Reliable Transporting and Optimal Routing on Rate-Based for Ad Hoc Networks

Ming-Hui Tsai[1], Tzu-Chiang Chiang[1,2], and Yueh-Min Huang[1]

[1] Department of Engineering Science, National Cheng-Kung University, Taiwan, ROC
[2] Department of Information Management, Hisng-Kuo University of Management, Taiwan
colux@tnfsh.tn.edu.tw, n9892105@mail.ncku.edu.tw,
huang@mail.ncku.edu.tw

Abstract. Owing to the inflexibility of the structure, the conventional layered protocols suffer from the inability to distinguish between losses due to route failures and congestion. Recently, in efforts to overcome the challenges of dynamic environment in ad hoc networks, there have been increased interests in protocols that rely on interactions between different layers. In this paper, we propose a cross-layer architecture combining the three layers of data link, network, and transport in order to provide reliable transporting and optimal routing on rate-based, termed as RTOR. In accordance with the rate computed at MAC layer, a path with the optimal rate is selected at network layer. While the flow control and reliability is performed by the determination of transmission rate and SACKs at transport layer, respectively. The transport protocol herein is not a variant of TCP but a rehash. In flow control, the transmission rate is adjusted by rate feedback instead of AIMD congestion windows. In reliability, SACK blocks appended to feedback packet are used rather than implication of packet lost by duplicate ACKs. In addition, choosing a path with the optimal rate can balance the network load in an appropriate degree. The evaluation results clearly indicate the significant performance improvement that RTOR provides over default TCP, TCP-ELFN, and ATP.

1 Introduction

An ad hoc network is a dynamic wireless network established by a group of mobile nodes on a shared wireless channel without any infrastructure. A communication session is achieved either through single-hop transmission if the recipient is within the transmission range of the source node, or by relaying through intermediate nodes otherwise. For this reason, ad hoc networks are also called multihop packet radio networks [12]. Those characteristics differentiate them from traditional computer networks including lack of infrastructures, mobility, shared channel, and limited bandwidth and, hence, the conventional layered OSI model applied in such circumstance suffers from varying channel and network conditions.

Recently, in efforts to overcome this kind of challenges and further improve the performance of wireless mobile networks, there have been increased interests in protocols that rely on interactions between different layers [1, 2]. Such cross-layer approaches to network design seek to enhance the performance by jointly designing

X. Zhou et al. (Eds.): EUC Workshops 2006, LNCS 4097, pp. 112 – 121, 2006.

multiple protocol layers. Certainly, on the other hand, protocol layers are excessively useful in allowing designing optimally a single protocol layer without the complexity and expertise associated with considering other layers. In [3], it shows that the conventional layered architecture is a reasonable way to operate wireless networks. Therefore, those are imperative to pay ripe deliberation between how to retain appropriately independency of separate layers and how to exploit the convenience from layers coordination. The key-influencing factor is the determination of parameters used as interacting information in the cross-layer architecture.

In this paper, we propose a distinct and adaptive cross-layer architecture combining the three layers of data link, network, and transport in order to provide reliable transporting and optimal routing on rate-based. Since the TCP suffers from the varying network conditions in ad hoc networks, the rate-based flow control is adopted instead of the congestion window. Furthermore, the congestion control and reliability mechanisms are decoupled. Congestion control is performed using rate feedback from the network, while reliability is ensured through selective ACKs (SACKs) [13] from the receiver. As to the metric of underlying routing protocols, we use the min-max rate instead of usual hop counts to choose the route supporting the optimal transmission rate. The scheme can balance flows among the whole network and, hence each connection can gain the higher transmission rate and avoid the excessive channel contention at intermediate nodes.

