ISSN 2278-3091
Volume 9, No.3, May - June 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse151932020.pdf
https://doi.org/10.30534/ijatcse/2020/151932020

A Novel Neural Network based Model Predictive Controller for Congestion Prevention in IP Networks

[1]Rashmi Baweja, Rajeev Gupta[2]
[1]PhD Scholar, Electronics Engineering Department, Rajasthan Technical University Kota, Rajasthan, India, baweja3rashmi@yahoo.co.in
[2]Professor, Electronics Engineering Department, Rajasthan Technical University Kota, Rajasthan, India, rajeev_eck@yahoo.com

ABSTRACT

Design of a neural network based model predictive controller for UDP(User Datagram Protocol) flow caused congestion, in IP(Internet protocol) networks is proposed in this paper. The objectives of congestion control are prevention of congestion collapse, maximum network bandwidth utilization, TCP-friendliness and smoothness for streaming media applications. Various approaches for controlling congestion in networks are present in the literature. Many of these are make use of network models, which are already identified. In this paper a neural network utilizing Levenberg-Marquardt learning algorithm for on-line identification of non-linear plant(network) model is implemented and combined with a model predictive optimization technique using back tracking line search routine over a specified time horizon. Simulations were carried out to prove the effectiveness of the designed controller. Significant increase in the network bandwidth utilization is also established.

Key words : Congestion Control, IP Networks, Model Predictive Control, Neural Network, System Identification**.**

1. INTRODUCTION

Today's IP networks are progressively being utilized for non-data/multimedia transmissions like voice and gaming/streaming videos, resulting the network flows varying from predominantly short burst type of traffic to the traffic that are time perceptive and of longer interval. The data oriented applications can effectively be controlled using reliable TCP protocol but at the same time the use of UDP for non-data applications does not provide end to end flow and congestion control. Due to the unreliable nature of UDP, it will continue to insert packets in the network despite the consequences of whether the packets are arriving at destination or not which may result in congestion collapse in IP networks. The highly nonlinear and dynamic nature of prevailing IP networks raise the need of more sophisticated and advanced methods of dealing with congestion issues.

Initially, Floyd and Fall [1] identified the high bandwidth TCP-unfriendly flow responsible for congestion in Internet and recognize the need of end to end congestion control mechanisms. Camacho and Bordons [2] in their book, established Model Predictive Control as a powerful controlling technique that does not require complex control algorithms and can be applied for a wide variety of process. Seungwan et al. [3] reviewed the concept of congestion control with a focus on the transmission control protocol/Internet protocol (TCP/IP) via AQM algorithms. Yang et al. [4] proposed an end-to-end transmission control protocol (TCP)-friendly multimedia streaming protocol for wireless Internet, namely WMSTFP, having only last hop as wireless hop. R. Srikant, [5] in his book, explains the basic mathematics necessary for internet Congestion Control. Welzl, M.[6] discussed the issue of traffic management with focus on network congestion control to improve Quality of Service(QOS). Pavlick [7] in his thesis addressed the congestion collapse issue by proposing a new mechanism to come up with a TCP friendly service for voice and video applications. Bartoszewicz [8] addressed the issue of flow control in fast connection-oriented communication networks supporting traffic produced by multiple sources. Rahnamai et al. [9] presented a neural network (NN) model predictive control (MPC) of TCP flows in AQM networks. Mamatas et al. [10] discussed various congestion control algorithms considering the packet networks as black, grey or green box. Rusmin et al. [11] explains internet congestion causes, weakness, and congestion control technique that researchers have been developed. Ignaciuk and Bartoszewicz [12] proposed a flow controller based on sliding-mode concept for congestion control in connection-oriented communication networks which are modelled as discrete-time n th-order systems. Chrysostomou et al.[13] again attempted fuzzy based congestion control in Communication Networks. Nannan et al. [14] analyze and control congestion in diffserv networks using second order sliding mode controllers to solve infinite switching problems. Discrete time Sliding Mode Control(SMC) [15,16,17] and fuzzy based [18,19] congestion control are also attempted effectively for various types of networks. Bazmi and Keshtgary [20] proposed neural network prediction algorithm based congestion control for content-centric networks.

