NN-WTP: Waiting Time Prediction for Mobile System using Neural Networks

Yingjie Fu[1], Puion Au[2]
[1]Department of Computer Science
City University of Hong Kong, Hong Kong SAR
[2]Department of Computer Engineering and Information Technology
City University of Hong Kong, Hong Kong SAR
Email: fuyingjie@tsinghua.org.cn;albert.au@alumni.cityu.edu.hk

Abstract—Admission control plays an important role for quality-of-service (QoS) in modern mobile telecommunication systems, and the waiting time prediction is expected in case of failure of making a connection due to the bandwidth reason. In this paper, we provided a waiting time prediction method for handling failed admission request in the mobile telecommunication system by neural networks (NN-WTP). Very simple input parameters, such as the day of a week, time, number of active connections that occupy the voice channel, and number of active connections that occupy the data channel are required for the back propagation neural networks to simulate a hyperplane. After a short training period, the waiting time prediction can be performed on this hyperplane. Finally, we did simulation with a whole day network request usage history to demonstrate the feasibility and high performance of this method.

Index—Waiting Time Prediction (WTP), Neural Networks (NN), Admission Control

I Introduction

In the last decades, the mobile telecommunication system has been developed very quickly. It affects our daily life in many aspects. From the beginning of mobile phone came into our daily life, people have many experiences that a call fails to be put through. There are cases that no signal, lack of electrical power, and system error, etc. However in most cases, we fail to put it through because of the air bandwidth limitation. This problem has being existing from the day, when wireless was use for telecommunication for the first time. Till recent years, this problem has been classified into wireless QoS problem [4, 11, 13, 21]. People have tried to solve model the admission control mechanisms from different angles [1, 2, 5, 7, 15, 17, 27, 28]. The method provided in this paper is an enhancement of Admission Control. We focus on the post-process of a request when it is denied by the Admission Controller. In this paper, we discuss a new technique on how to provide the waiting time estimation for failed requests. A caller that sent the request

can try again after the estimated time. The clients, whose requests are denied, do not need to retry aimlessly under this kind of mechanism. It is obvious that people would prefer to be informed about the minimum waiting time rather than continuously try to connect to the network frequently. On the other hand, frequent unsuccessful requests can cause system burden in both edge networks and core networks.

Former research[20, 24] have been done to detect the available bandwidth, through sending out a series of probing requests to get the round-trip-time, and then estimate how much bandwidth is available. The objective is to find out whether the bandwidth is wide enough to establish a connection and how much bandwidth can be provided to use. Further more, intelligence has been added in some of the research [4, 6. 8, 10]. Maybe these admission control methods are proper for the high-speed computer networks, but not for wireless telephone networks because it is power consuming, system resource consuming, and bandwidth consuming. A significant difference between telephone networks and computer networks is that telephone networks provide constant data rate using circuit switching while computer networks provides variable data rate using packet switching. In a wireless telephone system, allocated bandwidths are deferent level based constants, which mean different bandwidths will be allocated for different services. The initiator of a connection request only needs to know whether the connection can be established or not, but does not need to know how much bandwidth is available. NN-WTP provides exactly what we need with no additional consumption in comparison with bandwidth detection.

NN-WTP has been deeply intelligentized by using neural networks. Neural networks has been researched and practically implemented in many fields [29]. It is recognized as a useful method in pattern recognition and function approximation. In this paper, we used multilayer back propagation neural networks (BPNN) [30] to model a much complicated hyperplane to predict the waiting time of failed connection request. The training process is accomplished using very simple input parameters, such as the day of a week, time, voice channel user number (how many users are using the voice channel), and video channel user number. The NN-WTP training or the hyperplane formative process only takes a short time, which is presented in the following sections. Also we could see the performance of NN-WTP in our simulation, which indicates NN-WTP is a quite useful and practical technology.

II. Model and Analysis

A. Mobile Phone System Model

Wireless networks have a lot of configuration properties. For a mobile phone system, we suppose that the total bandwidth is B for a single base station, there are average N mobile phone users in one cell, and the

bandwidth consumption for each pair of user at phoning time is *b*. As we know that the number of paired-users and the phoning consistency time are random. We use the traditional traffic model in our modeling, which means the number of user pairs is Poisson distributed, as shown in Formula(1). And phone consistency time is exponentially distributed. As shown in Formula(2).

$$\begin{aligned} &\mathrm{P}[k \text{ users arrive in time interval } T] \\ &= \frac{(\lambda T)^k}{k!} e^{-\lambda T} \text{ where } \lambda > 0, T > 0 \end{aligned} \quad (1)$$

λ is mean arrival number in one time unit. Considering the fact that huge variation of calling tendency of mobile phone users in a day exists, e.g. people tend to use the phone at noon, and we considered this issue in our simulation.

$$\begin{aligned} &\mathrm{P}\ [\text{a pair of users phoning time less than } t] \\ &= 1 - e^{-t/\beta}, t \geq 0, \beta > 0 \end{aligned} \quad (2)$$

β is mean phone consistency time. The bandwidth constrain can be denoted as Formula(3).

$$\sum_{i=1}^{c} b_i \leq B \quad (3)$$

c is current paired-user number.

This inequality should be always true. In our mobile phone system model, signal interferences, fading or any other variable factors are not taken into account in order to simplify the discussion. Each user in our mobile phone system model will behave as the state diagram shown in Fig.1.

