A Cross layer Admission Control On-demand Routing Protocol for QoS Applications

Noureddine Kettaf, Hafid Abouaissa, Thang Vuduong† and Pascal Lorenz

University of Haute Alsace 34, rue de Grillenbreit 68008 Colmar Cedex, France
†France Telecom 2, Avenue Pierre Marzin, 22307 Lannion Cedex, France

Summary
Nowadays, the cross-layer design approach is the most relevant concept in mobile ad hoc networks which is adopted to solve several open issues. It aims to overcome ad hoc networks performance problems by allowing protocols belonging to different layers to cooperate and share network status information while still maintaining separated layers. In particular, the mechanisms on how to access the radio channel are extremely important to guarantee QoS (Quality of Service) and improve application performance.
In this paper, we propose a cross-layer routing protocol for mobile ad hoc networks based on the cooperation between the Admission Control enabled On-demand Routing (ACOR) protocol and the new QoS-based IEEE 802.11e MAC layer. This enhanced protocol aims to find a feasible route according to application QoS requirements. The performances of the proposed cross-layer protocol are extensively investigated by simulations. Results obtained show that compared to ACOR based on 802.11b, cross-layer ACOR provides an efficient QoS support in mobile ad hoc networks with low overhead and high reliability.
Key words:
Routing, QoS, Ad hoc, Cross layer, 802.11e, Bandwidth, delay.

Introduction

A mobile ad hoc network is a set of wireless mobile nodes dynamically forming a temporary network. The goal of this architecture is to provide communication facilities between end-users without any centralized infrastructure. In such a network, each mobile node operates not only as a host but also as a router.
Providing end-to-end quality of service (QoS) support guarantees in ad hoc networks is even more challenging for the following reasons. First, every node can randomly change position, the topology is generally unpredictable and the network status is imprecise. Second, the communication channel is wireless, so it shall suffer fading, time variation and multi-path effects, and third, non-centralization in the network and, thus, network resources cannot be assigned in a predetermined manner. Consequently, numerous reactive (On-demand) and proactive (table driven) [1], [2], [3], [4] QoS-based routing protocols were proposed, however, comparative investigations [5], [6] and [7] proved that proactive protocols are more liable to suffer performance deterioration than reactive protocols, due to stale route information. Though, on-demand routing approaches have been shown to perform well, they generally lack the support for QoS with respect to data transmission [8] but they suffer absence of resource management mechanisms such as admission control.
In QoS routing, admission control aim to provide a route containing enough unused resources to carry a flow, without interfering with nearby ongoing traffic and guarantees to satisfy a set of predetermined service performance constraints for the user in terms of bandwidth, delay, packet loss and so on.
Recently it has become evident that a traditional layering network approach (separating routing, scheduling, rate and power control) is not efficient for ad hoc wireless networks [9]. This is primarily due to the interaction of links through interference, which implies that a change in resources allocation on one link can induce changes in the capacities of all links in the surrounding area and changes in the performance of flows that do not pass over the modified link.
The main building blocks of a wireless network design are routing, rate control, medium access (scheduling) and power control. These building blocks are divided in layers. Typically, routing is considered in a routing layer and medium access in a MAC layer, whereas power control and rate control are sometimes considered in a PHY and sometimes in a MAC layer.
Enhanced Distributed Channel Access (EDCA) is a contention based Hybrid Coordination Function (HCF) channel access specified in 802.11e [10], [11]. The goal of this scheme is to enhance the DCF access mechanism of 802.11 [12] and provide a distributed access approach that can support service differentiation. The proposed scheme provides capability for up to four types of traffic classes. It assigns a short (Contention Window) CW to classes that should have high priority in order to ensure that in most cases, higher-priority classes should be transmitted before

Manuscript received September 5, 2006.
Manuscript revised September 25, 2006.

lower-priority ones. In essence, the CW_{min} parameter can be set differently for different priority classes, yielding higher classes with smaller CW_{min}. For further differentiation, in 802.11e different IFS Inter Frame Space (IFS) can be used according to traffic classes called in our paper Access Category (AC). Each AC within the host behaves like a virtual host: it contends for access to the medium and independently starts its *backoff* timer according to the basic scheme algorithm.

