AN ADMISSION CONTROL SCHEME FOR PROPORTIONAL DIFFERENTIATED SERVICES ENABLED INTERNET SERVERS USING SUPPORT VECTOR REGRESSION

CHENN-JUNG HUANG, YI-TA CHUANG and CHIH-LUN CHENG

Institute of Learning Technology, National Hualien Teachers College, Hualien, Taiwan 970

Abstract: Unpredictable response time is a common problem in contemporary web servers. Long response delays substantially cut company revenues owing to the large number of aborted e-commerce transactions. This work presents an admission control model and a traffic scheduler scheme of the web server under a proportional differentiated service, embedding a time series predictor to estimate the traffic load of the client in the next measurement time period. The experimental results indicate that the proposed models can effectively realize proportional delay differentiation service in multiclass Web servers.

Keywords: Proportional differentiated service, time series prediction, fuzzy logic systems, self-similarity, support vector regression

1. INTRODUCTION

With the increasingly popularity of the Internet worldwide, the use of web servers to advertise and sell merchandise in business is significantly increasing. Traditional web servers provide service for client requests using first-come-first-serve (FCFS) service model. FCFS introduces unpredictable response times for the clients when incoming traffic is bursty. Customers may become frustrated by a long response time and end the network connection with the web servers without finishing the transactions with the enterprises, leading to loss of revenue for the businesses and unsatisfactory quality of service (QoS). Although QoS provision in network transmission, such as Integrated Services (IntServ) and Differentiated Services (DiffServ), is currently of interest among researchers, network layer QoS guarantees alone might not be able to offer clients perceptible services when the servers are overloaded by unexpectedly significant rises in the number of connections.

Recent works have considered the issues of prioritized processing in web servers. Eggert and Heidemann [Eggert and Heidemann, 1999] attempted to provide QoS service at the application level by splitting client requests into two classes as in the network layer, and restricting the process pool size and response transmission rate for various priority groups. Bhatti and Friedrich [Bhatti and Friedrich, 1999] developed tiered service levels and overload management model, and implemented admission control mechanism by blocking low-priority tasks when the number of high priority jobs exceeded some predefined threshold. Both admission control models follow conventional fixed bandwidth leased line scheme that satisfies the requirement of bursty workload. Although the service quality of high priority tasks is guaranteed, some precious bandwidth is still unused under the average load. Vasiliou and Lutfiyya [Vasiliou and Lutfiyya, 2001] proposed a QoS architecture that permits the number of requests for different priority groups to be dynamically altered according to the performance of the high priority group during the runtime, the service quality of low priority tasks is still somewhat degraded. Chen et al. [Chen and Mohapatra, 2002] proposed so-called service differentiating Internet servers to manage the QoS service provided by the network layer. The servers also provide significantly better services to high priority tasks using prioritized scheduling and task assignment approaches. However, the starvation effect of the low priority task group is still significant owing to the monopolization of the resource allocated to the high priority tasks. To enhance the performance of the low priority tasks, Lee et al. [Lee et al, 2004] presented an admission control algorithm that permits proportional delay differentiated service (PDDS) [Dovrolis et al, 2002] at application level. Under PDDS, client requests are initially classified to different priority groups as in [Chen and Mohapatra, 2002], and the services achieved by classes are proportional to their ratios set in the service contracts. Clients can then be charged according to their maximum average waiting time QoS requirement. Kanodia and Knightly [Kanodia and Knightly, 2003] developed the latency-targeted multiclass admission control algorithm, which employs measurements of requests and service latencies to manage each class' QoS. Lee et al and Kanodia & Knightly [Lee et al, 2004; Kanodia and Knightly, 2003] might resolve the starvation issue for the low priority tasks as revealed by Chen & Mohapatra [Chen and Mohapatra, 2002]. However, compared with the approach using HTTP tags and HTML links in [Ritter et al, 2000], the algorithm of Lee et al [Lee et al, 2004] requires clients to feed two explicit parameters, maximum arrival rate and maximum average waiting time, into the server to launch the admission control mechanism. However, this scheme is impractical in reality. Furthermore, Lee et al and Kanodia and Knightly [Lee et al, 2004; Kanodia and Knightly, 2003] modeled the aggregate request rate from all clients as a Poisson process, which is inconsistent with the evidence shown in [Arlitt and Jin, 2000] that the traffic from World Wide Web (WWW) transfers has self-similarity. Thus, the validity of the performance evaluation reports given in [Lee et al, 2004; Kanodia and Knightly, 2003] is disputed.

This work presents two admission control models to enable PDDS at application level. Each proposed model predicts the total maximum arrival rate and maximum average waiting time of each priority task group for the next measurement period according to the arrival rate of each class during the current and the last three measurement periods. The admission control models can then apply the predicted values to determine the next client for service from one of the queues maintained for each priority task group. Significantly, the system automatically derives the two above-mentioned parameters, thus resolving the impractical problem of specifying the parameters by the clients as seen in [Lee et al, 2004]. Moreover, Arlitt and Jin [Arlitt and Jin, 2000] reveal that the WWW traffic possesses the characteristic of self-similarity, and self-similar time series are predictable. Meanwhile, the works of [Abraham, 2003; Dhyani et al, 2003; Bonino et al, 2003; Antoniol et al, 2001] also exhibit the effectiveness of predicting the number of web server access. Therefore, this work attempts to utilize the support vector regression (SVR) technique to realize the prediction mechanism, and compares this scheme with another well-known machine learning technique, fuzzy logic system, which is renowned by its mathematical framework to deal with real world imprecision, and permits decision-making by estimated values under incomplete or uncertain information. The SVR is specified to implement the proposed models because of its superior performance in many applications, such as time series prediction [Van Gestel et al, 2001], Internet traffic prediction, call classification for AT&T's natural dialog system, multi-user detection and signal recovery for a code division multiple access (CDMA) system [Chen et al, 2001; Gong and Kuh, 1999; Haffner et al, 2003; Kuh, 2001; Hasegawa et at, 2001]. VLSI chips also provide many solutions that enable the SVR to be hardware-computed. High-speed low cost SVR chips have been introduced recently, making implementation of SVR using hardware feasible [Anguita et al, 1999].

The rest of this paper is organized as follows. Section 2 introduces the proposed admission control models. Section 3 then describes the prediction algorithms adopted in the admission control models, namely the fuzzy logic system and support vector regression techniques. Section 4 introduces the simulation results, which compare the proposed algorithms with FCFS service model and with two representative time series predictors in the literature. Conclusions are finally drawn in Section 5.

2. ADAPTIVE ADMISSION CONTROL MODELS

WWW traffic has been observed to have self-similarity [Arlitt and Jin, 2000]. Liang [Liang, 2002] found that WWW traffic is predictable through self-similar time-series. Therefore, this work incorporate a prediction algorithm into the proposed admission control model to estimate the ratio of the expected average waiting time between different classes, and to investigate whether the service contract of each class is violated. Self-similarity is briefly defined and characterized below.

