](image)

**Figure 5: Optimal performance computation scheme.**

Cost function $J^*_N(\alpha, k) = \min J_N(\alpha, k)$

subject to

$$\begin{bmatrix} x_{k+1} \\ \xi_k \\ T_k \end{bmatrix} = \Lambda$$

where $k \in [k, k+N-1]$.

LPV system Model.

Constraints inside are used to calculate the two equations on slide 7.

Constraints to guarantee that control signals evolve within physical bounds.

$$J_N(\alpha, k) = \alpha J_{QoS}(\bar{k}) + (1 - \alpha) J_{ES}(\bar{k})$$

where

$$J_{QoS}(\bar{k}) = \sum_{k=k}^{\bar{k}+N-1} \frac{|s_f - s_{f,k}|}{3_f - 3_f}$$

and

$$J_{ES}(\bar{k}) = \sum_{k=k}^{\bar{k}+N-1} \frac{|s_f - s_{f,k}|}{3_f - 3_f}$$
Experimental Results

• Simulations are carried out to analyze the influence of the parameter $\alpha$ in the cost function (7), i.e., the trade-off between QoS and Energy saving.

• Simulation constraints:
  • The prediction horizon length has been set to $N = 20$
  • The reference response time has been set to $T_{\text{ref}} = 1$ second and the maximum tracking error: $\Delta = \pm 5\% T_{\text{ref}}$
  • The admission probability limits are chosen so that the server system is at least constrained to accept $50\%$ of the incoming requests. (in reality, this depends on the QoS agreement)
  • Service time limits identified.
Experimental Results

Reasonable trade-off between QoS and ES is achieved when Alpha = 0.25

Alpha = 0 favors energy savings

Alpha = 1 favors QoS

Figure 6: Plot of the performance measures $P_{QoS}$ and $P_{ES}$ as functions of $\alpha$ for $N = 20$. 
Experimental Results

To study the performance analysis of the LPV-MPC

Simulation 1: N = 20 and & = 0; Energy saving objective

Simulation 2: N = 20 and & = 0.25; Trade-off between Energy saving and QoS

Same response time is achieved in both cases

Large # of requests (between time 9-16), probability of admission is lowered to achieve high effective response time (around 0.75)

Admission probability is always 1 for uncontrolled server (accept all requests)