RESEARCH ARTICLE OPEN ACCESS

# Enhancement in Packet Scheduling and Optimization Through Cross Layer Design in Mobile Adhoc Network

Steven Raj N*, Namratha**, Praveena***
*(Department of CSE, GNDEC Bidar 585401)
** (Department of ECE Dept, GNDEC Bidar 585401)
*** (Department of ECE Dept, *Sri MVIT, Bangalore*)

## ABSTRACT

Cross layer design is a promising approach in mobile ad hoc networks (MANET) to combat the fast time-varying characteristics of wireless links, network topology, and application traffic. In this paper, we employ cross layer design to develop a novel-scheduling scheme with optimizations aimed at service differentiation. The meaning of service differentiation is to offer different routing and priority services to different types of load. The scheduling scheme is executed at the network layer of every station according to the channel conditions estimated by the MAC layer. The optimizations are based on traffic property sharing and packet timeout period interaction to reduce the packet collisions and improve network performance. We evaluate the proposed scheme under different network loads in terms of packet delivery ratio, average end-to-end delay. The simulation results show that our scheme can provide different service differentiations for time-bounded and best effort traffics. In particular, we can guarantee the delay requirements of time-bounded traffic.

## I. INTRODUCTION

The term MANET stands for Mobile Ad-hoc Network. This new networking concept defines simple mechanisms, which enable mobile devices to form a temporary community without any planned installation, or human intervention. The idea is to form a totally improvised network that does not require any pre-established infrastructure. But, how can we make this possible The answer is very simple. Each node acts as a host and a router at the same time. This means that each node participating in a MANET commits itself to forward data packets from a neighbouring node to another until a final destination is reached. In other words, the survival of a MANET relies on the cooperation between its participating members—if a source node wants to communicate with another node which is out of its transmission range, the former will send its packets to a neighboring node, which will send them, in its turn, to one of its neighboring nodes, and so on, until the destination node is reached. Some specific applications of MANET are military communications, virtual classrooms, emergency search and rescue operations and communication setup in exhibitions, conferences, meetings, etc. we start with a general review of Ad-hoc On-demand Distance Vector (AODV) routing protocol, in MANETs. This protocol initiate route discovery only when a route is needed and maintain active routes only while they are in use. Unused routes are deleted.AODV is an improvement on DSDV (Destination sequenced Distance Vector) [1 ] because it typically minimizes the number of required broadcasts by creating routes on an on-demand basis, as opposed to maintaining a complete list of routes as in the DSDV algorithm. When a source node wants to send a packet to some destination node and does not have a valid route to that destination, it initiates a path discovery process to locate the destination. It broadcasts a RREQ (Route Request) packet to its neighbours, which forward the request to their neighbours, and so on, until the destination is located or an intermediate node with a "fresh enough" route to the destination is located. During the process of forwarding the RREQ, intermediate nodes record in their route tables the address of neighbours from which the RREQ was received, thereby establishing a reverse path. When the RREQ has reached the destination or intermediate node with a "fresh enough" route, the destination/intermediate node responds by unicasting a RREP (route reply) packet back to the neighbour from which it first received the RREQ. As the RREP is routed back along the reverse path, nodes along this path set up forward route entries in their route tables. Finally, the source node can send its packets to the destination via the established path.

A set of predecessor nodes are maintained for each routing table entry, indicating the set of neighboring nodes which use that entry to route data packets. These nodes are notified with RERR packets when the next-hop link breaks. Each predecessor node, in turn, forwards the RERR to its own set of predecessors, thus effectively erasing all routes using the broken link. The source node may re-initiate route discovery for that destination. Otherwise, when a link break in an active route occurs, the node upstream of that break may choose to repair the link locally if the destination was no far farther than MAX_REPAIR_TTL hops away.The proposed protocol offers different ways to deal with network constraints like the restricted Bandwidth or the congestion. The best effort data means the files or the raw data, which do not have the time constraints, meaning that they can be transferred with a delayed transmission. Where as the Time bound data like the voice and the videos must be transmitted fast. Hence the network layer schedules them based on a priority queue. This is a newer concept, as the Network queue is generally a non-priority and best effort queue. Therefore the technique can be used to transfer any types of data load with various constraints and parameters like deadlock, time out, channel synchronization etc. The cross layer design offers a better approach for QOS improvement through information sharing between the layers. In a normal layered approach the buffer length of a particular layer may be flooded if entire decision is left on a layer. Instead, if the routing time and the decision is distributed amongst the layers, it would become easier for the layers and the possibility of data over flow is minimized. The rest of the paper is organized as follows. In section 2, we review the cross layer design in MANET. Section 3 explains the related work. Section 4 explains the methodology of the Scheduling and optimization scheme for service differentiation through cross layer design. Section 5 explains the simulation. In section 6 results were analyzed. Section 7 concludes the paper with remarks on future work.

