

2008

An Intelligent Model To Control Preemption Rate Of Instantaneous Request Calls In Networks With Book-Ahead Reservation

Iftekhar Ahmad
Edith Cowan University

Joarder Kamruzzaman
Monash University

Daryoush Habibi
Edith Cowan University

Farzana Islam
Asian Institute of Technology, Thailand

[10.1109/ATNAC.2008.4783348](https://doi.org/10.1109/ATNAC.2008.4783348)

This article was originally published as: Ahmad, I., Kamruzzaman, J., Habibi, D., & Islam, F. (2008). An Intelligent Model To Control Preemption Rate Of Instantaneous Request Calls In Networks With Book-Ahead Reservation. Proceedings of Australasian Telecommunication Networks and Applications Conference. ATNAC 2008. (pp. 344-348). Adelaide Australia. IEEE Press. Original article available [here](#)

© 2008 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

This Conference Proceeding is posted at Research Online.

<http://ro.ecu.edu.au/ecuworks/1269>

An Intelligent Model to Control Preemption Rate of Instantaneous Request Calls in Networks with Book-Ahead Reservation

Iftekhhar Ahmad^{*}, Joarder Kamruzzaman[#], Daryoush Habibi^{*} and Farzana Islam^{##}

^{*} School of Engineering, Edith Cowan University, Australia

[#] Gippsland School of Information Technology, Monash University, Australia

^{##} Asian Institute of Technology, Thailand

Abstract— Resource sharing between Book-Ahead (BA) and Instantaneous Request (IR) reservation often results in high preemption rate of on-going IR calls. High IR call preemption rate causes interruption to service continuity which is considered as detrimental in a QoS-enabled network. A number of call admission control models have been proposed in literature to reduce the preemption rate of on-going IR calls. Many of these models use a tuning parameter to achieve certain level of preemption rate. This paper presents an Artificial Neural Network (ANN) model to dynamically control the preemption rate of on-going calls in a QoS-enabled network. The model maps network traffic parameters and desired level of preemption rate into appropriate tuning parameter. Once trained, this model can be used to automatically estimate the tuning parameter value necessary to achieve the desired level of preemption rate. Simulation results show that the preemption rate attained by the model closely matches with the target rate.

Keywords: quality of service, preemption, resource reservation.

I. INTRODUCTION

For the past decades, Quality of Service (QoS) provisioning has been a major research issue mainly because of increasing demand of multimedia and distributed applications. Resource reservation is one of the widely practiced techniques that are used to ensure guaranteed QoS of applications. Two types of reservation techniques have been proposed by researchers: i) Book-Ahead (BA) reservation ii) Instantaneous Request (IR) reservation. Multimedia and distributed applications that require long duration, high bandwidth demand and have time sensitive significance are good candidates for Book-Ahead reservation [1-5]. In BA reservation, resource is reserved well in advance from the announced starting time over the declared duration to ensure that the application will not experience scarcity of resource at the point of its activation. Contrarily, an IR call connection requests for immediate reservation and usage of resources. Resource sharing between BA and IR requests imposes a number of challenges. One of them is to keep the preemption rate of on-going IR calls very low as preemption is considered as an interruption to service continuity and is thus perceived as a serious issue from users' perceived QoS point of view. In this paper, a novel application of ANN is shown to maintain service continuity of on-going IR calls in a QoS-enabled network with provision for BA reservation.

Artificial Neural Network (ANN) has been a useful tool to solve a number of complex problems in relation to Quality of Service (QoS) provisioning in communication research. Chou and Wu proposed a neural network based model [6] which adaptively adjusts network parameters like threshold, push-out probability and incremental bandwidth size of virtual path to maintain guaranteed QoS in ATM networks. Kumar *et al.* demonstrated that reduction of assigned resources needed to maintain guaranteed QoS under network overload conditions could be successfully done in real time with the assistance of neural networks [7]. Rovithakis *et al.* [8] proposed a method which uses neural networks as a controller to map QoS parameters at the application level for multimedia services into appropriate values for the media characteristics in order to achieve the required user satisfaction without violating the available bandwidth constraints. Tong *et al.* [9] investigated the application of a multi-layer perceptron (MLP) network and a radial basis function (RBF) network to estimate packet loss rate which is considered as an important QoS parameter in a computer network. Application of Hopfield Neural Network (HNN) for dynamic channel allocation (DCA) in cellular radio network was shown by Lazaro *et al.* [10]. In this work, we use an ANN model for estimation of the value of a tuning parameter for a previously proposed call admission control scheme to achieve a target level of operating IR call preemption rate.

