Available online at www.elixirpublishers.com (Elixir International Journal)

Network Engineering

Elixir Network Engg. 62 (2013) 17501-17506

Elixir ISSN: 2229-712X

Synthesis of neuro-fuzzy systems for active management of packet queues in telecommunication networks

Polschykov K* and Korotenko Y

Donbass State Engineering Academy, Kramatorsk, Ukraine.

ARTICLE INFO

Article history:
Received: 24 July 2013;
Received in revised form:
20 August 2013;
Accepted: 31 August 2013;

Keywords

Neuro-fuzzy systems,
Packet queue,
Telecommunication network,
Router.

ABSTRACT

In this article neuro-fuzzy systems for active management of packet queue of the output port of a router of a telecommunication network are considered. The employment of this systems allows reduction of average packet delay in a queue of the output port of a router and lowering of the probability of packet discard.

Introduction

For modern telecommunication networks with packet switch overload is typical, caused by the increase of traffic intensity in these networks in terms of their limited bandwidth. Without implementation of an effective counter to network overloads it cannot provide high customer service quality. To overcome overloads many methods are used, among which management technology of packet queues in routers is of considerable importance [1].

There is passive and active queue management. Passive management involves discarding packets arriving in those moments when there are no vacancies in the appropriate channel queue. This method is called Tail Drop [2]. The main advantage of Tail Drop is its simplicity, but the application of the method of passive management of queues in routers has the following negative consequences. First, under conditions of overload, the packets, being in a long queue for a long time, are awaiting for the transfer. As a result at the host side high values of average packet delay are observed. Second, because of sharp fluctuations in the packet queue length there is a significant jitter. It should be noted that the queue length can also vary from time to time (the effect of global synchronization). Third, the implementation of Tail Drop often invokes unjust queue capture by packets of only one or more streams. Consequently the other streams are transmitted with a high percentage of packet loss. In this case the quality of network user service is significantly reduced.

In order to correct these deficiencies active queue management is proposed. The essence of this management is to prevent the occurrence of congestion by discarding some of the packets entering the router until the completion of the channel queue.

The most common method of active management of queues is the method of random early detection of overload (Random Early Detection, RED) [3]. According to this method the decision on the rejection of a package is accepted on the basis of computing the average queue size and the probability of discarding packets. To calculate these values analytical expressions containing a number of parameters, which values are chosen experimentally, are used. It should be noted that the calculation formulas are heuristic and, unfortunately, do not have sufficient theoretical justification. Therefore, in practice the use of RED though reduces the average packet delay compared to Tail Drop, but often results in more packet loss.

Imperfect RED method gave rise to a large number of its various modifications. Currently known methods are adaptive (Adaptive RED), dynamic (Dynamic RED), stabilized (Stabilized RED), streaming (Flow RED), weighted (Weighted RED) early detection of overload [3]. However, all of the advanced versions of active queue management also have the main drawback of classic RED: the decision to drop packet is accepted on the basis of very rough, approximate models. Therefore, application of existing methods of queue management in packet switching networks does not always allow an effective remedy against overloads, limiting opportunities to ensure customer service quality.

Analysis of the research results showed that packet queue management in routers was available in terms of incomplete, fuzzy, inaccurate information about the state of the network in real-time and in the future. Effective means of management in such circumstances is the use of neuro fuzzy systems [4; 5].

Fuzzy neural network is a multi-layered neural network in which the layers serve as the elements of fuzzy inference. Neurons of such a network are characterized by a set of parameters which are adjusted during the learning process like in conventional neural networks. Actually, such systems are hybrid in essence and combine the advantages of neural networks and fuzzy inference systems. On the one hand, they allow developing and providing models of systems in the form of rules of fuzzy output possessing visibility and ease of meaningful interpretation. On the other hand, for the construction of rules of fuzzy output the methods of neural networks are used, which makes this process less difficult for researchers.

