I J C T A, 9(6), 2016, pp. 2787-2792 ISSN: 0974-5572

Stabilizing Queue using Intelligent Neural Networks for Congestion Control

G. Maria Priscilla*

ABSTRACT

Congestion a major problem in today's world of internet drops packet when the queue is heavily loaded. Active queue management schemes proactively respond for congestion during bursty traffic. Parameter settings in AQM schemes become difficult and hence intelligent neural network learning methods are used to stabilize the queue and gives best results in dropping of packet ratio. Stabilization of queue is tested on a dumbbell topology in NS-2 for various traffic conditions and ART2 performs better in training time, packet delivery ratio, queue delay and jitter when compared to other competitive learning methods in neural network.

Keywords: Active Queue Management(AQM), Random Early Detection (RED), neural networks, KohonenSelf Organizing Map (KSOM), Learning Vector Quantization (LVQ), Adaptive Resonance Theory (ART)

1. INTRODUCTION

A model by Emmanuel Lochin and Bruno Talavera [1] based on Kohonen neural Network to overcome the problem of setting the tuning parameters in RED AQM is designed. Kohonen neural networks are a class of neural networks known to solve the pole balancing problem [Mak91]. Pole balancing is a control benchmark historically used in mechanical engineering. It involves a pole placed on a cart via a joint allowing movement along a single axis. The cart is able to move along a track with a ûxed length as represented in figure 1 (a)the aim of the problem is to keep this pole balanced by applying forces to the cart. The main idea of this is based on the analogy existing between the pole balancing problem and the RED queuing problem. In Kohonen RED (KRED) [1], the pole balancing is compared to the evolution of the queue occupancy which oscillates between both thresholds ($min_{th,}$ max_{th}). The physical force applied on the pole by moving the cart front and back is similar to stabilizing the queue on various packet arrival rates in the queue. Figure 1(b) [1] illustrates this view.

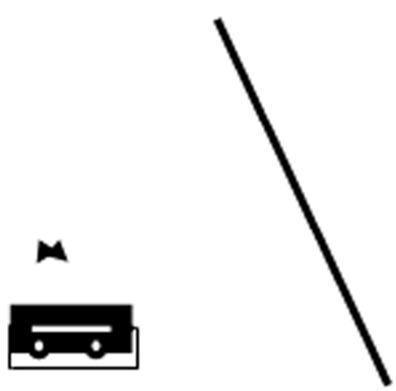

Figure 1 (a): Pole Balancing Problem

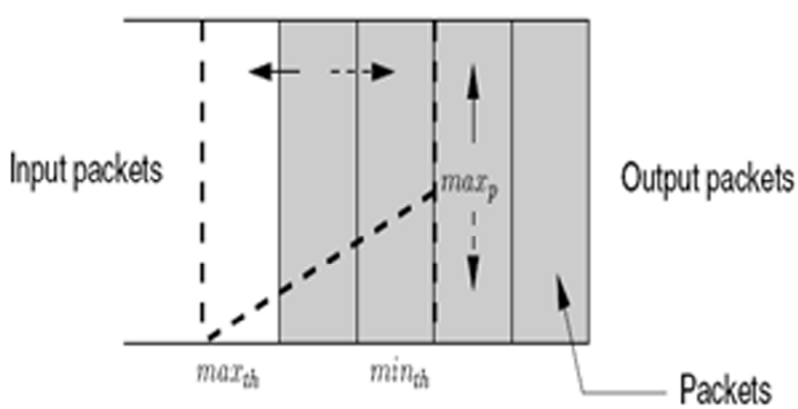

Figure 1 (b): Adaptive RED with Pole Balancing

* Associate Professor, Dept of Computer Science, SNR Sons College, Coimbatore, India, *Email: mariajerryin@yahoo.co.in*

2. ACTIVE QUEUE MANAGEMENT AND NEURAL SY

The table1 shows the input vector as the previous queue length and the current queue length and the output vector is the calculated value of the $max_{p.}$

The max_p is updated by using the value as a function of the arrival rate in order to stabilize the queue size between the min_{th} and $max_{th.}$ In[1], the authors explain the queue length variation by the need of dynamically changing max_p as function of the queue occupancy. They proposed to recomputed this probability following an AIMD algorithm. The update is done as a function of the average queue size. If the average queue size is around max_{th}, the algorithm increases max_p to drop more packets and decreases max_p if the value is around min_{th}.