To this end, the key parameter exchanging among those three layers is transmission rate which is computed and adapted at MAC layer, used as the metric for choosing routing path at network layer, and fed back to the sender for congestion (flow) control at transport layer. Rate-based transmission assists mainly in improving performance in three ways: first, it can avoid the drawbacks due to the burstiness resulting from windows-based transmission [5] adopted by the popular TCP. Second, it can provide the possibility of preserving stable transmission rate rather than over-raise the congestion window resulting in emergence of undesired congestion. Last, it can attain rapidly the suitable and available rate at a certain connection pair behind one single round-trip time (RTT), called as quick start which is superior than TCP's slow start.

The paper is organized as follows. In Section 2 we give a quite intensive overview of cross-layer literature. Section 3 presents our cross-layer design on rate-based including the dynamic rate measurement, optimal route determination, and congestion control and reliability achievement. In Section 4 we describe in details our simulation environment and we evaluate and comment on the performance of the protocols with respect to the selected metrics of interest. Finally, Section 5 concludes the paper.

2 Related Works

Recently, a lots of researches indicate that the conventional reliable transporting, e.g. TCP, applied in ad hoc networks suffers from the inability to distinguish between losses due to route failures and network congestion [1], i.e., all losses are just regarded as channel congestion. As we know, the losses are possibly as a result of the frequent route failure in ad hoc networks. In reality, the MAC and routing layer perceive most of the losses as due to route failure. So TCP treats losses as an

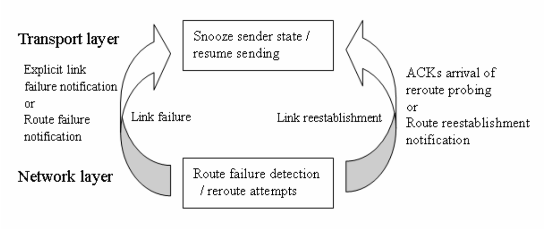

Transport layer
Snooze sender state / resume sending
Explicit link failure notification or Route failure notification
Link failure
Link reestablishment
ACKs arrival of reroute probing or Route reestablishment notification
Network layer
Route failure detection / reroute attempts

Fig. 1. Cross-layer design framework for distinguishing between losses due to route failures and network congestion

indication of congestion turns out to be inappropriate and, even merely slows down the sending rate by adjusting its congestion window resulting in severe performance degradation. There are many cross-layer researches [7, 8, 9, 10, and 11] proposed in recent years, hence, in order to defeat the inability and boost performance by exploiting the explicit notification from network layer, such as route failure notification (RFN) [7] or explicit link failure notification (ELFN) [8], as shown in figure 1.

Moreover, TCP's *additive increase multiplicative decrease* (AIMD) congestion window limits its ability to acquire spare bandwidth quickly, which is especially important under the varying conditions and often incur underutilization of network resource. In [6] Chen et al. propose an end-to-end rate-based flow control scheme (EXACT) which can react quickly and precisely to bandwidth variation and re-route events. But the end-to-end mechanism requires excessive computation and increases undesired complexity. Therefore, in [5] K. Sandaresan et al. propose a per-node basis scheme (ATP) instead of per-flow basis to compute the available transmission rate. It relies on intermediate nodes that a connection traverses to provide rate feedback information.

3 Cross-Layer Architecture of Reliable Transporting and Optimal Routing

Exploiting the coordination among the three layers of data link, network, and transport, we try to provide reliable transporting and optimal routing on rate-based. The proposed cross-layer architecture is shown as figure 2. In each connection pair, there are three roles involved that are sender, receiver, and intermediate node. The different role at different layer presents the different behavior. The concept of our architecture is that the sender's transmission rate is dependent on the rate feedback from receiver. As to the rate feedback information, the receiver attains it from the intermediate nodes traversed through by this connection pair.

In this section, we first highlight the specific form of rate computation at data link layer. We then elaborate the route determination upon the rate from the network layer, i.e., route initiation and recovery and, finally, present the congestion control and reliability mechanisms.