Tele:
E-mail addresses: konspol@rambler.ru

There are enough examples of successful systems using neuro-fuzzy inference in various fields of science and technology [4; 5]. Therefore an attempt to apply the apparatus of neuro-fuzzy systems for active management of packet queues in routers of telecommunication network is quite justified.

This article is dedicated to the solving of urgent scientific and technological challenges which consist in the synthesis of neuro-fuzzy systems for active management of packet queues in telecommunication network.

The aim of the paper is to reduce packet loss and average delay in routers using neuro-fuzzy active management of packet queues.

The essence of neuro-fuzzy active management of packet queues

The process of active management of packet queues of a router output port is given as a sequence of periodic cycles. Figure 1 shows a fragment of this sequence, in which the current cycle is e. In relation to cycle e, the cycles a, b, c and d are preliminary, and the cycle f is subsequent.

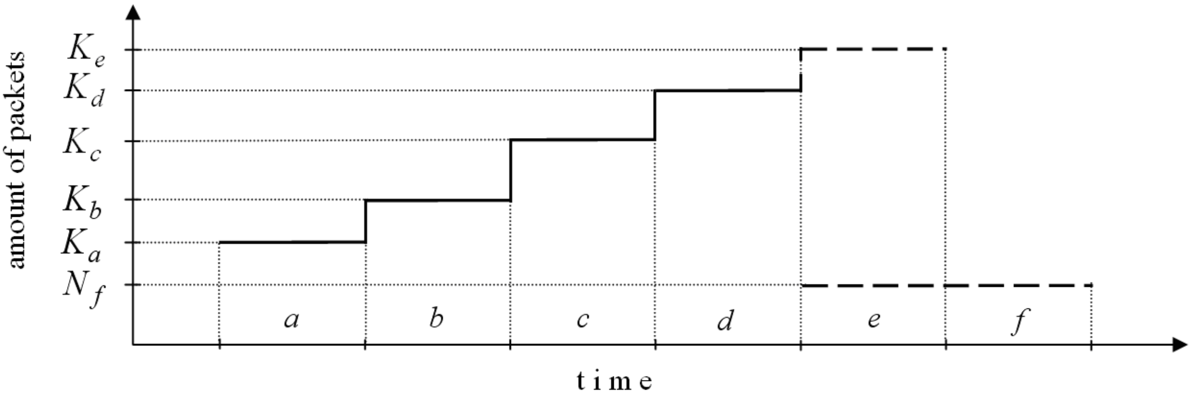

Figure 1: A fragment of the sequence of cycles of active management process of packet queue of a router output port

To the output port of a router in the previous four cycles packets K_a, K_b, K_c and K_d were directed respectively, and, in the current cycle package K_e is assumed to be directed. Furthermore, during the passive queue management the router output port in the cycle f is assumed to reject N_f packets. Then, if $K_e \geq N_f$, for the idea of active queue management it is necessary to randomly discard $N_e = N_f$ packets in the current cycle to anticipate in advance this occurrence of congestion in the next cycle.

The essence of neuro-fuzzy active management of packet queue of the source port of a router is as follows:

1) during each cycle calculation of packets directed to the output port of a router to be transmitted through the channel net work is executed;

2) at the beginning of a current cycle the values K_e and N_f are determined, which are the results of neuro-fuzzy prediction of quantities K_e and N_f accordingly (with the variables K_a, K_b, K_c and K_d used as input values), and then the probability of early package rejection is calculated:

$$p_e = \begin{cases} 1, & \widetilde{K}_e < \widetilde{N}_f; \\ \dfrac{\widetilde{N}_f}{\widetilde{K}_e}, & \widetilde{K}_e \geq \widetilde{N}_f. \end{cases} \tag{1}$$

3) value p_e before the end of the current cycle is used to resolve random early discard of packets; directed to the output port of the router packet is discarded when the condition is met:

$$p_e \geq P, \tag{2}$$

where P is a pseudorandom number with uniform distribution in the range [$\overline{0,\ 1}$].