Table 1
Input and Output values

Input_value [1]	Previous queue length
Input_value[2]	Current queue length
Output_value[1]	max_p

2.1. Kohonen-SOM based RED queue

The Kohonen SOM deals with a topological learning feature imply neural neighborhood generalization of a correct learning experience. Given the Kohonen SOM algorithm, the neural network can generalize its learnt experiences to other input vectors it has never seen before and produce adapted responses. The conservation of a direction, equilibrium or the correct parameter to adjust a RED mechanism is made possible although there is no way of predicting the way the neural network learns to solve this particular problem. The learnt sequences of input vectors are not the ones used in the tests, in order to prove that the learning method provides a general purpose neural network for the resolution of the problem dealt here. Once it has learnt, it can be used indeûnitely for the task it has been trained for. Neural network is well adapted to pattern and shape recognition problems, whilst a SOM such as the Kohonen SOM could be better suited to the task of stability preservation. Indeed, this Kohonen SOM algorithm preserves topological relationships between neighboring vectors. Each time a packet is enqueued, the Kohonen network computes a new max_p following the previous and the current average queue size. No other parameters are needed to perform this operation.

2.2. Modification to Kohonen-based RED queue Technique

Classification of the input vectors with a set of $m \times n$ weight matrix is the fundamental operation of Kohonen's network where '*m*' is the number of nodes in input layer and '*n*' is the grid size. Learning system in Kohonenbased RED queue technique considers the previously learned vectors while adopting the weight matrix for the current input vectors that is avoided by the proposed subsystem [3].

The following modifications are performed on existing system:

- ***Adoption of Weights***: Conventional recognition techniques deal with the previously stored vectors, the modified system only tries to operate on the recently given pattern sample. It avoids the previously learned vectors for the swiftness of learning process.
- ***Regular recognition***: This system using KRED offered the modification of weights for all the connections among the two layers. In the proposed MKRED system, the rapid change of neighborhood size was considered and hence the number of weights adopted easily decreased with the time. The Modified KRED [3] system proposes a function for changing the neighborhood size along with the change of the distance of winner node shown in the below equation (1)[3]

$$\varphi(t+1) = \varphi(t) - \eta(t)\,(d_J(t) - d_J(t-1)) \tag{1}$$

Where 't' is the time step, 'η' is a small variable called learning rate, 'φ' is the old weight and 'd_J' the input vector. The winning node is used as the maximum probability. Based on this probability, the queue is stabilized. The proposed method helps in achieving a better accuracy, increases training time, average queue delay and jitter is better and it stabilizes the queue with better performance.

2.3. Learning Vector Quantization

LVQ is an artificial neural network used in the field of computational intelligence [4]. It is a supervised neural network that uses competitive learning which applies winner-take-all Hebbian learning based approach. An LVQ system is represented by prototypes $W = (w(\text{i}), ..., w(n))$ which are defined in the feature space of observed data[7]. In winner-take-alltraining algorithms, one determines for each data point, the prototype which is closest to the input according to a given distance measure [5]. An advantage of LVQ is that it creates prototypes that are easy to interpret for experts in their respective application domain. LVQ systems can be applied to multi-class classification problems in a natural way. A key issue in LVQ is the choice of an appropriate measure of distance or similarity for training and classification [6].

2.4. Adaptive Resonance Theory 2

ART self–organizes a stable pattern recognition. This recognition code is done in real time based on the arbitrary input pattern sequences. Due to its self-stabilization ART does not possess the local minimum problem. It is an unsupervised paradigm and is based on competitive learning method, capable of finding categories and creating new categories when needed automatically. During the training process the old weights that were captured may be lost when new weights come in. The process of maintaining the old weights with that of the new weights is called the stability –plasticity dilemma which is the main concept used in this neural network.

Some of the properties of ART as in [8] are:

1. To normalize the total network activity.
2. Contrast enhancement of input pattern.
3. Short-Term Memory (STM) storage of the contrast-enhanced pattern.