## II. Overview of Cross Layer Design(CLD) in Manet

In this section, the primary design goal right from the start was to implement a system-wide CLD in a MANET protocol stack using 802.11, and the authors claim no other existing reference architecture has yet to accomplish this goal. The reference stack design can be seen in Figure 1 Protocols belonging to different layers can co-operate by sharing network status, while still maintaining the layer separation in the protocol design. The authors list 3 main advantages of their reference design: Full compatibility with existing standards as it does not modify each layer's Core function. A robust upgrade environment: adding or removing protocols belonging to different layers in the stack is possible without modifying the operations at the other layers. Maintaining the benefits of a modular architecture. Energy management, security and cooperation are cross-layered by nature, as seen in Figure 1. The core component of the reference design is the Network status repository. Whenever a protocol in the stack collects information, it will publish this to the repository and thus making it available for every other Protocol. The authors state that this avoids duplicating efforts to collect internal State information and leads to a more efficient design.

Since co-operation between the different protocols takes place in the network Status repository, this feature does not compromise the expected normal operation of the stack. We can even replace it with a legacy stack implementation without any other implication than loosing the optimization and performance gain provided by the CLD.

MANET reference architecture will offer the following performance advantages in an ad hoc network design:

- Cross-layer optimization for all network functions
- Improved local and global adaptation
- Full context awareness at all layers
- Reduced overhead

The approach aims to optimize overall network performance, by increasing local interaction among protocols, decreasing remote communications, and consequently saving network bandwidth.

The MANET implementation is what can be described as a system-wide CLD where stack-wide layer interdependencies are designed and implemented to optimize overall network performance. The knowledge has to be shared between all layers to obtain the highest possible adaptively. The different protocols will actively seek to use available state information throughout the stack to adapt their behaviour and thus maximize throughput and efficiency and minimize delay and power usage - all this while still being QOS efficient.

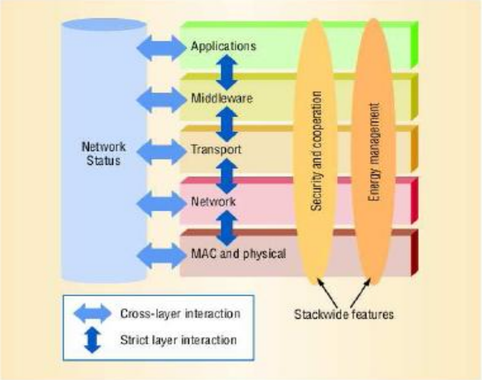

***Figure 1: MANET's cross-layered network stack[2]***

## III. Related Work

A mobile ad hoc network (MANET) consists of mobile stations. Which communicate with each other. Without any infrastructure, such as the base stations, as in cellular networks. Much prior research has shown that application traffic and wireless characteristics affect MANET design in each of the seven layers in the ISO/OSI reference model. Intuitively, through information sharing and interaction among different layers, one may take more efficient actions and respond more quickly to the time-varying changes of the network. This is the cross layer design method.

