Journal of Computer and Electronic Sciences

Journal of Computer and Electronic Sciences
Available online at jces.blue-ap.org
©2019 JCES Journal. Vol. 5(1), pp. 1-8, 20 May, 2019

The phenomenon of accumulation in computer networks and congestion control algorithms

Amin Yousefipoor*, Bijan Saedi, Barat Ebrahimi

Faculty Member of Khatam University
Corresponding Author: Amin Yousefipoor

Received: 08 March, 2019 *Accepted: 27 April, 2019* *Published: 20 May, 2019*

ABSTRACT

The accumulation phenomenon has been one of the major problems in computer networks in recent years, which is evident in the need to control this phenomenon in view of the growing Internet and its growing pace.Traffic on computer networks occurs when the amount of data input to a node in a network is greater than the output of that node. So far, there are many ways to control or prevent the phenomena of accumulation in computer networks that are mostly intuitive.These methods are examined from two different perspectives: in the first method, controllers are in the transmitter and the rest of the network is considered as a system in this approach. In the second method, the controller is in the router and the rest of the network is considered as a system. In this study, we present a model for predicting communications that occur in the next steps, and in this way, parts of the network with a high potential for congestion are already identified, and thus we can have better management. Consequently, by preventing, we will increase the loss of data due to network overcrowding.

Keywords: *traffic - computer networks - congestion - congestion control.*

©2019 JCES Journal All rights reserved.

INTRODUCTION

With the increasing use of the Internet network, the problem of congestion in computer networks has become more serious and has found much research. To prevent or control congestion, traffic sources set their send rates based on congestion indicators, and in routers, the active management algorithm sets up the congestion indicator queue based on the congestion level in the network. Along with the main purpose of controlling congestion, delaying and delaying vibration, high efficiency, response speed and simplicity in implementation and computation are introduced as important criteria for the efficiency of active queuing algorithms. Here, we first read about the congestion and how to control it in computer networks. These studies are mostly aimed at obtaining an overview of the congestion in the network and the solutions proposed to deal with it.

Congestion and congestion control in computer networks

Traffic congestion is One of the biggest problems that today's Internet users are facing. Controlling the flow and congestion in the Internet is considered since its inception, but it has become more important becoming more serious in various areas and their dimensions are expanded as Internet applications.The limited bandwidth of the output lines of a router, a limited number of packets can be removed from a router at a time unit, which is called the transmission link capacity. In fact, these limitations are the point where the phenomenon of congestion occurs in computer networks. If the arrival rate packet to the router is faster than the routing processing and exit rate, the queue length in the router begins to increase, and when the queue is filled, new packets will overflow resulting in lossing packet. The sender will attempt to retransmit after receiving no acknowledgment of receipt depending on the destination (receipt of duplicate receipt or expiration of the time period). Which imposes excess load on the network. An obvious solution for resolving the problem is to improve the speed of the processor in the router or to add a buffer length in the router.

Improving the processor speed may be able to improve the density control, but the mismatch of the processor speed and the bandwidth of the router does not improve the density of the network, as well as the advancement of technology and being faster processor speeds and increase the bandwidth of lines, transmitter speeds and bandwidth Routing input lines also increase

simultaneously, and the congestion problem remains. One way to control congestion is preventing congestion. In this way, before the congestion, notify the senders in a way that the congestion is approaching to reduce their rate of delivery in order to avoid congestion. The proposed algorithms for congestion control are generally the same.

Congestion control methods

In a general clasification, we can divide congestion control into two parts:

•Congestion control methods in the transfer layer

•Congestion control methods in the network layer

The two methods described can be called the control methods in the source and the control methods in the routers, respectively.