3. FINDINGS AND DISCUSSIONS

This shows the performance evaluation of the proposed approaches for which the experiments were carried out using Network Simulator (NS-2). The experimentation is performed upto 25 sample patternsto learn at a time. The dumbbell topology is used for the tests experiments. The TCP flows are NewReno with 10000 packets. Two types of traffic flow are used in the experiment. First the number of traffic flow is increasing from 50 to 250 flows with identical Round Trip Time (RTT) as in figure 2. The second type the traffic flow is varied every 50 seconds in which each flow has a RTT ranging from 64 to 102 ms shown in figure 2. It is

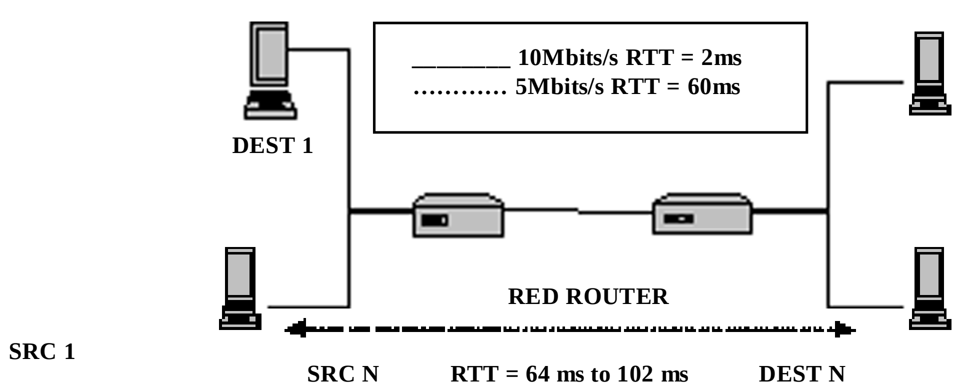

Figure 2: Dumbbell Topology

observed from the experimental results that the proposed approaches of the intelligent competitive learning methods are performing better than the approach in the existing system.

Table 2 shows the overall results of QOS parameters of the proposed approaches. From the table 2 it is clear that the results obtained in terms of training time, queue delay and jitter the performance is better in ART2 RED. Figure 3 gives the training time required for each neural network competitive learning methods like KSOM, LVQ, and ART2.

Table 2
Training Step, Average Queue Delay, Variance of Queue Delay (Jitter) and Packet Delivery Ratio (PDR)

Methods	Training Time *(Ms)*	Queue Delay *(ms)*	Jitter *(ms)*	PDR
KRED	301.4626	0.283853	0.003134	0.973
MKRED	248.0127	0.209761	0.002252	0.971
LVQRED	227.1097	0.210874	0.002295	0.971
ART2RED	110.5288	0.199942	0.001915	0.969

Figure 3: Training Time between KRED, MKRED[9], LVQ RED[10] and ART2 RED[11]