Some research has been done in cross layer design for ad hoc networks [2,3]. In [2], channel reservation control packets are used to estimate the channel condition, which is used by media access control (MAC) and routing to implement rate adaptation and optimal route selection. Reference [3] proposes a new multiple access control scheme composed of two phases: scheduling and power control, both based on the physical layer parameters, such as signal-to-interference-noise-ratio (SINR), etc. But this scheme assumes that there exists a separate feed back channel to send SINR measurements and a central controller to execute the scheduling algorithms, which may not be available in practical ad hoc networks. There is also much work in scheduling and priority provision in wireless networks [4-IO]. Chun et al. [4] evaluate the performance of different packet scheduling algorithms in the network layer. In addition, they also investigate the queuing dynamics at the mobile stations. In [5], three mechanisms to support message priorities in the MAC layer are presented: packets with different priorities stored in different FIFOs. assigned different access deferral and different backoff times. Reference [6] also proposes three different service differentiation schemes for IEEE 802.1 1 MAC layer – using different contention windows, assigning different inter frame spacings (IFSs), and using different maximum frame lengths for different users. However, only the network or MAC layer behavior is considered in [4-6], while the impacts of other layers are ignored. In [7], a busy tone priority-scheduling (BTPS) scheme is presented, in which two narrow-band busy tone signals are used to guarantee high priority stations preferred access to the wireless channel. But this scheme needs to divide the wireless spectrum into three channels: BTI, BT2 and Data channels. H. Luo, S. Lu and V. Bharghavan [8] propose a new model to address the trade-off between fairness and channel utilization, which concentrates on ensuring the fair allocation of channel bandwidth and maximizing the spatial reuse. Some other scheduling and priority algorithms are also proposed in wireless cellular networks or wireless LANs[10]

## IV. Methodology

### A. Cross Layer Design For Service Differentiation

In this work, we consider two traffic classes: best-effort traffic and time-bounded traffic (e.g., voice, video). When a packet arrives from the application layer at the network layer, it is put into different fist-in-first-out (FIFO) queues. Based on the different channel conditions estimated by the MAC layer (We have considered Bandwidth), the network layer takes different actions when reading packets from the queues. At the same time, the MAC layer takes different actions according to different traffic types and network packet timeout requirements

### B. Scheduling and Optimization Model in proposed Cross layer Design

Fig.2 shows the framework of our proposed scheduling and optimization scheme. Since we do not consider the impact of the transport layer, and just use simple transport protocol, such as UDP. The transport layer is ignored.

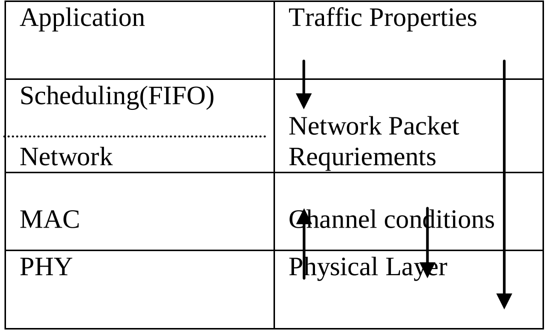

**Fig.2** ***Scheduling and optimization through cross layer Design***

Information exchanged between layers includes: traffic properties, network packet requirements, and channel conditions. Traffic property information is shared between the network and MAC layers. In our model, the traffic properties include: traffic type of Packet, transmission delay bound. Traffic type is set as the type-of-service (TOS) field and traffic category (TC) field in the network and MAC layers, respectively. Delay bounds are pre-defined values set by the application layer that are conveyed to the network and MAC layers through inter-layer interfaces. The application layer can determine whether to receive a packet based on the traffic properties after the packet arrives from the network layer.

In each station, after transmitting a packet, the network layer must wait for the MAC layer to inform it, if the packet is successful or not, Or for the transmission timeout. If MAC informs the network layer the transmission outcome before the packet timeout the network layer can initiate the re-transmission process, or just discard the packet and transmit the next one. However, if the packet timeout occurs first, the network layer will send the same or another packet to MAC when it is still transmitting the previous one, which will cause useless transmissions and severe network contention. Thus, to avoid MAC transmitting the packets that have already timed out at the network layer. packet requirements are conveyed to the MAC layer. Here, the requirements mainly refer to the packet timeout periods for the best-effort and time-bounded traffics denoted by $T_{rb}$ and $T_{iv}$ respectively. Therefore, based on the timeout periods, traffic delay bounds the MAC layer can determine whether to transmit or not after reading a packet from its FIFO.

The estimated channel condition at the MAC layer is conveyed to the network layer to aid the scheduling scheme to work efficiently.

Information exchanged between layers includes: traffic properties, network packet requirements, and channel conditions.