Congestion control and queue management algorithms

In the congestion control discussion, the removal algorithm from the end is one of the first ways to manage queues in routers, which is still widely used for its simplicity and ease of use. Although this approach seemed to be unnecessary at first when Internet traffic was not considered, the increase in traffic load caused problems such as synchronizing traffic flows, removing packets, swinging queues and filling queues[10]. AQM, by disassembling or marking the packets in the router's queue, and given the absence of the ACK of the packet, informs the source of the congestion and the sender also reduces the send rate according to the TCP algorithm. In the AQM, the controller is on the router, and the rest of the network (source TCP, network latency and destination dynamics) is considered as a process. (Figure 1-1)

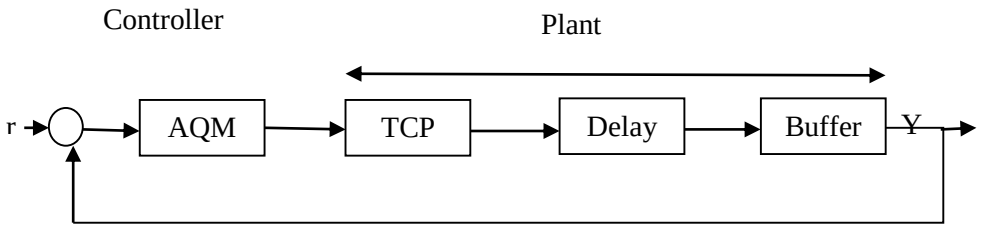


Figure 1- Network by a System Viewpoint with AQM Controller

Some AQM performance evaluation criteria

Through put - Loss Rate - Fairness

Active queuing management algorithms

TCP Accumulation Control Algorithms, while providing convenient services for users, are not capable of providing the right service quality in different traffic conditions in routers. In order to improve the control of the accumulation problem, routing algorithms have been proposed that called as active routing management algorithms. The most basic and obvious method is Drop tail, which is actually a FIFO buffer in the router, in which the new input packets are discarded after the buffer is filled in. And this continues until there is a blank buffer for the packages. Although this method is simple, it can lead to problems. In this way, the buffer function will depend on the length of the buffer. High-capacity buffers result in high latency in delivering packages to destinations, and low capacity buffers result in more packs being discarded, thereby reducing the use of existing resources or lower productivity.

On the other hand, according to the TCP algorithm in the source, discarding the packets after fulling the queue causes Global Synchronization. The Global Synchronization implication is that when large amounts of packets are discarded on the router, senders starting down at the same time and reducing the sending rate, and eventually the send rate will declines and the buffer length in the oscillating mode router will be unstable.

The Necessity of design a new AQM controller (Active Queue Management)

RED and ARED queue management algorithms are methods approved by the IETF and used in practice. These methods have been compared in several papers with methods based on control theory, especially the PI method. The performed studies show the superiority of controls based on control theory.

Given that the network parameters are constantly changing, the big problem of all these methods is that they are sensitive to changes in the network parameters and, as they move away from the nominal point, they find oscillating behavior and tend to become unstable. Rearranging them for new workpoints also requires experimentation and theoretical analysis. In addition, network latency and disturbances caused by non-responsive flows should also be considered in the design of the new controller. Several algorithms for managing the queuing are presented, which briefly explains each of the above algorithms.

The proposed algorithms in AQM

Drop tail: The easiest algorithm used in routers to roll the packets off the bottom of the buffer. Figure 2.1 shows the possibility of disposing packets based on the queue length. The main advantage of this algorithm is its simplicity and applicability to all traffic and its decentralized nature. The problems of this approach include the lack of Fairness in disposing of packets, unacceptable responses to non-responsive currents (non-responsive currents due to lack of feedback, do not cause a change in their rate of delivery after the accumulation of accumulation). As already mentioned, inclusive synchronization Another problem is this method.

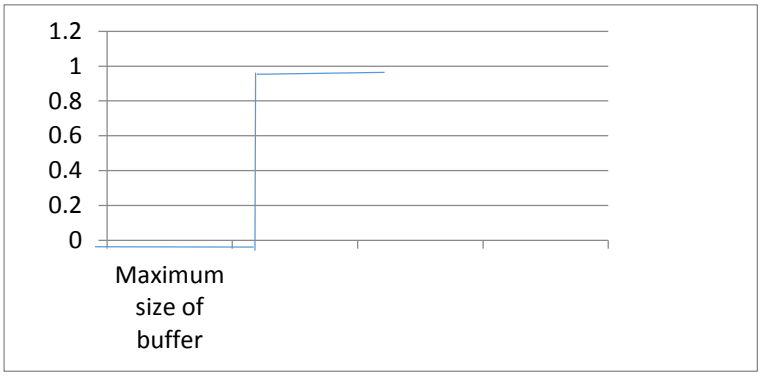


Figure 2- Drop tail algorithm in the router

RED mechanism (Random early detection)