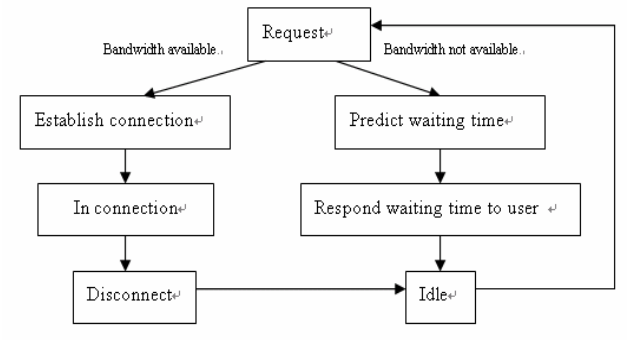


Fig.1.Connection State Diagram

When a user calls someone using a mobile phone, his phone will send signal to the base station in a cell to request for connection.

If the bandwidth is available at that moment, connection will be established and a constant bit rate connection will be kept until either one user sends disconnection request. After that, the user mobile phone will

keep in Idle state until a new connection request is sent.

If the bandwidth is not available at that moment, server side of the telecom's network will predict the waiting time for the sender, and respond to it. Then the caller will keep Idle till the predicted time.

B. Neural Network Model

Neural networks have been researched and built for decades. The most fantastic uses of them are in pattern recognition and function approximation. In this paper, our waiting time prediction approach is based on feed-forward neural network with back propagation.

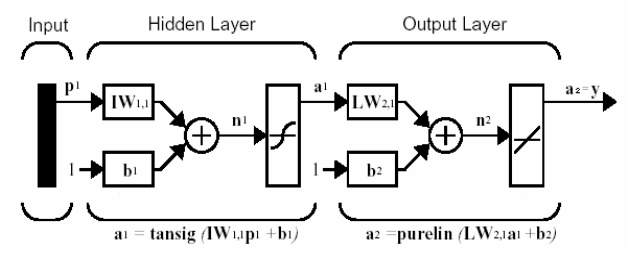


Fig.2. Feed forward neural network with back propagation

We used 2 layers of neurons to accomplish the hyperplane simulation tasks, one is a hidden layer and another one is an output layer, as shown in Fig.2. The inputs parameters are the day of a week, time, voice channel user number, and/or video channel user number, etc. All parameters are optional, but the most critical parameter(s) should be included, such as time because the change of data flow in mobile system is periodical in a dedicated area (mobile cell). The output is the predicted waiting time. The input parameters are denoted as an *R* by 1 vector *p1*. In the hidden layer, tangent sigmoid function has been used as transfer function. Suppose we use *S* neurons in this layer, the input weight matrix from input p1 to layer 1 can be denoted as $IW_{1,1}$, it is an *S* by *R* matrix. The bias vector *b1* is *S* by 1. The output of hidden layer *a1* is an *S* by 1 vector too, it can be presented as

$$a1 = tansig\,(n1)$$

$$= \frac{e^{IW_{1,1}p1+b1} - e^{-IW_{1,1}p1-b1}}{e^{IW_{1,1}p1+b1} + e^{-IW_{1,1}p1-b1}}, \qquad (4)$$

where $n1 = IW_{1,1}\,p1 + b1$.

In the output layer, pure linear function and only one neuron are used. The layer weight matrix from layer 1 to layer 2 could be denoted as $LW_{2,1}$, and it is a 1 by *S* vector. The bias vector b2 is a 1 by 1 scalar. The output of output layer *a2* is a 1 by 1 scalar too. It means the waiting time prediction result. And it is presented as Formula(5).

$$a2 = LW_{2,1} \times a1 + b2$$

$$= LW_{2,1}\ tansig\,(n1) + b2$$

$$= LW_{2,1} \times \frac{e^{IW_{1,1}\ p1+b1} - e^{-IW_{1,1}\ p1-b1}}{e^{IW_{1,1}\ p1+b1} + e^{-IW_{1,1}\ p1-b1}} + b2, \quad (5)$$

where $n1 = IW_{1,1}\ p1 + b1$.

After the training process of this neural network, $IW_{1,1}$ and $LW_{2,1}$ are S by R and S by 1 matrices respectively with determined factor values. The meaning of R is the parameter number in one input vector and S is the neuron number in the hidden layer.

There are many kinds of back propagation algorithms. Based on the large number of experiments, we selected Levenberg-Marquardt (LM) algorithm as our training algorithm, because LM has the fastest training speed on the same precision basis. The LM algorithm is a variation of Newton methods [22], and this algorithm was designed to approach second-order training speed without having to compute the Hessian matrix. As we used mean squared error (MSE) to be the performance function and the output vector a2 is 1 by 1, Hessian matrix can be approximated as $H=J^TJ$, where J is Jacobian matrix that contains first derivatives of the neural network error with respect to the weights and biases. The Jacobian matrix can be computed through a standard back propagation technique [26] that is much less complex than computing the Hessian matrix [22]. The LM algorithm update could be represented as Formula(6).

$$x_{k+1} = x_k - \alpha_k g_k, \quad \alpha_k = [J_k^T J_k + \mu I]^{-1},$$

$$g_k = J_k^T e_k$$

$$\text{so,} \quad x_{k+1} = x_k - [J_k^T J_k + \mu I]^{-1} J_k^T e_k, \quad (6)$$

$$J_k = \begin{bmatrix} \frac{\partial e_{1,1}}{\partial w_{1,1}^1} & \frac{\partial e_{1,1}}{\partial w_{1,2}^1} & \cdots & \frac{\partial e_{1,1}}{\partial w_{S^1,R}^1} & \frac{\partial e_{1,1}}{\partial b_1^1} & \cdots \\ \frac{\partial e_{2,1}}{\partial w_{1,1}^1} & \frac{\partial e_{2,1}}{\partial w_{1,2}^1} & \cdots & \frac{\partial e_{2,1}}{\partial w_{S^1,R}^1} & \frac{\partial e_{2,1}}{\partial b_1^1} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial e_{S^M,1}}{\partial w_{1,1}^1} & \frac{\partial e_{S^M,1}}{\partial w_{1,2}^1} & \cdots & \frac{\partial e_{S^M,1}}{\partial w_{S^1,R}^1} & \frac{\partial e_{S^M,1}}{\partial b_1^1} & \cdots \\ \frac{\partial e_{1,2}}{\partial w_{1,1}^1} & \frac{\partial e_{1,2}}{\partial w_{1,2}^1} & \cdots & \frac{\partial e_{1,2}}{\partial w_{S^1,R}^1} & \frac{\partial e_{1,2}}{\partial b_1^1} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