Baweja et al. [21] derived an improved version of congestion avoidance and resource allocation algorithm which is based on the AIMD feedback law for multiple users, transmitting at the same time on a single link. Among the various applications of Model Predictive Control(MPC), Novoselnik et al. [22] used nonlinear MPC for energy efficient housing with modern construction materials. Zhou et al. [23] presented a multi-agent model based predictive control using serial scheme for controlling congestion in large-scale urban traffic networks. Baweja and Gupta [24] proposed design and analysis of fuzzy logic congestion controllers for IP networks. Seder et al. [25] described a receding horizon control (RHC) algorithm for convergent navigation of a differential drive mobile robot and established the advantage of receding horizon control over other controlling approaches. Bazi & Bouchaib [26] compared the prevailing TCP varients for controlling congestion. Africa[27] proposed a neural network based control system for sensor based vehicle traffic control network.

Identifying the research gaps, it is being concluded that soft computing methods are rarely attempted to address the issue. This paper proposes a control mechanism to deal the problem by inclusion of flow and congestion controller based on neural network model predictive control technique for end-to-end streaming oriented protocol. Recently, neural networks have came out to be very useful and effective in identifying and controlling time-varying non-linear systems. Multilayer perceptron neural network (a type of feedforward artificial neural network) exhibits universal approximation features that make it suitable for modelling nonlinear and dynamic systems like IP networks and for implementing general-purpose nonlinear congestion controllers. The proposed control mechanism successfully achieve important objectives of preventing congestion collapse, providing smoothness for streaming media applications, acting TCP friendly and effective in network bandwidth utilization.

2. NETWORK MODEL

The model is built up using MATLAB Simulink software, it constitute essential components to simulate the NNMPC controller inside a defined network situation and is built up to prove the independent behaviour of TCP and controlled UDP data flow in similar conditions. It performs the test for congestion collapse, network bandwidth utilization and smoothness.In this model, NNMPC controlled UDP data flow and independent TCP flow, both run adjacent to an extremely varying bandwidth with a changing rate of ±5.9 Mbps. The imitated bandwidth of the link is taken as 100 Mbps. Thus, accessible bandwidth can vary in the range of 0 to 100 Mbps. The drop rate of packets is taken in the range of 0 to 5%. This model incorporate an application provided constant requested rate, an available bandwidth is also present which is taken as a MATLAB provided sinusoidal waveform customized by multiplying it with a constant to make it variable ranging from 0 to 100 Mbps. Although it differs from an actual network traffic model, however, it is used to represent an extremely fluctuating available bandwidth.

All the inputs are first normalized before being submitted to the NNMPC. A varying drop rate input is estimated based on the varying available bandwidth. A high accessible bandwidth generates a low drop rate and vice-versa. TCP response is generated using the drop rate and then normalized prior for giving as input to the NNMPC. Another variable i.e. the change rate is estimated as a rate of the change in output send rate in unit time and used as a feed-back input signal for NNMPC. Now, the NNMPC has all three necessary inputs, it provide the output in the form of a send rate signal (range 0 to 1) which is further multiplied with the un-normalized available bandwidth signal. The result of simulation graphically display all three inputs and the output.

3. CONTROLLER DESIGN

In this section, the neural network predictive controller which is implemented in Neural Network Toolbox[28] utilize the network model of IP network built up in section-2, as a nonlinear plant model to estimate future network performance. Designed controller then determine the control input to optimize network performance for a particular upcoming time period. System identification is the primary step in model predictive control which is used to find out the model of plant based on neural network. Whereas the next stage utilize this plant model to predict future plant behaviour.

3.1 System Identification

This design step generates a trained neural network which exactly characterize the forward dynamics of the plant. A neural network training signal is obtained in the form of prediction error between the plant output and the neural network output. This process of system identification is illustrated in Figure 1[28] and the layered structure of neural network based plant model is shown in Figure 2[28].

