Clustering and Classification (Time Series analysis) Based Congestion Control algorithm: Data Mining Approach

Manoj Devare and Ajay Kumar

Bharati Vidyapeeth University, Pune(MS), India, Modern College of Engineering, Pune(MS), India

Summary
The congestion controlling in the network is done by the various approaches. Here we have given a new approach to predict the congestion by historical data points. The time series analysis is done for solving the problem. The experimental analysis is done by using actual network test-bed. The data mining approach is used for discovery and inferring the future intensity of the traffic.
Keywords:
prediction, time series analysis, congestion, traffic intensity.

1. Introduction

The congestion is viewed as the resource allocation problem in the network. The congestion controlling, congestion detection, congestion avoidance algorithm is process of checking various factors across various layers of the network. The existing algorithms and approaches used are giving well performance across the various situations in the network. Analysis of the network congestion controlling algorithms is given in following section.

Congestion control and resource allocation are two sides of the same coin. On the one hand, if the network takes a active part in the resource allocation then congestion may be avoided. The congestion control is involves the both hosts and network elements such as routers. When too many packets are contending for the same link, the queue overflows and packets have to be dropped. The resource allocation can be viewed as router centric versus host centric or reservation based versus feedback based or window based versus rate based solution[17].

The TCP uses the Additive Increase Multiplicative Decrease (AIMD) with other schemes such as slow start for avoiding congestion. The concept of MSS and congestion window is used in these algorithms TCP Tahoe and Reno are two variations to the basic AIMD scheme. HighSpeed TCP (HSTCP) is a new congestion control algorithm for TCP protocol. With TCP Vegas, timeouts were set and round-trip delays were measured for every packet in the transmit buffer. In addition, TCP Vegas uses additive increases and additive decreases in the congestion window. [1] TCP New Reno improves retransmission during the fast recovery phase of TCP Reno. During fast recovery, for every duplicate ACK that is returned to TCP New Reno, a duplicate unacknowledged packet from the front of the transmit buffer is sent. In other words, several packets from the front of the transmit buffer may be retransmitted. For every ACK that makes progress in the sequence space, new unsent packets from the back of the transmit buffer are sent, to keep the halved congestion window full.[8][9][10].

TCP Hybla aims to eliminate penalization of TCP connections that incorporate a high-latency terrestrial or satellite radio link, due to their longer round trip times. It stems from an analytical evaluation of the congestion window dynamics, which suggests the necessary modifications to remove the performance dependence on RTT.[5]

Some other congestion avoidance algorithms are

- TCP Westwood
- TCP SACK
- H-TCP
- Scalable TCP
- HS-TCP
- BIC-TCP
- CUBIC TCP
- FAST TCP
- XCP
- LED

TCP Westwood+ is a sender-side only modification of the TCP Reno protocol that optimizes the performance of TCP congestion control over both wireline and wireless. TCP-Illinois is a variant of TCP congestion control protocol. It is especially targeted at high-speed, long-distance networks. A sender side modification to the standard TCP congestion control algorithm, it achieves a higher average throughput than the standard TCP, allocates the network resource fairly as the standard TCP, is compatible with the standard TCP, and provides incentives for TCP users to switch.[4][6][7] Some more papers are observed such as "one more bit is enough" has implemented simple low complexity protocol called variable structure congestion control protocol(VCP)[13]. The congestion removal based algorithm for the heterogeneous users is developed where no explicit feedback is required from network. The algorithm presented is totally window based[14]. Determining a near optimal policies when the available bandwidth is unchanging and when bandwidth is changing in restricted manner under the control of an adversary[15].

Manuscript received September 5, 2007
Manuscript revised September 20, 2007

A novel RED based hop-by-hop congestion control based on packet switched network is developed that is totally based on the coordination of the routers and hosts. [16]. The fairness and efficiency balancing is achieved by protocol design with binomial congestion control protocols is observed in some papers[18]. The TCP friendly congestion control mechanism for many-to-many communication environment targeted unicast WAN and multicast protocol(TFRC) has solved the sort of problem for competing resources[19].