S^M denotes for max row number in the output layer

$e_{p,q}$ denotes the error of output value indexed by (p, q) in the output matrix

$w_{q,r}^{p}$ denotes the value indexed by (q, r) in the weight matrix of layer p

b_{q}^{p} denotes p^{th} value in the biases vector of layer p

x_k is a vector of current weights and biases

g_k is the current gradient

α_k is the learning rate

J_k is the Jacobian matrix

III. Simulation and Analysis

Based on the mobile system model and the feed forward neural network, we did experiment to show the feasibility and high adaptability of NN-WTP. We used only one input parameter—time, and train the neural network with 9 different training algorithms. We compared these training algorithms' performance on the same waiting time prediction precision basis.

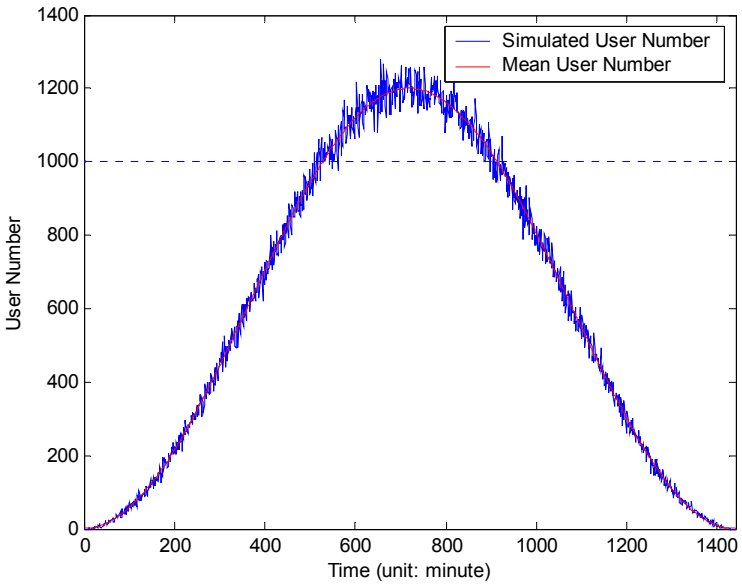


Fig.3. simulated user requests per minute

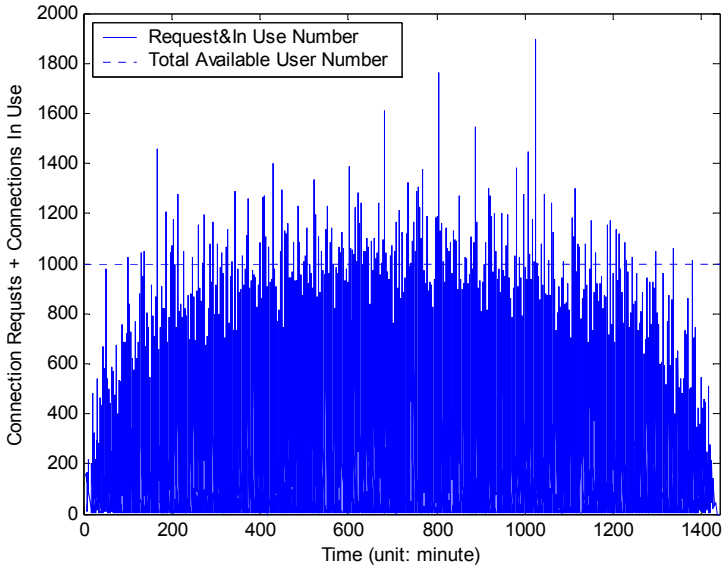


Fig.4. Request and In Use Number in Each Minute

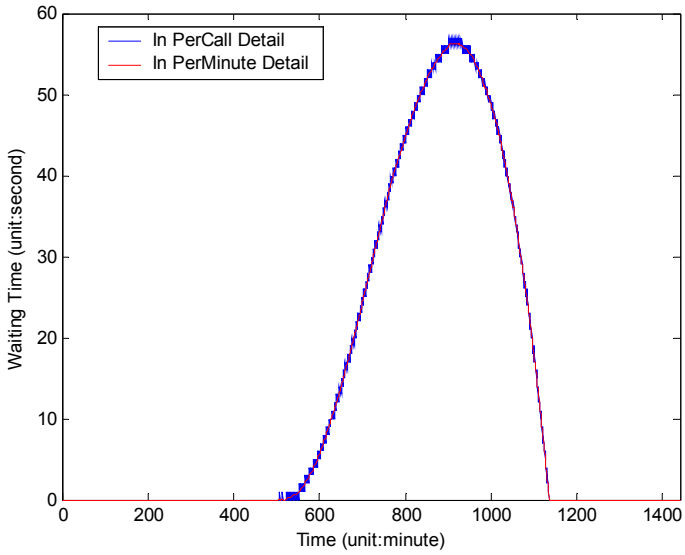


Fig.5. Waiting Time in PerCall Detail and PerMinute Detail

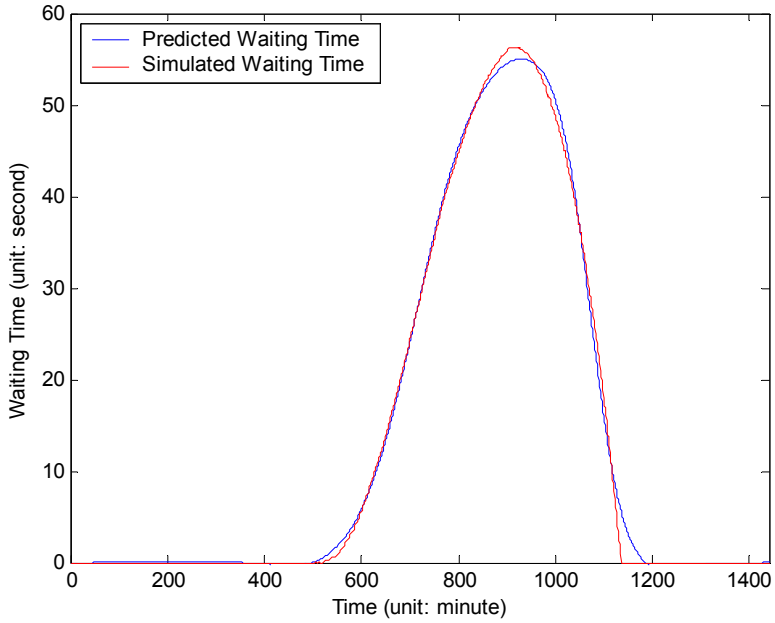


Fig.6.Waiting Time Prediction by LM Trained Neural Network with 5 Hidden Neurons

Suppose that the base station is capable to handle 1000 connections simultaneously. As we described in the mobile system model, the number of connection requests in one time unit follows Poisson distribution. We set a time unit as one minute, and mean arrival number is λ in one time unit. Considering there is significant variation of calling tendency of mobile phone users in a day, we suppose people tend to use the mobile phone at noon frequently but tend not to use mobile phone at night.

So it is supposed that there are 1200 connection requests coming in one minute at the peak request time, and the mean arrival number can be calculated using Formula(7).

$$\lambda = 600 \times (\cos(\frac{2 \cdot \pi \cdot time}{1440} - \pi) + 1) \text{ . (7)}$$

It is shown in Fig.3. When the request number plus the currently connected number is greater than 1000, some of the callers have to wait for a little while.

As described in the mobile system model, phone consistency time is exponentially distributed. We used mean phone consistency time β =120 seconds. So the

number of connection requests plus the connections in use for each minute can be calculated, as shown in Fig.4.

There are totally 861878 phone calls put through in 1440 minutes (24 hours) in our simulation. The waiting time can be calculated to be the neural network-training target. It is shown is Fig. 5. For a random call, if it cannot be put through, the waiting time is in calculated for each calling request. But it is too specific,

which appears redundant, we used the waiting time in for per minute to simplify the training data number from 861878×2 to 1440×2 (both training input and training target). Each PerMinute data is the mean value of the PerCall data in the current minute.

We used 9 feed forward neural network training algorithms to train the networks. But LM algorithm performs the best. Because we used mean squared error (MSE) as the performance function. So the Jacobian matrix can be computed through a standard back propagation technique that is much less complex than