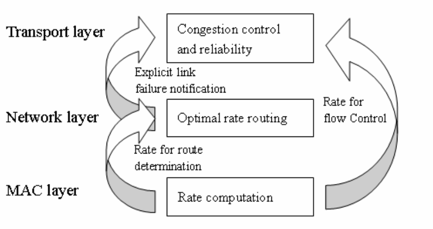

Transport layer
Network layer
MAC layer
Congestion control and reliability
Explicit link failure notification
Optimal rate routing
Rate for route determination
Rate computation
Rate for flow Control

Fig. 2. Cross-layer design framework on rate-based for reliable transport and optimal route

3.1 Dynamic Bandwidth Measurement at MAC Layer

As mentioned earlier, in consideration of varying channel condition and complexity of connection flows, we prefer per-node basis rate computation rather than per-flow basis. At the MAC layer, an exponential average of the queuing delay experienced by frames traversing and an exponential average of the transmission delay experienced by the head-of-line frames are computed at each node and denoted as Q and T, respectively. Q at a node is impacted by the contention between different flows traversing the node and, T is impacted by the contention between the different nodes in the vicinity of that node.

Rate Affected by Delay at MAC Layer

While a frame flows through the intermediate node, the MAC layer is required to calculate the queuing (Q_{sample}) and transmission (T_{sample}) delays experienced. The Q_{sample} and T_{sample} are the time spent by an outgoing frame from queued to dequeued and from dequeued to transmitted successfully, respectively. Both of the values are maintained using exponential averaging as follows:

$$\mathrm{Q}_t = \mathrm{Q}_{t-1} * \alpha + \mathrm{Q}_{sample} * (1-\alpha)$$

$$\mathrm{T}_t = \mathrm{T}_{t-1} * \alpha + \mathrm{T}_{sample} * (1-\alpha)$$

where Q_t and T_t are the exponential average after the outgoing packet transmitted successfully.

Computation of Transmission Rate and Delivery

The longer the delay at a node is, the smaller the transmission rate at a node can use. Each frame includes rate feedback field *R* (note that which will actually be an inverse of Q + T) that consists of the minimum 1/(Q + T) value at the upstream nodes that frame have traversed through. While the frame is dequeued for transmission, the intermediate node examine whether 1/(Q + T) value at that node is smaller than *R* or not. If positive, the intermediate node updates the *R* on the frame to the 1/(Q + T) value. Consequently, the rate is continuously processed along the downstream nodes until its receiver. The transmission rate *R* of a path is obtained as:

$$R_p = \min_{i \in p}\{\frac{1}{\mathrm{Q}_i + \mathrm{T}_i}\}$$

where *p* is one of the paths between the connection pair and *i* is an intermediate node along the path *p*.

A Special Case at an Idle Node

To deserve to be mentioned, a special case that there is no other traffic around a node, 1/(η* (Q + T)) is used as the rate instead of the normal 1/(Q+T). When the channel around the node is idle, the Q + T values will be determined by the actual delay experienced by the probe frame, i.e., route initiation. However, when the actual data flow begins, frames belonging to the flow will contend with other frames belonging to the same flow at both upstream and downstream nodes. For CSMA/CA, the typical value for η is 3. It can be as high as 5 for a path of length 5 or more with a large number hops.

3.2 Min-Max Rate Path Determinations at Network Layer

During connection initiation, or when recovering from a time out, RTOR uses the same mechanism called quick-start as used by ATP to probe for the available bandwidth within a single round-trip time. The sender uses the probing packet (i.e., *route discovery* in DSR or *route request* in AODV [14]) along its available paths through intermediate nodes to the receiver. The intermediate nodes, when they forward the probing packet, stamp on the packet the rate feedback in the manner described in section 3.1.

Which Path Is the Optimal One?