Researchers have proposed different models to keep the IR call preemption rate low in a QoS-enabled network where resources are shared between IR and BA calls. Greenberg *et al.* [1] proposed an approximate interrupt probability based admission control scheme that showed that resource sharing between IR and BA calls achieves better network performance than strict partitioning of resources proposed in [11], [12]. Schelen and Pink [2] proposed a model to reduce the number of preemption by introducing the concept of look-ahead time. Look-ahead time is defined as the pre-allocation time, i.e., the time for starting to set aside resources for advance reservations so that there is no resource scarcity at the starting time of a BA call. Ahmad *et al.* [3] proposed a Dynamic Look-Ahead Time (DLAT) based call admission control model that considered network dynamicity and attained better performance than CLAT based CAC model. Lin *et al.* [4] proposed an application aware look-ahead time based admission control scheme which considered different look-ahead time for

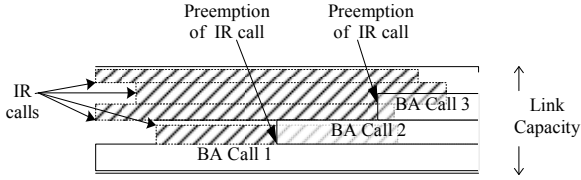


Figure 1: Preemption of IR calls in BA reservation.

different applications. Preemption rate is a key parameter in a QoS-enabled network. High preemption rate causes service degradation on users' side. From network perspective, high preemption rate causes extra over-head for re-routing, heavy signalling messages and congestion of traffic while very low preemption rate causes low utilization of network capacity. Desired level of preemption rate is not often a fixed value for all operating conditions in a particular network. Operating preemption rate is governed by the interests of both users and network provider and can be best addressed as an optimization problem. If most of the on-going applications are highly sensitive to service interruption, preemption rate should be maintained very low and vice versa. As a result, operating preemption rate is not a fixed value and should be adaptive with changing network conditions. CLAT model uses look-ahead time as the tuning parameter to achieve different values of preemption rate while DLAT model uses a tuning parameter 'c' to achieve operating preemption rate.

The model proposed in [4] is unable to adjust itself to achieve different values of preemption rate as it uses no tuning parameter. A model would be extremely useful to map the desired operating preemption rate and the current network traffic parameters into the tuning parameter which when fitted into CAC models attains the desired preemption rate in a real time network. The relationship among the tuning parameter, preemption rate and other network traffic parameters in DLAT model is very complex to determine using non-linear regression approach. Non-linear regression approach is thus restricted to a limited number of network parameters. This makes neural network a suitable candidate for this problem because of its capability of realizing any complex input-output relationship to an arbitrary degree of accuracy. This paper proposes an artificial neural network model to map the desired preemption rate and traffic parameters to the tuning parameter of DLAT based CAC model. Simulation results show that the proposed model successfully achieves the target preemption rate with different network conditions.

II. DYNAMIC LOOK-AHEAD TIME (DLAT) BASED CAC

A BA call request is characterized by two additional parameters, starting time and call holding time along with other QoS parameters. If the request is granted the application is guaranteed to have required resources at the time of its start. In contrast, starting time for an IR call is immediate and call holding time is open ended. Problem occurs when a BA call becomes active at certain point of lifetime of an on-going IR call and there does not exist enough available resources to support the BA call (Fig. 1). On-going IR calls are required to be preempted to meet BA bandwidth demand. IR call