Unlike Tail Drop, which does not provide any queuing management, RED is an active queuing management tool that is currently used in large IP networks. The RED algorithm is determined by calculating the average weighted length of the queue, the probable congestion found in the network, and if the network collapse is detected, the packet is deleted or marked. The main characteristic of the RED algorithm is that the average length of the queue is always kept low and, at the same time, does not collide with momentary explosions and fleeting overcrowding.

•The benefits of the RED algorithm include:

- There is no need to change the network layer protocol for the current hosts of the network.
- Identify the first stages of congestion and react with random disassembly of packets. If the congestion volume is rising, RED packs the packets more intensely to prevent full-fledged queues.
- RED does not wait to complete the pouring of packets until the full line is full,so the queue allows it to accept traffic blast and does not break all packets when the blast occurs. As a result, RED behavior is friendly to TCP streams.
- The queue traffic volume remains in the middle level.

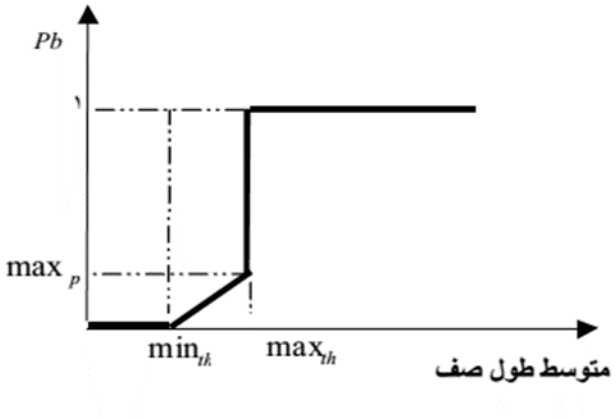


Figure 3- probability of packet dropping in RED

RED restrictions include:

- RED productivity is very sensitive to the selection of its parameters. Inappropriate selection of RED parameters can lead to productivity even lower than Tail Drop. Also, the average length of the RED queue varies depending on the level of collision it encounters.
- Configuring RED to achieve a predictable performance is a complex and difficult task.

FRED mechanism (Flow RED)

This method tries to get all the flows that go through the connection to get their fair share of the connection capacity. To accomplish this goal, this algorithm manages the queue based on each stream. FRED holds statistics for each flow that is in the routing lineup[22].

RIO mechanism (RED in input and output)

In this method, the packets are divided into two In and Out classes, the former representing the packets that are inside the traffic profile, and the latter, the characteristics of the packets that are located outside the profile and violated the traffic agreement. The mechanism used in this algorithm is similar to RED.

ARED mechanism (Adaptive RED)

ARED strives to maintain the average queue length between maxth and minth by setting the parameter max p based on the network traffic load. In this method, based on the intensity of the existing traffic load, if the average length of the queue is greater than maxth, maxp is multiplied by factor α and if the average length of the queue is less than minth, then the factor β decreases. [23].

BLUE mechanism

In order to eliminate the RED defects, a different algorithm* BLUE* was presented [25]. The BLUE Key Idea is the implementation of queue management directly based on the packet loss rate and connection efficiency. BLUE holds out a possibility of disposing in Pm, which is used to mark or disassemble queued logs.

Based on simulations, the BLUE mechanism has the following advantages:

-Need to a Lower buffer
-Ability to implement interactive applications on the Internet
-More memory allocation to high priority packets
-lower Latency

REM mechanism

This technique was presented by Sanjeewa Athuraliya and colleagues in 2001. [26] The main aspects of the REM mechanism are:

This method tries to adapt the user's sending rate to the network capacity while keeping the buffers as empty as possible.

The possibility of drowning / flagging is calculated as the sum of the values of the connections available in the package of the user.

The definition of congestion size and the probability of conveying in REM is different from RED.

