CONGESTION DETECTION AND CONTROL IN PARTIAL MESH USING BAYESIAN APPROACH

Tauseef Gulrez, Rajib Chakraborty, Rami Al-Hmouz, Zenon Chaczko,[1] & Md. Russell Iqbal[2]

1. Information & Communications Technology Group
Faculty Of Engineering, University of Technology Sydney, Australia.
2.Social Science & Professional Ethics Group
Faculty Of Social Sciences, University of NewSouthWales,Sydney, Australia.

{tgulrez, rchakraborty, ralhmouz, zchaczko@eng.uts.edu.au }
{ russelliqbal@yahoo.com.au }

Abstract

This paper presents the method of estimating of congestions in a partial mesh communication network comprises of many nodes such as the internet. The objectives are to characterize and model the non-stationary packet arrival process and to apply the model in adaptive routing algorithms. To achieve this, Bayesian estimation algorithms are implemented and used to determine the congestion of packet-queues in a prescribed interval time. The model is implemented in a state-space formation for estimating the arrival rate and occupancy at a network node. Bayesian method is used to predict the optimal size of the state vector in the proposed model with the aim to eliminate congestion through application of suitable adaptive routing algorithm.

Keywords: Congestions, partial mesh, nodes, adaptive routing, Bayesian estimation, packet-queues and state vector.

1. INTRODUCTION

Congestion control is one of the key components that have enabled the dramatic growth of the Internet. The original idea [4] was to adjust the transmission rate based on the loss probability. The first implementation of this mechanism, denoted TCP Tahoe, was later refined into TCP Reno. This algorithm (together with some of its siblings) is now the dominating transport protocol on the Internet. The throughput and delay experienced by individual users are depending on several factors, including the TCP protocol, link capacity and competition from other users. There are also lower layers that may affect the achieved delay and bandwidth, particularly if part of the end-to-end connection is a wireless link [5].

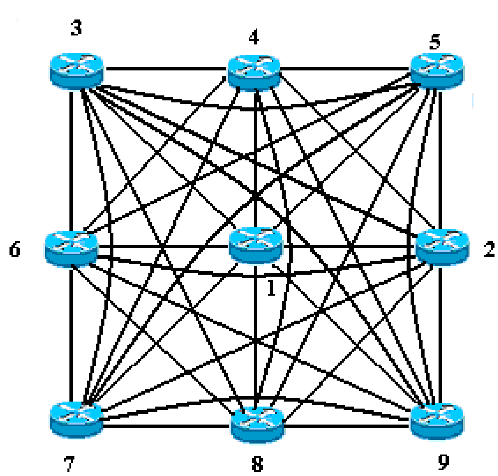
As the Internet expands in size (from millions to billions of nodes) and in capabilities (from mainly file-transfers and web traffic to an increasing proportion of streaming multimedia traffic), Network System researchers propose and investigate new protocols and control algorithms to provide assured quality of service, moreover the work is underway to implement effective methodology for Random Early Detection of Traffic Congestions in the nodes with inexpensive computations. Computer communications over the Internet requires the coordination of protocols, switching and routing functions. The delivery of packets from source to destination endpoints typically takes place using best-effort service. The packets are queued at a router for access to an outgoing link and are serviced on a first-in-first-out (FIFO) basis. There is work underway to add quality-of-service measures [6] to the best-effort service paradigm. These activities are being coordinated by the Internet Engineering Task Force (IETF) and include work related to (i) Traffic Engineering (TE) and Multi-protocol Label Switching (MPLS) [3] (ii) Integrated Services (Intserv)[1] (iii) Differentiated Services (Diffserv) [1] and (iv) IP Performance Metrics (IPPM) [2].

Traffic engineering is the application of technology and scientific rules to optimize and control network performance. A well engineered application must jointly address the performance requirements of end users (traffic management) and efficient usage of network resources (capacity management). Performance measures are typically the end-to-end delay, delay variation, packet loss and throughput. Network efficiency is measured with respect to the usage of buffer space, link bandwidth and computational resources.