2. Background

In many networks for traffic and telecommunications, minimizing delays from entry to exit is a major concern of users. In user-optimal routing, each user chooses a path minimize delay form entry to exit, given the existing paths chosen by all other users. Under user-optimal routing, at equilibrium all users experience the same delay. Many networks, especially data networks, are commonly modeled as networks of single-server queues. Single-server queuing networks with user-optimal routing in which adding servers or increasing the capacity of existing servers worsen the delay experienced by all users[3]. In continuation to this, the connection oriented protocols requires the virtual circuit connection for the duration of the entire communication of the both hosts.

The statistical data of rising numbers will indicate the congestion occurred. The factor under concentration under the measurement for the such congestion control are as average packet delay, the number of packets that time out and retransmitted standard deviation of packet delay, percentage of all packets discarded for lack of buffer space, average queue length. If we are able to reduce these quantities then it is definitely controlling a congestion in the network.

When users are connected they form a virtual circuit in connection oriented protocol. Some times the clients are idle and not communicating with the server i.e. wastage of the bandwidth resource. The time required to complete the request, duration of the entire communication, will gives the important information regarding communication scenario. While working with the congestion controlling on the homogeneous and heterogeneous networks we have tested the different queuing disciplines and their impact on the congestion control. The work we already presented in the role of markovian queuing discipline in congestion control.[19]

Mean while discussion with many researchers and after survey of the literature we came to the conclusion that how we can verify the results about their consistency and better output with respect to other algorithms? To resolve this query we have tested our simulations across two different simulators. The simulations technique we applied by simulators OMNeT++ and OPNET[20]. This technique we applied by Discrete event simulation of additive increase multiplicative decrease. Still the results on the actual network are always different than the simulated results.

To overcome the flaws in the simulation the work presented here is carried on the actual network. The dynamic(changing) behavior of the network elements may produces the different results.

3. Methodology

For experiment we have selected the five clients and one server. The network is heterogeneous network. Clients and servers are of connection oriented type. The clients and servers are created by using socket programming.. The time of clients requests and servers responses in the given time intervals is measured. The time differences are calculated so that the series shows the behavioral for the future predictions. The log files for recording the times of different events in the network are maintained at clients side and server side.

While experimentation we have choose the computer systems from the VIIT campus network. VIIT campus is well established network with the systems are connected to the server by the switches. So as the packets thrown are passed through the switches. While performing experiments we have created multithreaded server and many clients can connect to it. We maintained the time of events in the network activities. The aim is to avoid irregularities in the connection and unwanted connections for the long time of the span. We are maintaining the database of the different events. The database is used as the data warehouse. By applying the different operations on the data we can predict the future values of the traffic intensity and probability of the congestion in the network. Depending on the predictions the countermeasures can be applied for the future enhancements in the network.

The process of the data collection, selection of the data, preprocessing and transformations in the data, and inferring the knowledge base from the database is shown in figure 1.

Selection: The data needed for data mining process obtained form many the routing tables and times series stored at various hops in the network/subnet. We are obtaining data from different clients and servers in the network. The data collected by our programs that is in proper format, hence no necessary to *preprocess*. Similarly *transformations* for data reduction and reformatting is not necessary as the analyzed data we are storing in the data marts[2].

Data mining: As a part of KDD, we may observe the data mining as the main activity for the future predations about the traffic intensity and in particular period of time. The *interpretation and evaluation* is important because usefulness of results for the congestion control is depends on the same.

figure 1 shows data mining approach where during analysis the time series is observed creating a set of attributes values over period of time. Time series consist of only numeric values at specific, evenly spaced intervals. The values usually are obtained as evenly spaced time plots. There are three basic functions performed in time series analysis. In one case distance measures are used to determine the similarity between different time series. In second case the structure of the line is examined to determine and perhaps classify its behavior. Estimation and prediction may be viewed as types of classification. Prediction can thought of as classifying an attribute value into one of a set of possible classes. It is often viewed as forecasting a continuous value, while classification forecasts a discrete value.

All approaches to perform classification assume some knowledge of the data. Often a training set is used to develop the specific parameters required by the technique. Training data consist of sample input data as well as the classification assignment for the data. Domain experts may also be used to assist in the process.[2]

The classification can be defined as given database D={t1,t2,…..tn} of tuples(items, records) and a set of classes C={C1,C2…..Cm}, the classification problem is to define a mapping f : D→C where each *ti* is assigned to one class. A class C*j* , contains precisely those tuples mapped to it; that is Cj={ti │ *f*(ti)=Cj, 1 ≤ i ≤ n, and ti ε D}. The mapping from the database to the set of classes where each tuple in the database is assigned to exactly one class.