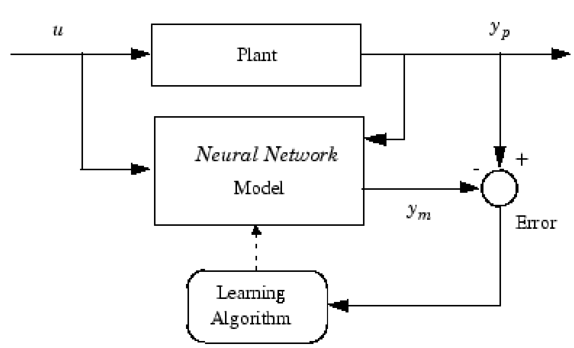

Figure 1: Block diagram of system identification process

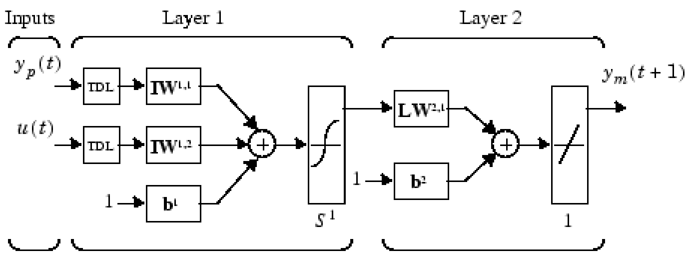

Figure 2: Layered neural network structure of plant model

Offline training of designed network is done in batch mode with the data obtained from the plant operation using trainlm training algorithm. Trainlm is a Matlab inbuilt network training function that modify weights and values of bias, using Levenberg-Marquardt optimization technique. Neural network plant model is designed by adjusting network architecture parameters as mentioned in Table 1 and the designed plant model is trained using the training data as given in Table 2. Applying these parameters along with a series of random step inputs generates training data for the simulating the plant model. Generated training data is analysed and accepted to create the final neural network plant model.

Table 1: Network architecture parameters

S.No	Parameters	Assigned value
1	No. of hidden layers	5
2	Sampling Interval in sec	0.2
3	Delayed plant inputs	2
4	Delayed plant outputs	2

Table 2: Training data / parameters

S. No.	Parameter	Assigned value
1	No. of training samples	200
2	Maximum plant input	100
3	Minimum plant input	0
4	Maximum plant output	100
5	Minimum plant output	0
6	No. of training epoches	1000
7	Training function	trainlm

After the training is complete, the response of the resulting plant model is displayed, as separate plots for testing, validation and training(Figure 3, Figure 4, Figure 5).