Proposed model

In the previous modeling, each input node tended to communicate with the node that had the highest score. In this case, some of the earlier created nodes have more privileges and, therefore, are more likely to be selected [34]. However, time is effective Factor in communication. With the weight on the edges, we consider the time of communication between the nodes. With this method, the nodes that are new to the network are more likely to be chosen for future communications.

In fact, by giving the weights to the edges, we can establish a balance between choosing an older node that has a large degree and a new node that is logged in. In this way, each edge created in the system gives a score based on its time, which gives it the basis for selecting the constituent nodes of that edge. The newer created edge has a higher score, so the probability of selecting the nodes of that edge is higher by the input node. This avoids the accumulation of edges around a certain node, and always nodes that are more active are chosen for the next connection. We can calculate the weight of each edge as:

$$W_i=\frac{\lambda}{t-t_i}$$

To calculate the degree of each node, we apply the points assigned to the edges attached to that node and assign it as a degree to the corresponding node. In this way, the node with the sum of the weight of the edges attached to it is more likely to be chosen for the next connection. This means that the nodes that have recently been involved in communications have a higher degree of relevance than the rest of their relationship, and these nodes are more active and are candidates for new communication. This method can be used to match the selection of a popular old node and a node that is new to the network.

According to studies, congestion in computer networks is a critical issue that affects the quality of service provided to users. There are various ways to predict congestion in the network, each has its own disadvantages and advantages. And our proposed model is a way to predict congestion around a particular router.According to the number of packages that passes through each router, it can be considered as its router grade, and packet sending time is added as a weight to the router, so more routers can be identified at a period of time. It can be assumed that more grade routing is more active and there is a occurrence for congestion, and we can examine the occurrence of congestion in that direction by calculating the degree that each router is in the network. We can identify and control it before the congestion, and thus prevent wasting of network resources. Instantly identifying congestion by sending a message can alert the sender to prevent subsequent packets being sent. In this way, it is no longer

necessary to implement mechanisms to select packets to remove from the overflow queue and it is not necessary to spend time. So, we prevent the occurrence of congestion by anticipation.

Research achievements

The results of this research are important in many aspects. First, this research is one of the real issues in computer networks, and the data used in this research were real.

The main purpose of this research is to examine communications in computer networks and to propose preventing the traffic creation that will cause the no accessing to entire network or a part of it. It also helps identifing and making decisions for communication without facing to traffic in the network and its problems. According to the results of the research, it is clear that the proposed strategy to identify traffic according to the dynamics of networks is a precise, effective and useful solution.

One of the most important achievements of the research is providing a framework for predicting the occurrence of traffic in computer networks that can be used in other computer networks and other similar topics. In the proposed method, taking the time as a factor for communication obtains better results than the Barbacy initial model.

In this research, a process of discovery of knowledge from actual data was designed and implemented to obtain verifiable results. In brief, this process includes preprocessing and data preparation, modeling and processing, and analyzing and interpreting the results. Each of these steps required a great deal of time and accuracy.

An overview of the research method

In this study, in order to prevent congestion in computer networks, a method is proposed to predict the probability of congestion based on the connection time as well as the degree at each node is located. In fact, the basis for choosing to communicate is being active. in the Barbacy initial model, the only degree of nodes was considered as a criterion for selection, which would create a congestion problem around a particular node.

Original database collection Building

To improve the study of the structure and dynamics of network traffic, a large database named Click Dataset was developed by Indian University users. Which includes a set of HTTP requests. The goal is better understanding the behavior of users to create real models of web traffic. Which improves the design of networks, sites and servers.

Data of this database includes

- Time stamp
- Requested URL
- referenced URL
- A logical classification of users
- A logical flag to specify whether the request is internal or requested from the outside.

To use this data, we need to convert them to decent and quality data. We also needed data conversion because of the textual structure of the data. We converted text data into numerical data by performing the operation. There was also a need for a clearing phase because in some communications one of the parties was unclear about the data being removed as noise from database.