2.1 Self-Similarity

Let $X=(X_t, t=0,1,2,\cdots)$ be a stationary stochastic process. If the average of the series X is computed over non-overlapping blocks of size m, an m-aggregated stationary time series $X^{(m)}=\left(X_k^{(m)}, k=0,1,2,\cdots\right)$ is obtained as follows:

$$X_k^{(m)}=\frac{\sum_{l=0}^{m-1}X_{km+l}}{m},\ m\in N \quad , \tag{1}$$

when the variances and the autocorrelations of $X^{(m)}$ and X satisfy the following relation:

$$\operatorname{Var}\left(X^{(m)}\right)=\frac{\operatorname{Var}(X)}{m^{2(1-H)}},\ 0.5<H<1, \tag{2}$$

$$r_{X^{(m)}}(i)=r_X(i),\ i\ge 0, \tag{3}$$

where H denotes the Hurst parameter. It is said that X is exactly self-similar. In addition, X is asymptotically self-similar if the following relation is satisfied:

$$\operatorname{Var}\left(X^{(m)}\right)\sim\frac{\operatorname{Var}(X)}{m^{2(1-H)}},\ 0.5<H<1, \tag{4}$$

$$r_{X^{(m)}}(i)\to r_X(i),\ m\to\infty. \tag{5}$$

The autocorrelations given in Eqs. (3) and (5) tell that the degree of variability or burstiness is identical at different time scales for self-similar stochastic process, and the autocorrelation does not drop to zero as $m\to\infty$. This is in contrast to the characteristic of the stochastic processes used in typical data models:

$$r_{X^{(m)}}(i)\to 0,\ m\to\infty. \tag{6}$$

As for the variances given in Eqs. (2) and (4), they decrease more slowly than $1/m$ when $m\to\infty$.

As the study in [Arlitt and Jin, 2000] showed that the self-similar traffic pattern generated by Web browsers fits very well to a Pareto-type distribution, our simulation model will thus assume the packet interracial times for each priority task group to be independent and identically distributed according to the infinite-variance Pareto distribution with shape parameter α and cut-off parameter k:

$$\begin{cases} f(t)=\dfrac{\alpha}{k}\left(\dfrac{k}{t}\right)^{\alpha+1} \\ F(t)=P(T\le t)=1-\left(\dfrac{k}{t}\right)^{\alpha}, \end{cases} \quad \alpha\ge 0,\ t>k \ , \tag{7}$$

$$f(t)=F(t)=0,\ t\le k. \tag{8}$$

2.2 Proportional Delay Differentiated Service

The fundamental principle of proportional delay differentiated service (PDDS) developed by Dovrolis *et al.* [Dovrolis et al, 2002] states that higher-class requests will receive better performance than the lower class requests. Specifically, $N>1$ service classes are assumed, where the priority of each class is set in a decreasing order, and the average waiting time is then set in inverse proportion to the priority for each class:

$$\frac{D_i}{D_j}=\frac{P_j}{P_i},\ 1\le i,j\le N\ , \tag{9}$$

where D_i and P_i denote the average waiting time and priority for class i, respectively. That is, the average waiting time is shorter for the priority task groups that pay higher usage costs.

The proposed admission control model forecasts the average waiting time of each class for next measurement period, and utilizes Eq. (9) to choose the class client with the largest gap in the ratio of the average waiting time to receive the next service from the server. Significantly, a new client requesting for service is placed in the client's class queue if the average waiting time for that class does not exceed a predefined threshold. Each class has its own average waiting time threshold value, and it is adjustable during operation according to the number of the clients who leave the class queue without service owing to long waiting time.

2.3 Parameters Required for Admission Control Model

The client is asked to supply some essential information before being admitted into the service. Each client has two options, its class and the maximum average waiting time that it can endure. The specification for the client's class is simple and consistent with the approach taken in the differentiated service enabled at the network layer, while the provision of the maximum average waiting time directly reflects the customer's requirement. The corresponding admission control models are developed based on the two parameter specifications.

2.3.1 *Using client class as the parameter*

As depicted in Fig. 1, each class client waiting for the service is placed in the corresponding class queue, which is managed using the FCFS service model.

According to conservation law [Bolch et al, 1998], if the average arrival rate is λ_i for a client of class *i* during the next measurement period, then the average waiting time for class *i*, given by D_i, should be:

$$\sum_{i=1}^{N} \lambda_i \cdot D_i = \sum_{i=1}^{N} \lambda_i \cdot \overline{D} \ , \tag{10}$$

where $\overline{D}$ represents the average waiting time for the aggregate traffic serviced by a work-conserving FCFS server.

The average waiting time for class *i* during the next measurement period can be derived from Eqs. (9) and (10) as follows:

$$D_i = \frac{\sum_{j=1}^{N} \lambda_j \cdot \overline{D}}{P_i \cdot \sum_{j=1}^{N} \frac{\lambda_j}{P_j}}, \ 1 \le i \le N \ . \tag{11}$$

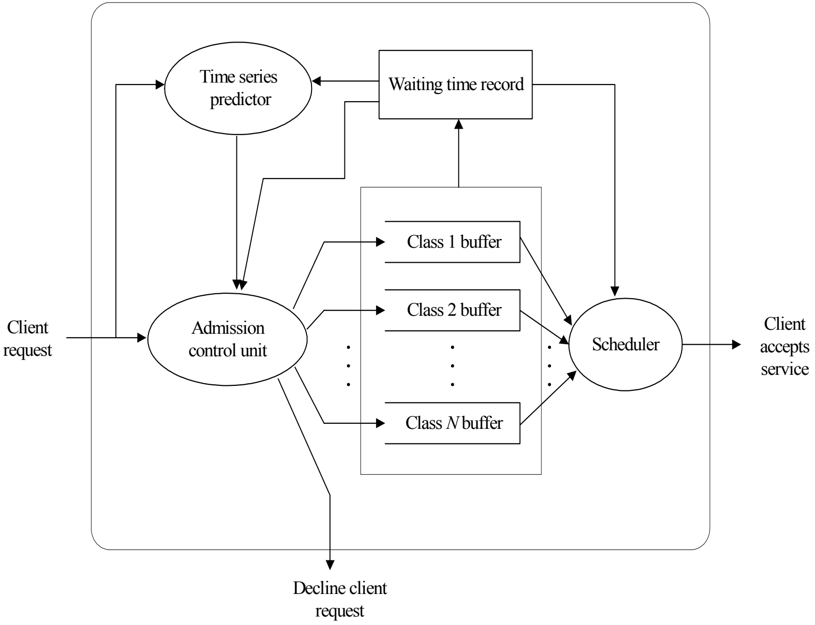

Figure 1: Admission control model with a parameter of each client's class specification.

Since the average arrival rate is difficult to obtain for each class during the next measurement period in Eq. (11), a time series predictor is incorporated into the proposed admission control model to foresee the average arrival rate and therefore resolve the issue of the unreasonable request for the arrival rate specified by each client as presented by Lee et al [Lee et al, 2004].

Class 1 clients are assumed to have the lowest priority, and the maximum waiting time for each class 1 client is assumed as D_1^{MAX}. According to Eq. (9), the maximum waiting time acceptable to a class *i* client is given by $\frac{D_1^{MAX} \cdot P_1}{P_i}$. Correspondingly, incoming class *i* client is permitted to use the server if the following relationship is fulfilled:

$$\frac{\sum_{j=1}^{N} \hat{\lambda}_j \cdot \overline{D}}{P_i \cdot \sum_{j=1}^{N} \frac{\hat{\lambda}_j}{P_j}} \le \frac{D_1^{MAX} \cdot P_1}{P_i} \ , \tag{12}$$

where $\hat{\lambda}_j$ represents the average arrival rate for class *i* aggregate traffic foreseen by the time series predictor.

When the server is ready to service the next client, the scheduler as shown on the right of Fig. 1 applies the following equation to decide which class client should be chosen for service:

$$k = \arg \max_{1 \le i \le N} \frac{W_i \cdot P_1}{W_1 \cdot P_i}, \quad (13)$$

where W_i denotes the waiting time for the client at the front of the class i queue, and P_i represents the priority of class i.