Traffic property information is shared between the network and MAC layers. In our model, the traffic properties include: traffic type of Packet, transmission delay bound. Traffic type is set as the type-of-service (TOS) field and traffic category (TC) field in the network and MAC layers, respectively. Delay bounds are pre-defined values set by the application layer that are conveyed to the network and MAC layers through inter-layer interfaces. The application layer can determine whether to receive a packet based on the traffic properties after the packet arrives from the network layer.

In each station, after transmitting a packet, the network layer must wait for the MAC layer to inform it, if the packet is successful or not, Or for the transmission timeout. If MAC informs the network layer the transmission outcome before the packet timeout the network layer can initiate the re-transmission process, or just discard the packet and transmit the next one. However, if the packet timeout occurs first, the network layer will send the same or another packet to MAC when it is still transmitting the previous one, which will cause useless transmissions and severe network contention. Thus, to avoid MAC transmitting the packets that have already timed out at the network layer. packet requirements are conveyed to the MAC layer. Here, the requirements mainly refer to the packet timeout periods for the best-effort and time-bounded traffics denoted by $T_{rb}$ and $T_{iv}$ respectively. Therefore, based on the timeout periods, traffic delay bounds the MAC layer can determine whether to transmit or not after reading a packet from its FIFO.

The estimated channel condition at the MAC layer is conveyed to the network layer to aid the scheduling scheme to work efficiently.

### C. Channel Condition Estimation at the MAC Layer

As described above, the MAC layer needs to estimate the channel condition and predict the packet re-transmission time.

In our scheme, we estimate the channel condition through Monitoring the delay of special MAC layer packets (RTS and CTS). Because they are very low load traffic we have assumed the ideal delay of the packets an inverse proportional function of the distance between the nodes. Predicting the amount of data already in the channel simulates excess delay. By using the delay we have estimated free bandwidth (Amount of packets a link can accommodate) at any instance. We denote the changes as presence of packets or packets reaching the destinations, during one specified period. Total buffer in the channel therefore is the sum of the length of all the data packets.

We use three channel states to represent the channel conditions: busy, normal, and idle, defined as follows.

- If there are significant amount of packets in the channel it is said to be busy.
- If there are no packets in the buffer, we denote the channel state as idle.
- The channel remains in the normal state if the channel is neither busy nor idle.

In conventional cross layer design, once the channel state changes, the MAC layer will notify the network layer about the channel state immediately. But in our work, MAC would sense the channel periodically or before a requested transmission and analyze if the amount of load network wants to transmit, can be transmitted or not. This information would be passed to the network layer.

With the help of this information, the network layer would estimate packet transmission time.

We have also proposed a modification in this scheme by the following theory. The network layer once knows from MAC that current length of buffer cannot be transmitted; it reduces the buffer and asks the MAC about the possibility of this new buffer to reach successfully. Hence delay can be minimized significantly.

There are three packet types: best-effort traffic packets, time-bounded traffic packets, and routing control packets (including route request, route reply and route error packets). For packet type t, as the control packet length is less, it would be assumed that the control packets can always reach the destination.

$T_d$ is used to denote the average delay. Which is computed as follows,

$$T_t=1/N\sum_{i=1}^{N} T_N$$

where N denotes the number of packets that the station has transmitted, $T_u$ represents the transmission time of the ith packet. In the following we use b, v, and c as the subscript to represent the best-effort, time-bounded and routing control packet types. Actually, the packet transmission time includes waiting time, deferral time, back off time and actual transmission time. Assuming that there are n packets remained in the Network buffer.

Where $n = n_b+ n_c+ n_t$ . (2)

As soon as a request arrives from the network layer, the MAC layer analyze the request type and if it is a data packet computes the probability of the current packet reaching the destination.

The throughput is defined as

Tp=Number of Packets Received (nb+nt) *100
/Number of Packets Transmitted(nb+nt) (3)

### D. Scheduling and Optimization Scheme

Once a packet arrives at the network layer, it will be put into a FIFO according to its traffic type. Then the station will schedule packet transmissions based on the different channel states estimated by the MAC layer. The scheduling algorithm is as follows. When the channel condition is busy, the network layer will transmit the best-effort packets only after finishing the transmissions of all time-bounded packets. In normal state, if the time-bounded FIFO size is smaller than $C_r$ , then the network layer will transmit the packets according to their insertion time into the two FIFOs. Otherwise, the station will transmit the packets in the time-bounded FIFO first. The station employs round robin to transmit the packets in FIFOs in the idle state.