For the values K_e and N_f the construction of appropriate neuro-fuzzy systems is provided.

Synthesis of neuro-fuzzy systems

It was noted above that the input variables of neuro-fuzzy prediction system for the quantity K_e are the values K_a, K_b, K_c and K_d, and the output variable is the value $\widetilde{K}_e$. To obtain the desired value of the output variable it is proposed to use one of the most common algorithms in fuzzy inference practice – the Sugeno algorithm of the 0th order, based on the knowledge base represented by the set of fuzzy rules [6]:

$$\left\{If\ \left(K_a=\alpha_1^A\right) and \left(K_b=\alpha_2^B\right) and \left(K_c=\alpha_3^C\right) and \left(K_d=\alpha_4^D\right), then \left(\tilde{K}_e=Y_r\right)\right\}, \qquad (3)$$

where α_1^A is the term (fuzzy set) number A of input variable K_a, $A\in\left[\overline{1, A_{\max}}\right]$;

α_2^B is the term number B of input variable K_b, $B\in\left[\overline{1, B_{\max}}\right]$;

α_3^C is the term number C of input variable K_c, $C\in\left[\overline{1, C_{\max}}\right]$;

α_4^D is the term number D of input variable K_d, $D\in\left[\overline{1, D_{\max}}\right]$;

Y_r is the crisp set of individual deduction of the rule number $r\in\left[\overline{1, \rho}\right]$.

For a neuro-fuzzy system to be built the easiest option of its initial parameters is selected: for the input variables two membership functions ($A_{\max}=B_{\max}=C_{\max}=D_{\max}=2$) of triangular shape are chosen. The proposed forecasting system of the value K_e structurally consists of four neural layers (Figure 2), by which the following procedures of fuzzy inference are performed: fuzzification, aggregation, activation and defuzzification.

The first layer performs fuzzification procedure, that is for certain values K_a^*, K_b^*, K_c^* and K_d^* the quantities $\mu_1\left(K_a^*\right)$, $\mu_2\left(K_a^*\right)$, …, $\mu_{A\max}\left(K_a^*\right)$, $\mu_1\left(K_b^*\right)$, $\mu_2\left(K_b^*\right)$, …, $\mu_{B\max}\left(K_b^*\right)$, $\mu_1\left(K_c^*\right)$, $\mu_2\left(K_c^*\right)$, …, $\mu_{C\max}\left(K_c^*\right)$, $\mu_1\left(K_d^*\right)$, $\mu_2\left(K_d^*\right)$, …, $\mu_{D\max}\left(K_d^*\right)$ – i.e. values of membership functions of input variables to relevant terms – are calculated.

The second layer of neuro-fuzzy system performs aggregation procedure, during which the truth degree of the conditions of each rule with specific values of input variables are determined:

$$\left\{G_r=\mu_A(K_a^*)\wedge\mu_B(K_b^*)\wedge\mu_C(K_c^*)\wedge\mu_D(K_d^*)\right\}. \qquad (4)$$

The third layer calculates the sum and weighted sum of output signals of the second layer. Besides, the activation is performed, the essence of which in the Sugeno algorithm of the 0th order is to define individual rule conclusions: Y_1, Y_2, …, Y_r, …, Y_ρ.