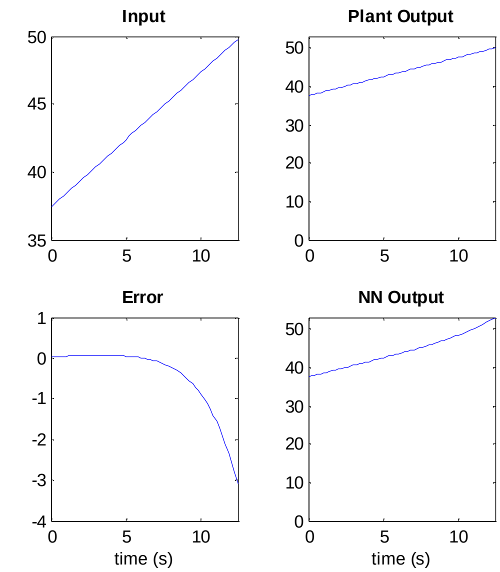

Figure 3: Testing data for NN Predictive Control

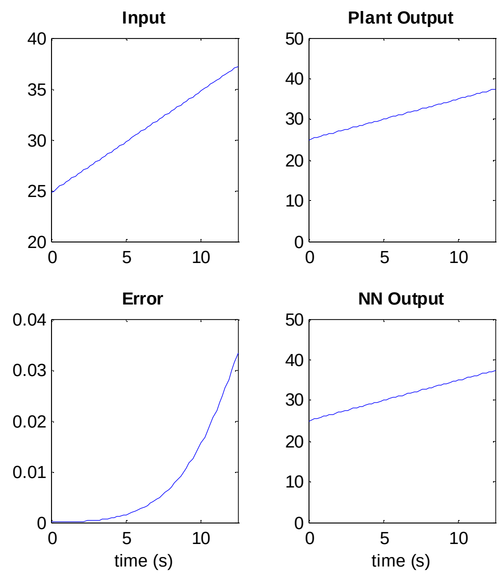

Figure 4: Validation data for NN Predictive Control

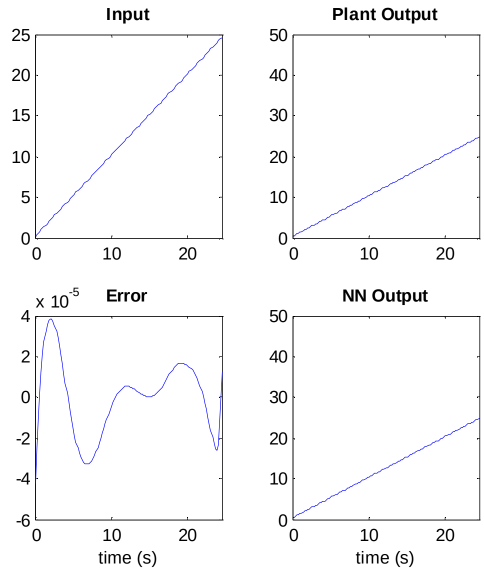

Figure 5: Training Data for NN Predictive Control

3.2 Control Design

In control design stage for model predictive control, the neural network plant model designed in first stage is used to predict future output of the plant. Based on the receding horizon technique, an optimization algorithm is used to opt for the control input that optimizes future performance. The neural network model anticipate plant response for a particular time duration. Anticipated plant response is used by a numerical optimization algorithm to calculate the control signal that minimizes the performance criterion given in (1) for a particular time period:

$$J = \sum_{j=N_1}^{N_2} (y_r(t+j) - y_m(t+j))^2 + \rho \sum_{j+1}^{N_u} (u'(t+j-1) - u'(t+j-2)) \quad (1)$$

here N_1, N_2, and N_u are the horizons for which the tracking error and the control increments are determined. The u′ variable is the provisional control signal, y_r is the desired output response, and y_m is the network model output response. The value of ρ that is the control weighing factor determines contribution of the sum of squares of the control increments over the performance index.

Figure 6[28] shows the complete block diagram of controller for model predictive control process. It consists of both, the neural network plant model and the optimization block. The optimization block calculates the values of u′ that minimize J, and then the optimal u is taken as input to the plant. Minimization is achieved using backtracking line search routine (srchbac) which is well-matched to use with the quasi-Newton optimization algorithms[28]. It starts with a step multiplier of 1 and then backtracks until a reasonable decrease in performance criteria(J) is achieved. Initially, it utilize performance value at the current point with a step multiplier of 1. The optimization block also does quadratic approximation of the performance along with the search direction by using the value of derivative of performance at the current point. The lowest value of quadratic approximation will be the new provisional optimum point (under specific conditions) and at this point performance is checked. In case the performance criteria(J) is not adequately decreased, a cubical interpolation is generated and the least value of the cubical interpolation will be the new provisional optimum point. Similar process is repeated until the required decrease in the performance criteria(J) is achieved. The neural network plant model generated using training data from first stage, is loaded into NNPC block of design window. The controller is designed by applying the control parameter values as given in Table 3.

Table 3: Control Parameters

S.No.	Control Parameters	Assigned values
1.	Cost Horizon	7
2.	Control weighing factor	0.05
3.	Control Horizon	2
4.	Search Parameter	0.001
5.	Minimization routine	srchbac
6.	Iterations/sample time	2

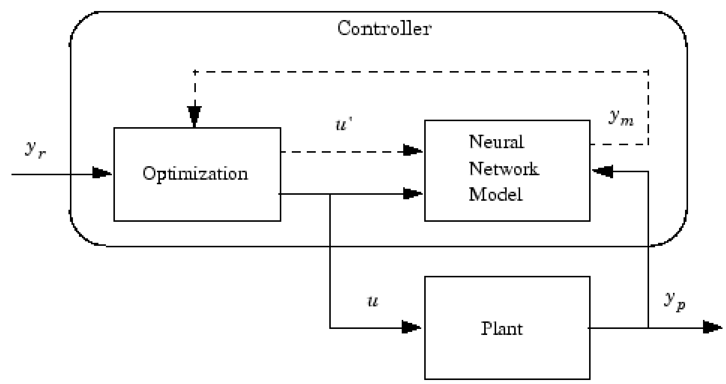

Figure 6: Block diagram of controller

In order to control the congestion caused by UDP data in dedicated IP network model, general input(pre-processed/ normalized) and output variables(post processed/ scaled) of the designed controller are defined by the network conditions as follows:

Inputs:

1. availableBW – Available Bandwidth input range is from 0 to 1. It is normalized as a percentage of the application provided requested rate.