The classification algorithms are available with statistical based, distance based, decision tree based, neural network based, rule based and combined techniques. While looking in deep for the clustering technique the hierarchical algorithms, partition algorithms, clustering large databases are available for the predictions.[2]

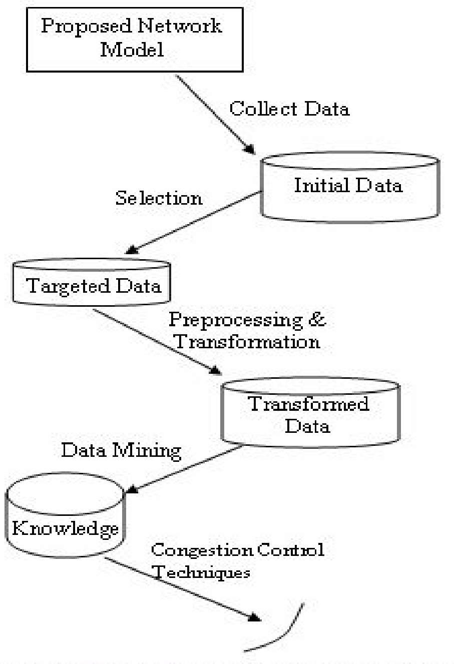


Fig.1 -Data Mining(KDD) in Congestion Control

A third application would be to use the historical time series plots to predict future values. The time series data may be continuous or discrete. Time series prediction is the use of model to predict future events based on known past events. For investigation purpose a time series is considered as consisting only numeric values. After finding out similarity of the time series they classified or clustering. Given several time series we may want to determine which time series are like each other(clustering). Alternatively, we may given a time series to find which time series from a set are like this one(classification). A special type of similarity analysis is identifying patterns within time series.

Here we have created two variants of the same algorithm. In first algorithm we have used the one separate process/thread for collecting data in the data mart and another process/thread to refine data to create knowledge base. These both of the threads are running one after another.

In second algorithm we have proposed the thread model for both activities running simultaneously and to view the results to enhance network performance and remove congestion. The comparative results are showing compromise between resource allocation at operating system level and speed/performance in decision making.

4. Algorithm

1. Generate traffic patterns for 'n' number of clients at different time. Store the time of connection for whole communication. As well as store starting time and end time for connection (i.e. used for time series analysis.)
2. Create routing table with different hops in the path of destination.
3. Analyze the data stored in the *data mart* (here we have considered the information stored at the 'deflector' and time series as part of data mart). Create knowledge base for further congestion controlling algorithm. The clustering and/or classification can be done by using this approach.
4. The information is used for the decision making and predictions for the future traffic intensity. Apply counter measures for the congestion control in high traffic time spans.

Here the 'deflector' is 'cute young deflector' in the initial stages. As time passes the information form the data marts can be refined and used to become 'smarten the deflector'. The cuteness and smartness depends on the knowledge present in the knowledge base. "More knowledge more the smartness". More refinements on the database tuples will gives more useful knowledge for he future enhancements of the predictions by the classification and clustering of the time series. The control measures are applied to prevent congestion and proper balance of resource allocation by fairness criteria can be done for the further refinements in the criteria. The 'n' iterations of the above algorithm can gives more and more knowledge in the form of tuples.

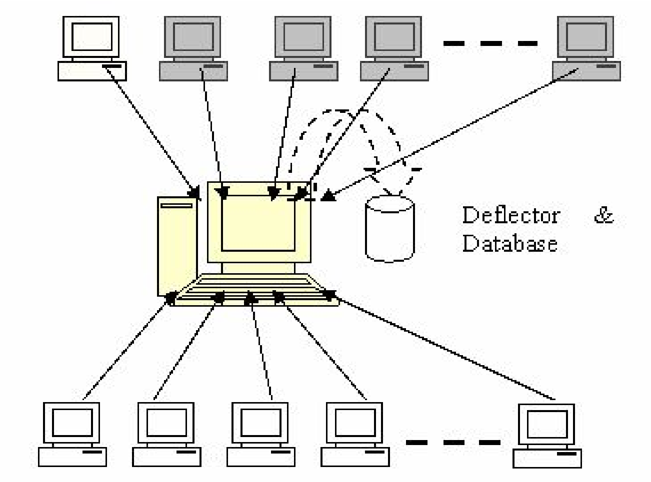


Fig.2. The Model with topology.