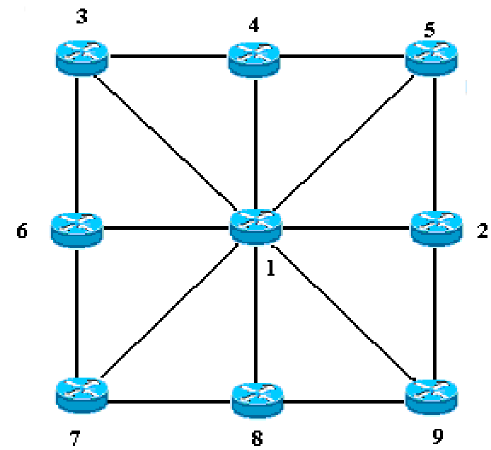
In this paper the estimation for congested nodes has been presented for a partial mesh network. The structure of this paper is as follows. In section 2 the topology of the network has defined with two subsections shows the mathematical model of node and the Bayesian estimation. In section 3 the simulation results are discussed, followed by the concluded remarks and future-work is in section 4 and 5 respectively.

2. THE PARTIAL MESH TOPOLGY

Also called mesh topology (figure 1) or a mesh network, mesh is a network topology in which devices are connected with many redundant interconnections between network nodes. In a true mesh topology every node has a connection to every other node in the network. There are two types of mesh topologies: full mesh and partial mesh.



(a)



(b)

Figure 1: a) Full Mesh Topology. b) Partial Mesh Topology

Full mesh topology occurs when every node has a circuit connecting it to every other node in a network. Full mesh is very expensive to implement but yields the greatest amount of redundancy, so in the event that one of those nodes fails, network traffic can be directed to any of the other nodes. Full mesh is usually reserved for backbone networks. Partial mesh topology is less expensive to implement and yields less redundancy than full mesh topology. With partial mesh, some nodes are organized in a full mesh scheme but others are only connected to one or two in the network. Partial mesh topology is commonly found in peripheral networks connected to a full meshed backbone.

2.1. Node queue length model

In a simplistic view of a network connected in partial mesh topology, queue length on each node on the grid of the topology is taken as a quality factor for that particular node. Let us assume there is a network consisting *'n'* nodes connected in a given way (figure.1). Now a parameter q_k^i is defined as the queue length at time k for node *i*, denoted in terms of its maximum threshold. Mathematically,

q_k^i - is proportional to the length of the queue on node *i* at time *k* (1)

The term q_k^i can be taken as a measure of node saturation for time *k*. Now it can be assumed that each node transmits some or all fraction of data, which were in its queue, to the other nodes connected to it and also receives data from the other connected nodes. Thus a new set of queue length is obtained for each of the nodes. Assuming a set of n nodes, this can be viewed as

$$q_k^i = (1 - f_k^{ii})_{q_{k-1}}^i + \sum_{j=1, j \neq i}^{n} f_k^{ij} q_{k-1}^j \qquad (2)$$

where ;

- *if* $j \neq i, f_k^{ij}$ refers to the fraction of queue-length at time k−1, on node j, sent to node *i.*
- f_k^{ij} , refers to the fraction of the queue-length sent from node i to the other nodes connected to it.

Thus the process model for node queues, can be taken as

$$\begin{bmatrix} q_k^1 \\ q_k^2 \\ \vdots \\ q_k^n \end{bmatrix} = \begin{bmatrix} 1-f_k^{11} & f_k^{12} & \cdots & f_k^{1n} \\ f_k^2 & 1-f_k^{22} & \cdots & f_k^{2n} \\ \vdots & \vdots & \ddots & \vdots \\ f_k^{n1} & \cdots & \cdots & f_k^{nn} \end{bmatrix} \begin{bmatrix} q_{k-1}^1 \\ q_{k-1}^2 \\ \vdots \\ q_{k-1}^n \end{bmatrix} + \begin{bmatrix} G_{11} & G_{12} & \cdots & G_{1n} \\ G_{21} & G_{22} & \cdots & G_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ G_{n1} & G_{n2} & \cdots & G_{nn} \end{bmatrix} \omega_k \qquad \text{--(3)}$$

Whatever transmitted by a node is supposed to be received by the other nodes connected to that given one. Hence a restriction parameter f_k^{ij} exists and that is

$$f_{ii}^k = \sum_{j=1, j \neq i}^{n} f_{ji} \qquad (4)$$

The process noise ω_k is normally distributed *n* vectors with mean zero variance Q_k . This noise model takes care of random error that may occur during the transmission or reception of data between the nodes. On the other hand the measurement model is simple and can be taken as

$$y_k = \begin{bmatrix} h_k^1 & 0 & \cdots & 0 \\ 0 & h_k^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_k^n \end{bmatrix} \begin{bmatrix} q_k^1 \\ q_k^2 \\ \vdots \\ q_k^n \end{bmatrix} + v_k \qquad (5)$$

Again v_k is the $n \times 1$ measurement noise model with mean zero and variance R_k.