2. tcpResponse –TCP response input range is also from 0 to 1. It is also normalized as a percentage of the application provided requested rate. The TCP Response function is determined using expression given by Floyd and Fall [1] used as the function, given in (2), where p is the packet loss rate, B is the packet size and R is the round trip time:

$$\frac{\sqrt{\frac{3}{2}}(B)}{R*\sqrt{P}} \tag{2}$$

3. changeRate – Change rate input range is -1 to 1and it is also normalized as a percentage of the application provided requested rate.

Output:

1. sendRate – Send rate output range is from 0 to 1 and it is multiplied by the un-normalized availableBW input before being taken as output by the application.

4. SIMULATION RESULTS

The proposed approach is tested using simulink software in MATLAB. The test is performed to assess the designed controller's congestion control, network bandwidth utilization and smoothness of output response. Figure 7 displays the controller output response with a fast changing available bandwidth input and simultaneously display the independently running TCP flow for the same available bandwidth. The application provided requested rate is assumed as 40 Mbps for the testing purpose. The TCP response is not fast enough to respond changes in available rate whereas immediately to react to congestion even in the presence of a large amount of available bandwidth. For multi-media or non-data applications, the rate fluctuations will have major effects on the user's required quality of service and should be smoothened out by the solution . Consequently, the controlled UDP send rate output obtained from the designed controller is more smoother and strictly follow the bandwidth available in the network and utilizes the network bandwidth much more effectively than TCP. For an available bandwidth of 40 Mbps for 50 seconds, ideally a total of 2000 Mb should have been delivered. In the proposed approach, it is evaluated that the designed NNMPC controlled UDP source deliver 1564 Mb that is 78.2% of available bandwidth utilization is achieved and the TCP Response is estimated at 320 Mb which is16% of available bandwidth utilization. It shows reported increase in network bandwidth utilization as compared to first order Sugeno type fuzzy logic controllers in the similar network conditions[24]. Most significantly, the send rate always remain less than the available bandwidth estimation. Hence, if estimation of available bandwidth is precise and well-timed, the proposed controller will never add any undeliverable packets into the network and completely prevent network congestion collapse. Figure 8 display result of another simulation based testing of designed controller for application provided requested rate of 60 Mbps and all other parameters remain unchanged.