The fourth layer implements the operation of division of output signals of the third layer, i.e. it determines the outcome of the defuzzification procedure, which results in determination of accurate values for the output variable by the centroid method for one-point aggregates as the expression:

$$\tilde{K}_e^*=\frac{\sum\limits_{r=1}^{\rho} Y_r G_r}{\sum\limits_{r=1}^{\rho} G_r}. \qquad (5)$$

To configure parameters of membership functions of input variables and values of individual conclusions of rules neuro-fuzzy system is to be taught to reveal dependence $K_e=f\left(K_a, K_b, K_c, K_d\right)$. For this purpose, through experimental studies, it is needed to collect data for neural self-learning in the form of a matrix:

$$\left\|\begin{matrix} K_1 & K_2 & K_3 & K_4 & K_5 \\ K_2 & K_3 & K_4 & K_5 & K_6 \\ . & . & . & . & . \\ K_{x-4} & K_{x-3} & K_{x-2} & K_{x-1} & K_x \\ . & . & . & . & . \\ K_{X-4} & K_{X-3} & K_{X-2} & K_{X-1} & K_X \end{matrix}\right\|, \qquad (6)$$

where K_x is the measured value of the number of packets directed to the output port of a router during the cycle number x $\left(x\in\left[\overline{1, X}\right]\right)$ at passive management of a respective packet queue.

Data for neural self-learning is obtained using the model of passive packet queue management of the output port of a router of a telecommunication network. This model is created in Simulink environment of computer mathematics system MATLAB [7].

Adequacy of the model is reasonably substantiated by a sufficient convergence of the results of simulation and analytical modeling [8].