In compact form (1) and (5) can be written as:

$$Q_k = F_k Q_{k-1} + G\omega_k \qquad (6)$$

$$y_k = HQ_k + v_k \qquad (7)$$

2.2 Bayesian Estimation of node queue

With a prior estimate at time $k-1$ of queue length denoted as $\hat{Q}_{k|k-1}$ and associated covariance $P_{k-1|k-1}$, the one-step prediction is done using the process model in (6) as

$$\hat{Q}_{k|k-1} = F_k \hat{Q}_{k-1|k-1} \qquad (8)$$

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k \qquad (9)$$

Innovation covariance and Kalman gain are defines by respectively.

$$S_k = HP_{k|k-1} H^T + R_k \qquad (10)$$

$$K_k = P_{k|k-1} H^T S_k^{-1} \qquad (11)$$

Finally the update estimate of queue length and covariance are given by

$$\hat{Q}_{k|k} = \hat{Q}_{k|k-1} + K_k \left[y_k - H\hat{Q}_{k|k-1} \right] \qquad (12)$$

$$P_{k|k} = \left[I - K_k H \right] P_{k|k-1} \qquad (13)$$

Expression in (12) gives the estimated length of queue at time *k*. The set of equations (8) through (13) gives a standard recursive Kalman filter process for estimating the queue length of a network modeled by (6) and 7).

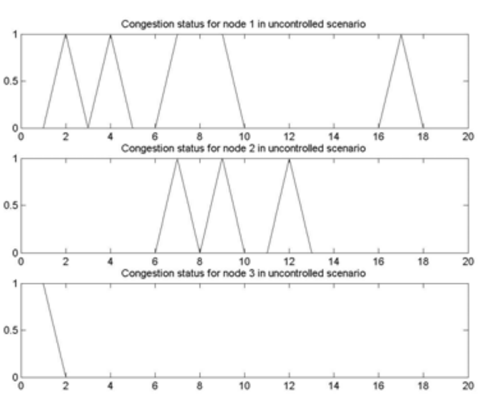
According to the requirement of the system and threshold of each node, a certain value can be assigned for each of the nodes and testing the estimated result of queue length given by (12) against that threshold, node saturation can be detected. Based on that, certain routing criterion can be fixed to handle the saturation. The new routing scheme will affect the F_k matrix of the process model given by equation (6). Any suitable algorithm can be taken to avoid the congestion status of the nodes for a longer period of time and this will be reflected in the model as a change of matrix F_k, keeping the constraint given by (4).

3. SIMULATION RESULTS

A partial mesh network (figure 1(b)), containing 9 nodes, is considered in the simulation environment. The interconnectivity among the network is based on the interconnection which governs the transmission and reception of data between nodes, a transition matrix F was modelled keeping the necessary condition of (4) true. The sensors are assumed to collect data of queue length at each node. Meanwhile the noise model was also set with mean zero and with variance 0.02 and 0.01 for process and measurement model respectively. The simulation was run under two conditions - the first one, where there is no congestion tackling criterion was implemented and the second one with some technique to tackle the congestion implemented. Both scenarios, the validity of the model and the estimation technique are explained.

3.1 Uncontrolled congestion in nodes

The first scenario was built such that the nodes will communicate with each other without taking the fact into consideration that they are getting congested. Irrespective of the node status in terms of queued data, new data arrives. Simulation result shows in (*figure 2&3*) that in this scenario, once a node reaches congested status, it does remain in it's previous status for some duration, which is normal given that no counter measure has taken to prevent the situation.