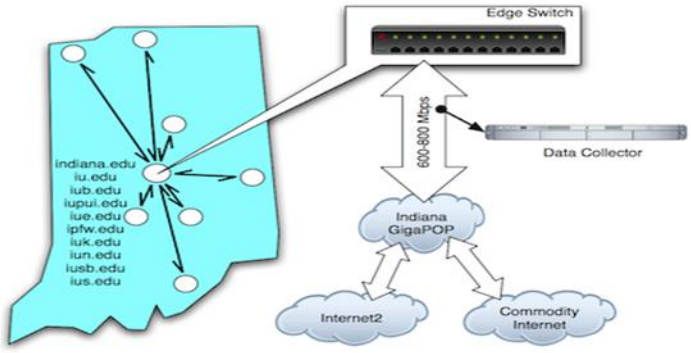


Figure 4-An overview of the data collector location

A small sample of data is as follows.

{"count": 5, "timestamp": 1257066002, "from": "sidereel.com", "to": "static.fmpub.net"}
{"count": 1, "timestamp": 1257066002, "from": "multiupload.com", "to": " golfbox.cn"}
{"count": 30, "timestamp": 1257066002, "from": "aznv.tv", "to": "whos.amung.us"}
{"count": 1, "timestamp": 1257066002, "from": "", "to": "games.sina.com.cn"}

{"count":1,"timestamp":1257066002,"from":"xmradio.com", "to": "xmro.xmradio.com"}
{"count":1,"timestamp":1257066002,"from":"aix1.uottawa.ca","to": "utstat.utoronto.ca"}
{"count":1,"timestamp":1257066002,"from":"supercheats.com","to": "view.atdmt.com"}

Table 1- Converting sample data for the initial time

count	time	from	To
3	1	465	397
1	1	320	154
1	1	56	320
2	1	40	158
12	1	167	484
4	1	32	175

Model Implementation

Implementation was done by using MATLAB. At this point, we wrote a program to help us see a graph of communication, and we used the program to measure the accuracy of the proposed method and the accuracy of the method.

Providing results from research and comparing methods

This research is aimed at improving the initial model of Barabasy in the field of predicting subsequent communications. It is increased due to the fact of congestion around the particular node, which is usually older nodes, and this was a weakness for this model, there were studies conducted to improve this weakness.

Also, given that the congestion problem is one of the problems that has always been on the computer network and many studies have been done to manage it, we decided to give an idea base on the model on the problem of traffic and congestion in a special switch and general congestion challenge.In the model we provided, we were considered a weight for each node according to it, in addition to the degree that each node has, the node arrival time affects the node's weight. In this way, nodes that are more active are always selected for subsequent communication. In this way, we can already predict which nodes are more compact in communicating. In a network, we can consider the switches as the nodes we want to predict subsequent communications. This method is in fact the category of methods to avoid congestion and it is important in this regard. Because the no data is losing and it does not need to be re-sent.

The shape and diagram of testing the model for about 2000 nodes are as follows.

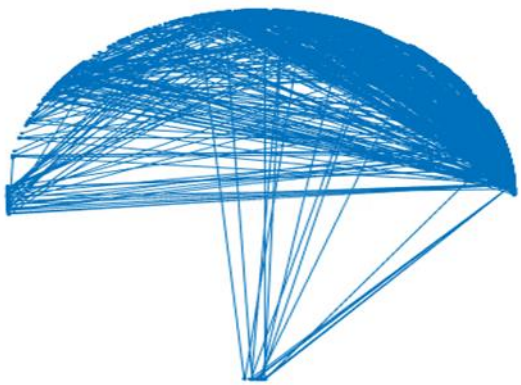


Figure 5-. Modeling form for 2000 nodes

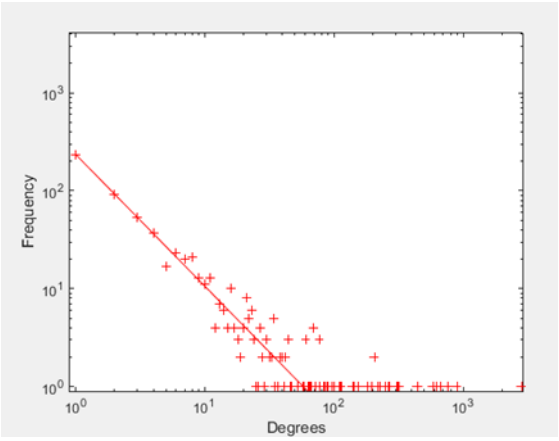


Figure 6- The 2000 Node Modeling Chart

We obtained reasonable results by calculating the accuracy and precision of the model.