Unlike the distinct layered routing protocols, which perform their forward path setup while receive the first route request (RREQ), in our system the receiver intuitively must wait for an interval to receive the probing packet with the same sequence number from all of the other routes. Of course, it is not essential that the first arrival of the probing packet is via either the shortest path (minimum hop counts) or the optimal path (maximum transmission rate). It only depends on the summation of queuing and transmission delay (*L*, latency) along its traversed path denoted as:

$$L_p = \sum_{i \in p} (\mathrm{Q}_i + \mathrm{T}_i)$$

To this regard, the smaller the *L* value of a path is, the earlier the probing packet is received via the path. The path with the optimal rate is indefinitely the same one with the minimum latency, but it is promised that the transmission rate along the path is definitely effective than others. In other words, there are more packets transmitted along the path during a certain period. In addition, the receiver prefers to the optimal rate path along that the intermediate nodes have born lower network load. The min-max rate path determination balances the network load in an appropriate degree. Refers to figure 3, it is obvious that the intermediate node of connection A is the key and influential node along the shorter path of connection B. The value aside node stands for its transmission rate (*R*). While the connection A is active, the sender B will choose the longer path as its routing path on our min-max scheme rather than the shorter one. Despite of the larger hop counts, the longer path can support better transmission rate. At the same time, the connection A is not affected by connection B and, hence sustains its better and available transmission rate.

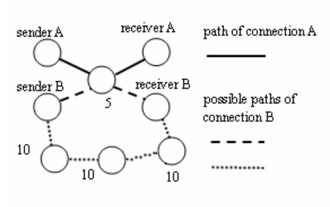
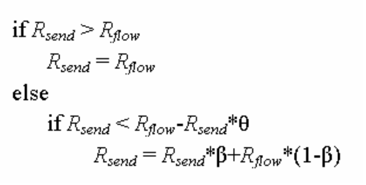


Fig. 3. Two possible paths of one connection



```
if $R_{send} > R_{flow}$
    $R_{send} = R_{flow}$
else
    if $R_{send} < R_{flow}$-$R_{send}$*θ
        $R_{send} = R_{send}$*β+$R_{flow}$*(1-β)
```

Fig. 4. Pseudo code for rate adjustment

Response to Link Failure

Whenever an intermediate node experiences a link failure at its MAC layer, the network layer at the node is informed about this failure and sends a path failure notification message to the sender of the packet. Similarly, the message is sent back to the sender as TCP-F or ELFN does. RTOR then send the probing packet in order to obtain the optimal transmission rate along a new route to the receiver. The primary diversity is the mechanism of resuming transmitting after route reestablishment in which RTOR uses the probing packet to get the explicit rate feedback, while TCP-F or ELFN just come off the snooze state using the previously frozen congestion window. Hence, performing rate estimation on the new route allows the connection to operate at the true available rate instead of either overutilizing or underutilizing the resource available along the new route.

3.3 Rate Feedback and Reliability at Transport Layer

Unlike TCP, which perform respectively flow control and reliability depending on AIMD congestion windows and duplicate ACKs reception, RTOR performs flow control and reliability mechanisms relying on the periodic feedback from the receiver. In order to send the feedback periodically, the receiver runs an expiration timer (*E*). Note that the *E* value should be larger than the round-trip time of a connection, but at the same time must be small enough to track the dynamics of the path characteristics.

Rate Feedback Influenced by Incoming Data Packets

Before the timer expiry, the receiver performs an exponential averaging of the rate piggybacked on every incoming data packet belonging to a flow:

$$R_{flow} = R_{flow} * \alpha + R_{sample} * (1 - \alpha)$$

where R_{sample} is the rate piggybacked on the incoming packet and α is exponential parameter between 0 and 1. As the timer expired, the R_{flow} value is piggybacked on the periodic feedback packet and fed back to the sender.

Rate Feedback Reset by Probing Packets

As to the probe packets of connection initiating or recovering from connection failure, those are sent out to elicit rate feedback from the receiver. While the receiver obtains all of the transmission rates through the possible routes, the route with the maximum rate value (R_{probe}) is determined and, then the rate is piggybacked on the response packet (RREP) to the sender on behalf of the data transmission rate. At the same time, the receiver's R_{flow} is set to R_{probe} while the expiration timer E is reset. Note that R_{flow} at receiver is on per-flow basis that is different from the R_p on per-node basis at intermediate nodes.