As the clients of some classes can tolerate longer waiting time, such as best-effort traffic, the proposed algorithm can pop up an interactive dialog box to ask clients if they wish to wait longer when the server is overloaded. The usage cost and the priority for the clients is reduced if they are willing to wait longer. This approach can lower the number of customers in the higher-class queues under a bursty workload, and thus maintain the stringent QoS requirement for higher class clients.

The algorithm for the admission control model is now summarized with each client's class as the parameter as follows.

1. When a class i client arrives, utilize the time series predictor to predict the average arrival rate of class i aggregate traffic during next measurement period.
2. Apply Eq. (11) to compute the average waiting time of class i during the next measurement window.
3. Employ Eq. (12) to determine whether the incoming client is admitted to use the server.
4. If admitted, place the client at the end of the class i queue.

If Eq. (12) is not satisfied, but the client is willing to wait longer, then search for the first lower class that satisfies the requirement of Eq. (12).

If found, place the client into the corresponding class queue.

2.3.2 *Using maximum waiting time as the parameter*

The proposed admission control model also permits the client to specify the maximum waiting time as revealed in Fig. 2. Notably, the proposed model requires a classifier to derive the incoming client's class, as displayed in Fig. 2.

Let D_1^{MAX} represent the maximum waiting time for class 1 clients with the lowest priority, and the maximum waiting time requested by the incoming client is set to ω. Equation (9) indicates that the longest waiting time that class i clients can bear is given by $\frac{D_1^{MAX} \cdot P_1}{P_i}$; the classifier can then apply the following equation to determine the incoming client's class:

$$l = \arg \min_{1 \le i \le N} \max\left(\frac{\omega \cdot P_i}{D_1^{MAX} \cdot P_1} - 1, 0 \right). \quad (14)$$

Notably, the above equation is applied to locate the highest priority task group with a maximum waiting time requirement longer than that is acceptable to the client.

The algorithm for the admission control model as shown in Fig. 2 can be summarized as follows:

1. Employ the classifier to classify the incoming client according to Eq. (14).
2. Apply the time series predictor to predict the average arrival rate of the incoming client's class, i, during the following measurement period.
3. Use Eq. (11) to calculate the average waiting time of class i during the following measurement window.
4. Employ Eq. (12) to determine whether the incoming client is admitted to use the server.
5. If admitted, place the client at the end of the class i queue.

If Eq. (12) is not satisfied, but the client is willing to wait longer, then search for the first lower class that satisfies the requirement of Eq. (12).

If found, then place the client into the corresponding class queue.

3. TIME SERIES PREDICTOR

The fuzzy logic technique has been used to solve several connection admission control in ATM and wireless networks and time series prediction problems efficiently in the literature [Liang, 2002]. We thus try to apply fuzzy logic controller concept to predict maximum arrival rate and maximum waiting time as shown in the scheme presented in the previous section.

3.1 Fuzzy Logic Predictor

As in [Ren and Ramamurthy, 2000], we use the average arrival rate for each class during the current and the last four measurement periods $\lambda(t-3)$, $\lambda(t\text{-}2)$, $\lambda(t\text{-}1)$, and $\lambda(t)$ to predict the average arrival rate during next measurement period $\hat{\lambda}(t+1)$. Fig. 3 shows the corresponding fuzzy logic time series predictor. The basic functions of the components employed in the predictor are described as follows.

- **Fuzzifier**: The fuzzifier performs the fuzzification function that converts crisp input data into suitable linguistic values that are needed in the inference engine.
- **Fuzzy rule base**: The fuzzy rule base is composed of a set of linguistic control rules and the attendant control goals.
- **Inference Engine**: The inference engine simulates human decision-making based on the fuzzy control rules and the related input linguistic parameters. The max-min inference method is used to associate the outputs of the inferential rules [Buckley and Eslami, 2002], as described later in this subsection.
- **Defuzzifier**: The defuzzifier acquires the aggregated linguistic values from the inferred fuzzy control action and generates a non-fuzzy control output, the foreseen average arrival rate of each class during next measurement period. The Mamdani defuzzification method is employed in this paper to compute the centroid of membership function for the aggregated output, where the area under the graph of membership function for the aggregated output is divided into two equal subareas [Buckley and Eslami, 2002].

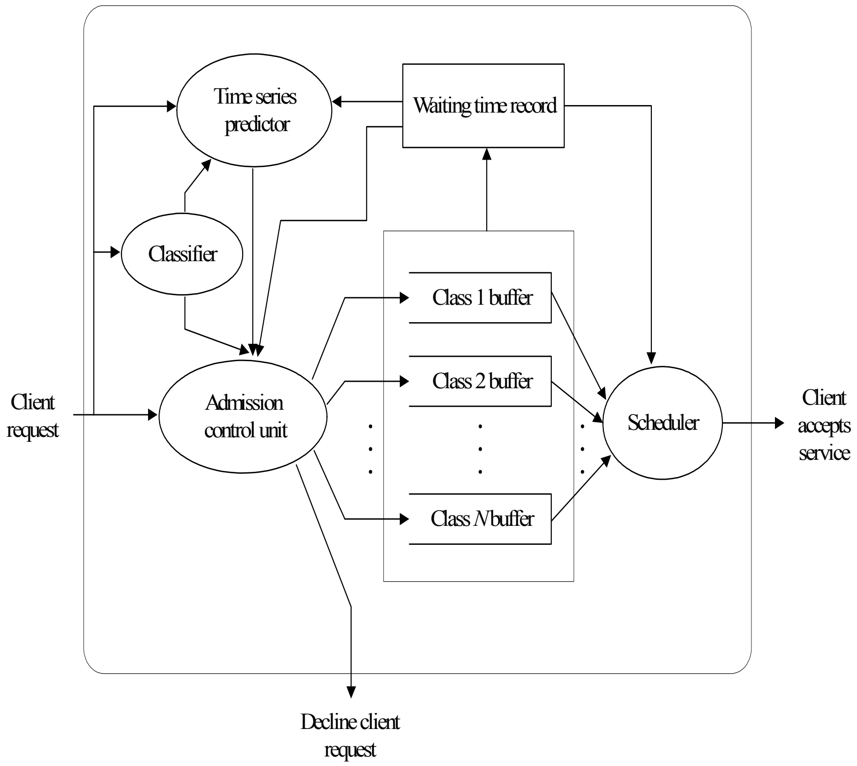

Figure 2: Admission control model with the parameter of the maximum waiting time.

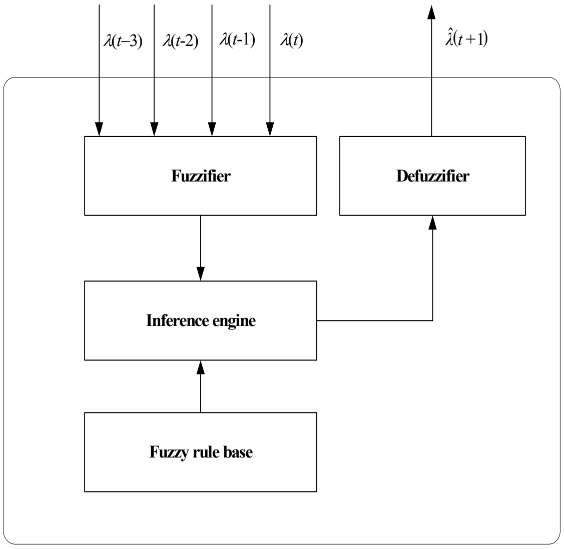

Figure 3: The fuzzy logic based time series predictor.