For example, we tested our model, which is supposed to predict communications at stage t = 3, thus comparing the number of communications that were predicted with the number of connections that really existed on the network.

In the t = 3 phase, we have 300 connections that we want to predict the same 300 connections using the model. The number of communications that our model predicted was 185, and the number of communications that were correctly identified was 112. You can measure precision according to existing formulas.

37% =Recall= $\frac{112}{300}$

= 60%Precision= $\frac{112}{185}$

Once again, this time, we implement the model by assuming that we want to predict 500 connections. The answer to this is that the 245 model predicts communications, of which there are actually 157 relationships and values such as we calculate beforehand.

= 53% Recall= $\frac{157}{300}$

= 64% Precision= $\frac{157}{245}$

This precision indicates that the model works well and delivers the desired performance to an acceptable level

REFERENCES

[1] C.-Q. Yang and A. Reddy, "A taxonomy for congestion control algorithms in packet switching networks," IEEE Network, vol.9, no.4, pp. 34–45, 1995.

[2] M. Sheela and J. Raja,"An effective mechanism for Congestion Control in High Speed Network," International Journal of Computer Science and Network (IJCSNS), vol.9, no.4, pp.62–70, 2009.

[3] S. Bauer, D. Clark, and W. Lehr, "The evolution of internet congestion," in 37th Research Conference on Communication, Information and Internet Policy, VA, USA, 2009.

[4] C. Hollot, V. Misra, D. Towsley, and W. Gong, "Analysis and design of controllers for AQM routers supporting TCP flows, " IEEE Transactions on Automatic Control, vol.47, pp. 945–959,Jun 2002.

[5] R. Adams, "Active Queue Management: A Survey," IEEE COMMUNICATIONS SURVEYS & TUTORIALS, vol.PP, no.99, pp. 1–52, 2012.

[6] J. Nagle, "RFC-896: Congestion Control in IP/TCP Internetworks," Internet RFCs, 1984.

[7] V. Jacobson, "Congestion avoidance and control," ACM SIGCOMM Computer Communication Review, vol.18, no.4, pp. 314–329, 1988.

[8] A. Afanasyev, N. Tilley, P. Reiher, and L. Kleinrock, "Host-to-Host Congestion Control for TCP," Communications Surveys & Tutorials, IEEE, vol.12, no.3, pp. 304–342, 2010.

[9] M. Zhang, M. Dusi, W. John, and C. Chen, "Analysis of UDP Traffic Usage on Internet
Backbone Links," in Ninth Annual International Symposium on Applications and the Internet (SAINT09), WA, USA, pp. 280–281, Jul 2009.

[10] B. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski, and L. Zhang. "Recommendations on Queue Management and Congestion Avoidance in the Internet". Internet RFCs, 1998.

[11] M. Christiansen, K. Jeffay, D. Ott, and F. D. Smith, "Tuning RED for Web traffic," IEEE/ACM Transactions on Networking, vol.9, no.3, pp. 249–264, 2001.