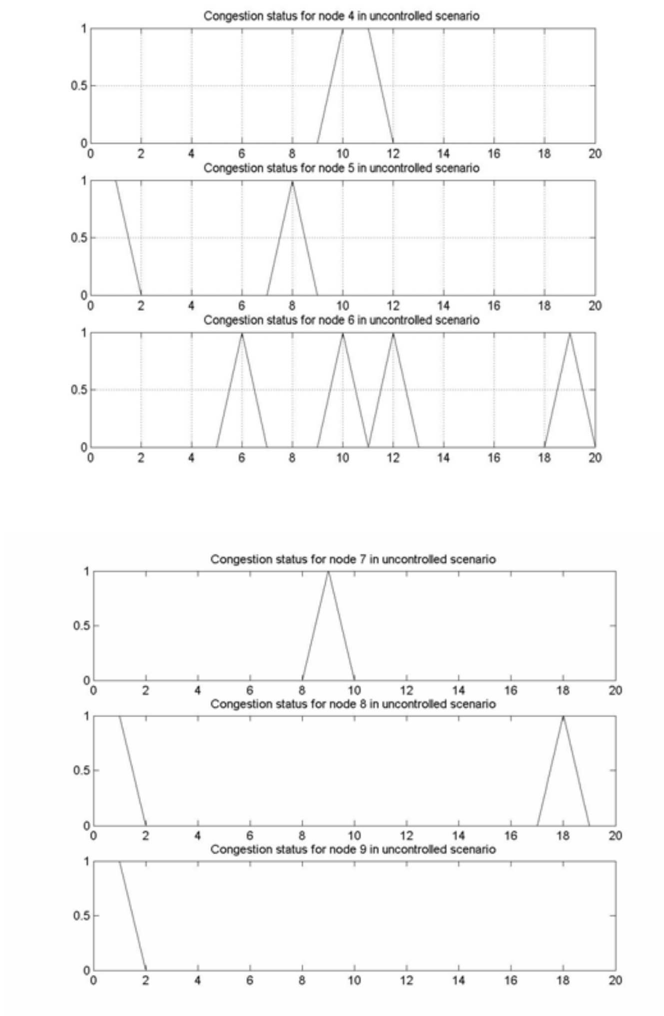
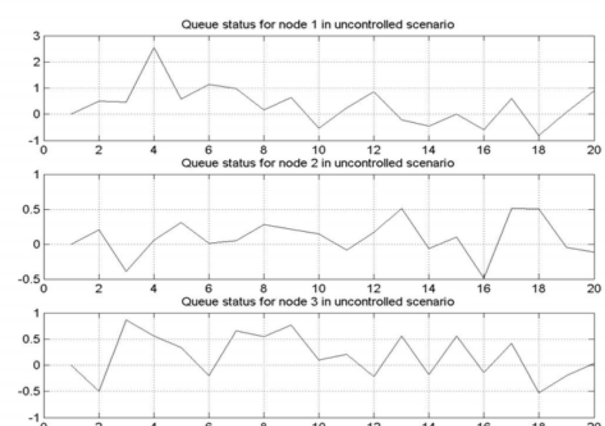