In order to improve service differentiation and network Performance we adopt the following two optimizations in our scheme.

The first is the traffic properties sharing between the network, and MAC layers described above. At the network layer, when the station reads one packet from a FIFO, it first checks whether the packet has violated its delay bounds. If bound is violated, it will be discarded. Otherwise, it will be transmitted. The MAC layer takes similar actions for different packets. That is, the network and MAC layers can both identify the different traffics of one packet through reading the TOS or TC field. At the network layer, once a packet arrives from the MAC layer, the station can determine whether to receive or discard it based on the delay bounds. Thus, the time delay can be guaranteed.

In general there is a second optimization which is as follows. In IEEE 802.11, every station needs to acknowledge every MAC protocol data unit (MPDU) (e.g. RTS, CTS, DATA, ACK or DATA/ACK data transmission process). Whenever the MAC layer succeeds or fails to transmit a packet, it will notify the network layer immediately. This is similar to the acknowledgement process in the DSR model. But, in our scheme we assume that once MAC transmits the packet, it will always be successful as part the design. If it is not, the packets would be considered to have lost. As we have suppressed the Transport layer, retransmission of the packets would not be considered. Hence the second optimization becomes an insignificant one in our work.

Through the proposed optimizations, we can reduce the transmissions that are useless for the destination station. Thus, the contention is reduced, which is important for lightening the network load. In addition this can reduce the memory requirements for the station because the optimizations eliminates storing useless packets in the FIFOs.

### A. Algorithm

```
initialize MAX=number of mobiles
Select Src, Dst
 Generate RREQ from src.
   For I=0;I<MAX;I++
      {
         if (BW[I][j]> RREQ Length)
// this is done by requesting the channel information from MAC
                Transfer Packet.
             Else
             Drop it
      }
 Obtain Shortest path (With Bandwidth filter)
 Transfer TB packet by checking the bandwidth.
       If bandwidth is not sufficient
         {
Reduce packet length and monitor.
         }
if length of packet is zero
   delay++
else
transmit
transmit 2 byte of BE also.
 A packet stays in channel for ideal time(delay)
 If simulation time> packet.end time
     packet.status=reached.(Dead)
 if BW<packet.length
  packet.status=Lost.
Performance graph is plotted.
```

Draw the graph of mobility v/s throughput, mobility v/s delay, node density v/s throughput, data v/s throughput.

## V. SIMULATION

The protocol that we have used is AODV. The existing protocol is updated using the cross layer design approach. Here the transmission is filtered based on the information passed by the MAC layer to the network layer. The MAC layer would sense the channel with the help of transmitting a beacon to the neighbors. It then measures the time for the beacons to come back. This time can be calculated based on either round trip delay or the end-to-end delay. We have used the concept of end-to-end delay as we have assumed that the channel is homogeneous. The Network layer waits for the response to come from the MAC layer. If the conditions are suitable, packets are forwarded else the network layer waits for the channel state to improve. The sensing is done periodically in order to avoid excess beacon overflow by the MAC layer.

## VI. RESULTS

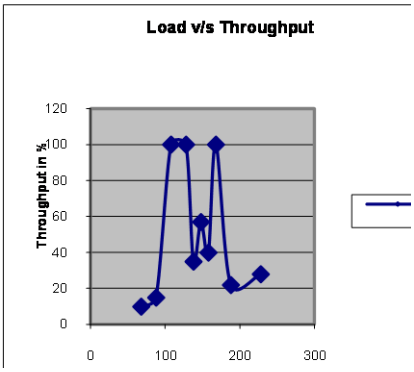

**Fig .3 Load v/s Throughput when 10 Nodes are distributed over an area of 100x100**

The performance demonstrates that the throughput never degrades exponentially like the one in AODV. Due to network variations, it may de very low at some point of time but the system recovers very fast and it can attain maximum throughput even when the load is high.