5. Experimental Analysis and Results

Table 1. time shown as data from log files

Server Starting Time : 10:47:40 IP-Address : 172.16.1.16					
IP Clients	New Request Received @ server	Client started	Sub Request Received @server	Request from client	End time
(1)	(2)	(3)	(4)	(5)	(6)
172.16.1.17	10:49:52	10:49:52	10:49:57	10:49:59	10:50:03
172.16.1.18	11:04:42	11:04:42	11:04:45	11:07:31	11:07:39
172.16.1.21	11:20:48	11:20:48	11:20:54	11:23:40	11:23:48
172.16.1.19	11:24:52	11:24:52	11:24:58	11:26:26	11:26:34
172.16.1.14	11:35:13	11:35:13	11:35:16	11:35:59	11:36:06

Table 2. Transformed data of time series from table1.

Server starting time : 10:47:40 IP-Address : 172.16.1.16							
IP Clients							
(1)	(4-3) sec	(5-3) sec	(5-4) Sec	(6-3) sec	(6-4) sec	(6-5) sec	(6-2) Sec
172.16.1.17	05	07	02	11	06	04	11
172.16.1.18	03	169	166	167	174	09	177
172.16.1.21	06	172	166	180	174	08	180
172.16.1.19	06	94	88	102	96	08	102
172.16.1.14	03	46	43	53	50	07	53

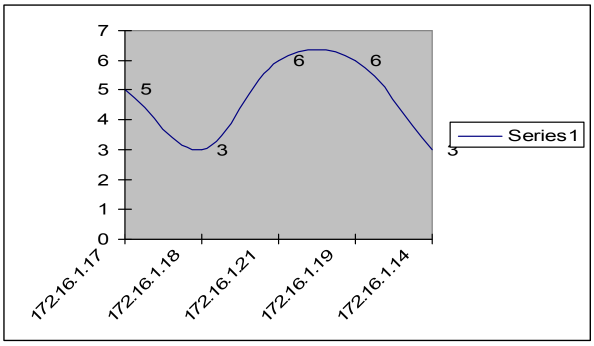


Fig. 3-Time series of (4-3) sec
The graph shown (4-3) column is the time elapsed between connection established with server and first request came to the server by that client.

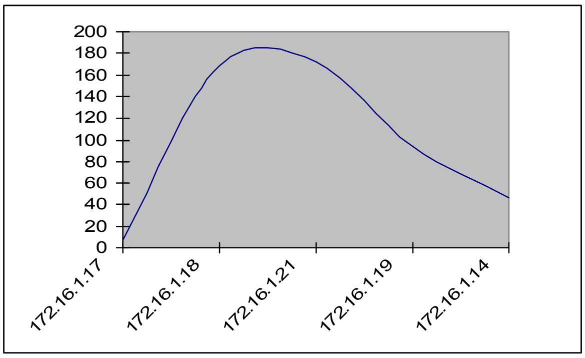


Fig 4-Time series of (5-3) sec
The graph shown(5-3) above is the time span between clients started and the first request arrived at the server. This time series is indication to the time of wastage of the resources.

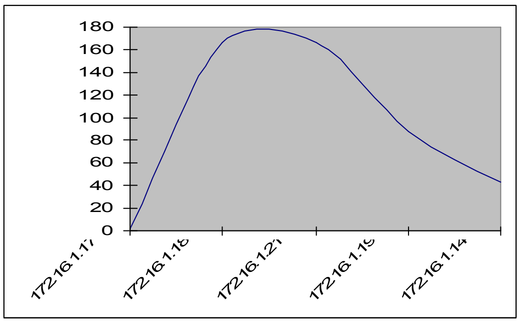


Fig. 5 Time series of (5-4) sec
This time series (5-4) is time elapsed between time of two consecutive requests.