Figure 2: Congestion status of nodes in uncontrolled scenario



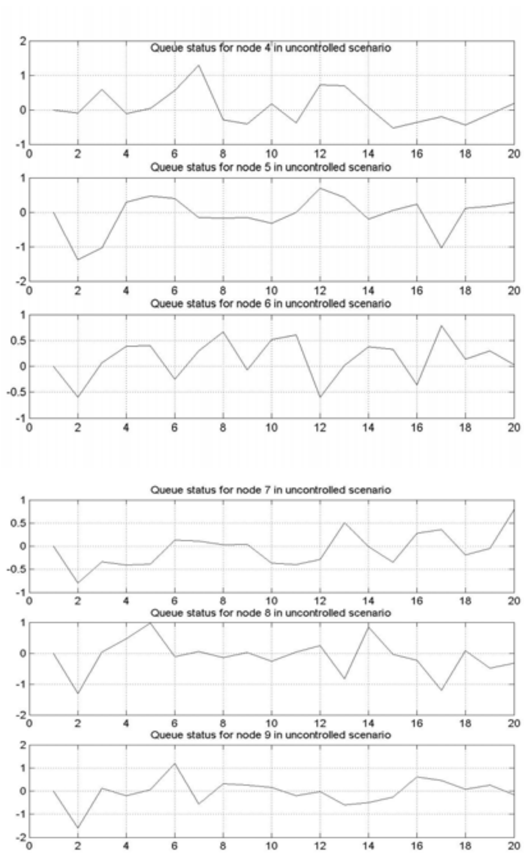


Figure 3: The Queue Status in uncontrolled scenario

3.2 Controlled congestion in nodes

In the second scenario as depicted in (*figure 4&5*), a simple approach has been taken to avoid the congestion in a certain node. When the estimated node queue goes above the threshold level, the node is stopped from receiving any data from other nodes while it continues to release its own data. This way it frees itself from congestion status and retains the original data flow once its queue length is below the threshold. This scheme is accomplished by retaining the f_k^{ij} term from the corresponding row and zeroing the other factors in the same row which refers to the fact that even though the node is sending the data, it is not receiving anything from the other nodes. Also to hold the constraint (4) true, the f_k^{ij} where $j \neq i$, is adjusted properly. From this (*figure 4&5*) simulation, the congestion status shows that once a node reaches that threshold level, it falls below the threshold due to the scheme applied to it and also on average it stays a shorter period of time at the saturation level. In the simulation, node 1 and 2 were put into strain condition. Node 1 has connection to all other nodes and therefore receives data from all the nodes and node 2 were tested against a condition that it transmits only 40% of data (*figure 4&5*).

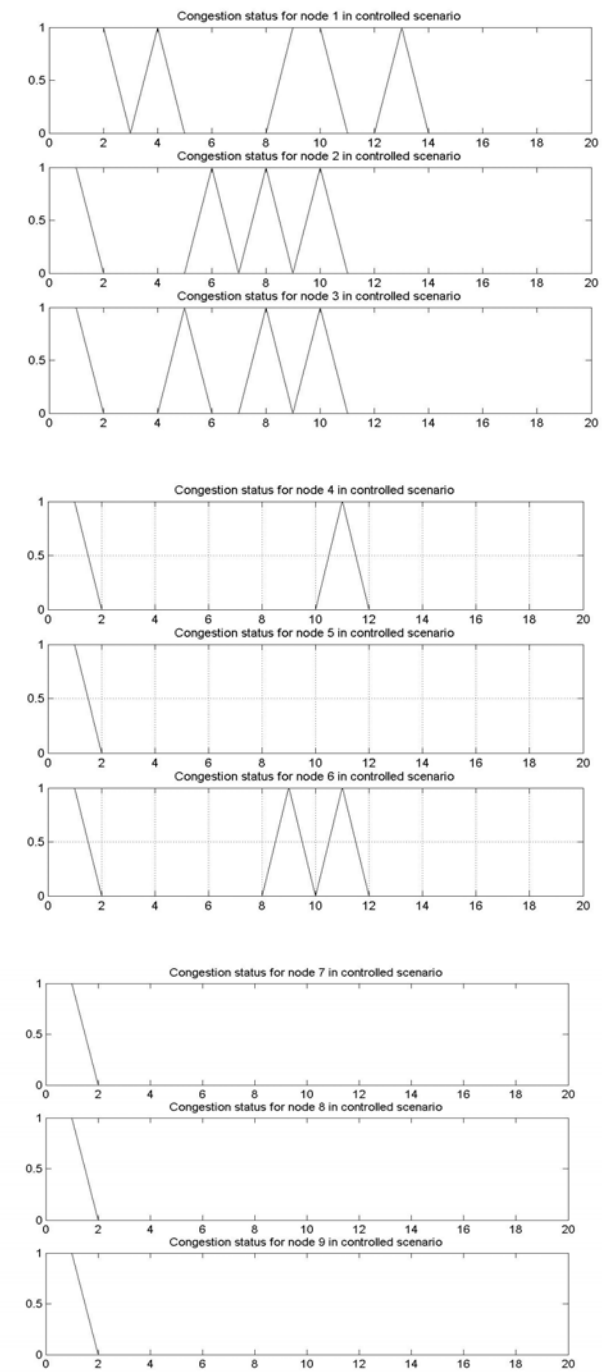


Figure 4: Congestion Status of nodes for controlled scenario.