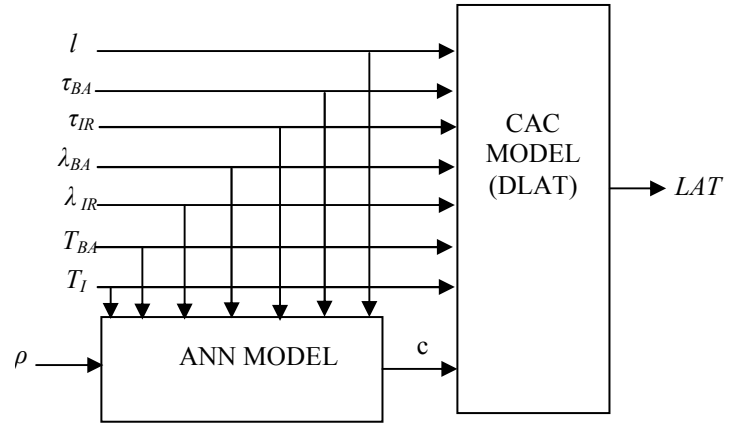


Figure 2: Proposed ANN model

preemption rate which indicates the ratio of preempted IR calls to accepted IR calls should be reasonably small in a QoS-enabled network. It is the responsibility of CAC model to keep the preemption rate low.

Dynamic Look-Ahead Time (DLAT) based CAC model calculates look-ahead time taking the dynamicity of traffic pattern and network state into consideration. It considers the current IR load, future BA load, IR call release rate, variation in load and arrival rate to calculate the look-ahead time. It dynamically updates the value of look-ahead time at regular time interval. Look-ahead time is calculated by the following equation ([3]):

$$LAT(t,s) = \frac{\max(A(s) + R(t) + (1+l)\sigma(\tau_{IR}) - C, 0)}{(1-b)\tau_{IR}\lambda_{IR}} + \sigma\left(\frac{1}{\lambda_{IR}}\right)c \quad (1)$$

Here, $LAT(t,s)$ is the look-ahead time w.r.t traffic condition at current time t and BA activation time s ($t < s$). $A(s)$ is the aggregate bandwidth reserved for BA calls to be activated at time s , $R(t)$ is the aggregate bandwidth used by IR calls at time t , τ_{IR} is the mean bandwidth demand of IR calls, λ_{IR} is the mean arrival rate of IR calls, l is the normalized BA limit which determines maximum allowable aggregate BA load, b is the call blocking probability for IR calls, $\sigma(\cdot)$ is the standard deviation and $c(>1)$ is the tuning parameter. The value of 'c' affects look-ahead time which, in turn, controls preemption rate.

Look-ahead time (LAT) is calculated at regular intervals of operating time and IR calls are checked against the following rule at call admission time.

$$C > \max_{s \in LAT} (r + R + A(s)) \quad (2)$$

Here C is the total capacity of the link, r is the bandwidth demand of the IR call, R is the aggregate bandwidth consumed by on-going IR calls.

Look-ahead time is computed at each interval for a number of entries in the Book-ahead table. Those entries are taken into calculation for which the following rule satisfies:

$$t > s_i - \frac{A(s_i) - A(t)}{(1-b)\tau_{IR}\lambda_{IR}} \quad (3)$$

The right hand side is conservatively computed on the worst case assumption that the network is completely utilized. In summary the algorithm is given as follows:

- Step 1:** At a time t of a time interval, for all entries s_i in Book-ahead table that satisfy Eq. (3) follow step 2.
Step 2: Calculate look-ahead time $LAT(t, s_i)$.
Step 3: Find the $LAT(t, s_i)$ for which $t > s_i - LAT(t, s_i)$ and $t - (s_i - LAT(t, s_i))$ is minimum.
Step 4: If no such $LAT(t, s_i)$ is found then LAT is set to zero. Go to step 6.
Step 5: Set LAT equal to $LAT(t, s_i)$.
Step 6: Quit the algorithm and go to step 1 when the next interval is due.

As mentioned earlier, DLAT model can successfully achieve low preemption rate in QoS-enabled network. However, the desired operating level of preemption rate is subject to change depending on users' demand and network management. For example, at a certain operating period if 80% of the total calls in the network are sensitive to service interruption, it is important for the network enterprise to maintain the preemption rate less than 0.2 to ensure that QoS guarantee is properly maintained. Otherwise it will result in user dissatisfaction due to excessive interruption of guaranteed service. Similarly, if the number of QoS-sensitive applications is small at a certain operating period, the network can operate at a reasonably high preemption rate. To assess and reflect the change in network condition, interval based monitoring is often used in a network which effectively determines the up-to-date traffic conditions and the desired level of preemption rate for the next operating interval. To achieve the desired but changing preemption rate in a real time network by using a model like DLAT model, it is important to devise an intelligent model which will automatically respond to the changing network condition and accordingly set the tuning parameter value so that the desired level of preemption rate is duly maintained. This work uses such an intelligent model using neural network to address this issue in the following section.