Adjustment of Sending Rate

While the sender receives the R_{flow} from the receiver, it adjusts its sending rate according to its original rate (R_{send}). To prevent undesired rate fluctuation, a threshold θ is applied to handle the adjustment. It is shown as the figure 4 how the rate is adjusted.

If R_{send} is larger than R_{flow}, it indicates that the original sending rate is over the sustainable rate at this connection. For avoiding the emergence of congestion, the sending rate should be adjusted down immediately. Oppositely, if R_{send} is smaller than R_{flow} in excess of R_{send}*θ, the sending rate should be adjusted up in an exponential averaging method. Specially, if the available rate R_{flow} lies within (R_{send}, R_{send}+R_{send}*θ), the sender preserve its sending rate R_{send} stably. It is of interest that RTOR operates in a state of poise under the stable network conditions.

Reliability

Owing to decouple of the flow control and reliability mechanisms, the reliability is achieved by providing SACKs from RTOR receiver to report back any holes observed in the data stream. Since the periodic feedback is not provided for every incoming data packet, but rather on a periodic basis, RTOR use a large number of SACK blocks which always identify the first sequence holes in the data stream. The number of SACK blocks is restricted within 20 while take account of the timer value and packet size as [5].

RTOR does not use a retransmission timeout at the sender and, hence, has to rely on the feedback from the receiver to perform correct error recovery. On timer expiry or receipt of probe packet, the receiver identifies all holes encountered and stamps SACK blocks on the feedback packet. While the sender receives the periodic feedback packet or response packet with SACK information, it updates its SACK scoreboard and identifies packets for retransmission. Of course, the packets marked for retransmission are sent preferentially.

4 Performance Evaluation

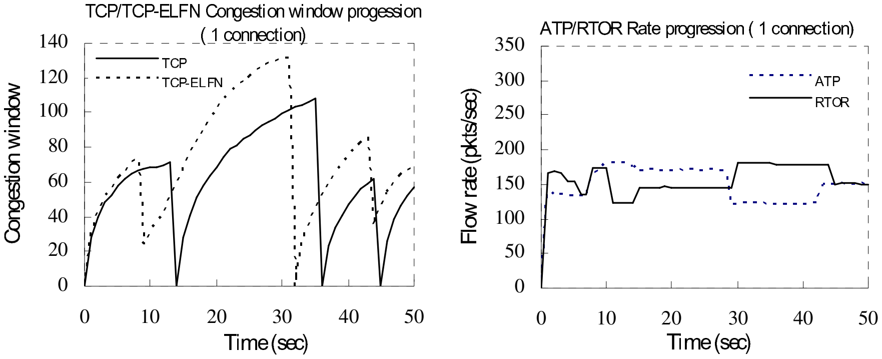
We use the ns2 network simulator for all the simulations. There are 100 nodes initially placed randomly by ns2 scenarios generator (NSG) within a predefined 1000m x 1000m grid area. The mobility model is the random waypoint model. The 1 and 20 connection pairs (sender-receiver) are randomly chosen from the set of 100 nodes in consideration of the effect of load. In our simulation, the nodal speed is 1m/s, 10 m/s, and 20 m/s within the both of different load scenarios. The underlying routing protocol is AODV which has been modified to suit our optimal rate scheme. FTP is the application that is used over TCP for all the flows in the network, while the packets generated are of size 512 bytes. The expiration timer *E* at receiver is set to one second. The performance of RTOR is evaluated and compared against default TCP (NewReno), TCP-ELFN [8], and ATP [5] for typical network scenarios outlined above. In particular, we have considered the following metrics:

1) congestion window/rate progression
2) aggregate throughput

4.1 Congestion Window/Rate Progression

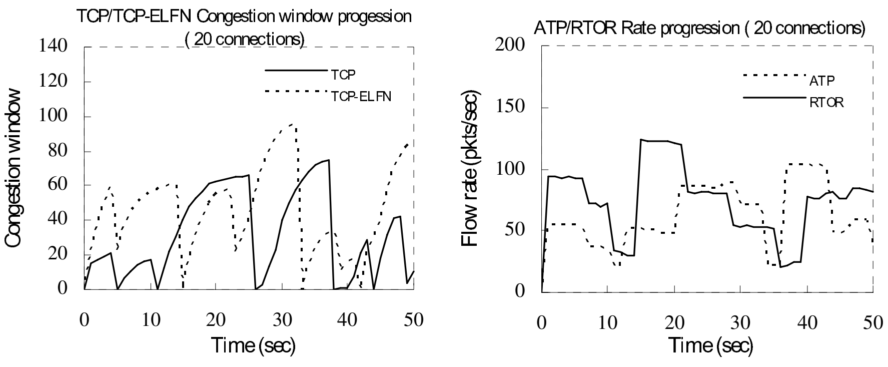
Snapshots of congestion window/rate progression results for default TCP, TCP-ELFN, ATP, and RTOR are presented for a single connection and a 20-connections scenario for a speed of 10 m/s in Figures 5 and 6, respectively. The focus here is to highlight the difference between the congestion window and rate adjustment. Obviously, both of default TCP and TCP-ELFN suffer from the congestion window mechanism (AIMD) and result in instantaneous throughput degradation, shown as figure 5(a) and 6(a). Oppositely, the both of rate-based ATP and RTOR can adapt their transmission rate to true available bandwidth on rate feedback, shown as figure 5(b) and 6(b).

The reasons which make the difference are: 1) RTOR does not decrease its rate on route failures unless dictated by its rate adjustment mechanism to do so. 2) RTOR is always able to reach to the available transmission rate while route initiation or reestablishment from route failures. 3) RTOR can operate in a state of poise under the stable network conditions. In addition, the observation that RTOR outperforms than ATP is that the former selects its path with the optimal rate while the last disregards the effect of routing paths with different rates.



(a) (b)

Fig. 5. Congestion window/rate progression (1 connection). (a)default TCP and TCP-ELFN (b) ATP and RTOR.

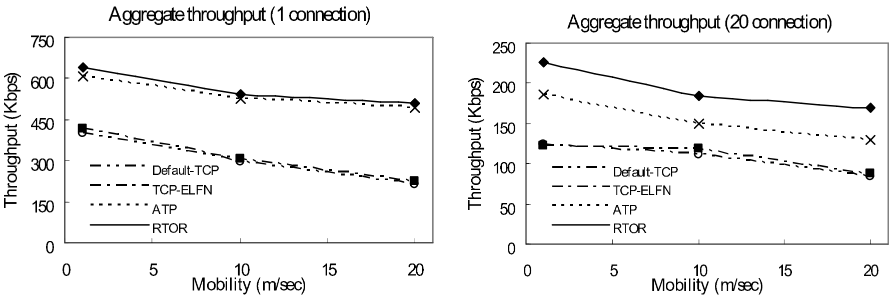


(a) (b)

Fig. 6. Congestion window/rate progression (20 connections). (a)default TCP and TCP-ELFN (b) ATP and RTOR.

4.2 Aggregate Throughput

In figure 7(a), we observe the aggregate throughput affected by different mobility for the single connection scenario. As the mobility increases, the throughput decreases no matter which protocol is used. In addition, it is obvious that the rate-based protocols have better performance due to its quick start and efficiency of network bandwidth utilization. For multiple (20) connections in figure 7(b), similarly, we find the same results in 7(a). It is worth mentioning that RTOR always has better performance than ATP in spite of mobility. Owing to the heavier connection load, RTOR appropriately balance flows among the whole network and, hence each connection can gain the higher transmission rate and avoid the excessive channel contention at intermediate nodes.