Fig. 4 shows the mapping of four inputs of the fuzzifier and the output parameter of the inference engine into some appropriate linguistic or membership values, which are expressed by the values within the range of 0 and 1. Three membership functions for each of four inputs and the output are given in Fig. 4, where the linguistic variables “low”, “medium” and “high” give the measure of the average arrival rate for each class. Note that the following Gaussian membership function is chosen for the antecedents and the consequent:

$$\mu_i(\lambda(t-i)) = \exp\left(-\frac{1}{2}\left(\frac{\lambda(t-i)-m_i}{\sigma_i}\right)^2\right), \quad i = 0,1,2,3 \text{ ,(15)}$$

where m_i denotes the mean, σ_i represents the variance.

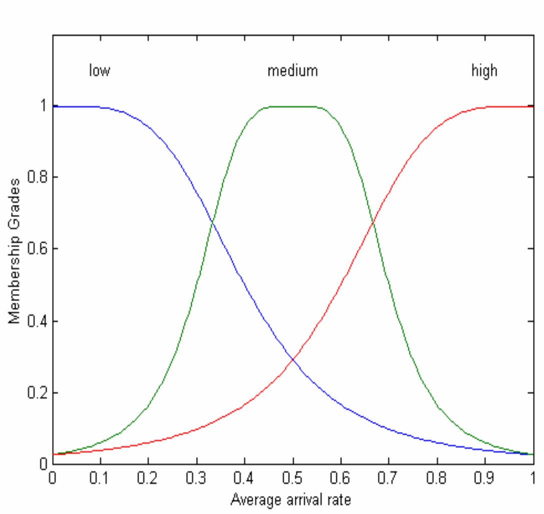

Figure 4: Membership function for the antecedents and the consequent.

The input and output fuzzy sets are correlated to establish the inferential rules of the fuzzy logic time series predictor. Note that three fuzzy sets are used for each antecedent, so the number of fuzzy rules is 3^4=81. By way of illustration, each fuzzy rule can be interpreted as:

Fuzzy rule R_j: IF $\lambda(t-3)$ is A_j and
$\lambda(t\text{-}2)$ is B_j and $\lambda(t-1)$ is C_j and
$\lambda(t)$ is D_j, **THEN** $\hat{\lambda}_j(t+1)$ is E_j. (16)

The inference engine then jumps to the following conclusion for fuzzy rule R_j:

$$E_j^{'} = \left(A_j^{'} \times B_j^{'} \times C_j^{'} \times D_j^{'}\right) \circ \left(A_j \times B_j \times C_j \times D_j \rightarrow E_j\right), (17)$$

where $A_j^{'}$, $B_j^{'}$, $C_j^{'}$ and $D_j^{'}$ stand for the membership grades of four inputs obtained from fuzzy rule R_j, respectively, and the expression inside the second parenthesis denote the simplified representation for Eq. (16).

Fig. 5 illustrates the reasoning procedure for a two-rule Mamdani fuzzy inference system. Note that the composition of minimum and maximum operations, which corresponds the ○ operator in Eq. (17), is employed in the evaluation of the fuzzy rules. The non-fuzzy output of the defuzzifier can then be expressed by the following algebraic expression:

$$\hat{\lambda}(t+1) = \frac{\int \left(\mu_A\left(\hat{\lambda}_A\right) \cdot \hat{\lambda}_A\right) d\hat{\lambda}_A}{\int \mu_A\left(\hat{\lambda}_A\right) d\hat{\lambda}_A} , \qquad (18)$$

where $\mu_A\left(\hat{\lambda}_A\right)$ denotes the membership function of the aggregated output $\hat{\lambda}_A$.

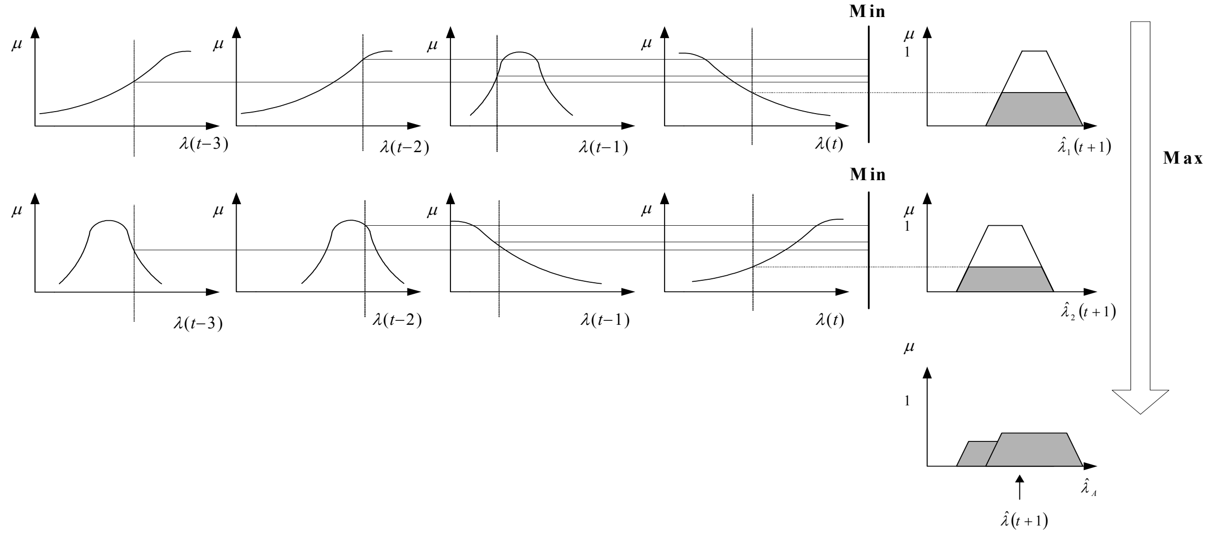

Figure 5: The reasoning procedure for Mamdani defuzzification method

3.2 Support Vector Regression Approach

Support vector regression (SVR) has recently gained popularity owing to its many attractive features and eminent empirical performance [Vapnik, 1995]. The major difference between the SVR and traditional regression techniques is that the SVR employs the structural risk minimization (SRM) approach, rather than the empirical risk minimization (ERM) approach typically adopted in statistical learning. The SRM attempts to minimize an upper threshold on the generalization rather than minimize the training error, and is expected to perform better than the traditional ERM approach. Furthermore, the SVR is a convex optimization, which guarantees that the local minimization is the unique minimization.

To solve a nonlinear regression or functional approximation problem, the SVR nonlinearly maps the input space into a high-dimensional feature space using an appropriate kernel representation, such as polynomials and radial basis functions with Gaussian kernels. This approach is utilized to build a linear regression hyperplane in the feature space, which is nonlinear in the original input space. The parameters can then be derived by solving a quadratic programming problem with linear equality and inequality constraints [Vapnik, 1995].

A training data set $D = \{(\mathbf{x}_i, y_i) \in \Re^n \times \Re, i = 1,...,l\}$ comprising l pair training data $(\mathbf{x}_i, y_i), i = 1,...l$, is given. The input $\mathbf{x}_i$ terms are n-dimensional vectors, and the system response y_i terms are continuous values. The SVR attempts to approximate the following function using data set D:

$$f(\mathbf{x}, \mathbf{w}) = \sum_{i=1}^{N} w_i \cdot \varphi_i(\mathbf{x}) + b , \qquad (19)$$

where b denotes the bias term, and the w_i terms represent the subjects of learning. Furthermore, a mapping $\mathbf{z} = \mathbf{\Phi}(\mathbf{x})$ is selected in advance to map input vectors $\mathbf{x}$ into a higher-dimensional feature space F, which is spanned by a set of fixed functions $\varphi_i(\mathbf{x})$.