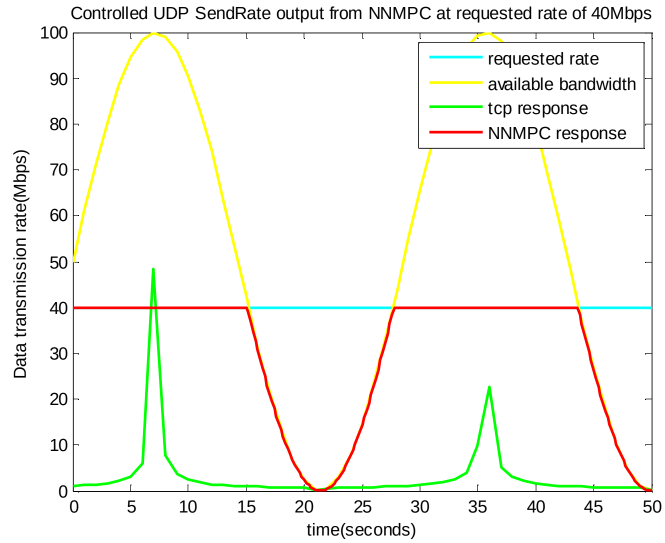

Figure 7: NNMPC based UDP controller output response at 40Mbps requested rate

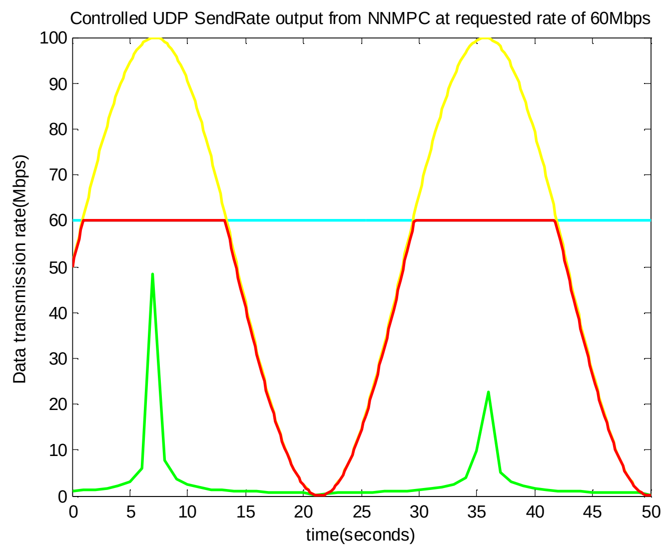

Figure 8: NNMPC based UDP controller output response at 60Mbps requested rate

5. CONCLUSION

In this paper a NNMPC controller design is presented and analysed for providing flow control based congestion control transport layer protocol that works well for non-data/multi-media applications. The proposed neural network based model predictive controller is designed to solve vital issues in the IP networks, such as congestion collapse, TCP-friendliness and smooth output response. With the help of simulations, it is proved that the controller completely eliminates the risk of congestion collapse from undelivered packets and simultaneously utilizes the network better than TCP by improving network bandwidth utilization to 78.2%. The designed NNMPC controller reacts to fluctuations in network conditions smoother than the AIMD mechanism used for TCP congestion control, simultaneously remain independent and friendly with the TCP flows. The designed controller is also proved to be accurate and appropriately react to fast variations in the network bandwidth.

ACKNOWLEDGEMENT

The author thankfully acknowledge department of Electronics Engineering, Rajasthan Technical University, Kota, for providing research opportunity.