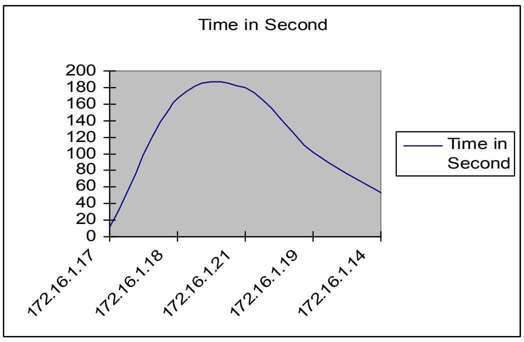


Fig.6 Time series of (6-3) sec
Time elapsed between client started to the request satisfied

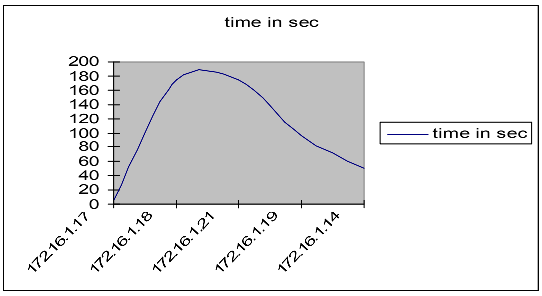
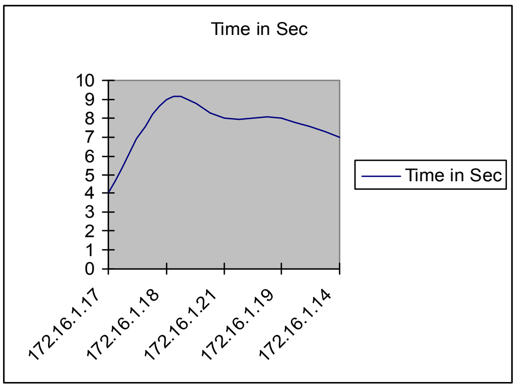
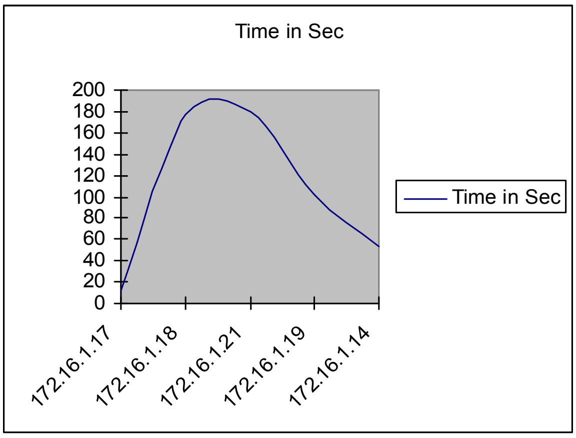


Fig. 7. Time series of (6-4) sec
Graph shown (6-4) is time elapsed between client started to the request satisfied.



Time series of (6-5) sec

Graph shown above(6-5) is time required to transfer the file from server to the client. This is actual time of transfer.



Time series of (6-2) sec
The graph shown above(6-2) is the total time elapsed from connection established to the sub request satisfied.

6. Conclusion

Data mining approach used in the congestion is for clustering the time series is useful for the future predictions. After adding more clients with time series as the output of the data is used for the condition after some time span. Association/pattern matching rules to optimize the network performance. After applying one of the algorithm on the data of log files of client and server.(user log files as data warehouse). Use the KDD(Knowledge Discovery over the Database) technique and improve the performance of the network.

Future work

The work carried is measuring the time in seconds. If the time is measured in milliseconds then it may produces the better and useful information for the connections and relatively for congestion control. For comparative study of the algorithms the simulations may be done in future.