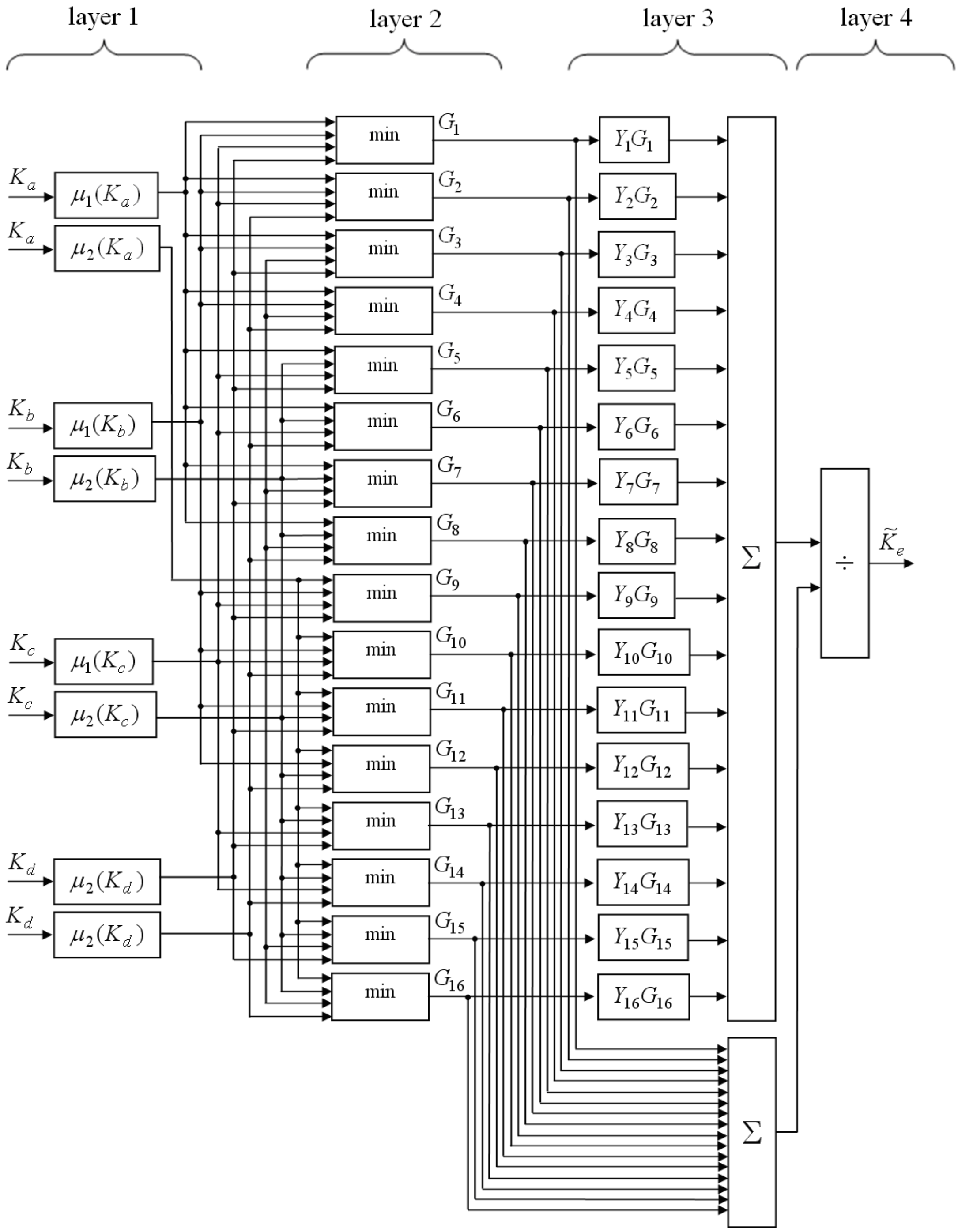

Figure 2: Structure of neuro-fuzzy prediction system of value K_e

As a result of neural self-learning the expressions for the membership functions of input variables are obtained:

$$\mu_1(K_a) = \begin{cases} 1, & if \quad K_a < 0{,}017; \\ \dfrac{9{,}918 - K_a}{9{,}901}, & if \quad 0{,}017 \le K_a \le 9{,}918; \\ 0, & if \quad K_a > 9{,}918; \end{cases} \tag{7}$$

$$\mu_2(K_a) = \begin{cases} 0, & if \quad K_a < 0{,}029; \\ \dfrac{K_a - 0{,}029}{9{,}991}, & if \quad 0{,}029 \le K_a \le 10{,}02; \\ 1, & if \quad K_a > 10{,}02; \end{cases} \tag{8}$$

$$\mu_1(K_b) = \begin{cases} 1, & if \quad K_b < 2{,}06; \\ \dfrac{9{,}904 - K_b}{7{,}844}, & if \quad 2{,}06 \le K_b \le 9{,}904; \\ 0, & if \quad K_b > 9{,}904; \end{cases} \tag{9}$$

$$\mu_2(K_b) = \begin{cases} 0, & if \quad K_b < 2{,}149; \\ \dfrac{K_b - 2{,}149}{7{,}911}, & if \quad 2{,}149 \le K_b \le 10{,}06; \\ 1, & if \quad K_b > 10{,}06; \end{cases} \tag{10}$$

$$\mu_1(K_c) = \begin{cases} 1, & if \quad K_c < 2{,}063; \\ \dfrac{9{,}793 - K_c}{7{,}73}, & if \quad 2{,}063 \le K_c \le 9{,}793; \\ 0, & if \quad K_c > 9{,}793; \end{cases} \tag{11}$$

$$\mu_2(K_c) = \begin{cases} 0, & if \quad K_c < 2{,}17; \\ \dfrac{K_c - 2{,}17}{7{,}89}, & if \quad 2{,}17 \le K_c \le 10{,}06; \\ 1, & if \quad K_c > 10{,}06; \end{cases} \tag{12}$$

$$\mu_1(K_d) = \begin{cases} 1, & if \quad K_d < 1{,}951; \\ \dfrac{9{,}833 - K_d}{7{,}882}, & if \quad 1{,}951 \le K_d \le 9{,}833; \\ 0, & if \quad K_d > 9{,}833; \end{cases} \tag{13}$$

$$\mu_2(K_d) = \begin{cases} 0, & if \quad K_d < 2{,}006; \\ \dfrac{K_c - 2{,}006}{7{,}945}, & if \quad 2{,}006 \le K_d \le 9{,}951; \\ 1, & if \quad K_d > 9{,}951; \end{cases} \tag{14}$$