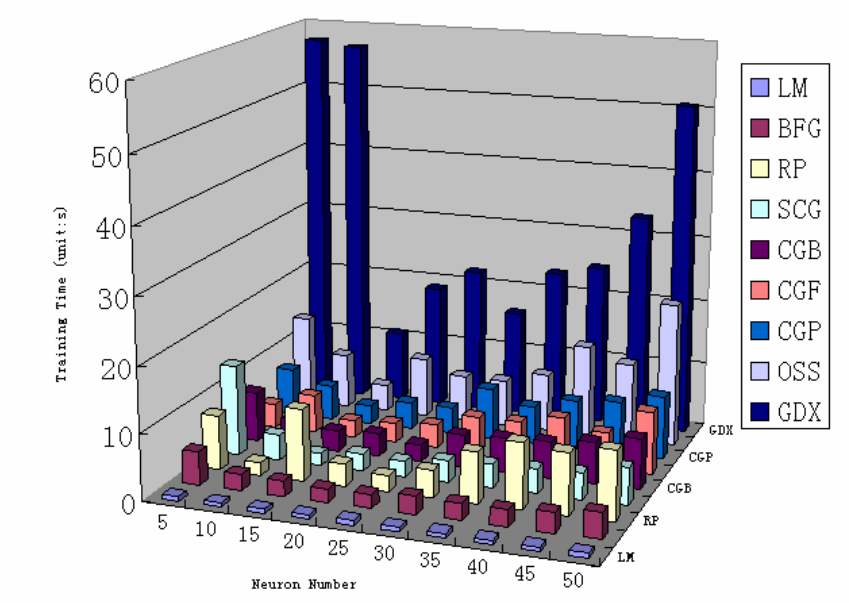


Fig.7. Training Time Comparison

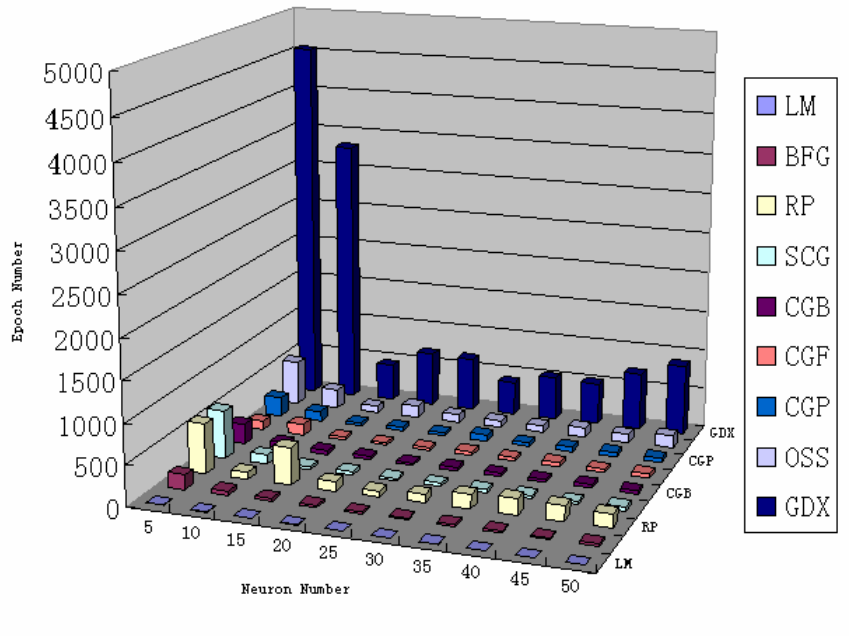


Fig.8.Training Epoch Comparison

computing the Hessian matrix. We first provide the training result from a neural network with only 5 hid-

den neurons. And then provide performance comparison among different algorithms with different hidden neuron numbers.

Fig.6 shows the prediction result of neural network with 5 hidden neurons. It was trained with LM algorithm, and only took 0.672 second to accomplish the training process. The training process stopped when MSE achieved 0.001 s^2.

For each training algorithm, we did the experiment 10 times to evaluate each algorithm's performance under different neuron number in the hidden layer. In each experiment, the hidden neuron number will be set from 5 to 50, and add 5 for each experiment. We set MSE goal to 0.001 (s^2). When MSE reaches 0.001, training will stop automatically. We can evaluate these algorithms on time complexity and epoch complexity. Fig.7 shows LM algorithm has the best performance in terms of time complexity comparing with other algorithms. Fig.8 shows LM, BFG, SCG, CGB and CGF have almost the same epoch complexity when neuron number is larger than 10. But LM is still a little better than others. Fig.9 shows the Epoch speed of each algorithm. It is obvious that LM algorithm has the lowest epoch speed. That is to say, when there are more neurons in the hidden layer, and epoch number increases, LM will not perform very well. In fact, when neural network has more than one hundred, LM will behave worse than some other algorithms. In these figures, BFG is for FGS-Quasi-Newton, RP for Resilient Back propagation, SCG for Scaled Conjugate Gradient, CGB for Conjugate Gradient with Powell/Beale Restarts, CGF for Fletcher-Powell Conjugate Gradient, CGP for Polak-Ribiére Conjugate Gradient, OSS for One-Step Secant and GDX for Variable Learning Rate Back propagation. They are all famous training algorithms [22].

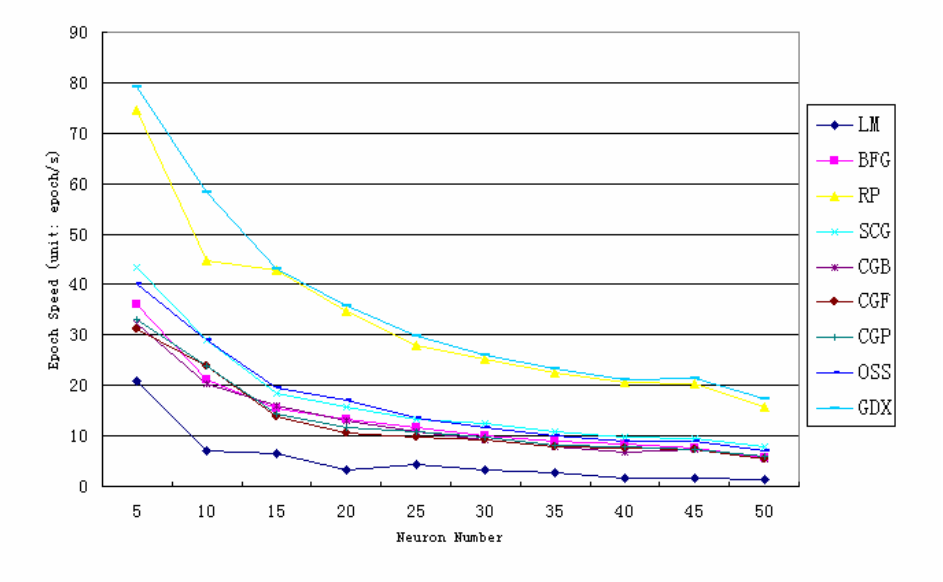


Fig.9.Epoch Speed Comparison

It is found that, the series of training processes by LM take the similar time to achieve the same precision. The training time ranged from 0.578 to 0.719 seconds, and the network with 50 hidden neurons took 0.719 seconds. The waiting time predictions with the 10 different neuron numbers are shown in Fig.10 and Fig.11. We could see that, all of the hidden neuron number setups perform well. On this basis, we will choose the one with less neuron number.