By defining a linear loss function with the following ε-insensitivity zone as illustrated in Fig. 6:

$$|y_i - f(\mathbf{x}_i, \mathbf{w})|_\varepsilon = \begin{cases} 0 & \text{if } |y_i - f(\mathbf{x}_i, \mathbf{w})| \le \varepsilon \\ |y_i - f(\mathbf{x}_i, \mathbf{w})| - \varepsilon & \text{otherwise} \end{cases} ,(20)$$

The w_i terms in Eq. (19) can be estimated by minimizing the risk:

$$R = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{l}\left(\sum_{i=1}^{l} |y_i - f(\mathbf{x}_i, \mathbf{w})|_\varepsilon\right), \qquad (21)$$

where C denotes a user-chosen penalty parameter that determines the trade-off between the training error and VC dimension of the SVR model. Significantly, the VC dimension is a scalar value that measures the capacity of a set of functions [Vapnik, 1995].

Equation (21) can be further derived as the following constrained optimization problem:

$$R\left(\mathbf{w},\xi,\xi^*\right)=\frac{1}{2}\|\mathbf{w}\|^2+\frac{C}{l}\left(\sum_{i=1}^{l}\xi_i+\sum_{i=1}^{l}\xi_i^*\right), \qquad (22)$$

subject to constraints:

$$\begin{cases} y_i-\mathbf{w}^T\mathbf{x}_i-b\le\varepsilon+\xi_i^* \\ \mathbf{w}^T\mathbf{x}_i+b-y_i\le\varepsilon+\xi_i \\ \xi_i,\xi_i^*\ge 0 \end{cases}, \qquad (23)$$

where ξ_i and ξ_i^* denote the respective measurements above and below the zone with the radius ε in Vapnik's loss function as given in Eq. (20).

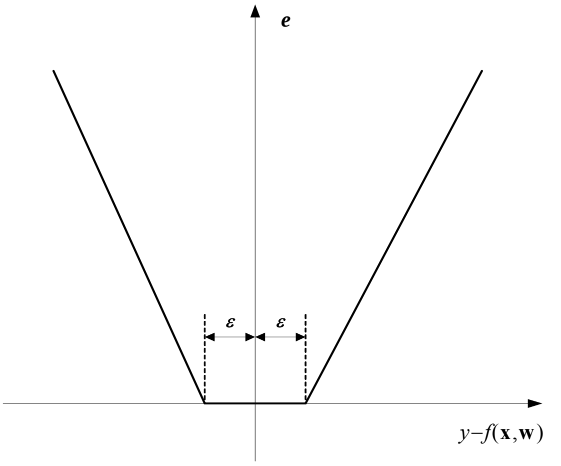

Figure 6: ε-insensitivity loss function.

Schölkopf *et al.* developed a modification of original Vapnik's SVR algorithm, called ν-SVR, and claimed that it can automatically minimize the radius ε [Schölkopf et al, 2000]. Lagrange multiplier methods can be employed to demonstrate that the constrained optimization problem in Eqs.(22) and (23) maximizes the solution of the following equation:

$$W\left(\boldsymbol{\alpha},\boldsymbol{\alpha}^*\right)=\sum_{i=1}^{l}\left(\alpha_i^*-\alpha_i\right)y_i-\frac{1}{2}\sum_{i,j=1}^{l}\left(\alpha_i^*-\alpha_i\right)\left(\alpha_i^*-\alpha_i\right)k\left(\mathbf{x}_i\cdot\mathbf{x}_j\right), \qquad (24)$$

under constraints:

$$\begin{cases} \sum_{i=1}^{l}\left(\alpha_i^*-\alpha_i\right)=0 \\ 0\le\alpha_i,\alpha_i^*\le\frac{C}{l},\ i=1,\ldots,l, \\ \sum_{i=1}^{l}\left(\alpha_i^*+\alpha_i\right)\le C\cdot\nu \end{cases} \qquad (25)$$

where (α_i,α_i^*) denotes one of l Lagrange multiplier pairs; C represents a regularization constant specified *a priori*; ν is a constant greater than or equal to zero, and $k(\mathbf{x}_i.\ \mathbf{x}_j)$ denotes normally a Gaussian kernel or polynomial kernel.

The best nonlinear regression hyperfunction is then represented as:

$$f(\mathbf{x})=\sum_{i=1}^{l}\left(\alpha_i^*-\alpha_i\right)\cdot k\left(\mathbf{x}_i,\mathbf{x}\right)+b, \qquad (26)$$

where b denotes the optimal bias.

4. PERFORMANCE EVALUATION

A series of simulations was performed to measure the performance and behavioral specifics of the proposed admission control models. Only the admission control model utilizing the maximum waiting time as the parameter is implemented in this study, since it is very similar to the algorithm that utilizes each client's class specification as the parameter except in the presentation of the parameters. The performance metrics of most interest include the throughput of the admitted clients from the premium class, the percentage of infringement of QoS requirement for admitted clients from each priority task group and the achieved waiting time ratio for different classes of requests.

4.1 Simulation Scenario

An event-driven simulator is developed to examine the admission control models proposed in this work. Although previous works report that the WWW traffic pattern is self-similar characteristic, the precise generation of representative self-similar WWW traffic for performance evaluation remains an open problem. Various offered loads on the Web server were therefore simulated, maintaining fixed targeted waiting time ratios by utilizing the real trace based on the Web server logs for the 1998 World Cup Soccer web site. The trace includes requests for 5200 unique files, and the average request size is 7KB. The access logs provide the request timestamp, client ID, object URL, service status and reply size of each request. Table 1 shows the simulation parameters. To simplify the study during the experiments, This work only considered two priority levels, premium and basic clients, because the major concern of the simulations is to examine the effectiveness of the proposed admission control models. The waiting time differentiation of the two class clients is 2. The maximum waiting times of all clients are drawn uniformly between 2 and 11 seconds for the algorithm using the maximum waiting time as the parameter. The service times of all requests are exponentially distributed with a mean equal to 18ms. The priority of each incoming request is assigned randomly, and the number of high priority tasks and low priority tasks is almost the same.

4.2 Simulation Results

A series of simulations was conducted for the proposed admission control models embedded with the two time series predictors, i.e. support vector regression (SVRAC) and fuzzy logic system (FLAC). The simulation results were compared with those of the first-come-first-served service model (FCFS).

Parameter	Value
Priority Level	2
Measurement period	1000 seconds
Disk seeking overhead	0.1 ms.
Disk bandwidth	100 Mbps
Network bandwidth	100 Mbps
Maximum server process number	1000
Maximum queue length	1000
Waiting time differentiations	2

Table 1: Simulation parameters

Figures 7–9 display the throughputs of admitted clients from the premium and basic classes for three admission control models. Table 2 lists the throughput ratios of the three algorithms. Table 2 reveals that the average throughput of the admitted clients from premium class in SVRAC is significantly better than that in FLAC, while FCFS does not distinguish between the two priority task groups as expected. Meanwhile, although all three algorithms attain about the same performance under low workloads, Figs. 7–9 indicate that the machine learning techniques, such as support vector regression and fuzzy logic system, effectively predict self-similar time series for admission control models under high traffic load.