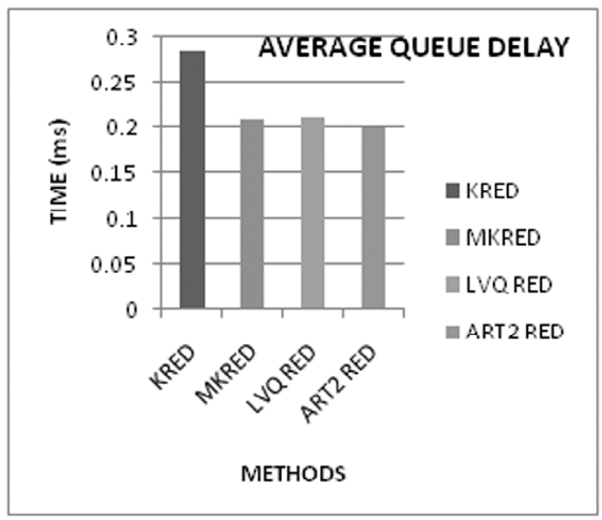

Figure 4: Average Queue Delay between KRED, MKRED, LVQ REDand ART2 RED

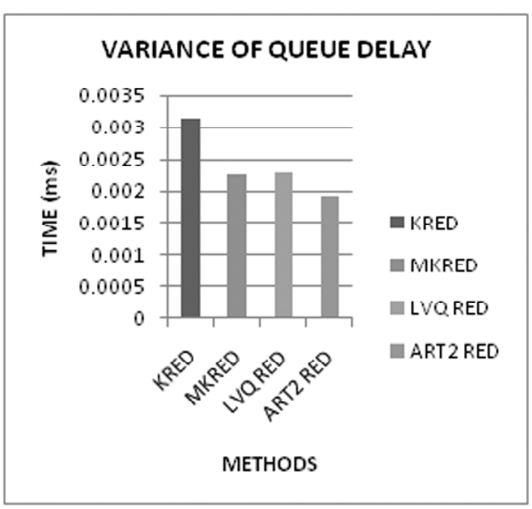

Figure 5: Variance of Queue Delay between KRED, MKRED, LVQ RED and ART2 RED

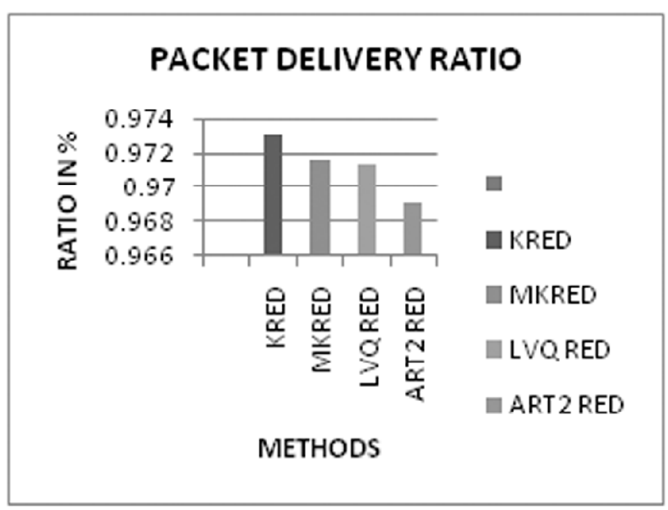

Figure 6: Packet Delivery Ratio between KRED, MKRED, LVQ RED AND ART2 RED

Figures 4 and 5 show the average queue delay and the jitter performance of the various competitive learning approaches. The ART2 RED performs better in average queue delay and in the variance of queue delay (jitter) than the other approaches.

Figure 6 show the packet delivery ratio of the various approaches among which KRED delivers more packet when compared to the other approaches.

Table 3 and figure 7 discuss about the number of bytes sent and the total number of bytes received in various approaches. When compared to all the approaches the proposed MKRED intelligent learning performs better than the other approaches.

4. CONCLUSION

This focuses on developing an intelligent active queue management for congestion control. The present wok proposed three competitive learning methods for maintaining the queue stability in congestion control. The MKRED uses the conventional recognition technique which increases learning time exhaustively. This method avoids the previously learnt vectors for the swiftness of learning process. In the regular recognition technique, the weights are updated by considering the rapid change of neighborhood size and hence the number of weights adopted easily decreases with the time. The LVQ approach uses an adaptive data classification method and generalizes its learnt experiences to the other input vectors which has never

Table 3
Number of Bytes Sent and Received

Methods	*BYTES Sent (Bytes)*	*BYTES received (Bytes)*
KRED	565365	545740
MKRED	574247	557947
LVQ RED	574606	558134
ART2 RED	576472	558710