and also the values of individual rules: $Y_1 = 8{,}452$, $Y_2 = -6$, $Y_3 = 1{,}284$, $Y_4 = 18{,}96$, $Y_5 = 25{,}31$, $Y_6 = 9{,}983$, $Y_7 = 8{,}011$, $Y_8 = -4{,}957$, $Y_9 = 4{,}676$, $Y_{10} = -6{,}303$, $Y_{11} = 0{,}447$, $Y_{12} = 24{,}32$, $Y_{13} = 2{,}996$, $Y_{14} = 21{,}38$, $Y_{15} = -4{,}183$, $Y_{16} = 4{,}737$.

Neuro-fuzzy system for prediction of the value N_f by its construction and operation is similar to neuro-fuzzy prediction system of the value K_e. The main difference of these systems is in output variables. In the first case the output parameter is value $\tilde{N}_f$ and the second one is $\tilde{K}_e$.

The results of simulation modeling

To study the process of premature packet dropping, taking place in accordance with the RED method, a simulation model was used, to the development of which paper [9] is dedicated. In the environment of MATLAB + Simulink a simulation model of neuro-fuzzy active packet queue management of the output port of a router has been developed.

To investigate the packet queue management hundreds of simulation experiments were conducted, which used a variety of incoming traffic data. The effectiveness of the investigated process was assessed with a regard to indices t_d (average delay of packets in the queue) and p_d (probability of dropping a packet directed to the output port of a router).

As a result of the simulation it is revealed that by the employment of neuro-fuzzy active packet queue management the value t_d is reduced by 4% – 10% compared to the employment of RED and by 8% – 24% compared to the employment of Tail Drop.

Implementation of neuro-fuzzy active packet queue management allows obtaining value p_d at a level corresponding to the results of passive packet queue management. Compared to the employment of RED the application of the proposed method can reduce the probability of packet discarding by 5% – 12%.

Conclusion

Neuro-fuzzy systems for active management of packet queue of a router output port of a telecommunication network has been synthesized, the employment of which allows reduction of average packet delay in a queue of a router output port of by 4% – 10% and lowering of the packet probability discard by 5% – 12%.

References

[1] Floyd S. A Report on Some Recent Developments in TCP Congestion Control. IEEE Communications Magazine. April 2001. Available from: http://www.aciri.org/floyd/papers/report_Jan01.pdf.

[2] Braden B., Clark D., Cmwcrofi J., Davie B., Peering S., Estrin D, et al. Recommendations on Queue Management and Congestion Avoidance in the Internet. RFC 2309. April 1998. Available from: http://tools.ietf.org/html/rfc2309

[3] Floyd S. Jacobson V. Random early detection gateways for congestion avoidance, IEEE/ACM Transactions on networking. 1993; 1(4): 397–413.

[4] Castellano G., Castiello C., Fanelli A. M., Jain L. Evolutionary neuro-fuzzy systems and applications. Advances in Evolutionary Computing for System Design, 2007; 66: 11–45

[5] Polschykov K.O. Neuro-fuzzy control method of data sending intensity by sources in mobile radio network of the special setting. Science and technology of the Air Force of Ukraine. 2012; 3(9): 118–122

[6] Takagi T., Sugeno M. Fuzzy identification of systems and its applications to modeling and control. IEEE Transactions on Systems, Man, and Cybernetics, 1985, 15(1): 116–132.

[7] Simulink – Simulation and Model Based Design. Available from: http://www.mathworks.com/products/simulink/index.html?s_cid=BB

[8] Polschykov K.O., Lavrut O.O., Druzhinin S.V. Simulation model of flows control in information network by change of segment interval. Radioelectronic and Computer Systems. 2009; 7(41): 304–8.

[9] Rvachova N. V., Polschykov K. O., Voloshko S. V. The method of choice intersegment interval in the transport protocol of the telecommunications network based on fuzzy system. Problems of Telecommunications. 2011; 2(4): 72–82.