Scheme	Average throughput		Throughput ratio (%)	
	Premium class	Basic class	Premium class	Basic class
SVRAC	520.119	377.995	95.29881	69.36363
FLAC	507.335	393.83	92.95272	72.26941
FCFS	450.102	450.616	82.60571	82.67364

Table 2: Comparison of throughputs for two priority task groups

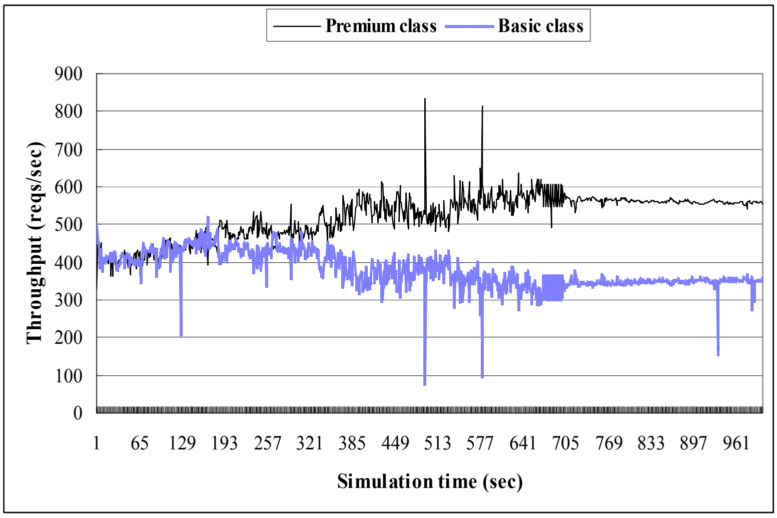

Figure 7. Throughput of the admitted clients from two classes in the SVRAC scheme.

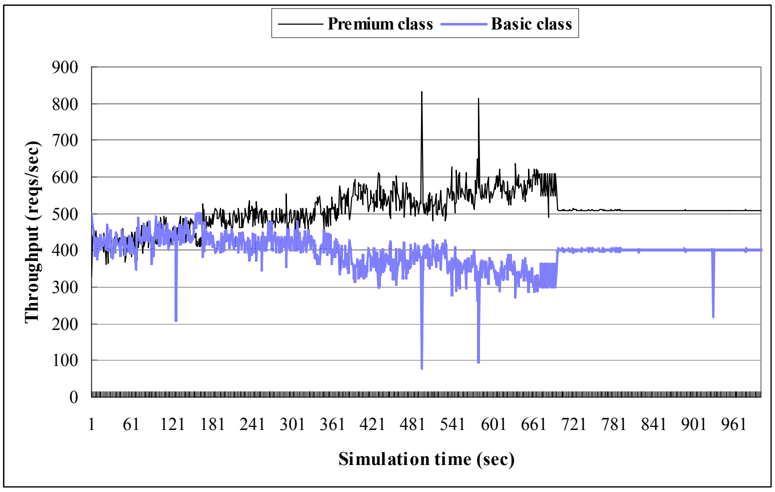

Figure 8. Throughput of the admitted clients from two classes in the FLAC scheme.

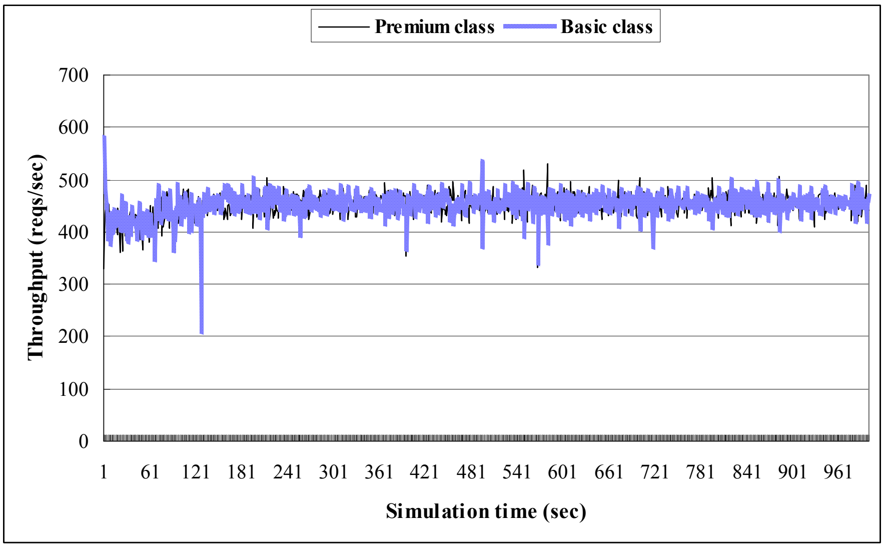

Figure 9. Throughput of the admitted clients from two classes in the FCFS scheme.

Table 3 lists the percentage of the clients that infringe their QoS requirement (maximum waiting time), while Figs. 10–12 show the number of declined clients' requests owing to the violation of maximum waiting time requirement. The analytical results reveal that the SVRAC model provides the best service for the admitted premium class clients. Although the other two algorithms might accept more basic-class clients for service than the SVRAC algorithm, the higher percentage of violations of QoS requirements for the admitted clients of premium class as listed in Table 3 is highly undesirable for the realization of Internet servers providing proportional differentiated services.

Scheme	Reject ratio (%)	
	Premium class	Basic class
SVRAC	4.7012	30.6364
FLAC	7.0473	27.7306
FCFS	17.3943	17.3264

Table 3: Percentage of infringement QoS requirement for two priority task groups

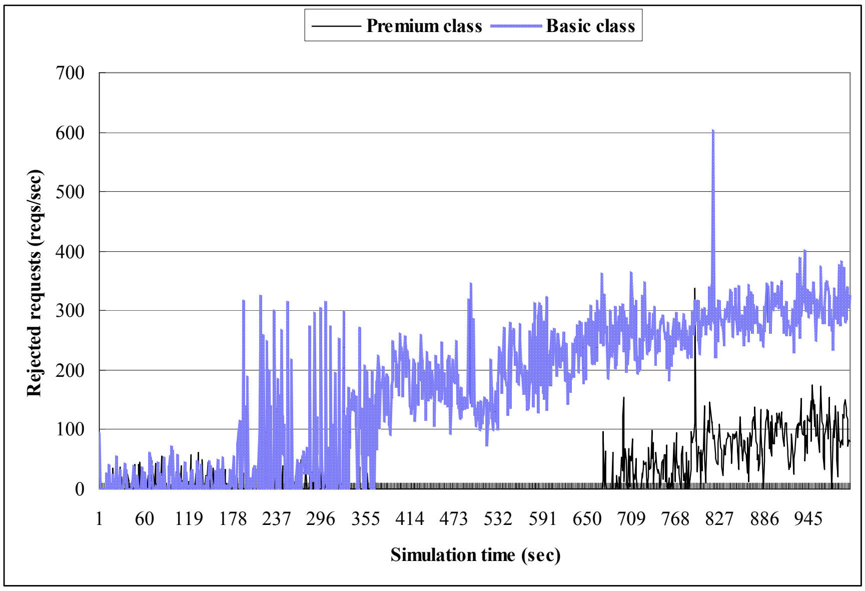

Figure 10. Rejected clients' requests from two classes in the SVRAC scheme.