Average length of the queue

[12] T. Bonald, M. May, and J. C. Bolot, "Analytic evaluation of RED performance," in Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM),vol.3, pp. 1415–1424, Mar 2000.
[13] C. Hollot, V. Misra, D. Towsley, and W. Gong, "Analysis and design of controllers for AQM routers supporting TCP flows," IEEE Transactions on Automatic Control, vol.47, pp. 945–959,Jun 2002.
[14] K. Jong-hwan and Y. Ikjun, "Reducing Queue Oscillation at a Congested Link," IEEE Transactions on Parallel and Distributed Systems, vol.19, no.3, pp. 394–407, 2008.
[15] P. Wang, H. Chen, X. Yang, and Y. Ma, "Design and analysis of a model predictive controller for active queue management," ISA Transactions, vol.51, no.1, pp. 120–131, 2012.
[16] X. Guan, B. Yang, B. Zhao, G. Feng, and C. Chen, "Adaptive fuzzy sliding mode active queue management algorithms," Telecommunication Systems, vol.35, no.1, pp. 21–42, 2007.
[17] P. Yan, Y. Gao, and H. Ozbay, "A variable structure control approach to active queue management for TCP with ECN," IEEE Transactions on Control Systems Technology, vol.13, pp.203–215, Mar 2005.
[18] Q. Chen and O. W. W. Yang, "Robust Controller Design for AQM Router," IEEE Transactions on Automatic Control, vol.52, pp. 938–943, May 2007.
[19] B. Marami and M. Haeri, "Implementation of MPC as an AQM controller," Computer- Communications, vol.33, no.2, pp. 227–239, 2010.
[20] S. Manfredi, M. di Bernardo, and F. Garofalo, "Design, validation and experimental testing of a robust AQM control," Control Engineering Practice, vol.17, no.3, pp. 394–407, 2009.
[21] P.-F. Quet and H. Ozbay, "On the design of AQM supporting TCP flows using robust control theory," IEEE Transactions on Automatic Control, vol.49, pp. 1031–1036, Jun 2004.
[22] W.-J. Kim and B. G. Lee, "FRED: Fair random early detection algorithm for TCP overATM networks," Electronics Letters, vol.34, pp. 152–154, Jan 1998.
[23] Y.-D. Xu, Z.-Y. Wang, and H. Wang, "ARED: a novel adaptive congestion controller," in International Conference on Machine Learning and Cybernetics, vol.2, pp. 708–714, Aug 2005.
[24] P. Yan, Y. Gao, and H. Ozbay, "A variable structure control approach to active queue management for TCP with ECN," IEEE Transactions on Control Systems Technology, vol.13, pp.203–215, Mar 2005.
[25] W. chang Feng, K. Shin, D. Kandlur, and D. Saha, "The BLUE active queue management algorithms," IEEE/ACM Transactions on Networking, vol.10, pp. 513–528, Aug 2002.
[26] S. Athuraliya, S. Low, V. Li, and Q. Yin, "REM: active queue management," IEEE Network,vol.15, no.3, pp. 48–53, 2001.
[27] J. Sun, S. Chan, and M. Zukerman, "IAPI: An intelligent adaptive PI active queue management scheme," Computer Communications, vol.35, no.18, pp. 2281–2293, 2012.
[27]] Z. Na, Q. Guo, Z. Gao, J. Zhen, and C. Wang, "A novel adaptive traffic prediction AQM algorithm," Telecommunication Systems, vol.49, no.1, pp. 149–160, 2012.
[28] C. Zhou, J. He, and Q. Chen, "A robust active queue management scheme for network congestion control," Computers & Electrical Engineering, vol.39, no.2, pp. 285–294, 2013.
[29] Albert-Laszlo Barabasi, Reka Albert, statistical machanics of complex networks,Department of Physics, College of Science, 225 Nieuwland Science Hall, University ofNotre-Dame,Notre-Dame, USA,Elsevier,51pages, 2002.
[30] Tao Jia Albert-Laszlo Barabasi,Control Capacity and A Random Sampling Method inExploring Controllability of Complex Networks, ١Center for Complex Network Researchand Department of Physics,6 pages,2013.
[31] Albert-Laszlo Barabasi Reka Albert, Hawoong Jeong, Mean-field theory for scale-free random networks,Department of Physics, University of Notre-Dame, Notre-Dame, USA,Elsevier,15 pages, 1999.
[32] Albert-Laszlo Barabasi, Reka Albert, Hawoong Jeong, Scale-free characteristics of random networks, Department of Physics, College of Science, 225 Nieuwland Science Hall, University of Notre-Dame,Notre-Dame, USA,Elsevier,9pages,2000.
[33] albert barabasi ,network science(book chapter 5) ,2013.
[34] Gourab Ghoshal, Liping Chi & Albert-Laszlo Barabasi,Uncovering the role of elementary processes in network evolution, 6 page