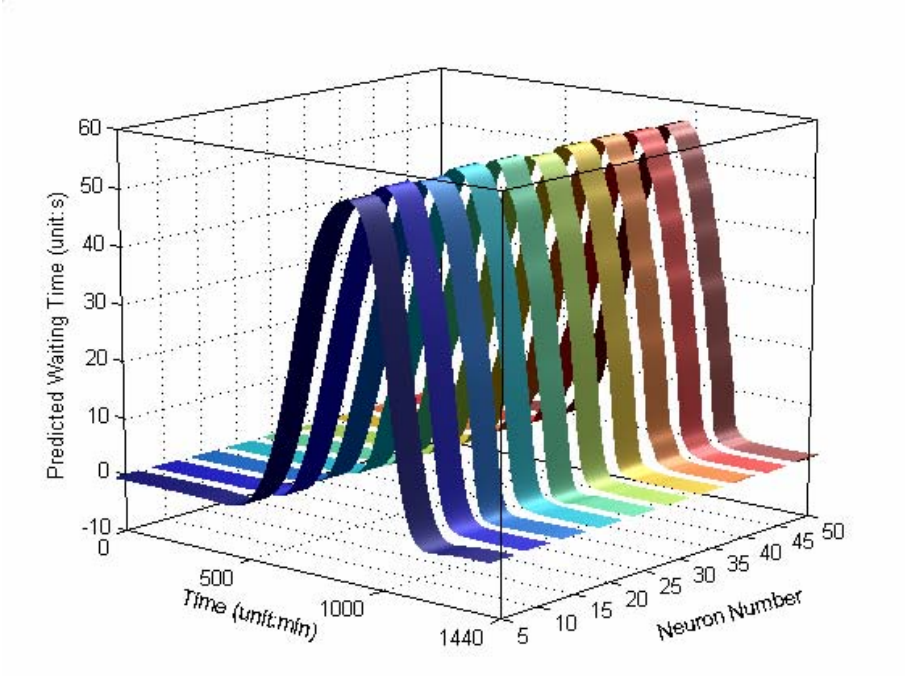


Fig.10. Waiting Time Prediction with Different Hidden Neuron Numbers

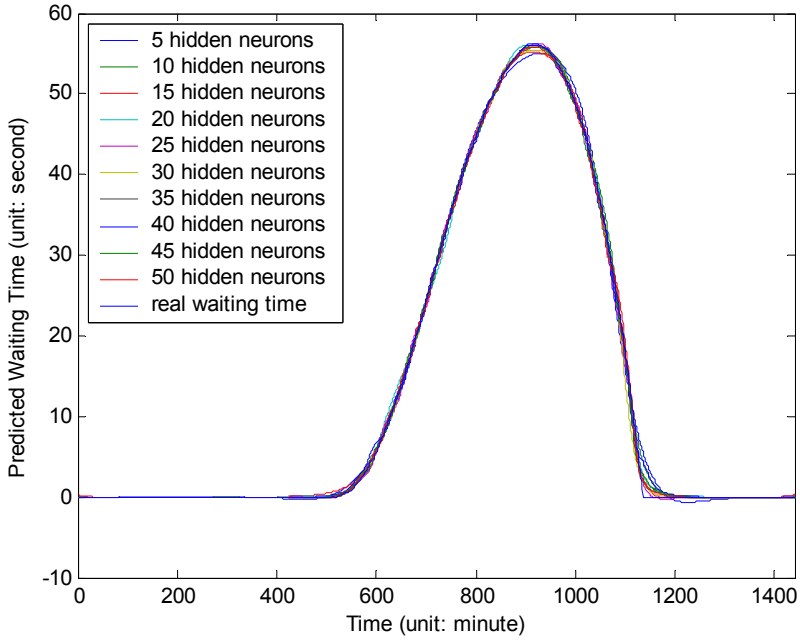


Fig.11. Flat Shape of Fig.10

IV. Conclusion

As congestion will happen in mobile system networks, and mobile phone users are willing to be notified the estimated waiting time in case congestion happens. We provided a waiting time prediction method using feed forward neural network with back propagation. We discussed the mobile system model and neural network model, and then provided simulation result to show the feasibility and high performance of the model. In the simulation, waiting time prediction of a whole day was predicted and satisfactory experiment result has been got by using LM algorithm.

V. Future Work

It could be found that neural network can provide us high performance, adaptive and accurate waiting time prediction. We did simulation for a whole day waiting time prediction to prove its feasibility. It has been mentioned in model and analysis section that multi-parameters could be used to form hyperplane to do waiting time prediction. However, in order to simply the simulation and provide better and clear presentation, we only used one parameter. We only used time as the input parameter in our simulation. Probably more works could be done when we use multi-parameters as the input and the experiment could be done to simulate a month or a yearlong waiting time prediction. Further more, although we proposed LM algorithm to be the training algorithm, other more stable algorithms are expected to takeover LM when the neural network has more than one hundred weights, e.g. GDX and CGF etc.

Reference

[1]. Bin Li; Lizhong Li; Bo Li; Sivalingam, K.M.; Xi-Ren Cao; *Call admission control for voice/data integrated cellular networks: performance analysis and comparative study*, IEEE Journal on Selected Areas in Communications , Volume: 22 , Issue: 4 , May 2004 ,Pages:706 – 718

[2]. Jihui Zhang; Jinpeng Huai; Renyi Xiao; Bo Li; *Resource management in the next-generation DS-CDMA cellular networks*, IEEE Wireless Communications, Volume: 11 , Issue: 4 , Aug. 2004 , Pages:52 – 58

[3]. Ramiro-Moreno, J.; Pedersen, K.I.; Mogensen, P.E.; *Capacity gain of beamforming techniques in a WCDMA system under channelization code constraints*, IEEE Transactions on Wireless Communications, Volume: 3 , Issue: 4 , July 2004 ,Pages:1199 – 1208