III. ANN BASED MODEL OF LOOK-AHEAD TIME

A. The Proposed Model

A network is characterized by network parameters like mean bandwidth demand of BA calls τ_{BA} , mean bandwidth demand of IR calls τ_{IR} , mean arrival rate of BA calls λ_{BA} , mean arrival rate of IR calls λ_{IR} , mean call holding time of

BA calls T_{BA} , mean call holding time of IR calls T_{IR} , BA limit l and the IR call preemption rate ρ . The nature of distribution for bandwidth demand, arrival rate and call holding time can be found from proper periodic traffic analysis. Preemption rate is a complex non-linear function of the network parameters. The aim of this work is to find the value of tuning parameter 'c' which when fitted into Eq. (1) provides the desired operating preemption rate. The variable network parameters, as shown in Fig. 2, are the inputs to the ANN and produces tuning parameter 'c' as the output. Parameters τ_{IR} , λ_{IR} , τ_{BA} , λ_{BA} , T_{IR} , T_{BA} , l and 'c' are the inputs to the DLAT model which again provides look-ahead time LAT as the output. New calls are checked against the LAT according to Eq. (2).

The ANN is trained with a wide range of network parameters for different network conditions. The training dataset is created by simulating the DLAT model as follows: for a particular set of τ_{IR} , τ_{BA} , λ_{IR} , λ_{BA} , T_{IR} , T_{BA} , l and tuning parameter 'c', the simulation is done for 2.5×10^6 sec calculating look ahead time at 10s interval. At the end of each simulation the number of calls preempted is calculated and preemption rate is determined. To cover a large spectrum of network operating points, the parameters were varied over a wide range. For each combination of parameters, the value of 'c' was varied from 1~21 (c = 21 provides acceptably low preemption rate for different traffic parameters in our simulation) yielding different preemption rate. This procedure generates a data set where each set of traffic parameters and preemption rate is associated with a 'c' value, necessary to train an ANN model. Once the ANN model is trained with sufficient amount of training data, the network is expected to estimate the appropriate value of 'c' in response to the network state.

B. Learning Algorithms

Neural networks are a class of nonlinear model that can approximate any nonlinear function to an arbitrary degree of accuracy and have been used to realize complex input-output mapping in different domains. The most commonly used neural network architecture is multilayer feedforward network. It consists of an input layer, an output layer and one or more intermediate layer called hidden layer. All the nodes at each layer are connected to each node at the upper layer by interconnection strength called weights. A training algorithm is used to attain a set of weights that minimizes the difference between the target and actual output produced by the network. For supervised learning, Backpropagation [13] is the most commonly used algorithm to train multi-layer feedforward network. In this study, we experimented with two improved variants of backpropagation algorithm: Scaled Conjugate Gradient (SCG) [14] and Backpropagation with Bayesian Regularization (BR) [15].

IV. SIMULATION RESULTS

Both training and testing data have been collected from the simulation results carried by Ahmad *et al.* [3]. A dataset consisting of 880 data has been used in the ANN model. The

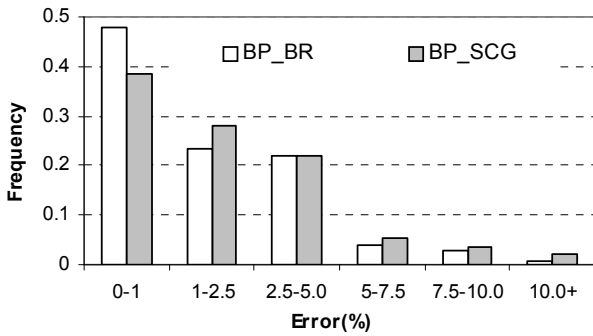


Figure 3: Frequency distribution of percentage error on test data.