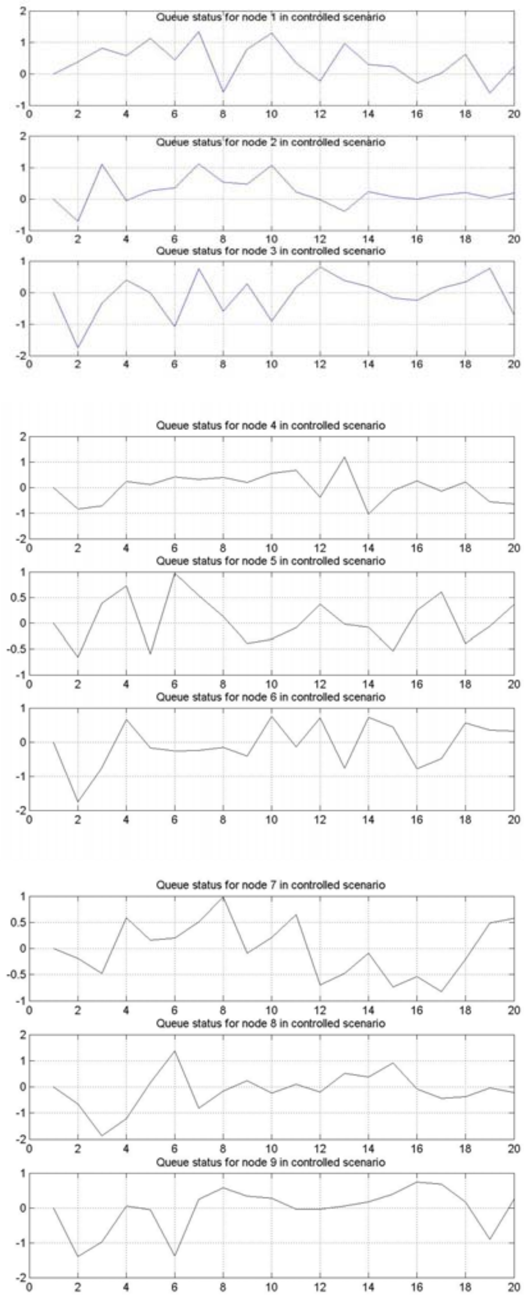


Figure 5: Queue status of nodes for controlled scenario.

4. CONCLUSION

In this paper we have studied estimation that could be suitable for congestion control applications. The network model proposed here is simple but enough to handle the nodes congestion or saturation status. The Bayesian approach is taken to estimate the status of the node's queues which have tested in simulation. In both the scenarios of simulation the estimation approach is successful to estimate the queue length and therefore detects the congestions or saturations in the nodes. Moreover, we tested the same scheme on the network equipped with nine nodes of Cisco 7600 series routers, it show us the very good results and at every instant of time nodes saturation status was determined in a very simple way. An effective routing algorithm can also be done by properly modifying the transition matrix F_k, after the congestion is detected.

5. FUTURE WORK

Prediction of congestion in a network based on the same model of nodes can be achieved and hence a complete routing framework can be built. We are planning to implement our model in more efficient way with a more complex structure of network.

REFERENCES

[1] Clemm, A., P. Leung: Model-Driven Open Packet Telephony Management. IEEE/IFIP NOMS 2002, Florence, Italy, 4/2002.

[2] Dreo Rodosek, G., L. Lewis: Dynamic Service Provisioning: A User Centric Approach. IEEE/IFIP DSOM 2001, Nancy, France, 10/2001.

[3] D.0. Awduche, "MPLS and Traffic Engineering in IP Networks," IEEE Communications Magazine., p42-48., December 1999.

[4] Jacobson, V. (1988*a*). Congestion avoidance and control. *ACM Computer Communication Review*18, 314–329.

[5] Mascolo, S., C. Casetti, M. Gerla, M. Y. Sanadidi and R.Wang (2001). TCP Westwood: bandwidth estimation for enhanced transport over wireless links. In: *MobiCom*. Rome, Italy.

[6] X. Xiao and L.M. Ni, "Internet QOS: A Big Picture," IEEE Network, p8-18, March/April 1999.