[4]. Chang Wook Ahn; Ramakrishna, R.S.; *QoS provisioning dynamic connection-admission control for multimedia wireless networks using a Hopfield neural network*, IEEE Transactions on Vehicular Technology , Volume: 53 , Issue: 1 , Jan. 2004, Pages:106 – 117

[5]. Leong, C.W.; Weihua Zhuang; Yu Cheng; Lei Wang; *Call admission control for integrated on/off voice and best-effort*

data services in mobile cellular communications, IEEE Transactions on Communications, Volume: 52 , Issue: 5 , May 2004, Pages:778 – 790

[6]. Ormeci, E.L.; *Dynamic admission control in a call center with one shared and two dedicated service facilities*, IEEE Transactions on Automatic Control, Volume: 49 , Issue: 7 , July 2004, Pages:1157 – 1161

[7]. Rao, R.M.; Comaniciu, C.; Lakshman, T.V.; Poor, H.V.; *Call admission control in wireless multimedia networks*, IEEE Signal Processing Magazine, Volume: 21 , Issue: 5 , Sept. 2004 , Pages:51 – 58

[8]. Levendovszky, J.; Fancsali, A.; *Real-Time Call Admission Control for Packet-Switched Networking by Cellular Neural Networks*, IEEE Transactions on Circuits and Systems I: Regular Papers, Volume: 51 , Issue: 6 , June 2004, Pages:1172 – 1183

[9]. Dragan, V.; Morozan, T.; *The linear quadratic optimization problems for a class of linear stochastic systems with multiplicative white noise and Markovian jumping*, IEEE Transactions on Automatic Control , Volume: 49 , Issue: 5 , May 2004 , Pages:665 – 675

[10]. Lei Huang; Kumar, S.; Kuo, C.-C.J.; *Adaptive resource allocation for multimedia QoS management in wireless networks*, IEEE Transactions on Vehicular Technology, Volume: 53 , Issue: 2 , March 2004 , Pages:547 – 558

[11]. Hu, F.; Sharma, N.K.; *Priority-determined multiclass handoff scheme with guaranteed mobile QoS in wireless multimedia networks*, IEEE Transactions on Vehicular Technology, Volume: 53 , Issue: 1 , Jan. 2004 , Pages:118 – 135

[12]. Jianxin Yao; Mark, J.W.; Tung Chong Wong; Yong Huat Chew; Kin Mun Lye; Kee-Chaing Chua; *Virtual partitioning resource allocation for multiclass traffic in cellular systems with QoS constraints*, IEEE Transactions on Vehicular Technology , Volume: 53 , Issue: 3 , May 2004 , Pages:847 – 864

[13]. Lee, C.-G.; Sha, L.; Avinash Peddi; *Enhanced utilization bounds for QoS management*, IEEE Transactions on Computers , Volume: 53 , Issue: 2 , Feb 2004 ,Pages:187 – 200

[14]. Shengquan Wang; Dong Xuan; Bettati, R.; Wei Zhao; *Providing absolute differentiated services for real-time applications in static-priority scheduling networks*, IEEE/ACM Transactions on Networking, Volume: 12 , Issue: 2 , April 2004, Pages:326 – 339

[15]. Duan-Shin Lee; Yun-Hsiang Hsueh; *Bandwidth-reservation scheme based on road information for next-generation cellular networks*, IEEE Transactions on Vehicular Technology, Volume: 53 , Issue: 1 , Jan. 2004 , Pages:243 – 252

[16]. Cruz-Perez, F.A.; Ortigoza-Guerrero, L.; *Capacity optimization in wireless communication systems with mixed platforms*, IEEE Communications Letters, Volume: 8 , Issue: 4 , April 2004 , Pages:217 – 219

[17]. Foo, Y.-L.; Takahashi, K.; Lee, S.-W.; *Cell admission control in multiservice satellite systems*, 10th International Conference on Telecommunications, Volume: 2 , 23 Feb.-1 March 2003 , Pages:1605 - 1609 vol.2

[18]. Osseiran, A.; Ericson, M.; *On downlink admission control with fixed multi-beam antennas for WCDMA system*, Vehicular Technology Conference 2003-Spring, Volume: 2 , 22-25 April 2003 , Pages:1203 - 1207 vol.2

[19]. Outes, J.; Nielsen, L.; Pedersen, K.; Mogensen, P.; *Multi-cell admission control for UMTS*, Vehicular Technology Conference 2001 Spring. Volume: 2 , 6-9 May 2001 , Pages:987 - 991 vol.2

[20]. Lai, K.; Baker, M.; *Net timer: A tool for measuring bottleneck link bandwidth*, Proceedings of the USENIX Symposium on Internet Technologies and Systems, March 2001.

[21]. Ei-Kadi, M.; Olariu, S.; Abdel-Wahab, H.; *Rate-based borrowing scheme for QoS provisioning in multimedia wireless networks*, IEEE Transactions on Parallel and Distributed Systems, Volume: 13 , Issue: 2 , Feb. 2002 , Pages:156 – 166

[22]. Demuth, H.; Beale, M., *Neural Network Toolbox*, MathWorks Cooperation, 2000.

[23]. Beming, P.; Frodigh, M.; *Admission control in frequency hopping GSM systems*, 1997 IEEE 47th Vehicular Technology Conference, Volume: 2 , 4-7 May 1997 , Pages:1282 - 1286 vol.2

[24]. Carter, R. L.; Crovella, M. E.; *Dynamic server selection using bandwidth probing in wide-area networks*, Technical Report BU-CS-96-007, Computer Science Department, Boston University, March 1996

[25]. Hiramatsu, A.; *Training techniques for neural network applications in ATM*, IEEE Communications Magazine, Volume: 33 , Issue: 10 , Oct. 1995 , Pages:58, 63 – 67

[26]. Hagan, M. T., and Menhaj M.; *Training feedforward networks with the Marquardt algorithm*, IEEE Transactions on Neural Networks, vol. 5, no. 6, pp. 989–993, 1994.