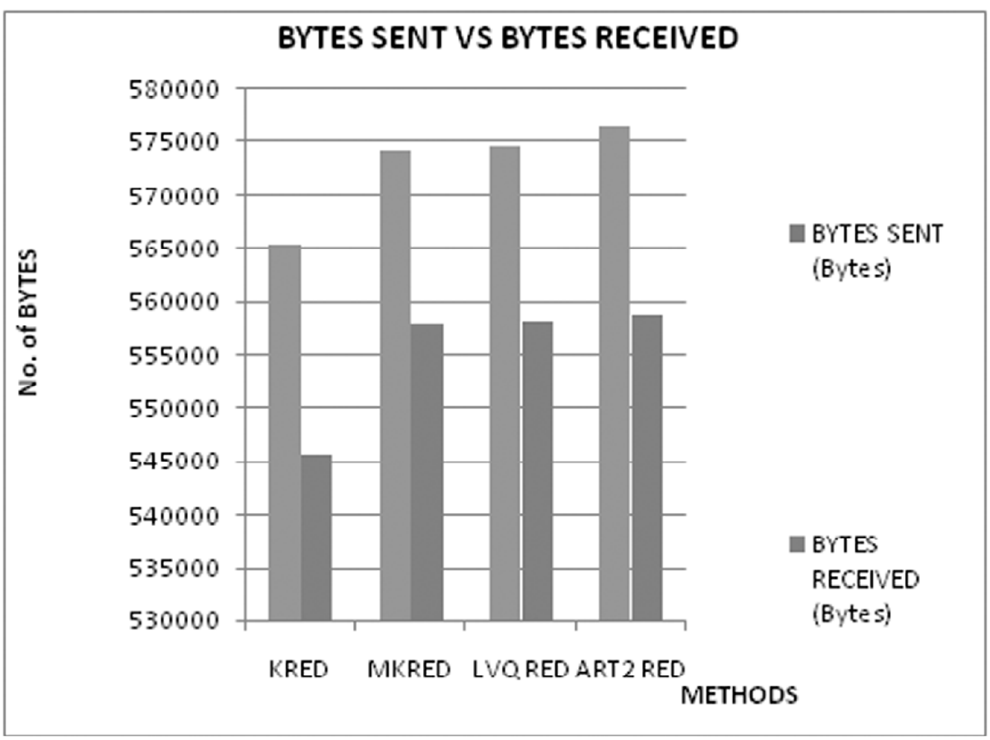

Figure 7: Bytes Sent vs Bytes Received betweenKRED, MKRED, LVQ RED AND ART2 RED

aroused before. So that adaptive responses are received in a faster manner. The ART2 approach, the intelligent learning uses the stability plasticity dilemma henceforth having the previously learnt knowledge in the short term memory and long term memory for classification. This speeds up the learning time obtained for various input vectors. The intelligent learning capability generates the previous knowledge about the given traffic as input vectors and speeds up the training process. Clustering of the traffic and the categorization process of the intelligent learning methods are highly performed for bursty traffic. The results confirm that ART2 RED trains the input vectors in a faster manner and stabilizes the queue with less queue delay and jitter for varying traffic with different round trip time. The parameter settings of the RED queue algorithm are easier and the stability of the queue is maintained. The drop probability of the ART2 RED is decreased so that the dropping of the packets is comparatively low. The QoS parameters are discussed to provide a more generalized learning method suitable for all types of traffic.

REFERENCES

[1] Emmanuel Lochin and Bruno Talavera, "Managing Network Congestion with a Kohonen-Based RED Queue", Engineering Applications of Artificial Intelligence, Vol. 24, No. 1, pp. 77-86, February 2011.

[2] A. Makarovic, "Machine Intelligence 12: Towards an AutomatedLogic of Human Thought", In Clarendon Press, New York, NY, USA, pp. 241-258, 1991.

[3] S.M. Kamrulhasan, MD. Nazrul Islam Mondal, NafisaTarannum, MD. Tanvir Ahmed Chowdhury, "Recognition of Facial Patternby Modified Kohonen's Self Organizing Map(MKSOM) andAnalyze the Performance", Proceeding of the 3rd AsianApplied Computing Conference, Nepal engineering college, Kathmundu, Nepal, December 2005.

[4] Jason Brownlee, "Clever Algorithms: Nature-InspiredProgramming Recipes", First Edition, Lulu Enterprises, ISBN:978-1-4467-8506-5, January 2011. www.svs-vistek.com

[5] Michael Biehl, Anarta Ghosh1, and Barbara Hammer2, "Learning Vector Quantization: The Dynamics of Winner-Takes-All Algorithms", Neuro computing, Vol. 69(7-9), pp. 660-670, 2006.

[6] A.H. Chen, S.Y. Huang, P.S. Hong, C.H. Cheng and E.J. Lin, "HDPS: Heart Disease Prediction System" Department of Medical Informatics, Taiwan, ISSN 0276-6584, Computing in cardiology

[7] Wikipedia.org/wiki/Learning Vector Quantization

[8] www.learnartificialneuralnetworks.com/art.html

[9] G. Maria Priscilla and Antony SelvdossThanamani, "Managing Network Congestion With A Modified Kohonen Based Queue", International Journal Of Engineering Science And Technology, Vol. 2, No. 11, pgs: 6747 – 6752, ISSN: 0975-5462, Nov. 2010.

[10] G. Maria Priscilla and C.P. Sumathi, "An Intelligent Active Queue Management for Congestion Control", International Journal of Computer Applications, Vol. 41, No. 16, pg. 25-28, ISSN: 0975-8887, March 2012.

[11] G. Maria Priscilla and C.P. Sumathi, "Managing Queue Stability using Art2 in Active Queue Management for Congestion Control", ARPN Journal of Engineering and Applied Science, Vol. 8, No. 8, pgs. 663-667, ISSN 1819-6608, August 2013