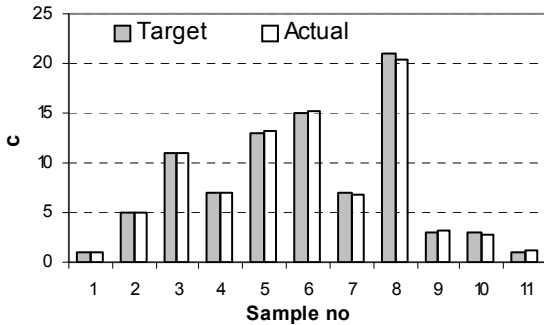


Figure 4: Comparison of 'c' values on test data.

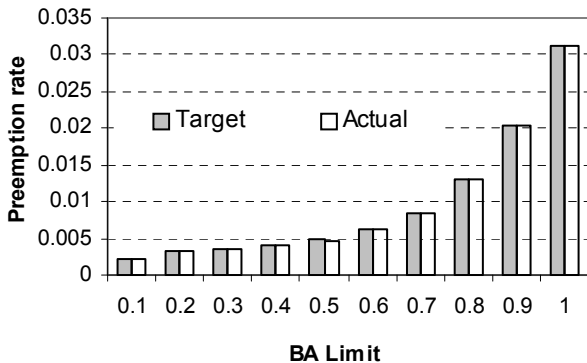


Figure 5: Comparison of preemption rate on test data.

dataset covers wide range of combinations of network and traffic parameters. 85% of the total data set was used for training purpose and the rest 15% was used to test the ability of the model to produce correct value of 'c' (tuning parameter). Scale conjugate gradient (SCG) and Bayesian Regularization (BR) back propagation algorithm has been used for training the data. The network consists of 8 inputs and 1 output (Fig. 2). The experiment was conducted using different combination of hidden layers and hidden units. Three hidden layers consisting of 40, 25, 15 neurons, respectively were found to provide the best match on the test data measured in terms of mean squared error calculated over the target and estimated 'c' values. Since the performance of an ANN network depends on the initial weights and other learning parameters, a number of trails were conducted; each trail was continued for 40,000 epochs unless

Table 1: Accuracy of Estimation in ANN model.

Training Algorithm	Mean Square Error	Maximum Deviation (%)	Mean % Error
SCG	0.0764	12.7	2.51
BR	0.0610	11.2	1.88

terminated at a predefined mean squared error. For simulation of this model we used the standard ANN tools in MATLAB.

Table 1 shows the mean squared error, maximum deviation with respect to target 'c' and mean error on test data for the best trial in SCG and BR algorithms. The table indicates that BR algorithm is more suitable for estimation of 'c' values (3.4% less mean error) compared to SCG algorithm. Results found on the test data for SCG and BR algorithm are also reported in Fig. 3. It shows that in almost 47% of the test data, the 'c' value estimated by BR algorithm lie within 1% error margin with respect to the target value while for SCG algorithm around 38% of the test data lie within 1% error. In 23% of the test data, estimated 'c' value differs from the target value in BR algorithm by an error margin in the range of 1.0 ~ 2.5% while in almost 21% cases, error lies within the range of 2.5~5.0%. Around 5% of the total output was found to deviate by more than 5% error on target value. These data causing high deviation are mainly for higher target value of 'c' ($c > 11.0$) for which preemption rate approaches very close to zero and larger 'c' values result in very small difference in preemption rate. In only 0.7% of test data, the estimated 'c' value deviates 10% or more from its actual value in BR algorithm. From Table 1 and Fig. 3, it can be concluded that the BR algorithm is best suited for the current problem and hence the rest of the simulation results presented in this section are based on BR algorithm. A potential problem in learning is the lack of smoothness of the trained weights which may contribute to a network's poor performance in generalization. Bayesian regularization technique incorporate the magnitude of weight values into the objective function to minimize leading to a network with less variation among the trained weights which results in better performance.

The value estimated by the ANN model with BR algorithm is then fed to DLAT model. Because of the deviation of ANN estimated value from the target value, network performance parameters like preemption rate, utilization and call blocking rate also deviate from the desired values. Further investigation was done to assess this impact. Since the test data is quite large (132), a certain number of test samples were chosen for presentation following the frequency distribution shown in Fig. 3. These samples, target and predicted 'c' values and their deviations are shown in Fig. 4: sample number 1-4 (<1%), 5-6 (1~2.5 %), 7-8(2.5~5%), 9(5~7.5%), 10(7.5~10%) and 11(>10%), the number within the bracket indicates the error range.