[27]. Cooper, C.A.; Park, K.I.; *Toward a broadband congestion control strategy*, IEEE Network, Volume: 4, Issue: 3, Pages: 18 – 23, May 1990

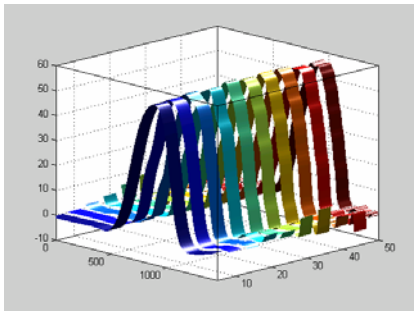
[28]. Murata, M.; Oie, Y.; Suda, T.; Miyahara, H.; *Analysis of a discrete-time single-server queue with bursty inputs for traffic control in ATM networks*, IEEE Journal on Selected Areas in Communications, Volume: 8, Issue: 3, Pages: 447 – 458, April 1990

[29]. *DARPA Neural Network Study*, Lexington, MA: M.I.T. Lincoln Laboratory, 1988.

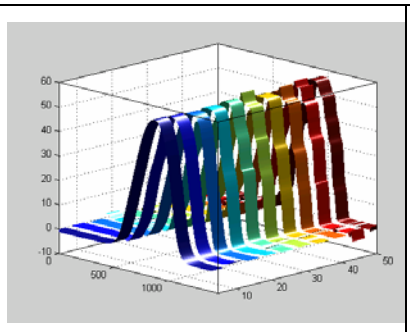
[30]. Parker, D. B.; *Learning-logic: Casting the cortex of the human brain in silicon*, Technical Report TR-47, Center for Computational Research in Economics and Management Science, MIT, Cambridge, MA, 1985.

Appendix: Performance Comparison with

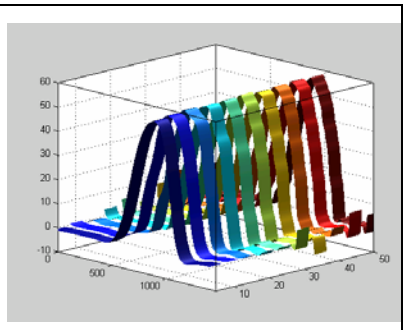
Different Neural Network Training Algorithms



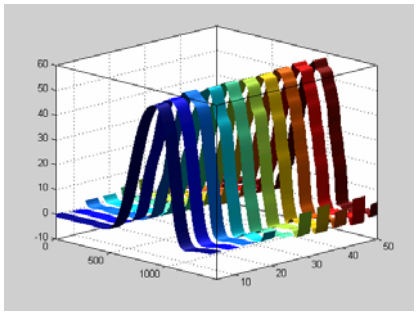
BFGS Quasi-Newton



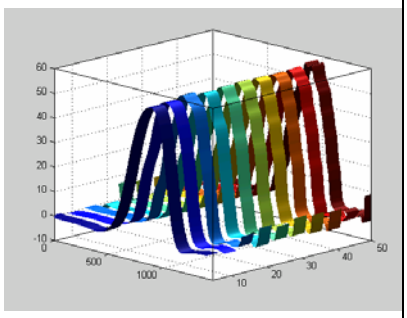
Resilient Back Propagation



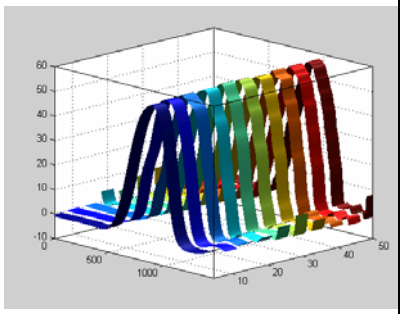
Scaled Conjugate Gradient



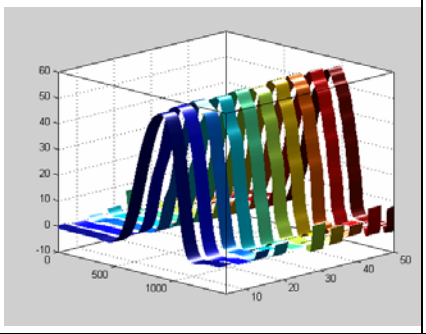
Conjugate Gradient with Powell/Beale Restarts



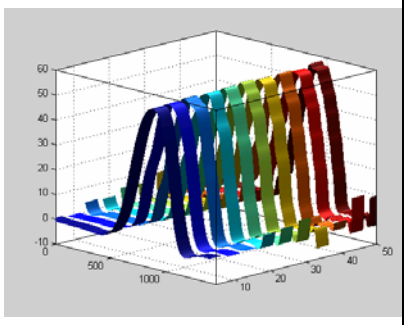
Fletcher-Powell Conjugate Gradient



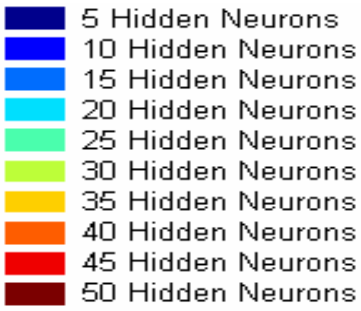
Polak-Ribiére Conjugate Gradient



One-Step Secant



Variable Learning Rate Back Propagation



Legend of Different Neuron Number

Fig.12.Other 8 Waiting Time Prediction Results with Different Neuron Number