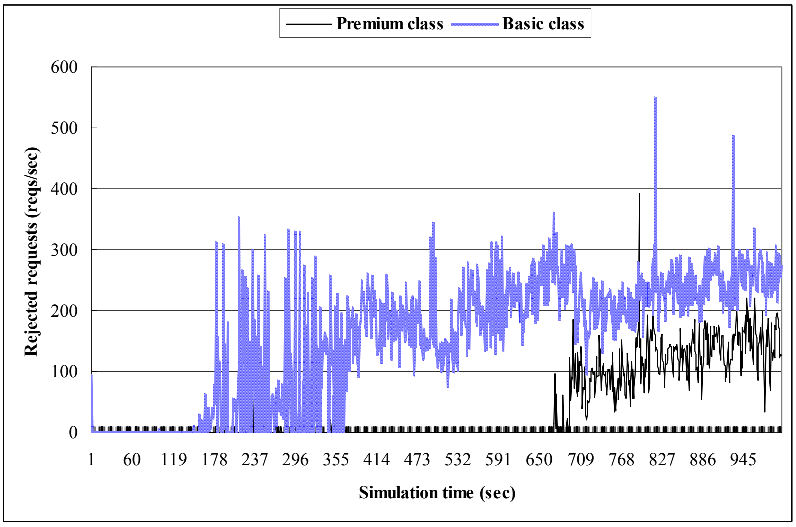

Figure 11. Rejected clients' requests from two classes in the FLAC scheme.

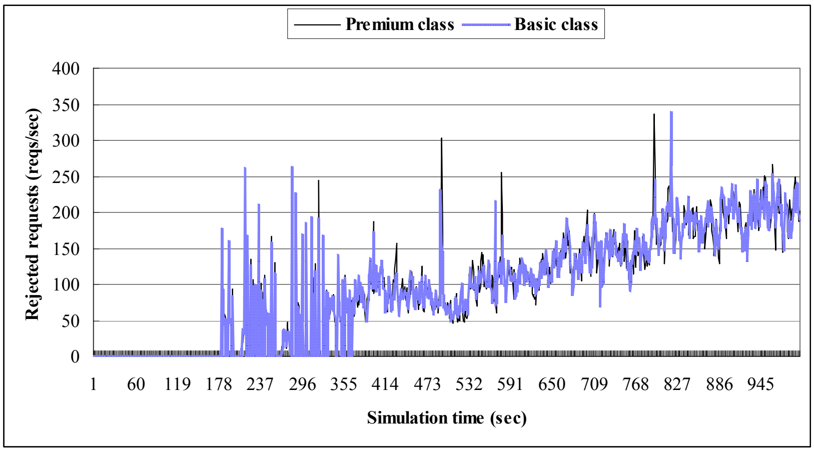

Figure 12. Rejected clients' requests from two classes in the FCFS scheme.

Figures 13–15 show the average waiting times of admitted clients of premium and basic classes for the three admission control models. Table 4 lists the waiting time ratios for the three algorithms. Table 4 reveals that the ratio of SVRAC is slightly closer to the preset ratio of 2, than that of FLAC, while FCFS does not discriminate between the two priority task groups. The throughput of the admitted premium class clients is significantly higher than those of the basic class as revealed in Figs. 7 to 12, but the differences between the support vector regression and fuzzy logic system in the average waiting time ratio for the clients of two classes were found to be insignificant.

Scheme	**Ratio**
SVRAC	**2.315317**
FLAC	**2.344134**
FCFS	**0.999585**

Table 4: The waiting time ratio for the three schemes

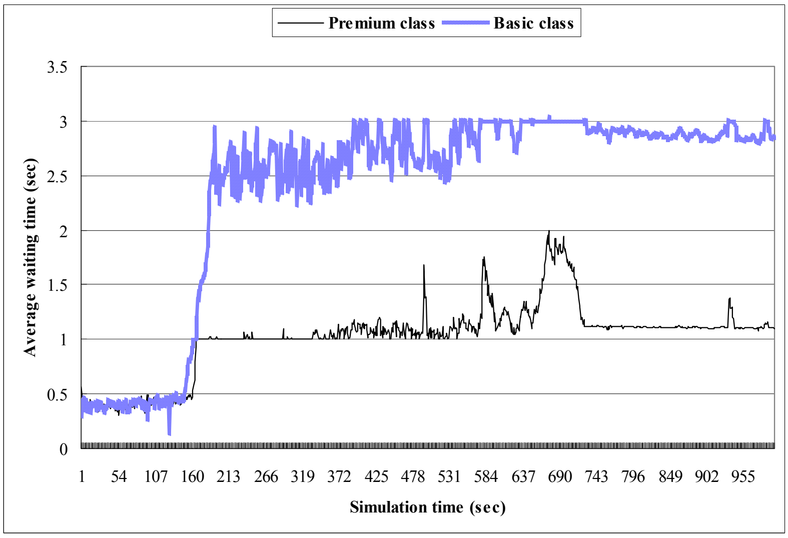

Figure 13. Average waiting time of two classes in the SVRAC scheme.

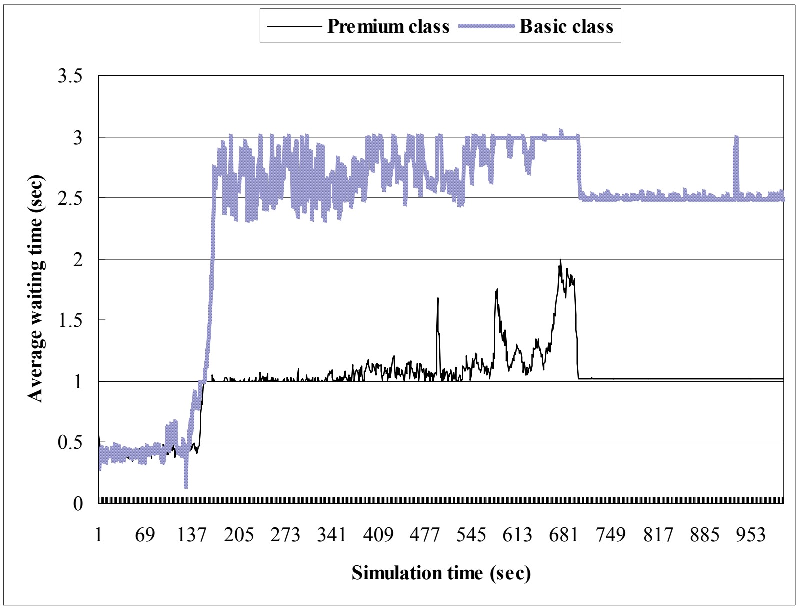

Figure 14. Average waiting time of two classes in the FLAC scheme.

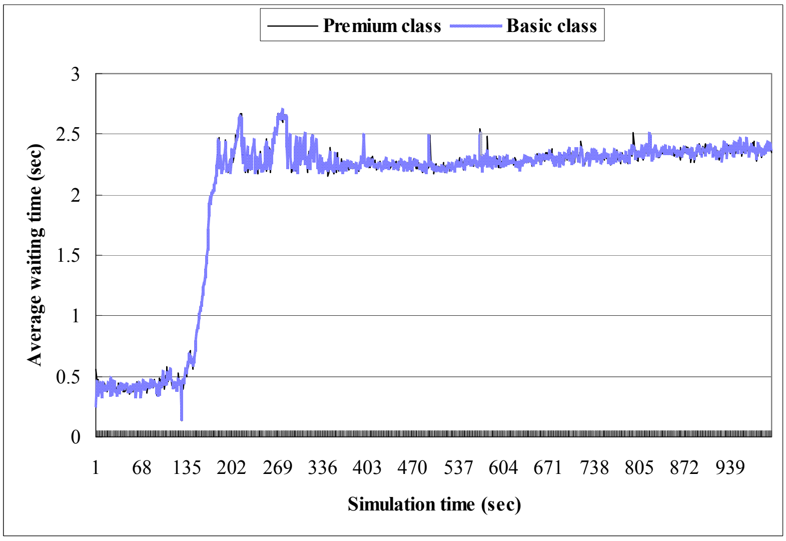

Figure 15. Average waiting time of two classes in the FCFS scheme.