(a) (b)

Fig. 7. Aggregate throughput under (a)1 connection (b)20 connection

5 Conclusion

In this paper, we presented a cross-layer architecture combining the three of MAC, network, and transport layers for seeking a solution to the problem of performance degradation in mobile ad hoc networks. The frequent route failures and route reestablishments in these environments introduce a new challenge to the TCP congestion control, and lead us to adopt the rate-based flow control despite TCP unfriendness. The limited bandwidth under channel shared is another challenge to the common routing protocols, and induce us to select a path with the optimal rate from the consideration of equilibriums of connection load. Topics for further research include the investigation of TCP friendness on rate-based, furthermore, support of real-time video stream especially taking account of delivery latency.

References

1. A.A. Hanbali, E. Altman, and P. Nain, “A Survey of TCP over Ad Hoc Networks,” IEEE Comm. Surveys & Tutorials 3rd Quarter 2005, vol. 7, no. 3, pp. 22-36.
2. X. Chen, H. Zhai, J. Wang, and Y. Fang, “TCP Performance over Mobile Ad Hoc Networks,” Can. J. Elect. Comput. Eng., Vol. 29, No. 1/2, January/April 2004, pp. 129-134.
3. V. Kawadia and P.R. Kumar, “A Cautionary Perspective on Cross-Layer Design,” IEEE Wireless Comm. Feb 2005, pp. 3-11.
4. E. Setton, T. Yoo, X. Zhu, A. Goldsmith, and B. Girod, “Cross-Layer Design of Ad Hoc Networks for Real-Time Video Streaming,” IEEE Wireless Comm. Aug 2005, pp. 59-65.
5. K. Sundaresan, V. Anantharaman, H.-Y. Hsieh, and R. Sivakumar, “ATP: A Reliable Transport Protocol for Ad Hoc Networks,” IEEE Trans. on Mobile Computing, VOL. 4, NO. 6, NOV/DEC 2005, pp. 588-603.
6. K. Chen, K. Nahrstedt, and N. Vaidya, “The Utility of Explicit Rate-Based Flow Control in Mobile Ad Hoc Networks,” WCNC 2004 / IEEE Comm. Society, pp. 1921-1926.
7. K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash, “A Feedback-Based Scheme for Improving TCP Performance in Ad Hoc Wireless Networks,” IEEE Personal Comm. Feb 2001, pp. 34-39.
8. G. Holland and N. Vaidya, “Analysis of TCP Performance over Mobile Ad Hoc Networks,” ACM Wireless Networks, vol. 8, no. 2, Mar 2002, pp. 472-479.
9. T.C. Chiang, M.H. Tsai, and Y.M. Huang, "Adaptive Clustering with Virtual Subnets Support in Ad Hoc Networks", Lecture Notes in Computer Science, Volume 3992 ,pp. 1008–1015, 5/2006.
10. D. Kim, C. Toh, and Y. Choi, “TCP-BuS: Improving TCP Performance in Wireless Ad Hoc Networks,” J.Comm and Net., vol. 3, no. 2, June 2001, pp. 175-186.
11. J. Liu and S. Singh, “ATCP: TCP for Mobile Ad Hoc Networks,” IEEE JSAC, vol. 19, no. 7, July 2001, pp. 1300-1315.
12. M.H. Tsai, T.C. Chiang and Y.M. Huang, "On Scalability and Mobility Management of Hierarchical Large-Scale Ad Hoc Networks", Lecture Notes in Computer Science, Volume 3823 ,pp714 - 723, Dec. 2005.
13. M. Allman, D. Glover, and L. Snachez, “Enhancing TCP over Satellite Channels using Standard Mechanisms,” RFC 2488, Jan.1999.
14. Charles. E. Perkins, AD HOC NETWORKING, Addison-Wesley, 2001. Chapter 5 and 6.