Appendix

(Contents of auto generated log files)
Sat Mar 31 10:47:40 GMT+05:30 2007
ServerSocket[addr=0.0.0.0/0.0.0.0,port=0,localport=8189]
Client1
SocketSocket[addr=/172.16.1.17,port=1182,localport=8189]
New Request Received :Sat Mar 31 10:49:52 GMT+05:30 2007
Sat Mar 31 10:49:52 GMT+05:30 2007
Sub Request Received :Sat Mar 31 10:49:52 GMT+05:30 2007
file size in bytes:2411
Sub Request Received :Sat Mar 31 10:49:53 GMT+05:30 2007
Client2
SocketSocket[addr=/172.16.1.18,port=1197,localport=8189]
New Request Received :Sat Mar 31 11:04:42 GMT+05:30 2007
Sat Mar 31 11:04:42 GMT+05:30 2007
Sub Request Received :Sat Mar 31 11:04:42 GMT+05:30 2007
file size in bytes:2411
Sub Request Received :Sat Mar 31 11:07:28 GMT+05:30 2007
Client3
SocketSocket[addr=/172.16.1.21,port=1133,localport=8189]
New Request Received :Sat Mar 31 11:20:48 GMT+05:30 2007
Sat Mar 31 11:20:48 GMT+05:30 2007
Sub Request Received :Sat Mar 31 11:20:48 GMT+05:30 2007
file size in bytes:2411
Sub Request Received :Sat Mar 31 11:23:35 GMT+05:30 2007
Client4
SocketSocket[addr=/172.16.1.19,port=1128,localport=8189]
New Request Received :Sat Mar 31 11:24:52 GMT+05:30 2007
Sat Mar 31 11:24:52 GMT+05:30 2007
Sub Request Received :Sat Mar 31 11:24:52 GMT+05:30 2007
file size in bytes:2411
Sub Request Received :Sat Mar 31 11:26:20 GMT+05:30 2007
Client5
SocketSocket[addr=/172.16.1.14,port=1110,localport=8189]
New Request Received :Sat Mar 31 11:35:13 GMT+05:30 2007
Sat Mar 31 11:35:13 GMT+05:30 2007
Sub Request Received :Sat Mar 31 11:35:13 GMT+05:30 2007
file size in bytes:2411
Sub Request Received :Sat Mar 31 11:35:55 GMT+05:30 2007

Client1 Logs

C:\j2sdk1.4.1\bin>java echoclient itlab1pc16
LocalHost IP-Address=itlab1pc17/172.16.1.17
Connection to
ServerSocket[addr=itlab1pc16/172.16.1.16,port=8189,localport=1182]
Sat Mar 31 10:47:57 GMT+05:30 2007
requiredfile.java
File Request from client sent onSat Mar 31 10:49:32 GMT+05:30 2007
End time: Sat Mar 31 10:50:03 GMT+05:30 2007

Client2 Logs

C:\j2sdk1.4.1\bin>java echoclient itlab1pc16
LocalHost IP-Address=itlab1pc18/172.16.1.18
Connection to
ServerSocket[addr=itlab1pc16/172.16.1.16,port=8189,localport=1197]

Sat Mar 31 11:04:45 GMT+05:30 2007
File request from client sent onSat Mar 31 11:07:31 GMT+05:30 2007
End time :Sat Mar 31 11:07:39 GMT+05:30 2007

Client3 Logs

C:\j2sdk1.4.1\bin>java echoclient 172.16.1.16
LocalHost IP-Address=itlab1pc18/172.16.1.21
Connection to
ServerSocket[addr=/172.16.1.16,port=8189,localport=1133]
Sat Mar 31 11:20:54 GMT+05:30 2007
File request from client sent onSat Mar 31 11:23:40 GMT+05:30 2007
End time :Sat Mar 31 11:23:48 GMT+05:30 2007

Client4 Logs

C:\j2sdk1.4.1\bin>java echoclient itlab1pc16
LocalHost IP-Address=itlab1pc18/172.16.1.19
Connection to
ServerSocket[addr=itlab1pc16/172.16.1.16,port=8189,localport=1128
Sat Mar 31 11:24:58 GMT+05:30 2007
File request from client sent onSat Mar 31 11:26:26 GMT+05:30 2007
End time :Sat Mar 31 11:26:34 GMT+05:30 2007

Client5 Logs

C:\j2sdk1.4.1\bin>java echoclient itlab1pc16
LocalHost IP-Address=itlab1pc18/172.16.1.18
Connection to
ServerSocket[addr=itlab1pc16/172.16.1.16,port=8189,localport=1110]
Sat Mar 31 11:35:16 GMT+05:30 2007
File request from client sent onSat Mar 31 11:35:59 GMT+05:30 2007
End time :Sat Mar 31 11:36:06 GMT+05:30 2007

References

[1] Lefteris Mamatas, Tobias Harks et.al, "Approaches to Congestion Control in packet Network", Journal of Internet Engineering, Vol. 1 No.1, January 2007.pp.22-33.