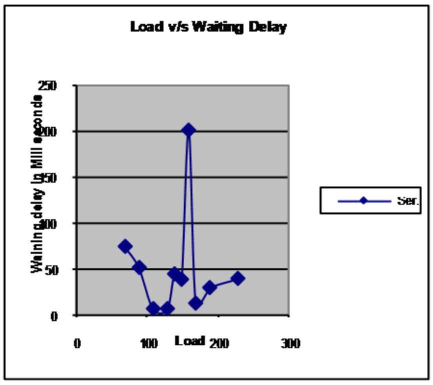

**Fig 4. Shows the Delay v/s Load**

The performance demonstrates that the delay is always kept at a lower level. But due to lack of bandwidth, sometimes the delay may be huge, but those situations are quickly recovered. In AODV, the delay is exponential against load. But the designed approach succeeds at controlling the delay. It must be noted here that the delay is the total waiting delay of best effort and time bound. The experimental results shows that the Time bound delay is much lower than that of Best effort Delay and Time bound throughput is always higher than that of best effort.

## VII. CONCLUSION

The Cross layer design approach solves the problem of transmission without any prior knowledge about the channel. Due to this, the QOS is improved. As the network layer maintains the queue the possibility of buffer overflow is also very less.

The cross layer design defines a way for communication between various layers. In our approach the work depicted the cross communication between the MAC and the Network layer. As the time stamp associated with both the layers is independent of each other, the sensing and transmission modes are independent. It has restricted the channel sensing to requirement specification only. i.e. Rather than checking the channel continuously, MAC layer senses the channel only when the network layer asks it to do so. Therefore the channel is never over flooded by the beacons. It is observed that the efficiency or the performance of this protocol is significantly high over the AODV protocol.

For Packet lifetime ideal channel delay time is considered. In real time it would have to consider the congestion status of the channel also. Hence considering the effect of network layer beacons and at the same time the congestion state delay can do further modification on the work.

## REFERENCES

[1] J. Broch. D. B. Johnson, D. A. Maltz. "The Dynamic Source Routing protocol for mobile ad hoc networks." Internet Draft. draft-ietf-manet-dsr-* .txt.

[2] W. H. Yuen, H. Lee, T. D. Andersen. "A simple and effective cross layer networking system for mobile ad hoc networks," IEEE PiMRC. Lisboa. Portugal, September 2002.

[3] T. ElBatt and A. Ephremides. "Joint scheduling and power control for wireless ad-hoc networks," IEEE InfoCom. New York, NY, USA. June 2002.

[4] G. Chun, D. R. Jaisinghani. M. Baker, "Evaluation of scheduling algorithms in mobile ad hoc networks," ACM MobiHoc. Lausanne. Switzerland. June 2002.

[5] X.Pallot, L. E. Miller. "Implementing message priority policies over an 802. I I based mobile ad hoc network,.' IEEE MILCOM. Washington. D.C., USA, October 200 I .

[6] I. Aad, C. Castelluccia, "Differentiation mechanisms for IEEE 802.1 I . IEEE InfoConi, Anchorage, Alasaka, 1JSA. April 200 1.

[7] X. Yang, N. H. Yaidya, "Priority scheduling in wireless ad hoc networks," ACM MobiHoc, Lausanoe, Switzerland, June 2002.

[8] H. Luo, S . Lu, V. Bharghavan, "A new model for packet scheduling in multihop wireless networks," ACM MobiCom, Boston. Massachusetts. USA, August 2000.

[9] S. Lu. V. Bharghavaii. R. Srikant. "Fair scheduling in wireless packet networks." 1 EEE/ACM Transactions on Networking . Volume 7. Issue 4 August 1999

[10] A. Dugar, N. Vaidya. P. Bahl. "Priority and fair scheduling in a wireless LAN," Technical Report TRO 1-006. Computer Science Department. Texas A&M University, April 7, 2001.

[11] X. Pallot, N. Rous, J. S. Pegon, "README File for NlST DSR Model." File and Model found at http://w3 .antd.nist.gov/wctg/DS Rreadme.pdf. December 2000.

[12] J. Yoon, M. Liu, B. Noble, "Random waypoint considered harmful." IEEE InfoCom, San Francisco. California, USA, March 2003.