REFERENCES

1. S. Floyd and K. Fall. **Promoting the use of end-to-end congestion control in the Internet**, in *IEEE/ACM Transactions on Networking*, vol. 7, no. 4, pp. 458-472, Aug. 1999, doi: 10.1109/90.793002.
2. E.F. Camacho and C. Bordons, ***Model Predictive Control***. London: Springer-Verlag, 1999. https://doi.org/10.1007/978-1-4471-3398-8
3. S. Ryu, C. Rump and C. Qiao. **Advances in internet congestion control**, in IEEE Communications Surveys & Tutorials, vol. 5, no. 1, pp. 28-39, Third Quarter 2003, doi: 10.1109/COMST.2003.5342228.
4. Fan Yang, Qian Zhang, Wenwu Zhu and Ya-Qin Zhang. **End-to-end TCP-friendly streaming protocol and bit allocation for scalable video over wireless Internet**, *in IEEE Journal on Selected Areas in Communications*, vol. 22, no. 4, pp. 777-790, May 2004, doi: 10.1109/JSAC.2004.826008.
5. R. Srikant. ***The Mathematics of Internet Congestion Control (Systems and Control: Foundations and Applications)***. London: Springer Verlag, 2004. https://doi.org/10.1007/978-0-8176-8216-3
6. M. Welzl. ***Network Congestion Control: Managing Internet Traffic***, New Jersy: John Wiley & Sons, 2005.
7. B. Pavlick. **A Fuzzy Logic based controller to provide end-to-end Congestion Control in Streaming Media Applications**, M.S. Dissertation, University of South Florida, Tampa, FL, 2005.
8. A. Bartoszewicz. **Nonlinear flow control strategies for connection-oriented communication networks**, *IEE Proceedings on Control Theory and Applications*, vol. 153, no. 1, pp. 21-28, 2006. https://doi.org/10.1049/ip-cta:20045247
9. K. Rahnamai, K. Gorman and A. Gray. **Model Predictive Neural Control of TCP Flow in AQM Network**, *NAFIPS 2006 - 2006 Annual Meeting of the North American Fuzzy Information Processing Society*, Montreal, Que., 2006, pp. 493-498, doi: 10.1109/NAFIPS.2006.365459.
10. L. Mamatas, T. Harks and V. Tsaoussidis. **Approaches to Congestion Control in Packet Networks**, *Journal of Internet Engineering*, vol. 1, pp. 22-33, 2007.
11. P.H. Rusmin, C. Machbub, A.Harsoyo and D. Hendrawan. **Internet Congestion Control System**, *Makara Journal of Technology*, vol. 12, no.1, pp.19-26, 2010.
12. P. Ignaciuk and A. Bartoszewicz. **Linear Quadratic Optimal Discrete-Time Sliding-Mode Controller for Connection-Oriented Communication Networks**, in *IEEE Transactions on Industrial Electronics*, vol. 55, no. 11, pp. 4013-4021, Nov. 2008, doi: 10.1109/TIE.2008.921464.
13. C. Chrysostomos and A. Pitsillides. **Fuzzy Logic Control in Communication Networks**, *Studies in Computational Intelligence*, vol. 202, no.5, pp. 197-236, 2009. doi./10.1007/978-3-642-01533-5_8.
14. Nannan Zhang, Yuanwei Jing, Yucheng Zhou and Siying Zhang. **Sliding mode congestion control for DiffServ Networks**, *2009 American Control Conference, St. Louis, MO, 2009*, pp. 3314-3318, doi: 10.1109/ACC.2009.5160278.
15. P. Baburaj and B. Bandyopadhyay. **Sliding mode based AQM for robust control of IP based network**, *2011 11th International Conference on Control, Automation and Systems*, Gyeonggi-do, 2011, pp. 149-154.
16. R. C. Joshi and V. K. Thakar. **Congestion control in communication networks using discrete sliding mode control**, *Proceedings of the 31st Chinese Control Conference*, Hefei, 2012, pp. 5553-5557.
17. P. Baburaj and B. Bandyopadhyay. **Discrete-time integral sliding-mode flow control for connection-oriented communication networks**, *2012 12th International Conference on Control Automation Robotics & Vision (ICARCV)*, Guangzhou, 2012, pp. 205-210, doi: 10.1109/ICARCV.2012.6485159.
18. W. Chang, P. Chen and Yang Cheng. **Robust Fuzzy Congestion Control of TCP/AQM Router via Perturbed Takagi-Sugeno Fuzzy Models**, *International Journal of Fuzzy Systems*, vol. 15, no.6, pp. 203-213, 2013.
19. P.H. Rusmin. **Internet Congestion Control Using Fuzzy Integral Controller**, *International Journal on Electrical Engineering and Informatics*, vol. 6, no.9, pp. 497-510, 2014, doi: 10.15676/ijeei.2014.6.3.4.
20. P. Bazmi and M. Keshtgary. **A Neural Network based Congestion Control Algorithm for Content-Centric Networks**, *Journal of Advanced Computer Sci. and Tech.*, vol. 3, pp. 214-220, 2014, doi: 10.14419/jacst.v3i2.3696.
21. R. Baweja, R. Gupta and N. K. Bhagat. **Improved congestion avoidance and resource allocation algorithm**, *2014 2nd International Conference on Emerging Technology Trends in Electronics, Communication and Networking*, Surat, 2014, pp. 1-5, doi: 10.1109/ET2ECN.2014.7044991.
22. B. Novoselnik, J. Ćesić, M. Baotić and I. Petrović. **Nonlinear model predictive control for energy efficient housing with modern construction materials,** *2015 IEEE Sensors Applications Symposium (SAS)*, Zadar, 2015, pp. 1-6, doi: 10.1109/SAS.2015.7133614.
23. Z. Zhou, B.D. Schutter, S. Lin and Xi Y. **Multi-Agent Model based Predictive Control for Large-Scale Urban Traffic Networks Using a Serial Scheme**, *IET Control Theory Appl.*, vol. 9, pp. 475–484, 2015,

doi: 10.1049/iet-cta.2014.0490.

24. R. Baweja and R. Gupta. **Design and Analysis of Fuzzy Logic Congestion Controllers for Networks with both TCP and UDP Data Flow**, *International Journal of Ciência e Técnica Vitivinícola*, vol. 32, pp. 20-32, 2017.
25. M. Seder, M. Baotić and I. Petrović. **Receding Horizon Control for Convergent Navigation of a Differential Drive Mobile Robot**, *in IEEE Transactions on Control Systems Technology*, vol. 25, no. 2, pp. 653-660, March 2017, doi: 10.1109/TCST.2016.2558479.
26. k. Bazi & N. Bouchaib. **Comparative study of TCP congestion control algorithms**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 6, pp. 3560-3564, 2019, doi: 10.30534/ijatcse/2019/137862019.
27. A. Africa. **Sensor-based Traffic Control Network with Neural Network Based Control System**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8 no.4, pp. 983-989, 2019, doi: 10.30534/ijatcse/2019/01842019.
28. MATLAB and **Neural Network Toolbox Release 2012b**. The MathWorks Inc., Natick, Massachusetts, United States.