[2] Margaret H. Dunham,"Data Mining-Introductory and Advanced Topics" person Education, LPE, ISBN 81-7808-996-3.Year 2003.

[3] Joel E. Cohen and Clerk Jeffries, "Congestion Resulting form Increased Capacity in Single-Server Queuing Network", IEEE/ACM Transactions on Networking, vol.3 No.2, April 1997.pp.305-310.

[4] www.wikipedia.org

[5] R. Jain ,"Myths in Congestion Control in High Speed Networks", Digital Equipment corporation, pp1-24.

[6] Sally Floyd, Sylvia Ratnasamy, Scott Shenker, "Modifying TCP's Congestion Control for High Speeds Networks", May 2002. http://www.aciri.org/tfrc/.

[7] Arjan Durresi et.al, "Load Early Detection (LED): A Congestion Control Algorithm Based on Routers' Traffic Load", IPSJ Vo. 2, Feb. 2006.

[8] M. Gerla, R. Lo Cigno, S. Mascolo,W. Weng, "Generalized Window Advertising for TCP Congestion Control", ETT, Feb 2000,PP 1-14.

[9] Saverio Mascolo et. al, "TCP Westwood: Bandwidth Estimation for Enhanced Transport over Wireless Links", Research NSF.

[10] Saverio Mascolo and Giuseppe Racanelli, " Testing TCP Westwood + over Transatlantic Links at 10 Gigabit/Second rate", Datatag Project and the IT division of CERN (Switzerland).

[11] Saverio Mascolo and Francesco Vacirca, "Issues in Performance Evaluation of New TCP Stacks in High Speed Networks", Research project

[12] Saverio Mascolo, " Congestion control in high-speed communication networks using the Smith principle", Elsevier Automatica 35 (1999) 1921}1935

[13] Yong Xia, etl al."One More bit is Enough ", SIGCOMM 2005.

[14] Richard J. La, Venkat Anantharam, "Window Based Congestion Control with Heterogeneous Users", IEEE INFOCOM 2001.

[15] Richard Karp, "Optimization Problems in Congestion Control", IEEE 2000.

[16] Bin Pang et. al,"A Noval RED-Based hop-by hop Congestion Control", IEEE 2001.

[17] Larry L.Paterson, Bruce S. Davie, "Computer Networks- A System Approach ", 3rd Ed., Elsevier Publications 2006. ISBN 0-55860-833-8. PP. 450-533.

[18] Nishanth R. Sastry, Simon S. Lam, "Theory of Window based unicast Congestion Control", Proceedings of IEEE International Conference on Network Protocol (ICNP 02), PP. 1092-1648.

[19] Manoj Devare, Ajay Kumar, "Role of Markovian Queuing Discipline in Congestion Control in Homogenous and Heterogeneous Networks ", Proceedings of International Conference on Systemics, Cybernatics and informatics", ICSCI-2007. PP.436-439.

[20] Manoj Devare, Ajay Kumar, "Congestion and flow Control in Homogeneous Heterogeneous Networks: Discrete Event Simulation Model ", International Journal of Computer Science & Security IJCSS Vol. 1 Issue 2.

Manoj Devare: He has completed M.Sc in Computer Science from North Maharashtra University, Jalgaon, India. Presently he is working as a Lecturer in Computer Applications in VIIT, Baramati(MS). He is doing Ph.D. in Computer Science from Bharati University, Pune. He has published more than 15 papers in International Journals /National-International Conferences. His area of interest is Simulation of Computer Networks and Expert Systems.

Dr. Ajay Kumar: He has completed B.E., M.Sc.Engg. in Computer Science and Ph.D. in Computer Science. He is having 17 years of teaching experience. Presently he is working as Professor and Head of Dept of MCA in Modern College of Engineering Pune-India, He has published 4 books in Information Technology. He has also published more than 25 papers in National / International Conferences and Journals. His area of interest is Software engineering and Computer Networks