5. CONCLUSION

This work presents two adaptive admission control models to provide a proportional delay differentiated services from an Internet server. Two different time series predictors, namely support vector regression and fuzzy logic, are embedded in the admission control models to estimate the traffic load of the client in the next measurement period. The prediction is needed to determine whether the client can be accepted in the admission control, and the forecast is promising because a self-similar time series is predictable. Simulation results demonstrate that the implementation of time series prediction algorithm with support vector regression (SVR) is significantly better then fuzzy logic system and first-come-first-serve service model when the performance metrics of the throughput and the ratio of rejected clients from premium class are compared. Meanwhile, the average waiting time ratio of the clients from the two classes for both the SVR algorithm and the fuzzy logic system is also maintained within a reasonable range of the predetermined ratio. In the subsequent research, we will not only incorporate other intelligent tools, including neuro-fuzzy and genetic algorithms, into the proposed admission control model, but also explore some ensemble learning method to combine the results of individual machine learning techniques so that the accuracy of prediction for the arrival rate of the aggregate traffic can be further enhanced.

REFERENCES

Abraham A. 2003, “Business intelligence from web usage mining,” *Journal of Information & Knowledge Management*, vol. 2, no. 4, pp. 375-390.

Anguita, D., Boni, A. and Ridella, S. 1999, “Learning algorithm for nonlinear support vector machines suited for digital VLSI,” *Electronics Letters*, vol. 35, no. 16, pp. 1349 – 1350.

Antoniol, G., Casazza, G., Di Lucca, G., Di Penta, M. and Merlo, E. 2001, “Predicting Web site access: an application of time series,” *The 3rd International Workshop on Web Site Evolution*, vol. 1, pp. 57-61.

Arlitt, M. F. and Jin, T. 2000, “A workload characterization study of the 1998 world cup web site,” *IEEE network*, pp. 30-37.

Barford, P. and Crovella, M. E. 1998, “Generating representative Web workloads for network and server performance evaluation,” *ACM SIGMETRICS Performance Evaluation Review*, vol. 26, no. 1, pp. 151-160.

Bhatti, N. and Friedrich, R. 1999, “Web server support for tiered services,” *IEEE Network*, vol. 13, no. 5, pp. 64-71.

Bolch, G, Greiner, S., de Meer, H. and Trivedi, K. S. 1998, Queueing networks and Markov chains: modeling and performance evaluation with computer science applications. New York: John Wiley & Sons, Inc.

Bonino, D., Corno, F. and Squillero, G. 2003, “Dynamic prediction of Web requests” *The 2003 Congress on Evolutionary Computation*, vol. 3, pp. 2034-2041.

Buckley, J. and Eslami, E. 2002, An introduction to fuzzy logic and fuzzy sets (advances in soft computing). Physica Verlag.

Chen, S.; Samingan, A.K. and Hanzo, L. 2001, “Support vector machine multiuser receiver for DS-CDMA signals in multipath channels,” *IEEE Transactions on Neural Networks*, vol. 12, no. 3, pp. 604 – 611.

Chen, X. and Mohapatra, P. 2002, “Performance evaluation of service differentiating Internet servers,” *IEEE Trans. Computers*, vol. 51, no. 11, pp. 1368-1375.

Dhyani, D., Bhowmick, S. and Ng, W.-K. 2003, “Modelling and predicting a Web page accesses using Markov processes,” *14th International Workshop on Database and Expert Systems Applications*, pp. 332 – 336.

Dovrolis, C., Stiliadis, D. and Ramanathan, P. 2002, “Proportional differentiated services: delay differentiation and packet scheduling,” *IEEE/ACM Trans. Networking*, vol. 10, no. 1, pp. 12-26.

Eggert, L. and Heidemann, J. 1999, “Application-level differentiated services for web servers,” *World Wide Web Journal*, vol. 3, no. 3, pp. 133-142.

Gong, X. and Kuh, A. 1999, “Support vector machine for multiuser detection in CDMA communications,” *The Thirty-Third Asilomar Conference on Signals, Systems, and Computers*, vol. 1, pp. 680 – 684.

Haffner, P., Tur, G., and Wright, J. H. 2003, “Optimizing SVMs for complex call classification,” *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. I-632̃ I-635.

Hasegawa, M., Wu, G. and Mizuno, M. 2001, “Applications of nonlinear prediction methods to the Internet traffic,” *The 2001 IEEE International Symposium on Circuits and Systems*, vol. 2, pp. III-169̃III-172.

Kanodia, V. and Knightly, E. W. 2003, “Ensuring latency targets in multiclass Web servers,” *IEEE Trans. Parallel and Distributed Systems*, vol. 14, no. 1, pp. 84-93.

Kuh A. 2001, “Adaptive kernel methods for CDMA systems,” *2001 IEEE International Joint Conference on Neural Networks*, vol. 4, pp. 2404 – 2409.

Lee, S. C., Lui, J. C. and Yau, D. K. 2004, “A proportional-delay DiffServ-enabled Web server: admission control and dynamic adaptation,” *IEEE Trans. Parallel and Distributed Systems*, vol. 15, no. 5, pp. 384-400.

Liang, Q. 2002, “Ad hoc wireless network traffic-self-similarity and forecasting,” *IEEE Communication Letters*, vol. 6, no. 7, pp. 297-299.

Paxson, V. and Floyd, S. 1997, “Why we don't know how to simulate the Internet,” *1997 Winter Simulation Conference*, pp. 1037-1044.

Ren, Q. and Ramamurthy, G 2000, “A real-time dynamic connection admission controller based on traffic modeling, measurement, and fuzzy logic control," *IEEE J. Selected Areas in Comm.*, vol. 18, no. 2, pp. 184-196.

Ritter, H., Pastoors, T. and Wehrle, K. 2000, "DiffServ in the web: different approaches for enabling better services in the World Wide Web," *Joint Conf. of Broadband Communications, High Performance Networking and Performance of Communication Networks*, pp. 555-566.

Schölkopf, B., Smola, A., Williamson, R. and Bartlett, P. L. 2000, "New support vector algorithms," *Neural Computation*, vol. 12, pp. 1207-1245.

Van Gestel, T., Suykens, J.A.K. and Baestaens, D.-E., Lambrechts, A., Lanckriet, G., Vandaele, B., De Moor, B. and Vandewalle, J. 2001, "Financial time series prediction using least squares support vector machines within the evidence framework," *IEEE Transactions on Neural Networks*, vol. 12, no. 4, pp. 809 – 821, 2001.

Vapnik, V. 1995, The nature of statistical learning theory. New York: Springer-Verlag.

Vasiliou, N. and Lutfiyya, H. 2001, "Managing a differentiated quality of service in a World Wide Web server," *IEEE Inter. Symp. Integrated Network Management*, vol. VII, pp. 309-312.

AUTHOR BIOGRAPHIES

Chenn-Jung Huang was born in Hualien, Taiwan, in 1961. He received a B. Sc. degree in Electrical Engineering from National Taiwan University, Taiwan and an M. S. degree in Computer Science from University of Southern California, Los Angeles, in 1984 and 1987. He received a Ph. D degree in Electrical Engineering from National Sun Yat-Sen University, Taiwan, in 2000. He is currently an Associate Professor at the Institute of Learning Technology, National Hualien University of Education, Taiwan. His research interests include computer communication networks, data mining and machine learning applications.

Yi-Ta Chuang and Chih-Lun Cheng are pursuing a Master's degree at the Institute of Learning Technology, National Hualien University of Education, Taiwan. Their research interests include computer communication networks, data mining and applications of machine learning techniques.