Figure 4 shows that actual value of 'c' provided by the ANN model is very close to the target value of 'c'. Impact of actual and target value of 'c' on preemption rate is reported in Fig. 5. In the worst case for sample #11 which is the representative of ANN estimated 'c' value with the maximum percentage (11.2%) error, the difference in preemption rate is found to be 2.29%. For most of the data, the achieved preemption rate matches very closely with the preemption rate corresponding to the target value of 'c'.

V. CONCLUSION

This paper demonstrates the use of neural networks in modelling of look-ahead time in BA reservation for controlling preemption rate of on-going IR calls in a QoS-enabled network. The data set to train the model was created by simulating the previously proposed DLAT based CAC model with different traffic and tuning parameter values that govern the preemption rate. Once trained, the ANN model can estimate the value of tuning parameter under all network operating conditions to achieve the desired preemption rate. Simulation results demonstrate close match between the target and actual value of tuning parameter 'c' and preemption rate. The ANN model when used in conjunction with DLAT model can further improve its potential to maintain QoS by appropriately controlling the IR call preemption rate.

REFERENCES

- [1] A. Greenberg, R. Srikant, and W. Whitt, "Resource sharing for book-ahead and instantaneous-request calls," *IEEE/ACM Trans. Networking*, vol.7, pp.10-22, 1999.
- [2] O. Schelen, and S. Pink, "Sharing resources through advance reservation agents," *Journal of High Speed Networks*, Special issue on Multimedia Networking, vol. 7, no. 3-4, 1998.
- [3] I. Ahmad, J. Kamruzzaman, and S. Ashwathanarayaniah, "A dynamic approach to reduce preemption in Book-Ahead Reservation in QoS-enabled networks," *Computer Communications*, Elsevier Science, 2006.
- [4] Y. Lin, C. Chang, C and Y. Hsu, "Bandwidth brokers of instantaneous and book-ahead requests for differentiated services networks," *ICICE Trans. Commun.*, vol. E85-B, no.1, 2002.
- [5] F. Breiter, S. Kühn, E. Robles, and A. Schill, "The usage of advance reservation mechanisms in distributed multimedia applications," *Computer Networks and ISDN Systems*, vol. 30, no. 16-18, pp. 1627-1635, 1998.
- [6] L. Chou and J. Wu, "Parameter adjustment using neural network based genetic algorithms for guaranteed QoS in ATM networks," *IEICE Transactions on Communications*, vol. E78-B, pp. 572-9, 1995.
- [7] V. Kumar, and P. Venkataram, "An efficient resource allocation scheme for mobile multimedia networks," *Proc. Mobile and Wireless Communication Networks 2002*, pp. 88-92, 2002.
- [8] G. Rovithakis, A. Matamis, and M. Zervakis, "Controlling QoS at the application level of multimedia applications using artificial neural networks: experiment results," *Proc. EUSIPCO 2000*, vol. 2, pp.849-52, 2002.
- [9] H. Tong, and T. Brown, "Estimating loss rates in an integrated service network by neural networks," *Proc. IEEE GLOBECOM 1998*, vol. 1, pp. 19-24, 1998.
- [10] O. Lazaro, and D. Girma, "A Hopfield neural-network-based dynamic channel allocation with handoff channel reservation control," *IEEE Transaction on Vehicular Technology 2000*, vol. 49, pp. 1578-1587, 2000.
- [11] D. Ferrari, A. Gupta, and G. Ventre, "Distributed advance reservation of real-time connections," *Proc. NOSSDAV*, pp. 15-26, 1995.
- [12] K. Ross, *Multiservice Loss Models for Broadband Telecommunication Networks*. U.K.: Springer-Verlag, 1995.
- [13] D.E Rumelhart, J.L. McClelland and the PDP research group, *Parallel Distributed Processing*, vol. 1, MIT Press, 1996.
- [14] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, NY, 1995.
- [15] D. Mackay, "Bayesian interpolation," *Neural Computation*, vol. 4, pp. 415-447, 1992.