Despite of many enhancement mechanisms that have been proposed to achieve QoS support, performance evaluation results in mobile networks show that EDCA still suffer from significant throughput degradation and high latency caused by the increasing time used for channel access negotiation and network characteristics [13]. In this context, the route quality plays an important role in the success of application of delivery and QoS support.

Our contributions in this paper are as follows. First, we give an overview of the ACOR [14] (Admission Control enabled On-demand Routing) protocol which is designed to support Soft QoS [15] or better than BE (Best-Effort) service, rather than guaranteed Hard QoS. We introduce detailed computations of available bandwidth and end-to-end delay. The bandwidth and delay metrics are represented by elementary local cost functions at each node to ensure implicit resource reservation. Furthermore, a global function to represent the end-to-end cost of a route is accumulated from the source node to the destination one; it contains the addition of the elementary cost functions. Second, we describe the cross-layer algorithm, which is based on the marking technique. We define four types of Access Categories, where AC0 is the highest-priority and AC3 is the lowest one.

Routing with multiple QoS metrics is a NP-complete problem [16]; however, this problem is beyond the scope of this paper and we will deal with it in a future work.

The rest of this paper is organized as follows. Section II describes the proposed Admission Control enabled On-demand Routing (ACOR) protocol. Section III presents cross-layer QoS-based algorithm. We also devote Section IV to performance evaluation. Finally, Section IV draws conclusions and highlights ideas for future works.

2. Admission Control enabled On-demand Routing protocol: An Overview

The limited resources in mobile ad hoc networks have made designing of an efficient and reliable QoS routing strategy a challenging problem. However, a simple routing strategy is required to efficiently use the limited resources while at the same time being adaptable to the changing network conditions such as: network size, traffic density and network partitioning. In parallel, the routing protocol may need to provide different levels of QoS to different types of applications and users. Motivated by the latter, ACOR was proposed to efficiently provide end-to-end support for QoS by introducing simple functions which represent QoS metrics and ensure implicit resource reservation. The aim of admission control is to determine whether the available resources can meet the requirements of a new flow maintaining bandwidth and delay levels for existing flows.

Specifically, an ad hoc network can be modeled as an undirected graph *G(V,E)*, where *V* is the set of nodes, and *E* is the set of links. Motivated by the *colored* sub-graphs formulation presented in [17], we divide each link to sub-links represented by elementary cost functions for QoS metrics (i.e.: bandwidth, delay, energy, bit error, packet loss probability, security, etc. For purposes of clarity in this paper, we only focus on bandwidth and delay. We will examine other metrics like delay jitter, packet loss probability, security and energy, in a future research. On one hand, the bandwidth at each node is represented by $F_{\updownarrow}$ which is a ratio of the requested bandwidth *B* by an application and the supported bandwidth by a link $B_{\mp\to\square}$ in addition to the residual bandwidth $B_{\square}$ at a node, on the other, the delay is represented by $F_{\square}$ which is also a ratio of supported delay *D* by an application and the accumulated hop-by-hop delays with the upper bound of delay $D_{\mp\to\square}$. The sum of the elementary cost functions ($F_{\updownarrow}$ and $F_{\square}$) at each node is added to the global cost function $F_{\square}$ received in the route request packet during the route discovery to represent a route's end-to-end cost.

ACOR is a reactive routing protocol. Therefore, routes are determined only when needed. A "Hello" packet is used to detect and monitor links to neighbors and to estimate delay.

2.1 Bandwidth and Delay Estimation

To offer bandwidth guaranteed QoS, the residual (unused) end-to-end bandwidth must be known. In wired networks this is a trivial task since the underlying medium is a dedicated point-to-point link with fixed capability. However, in wireless networks the radio channel of every node is shared with all its neighbors. Due to the shared medium, a node can successfully use the channel only when all its neighbors do not transmit and receive packets simultaneously. We use a simple and efficient method to estimate residual bandwidth B_{res} by listening to the channel of the IEEE 802.11 MAC. This method is based on the ratio of free and busy times.

Specifically, the DCF mode is based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) algorithm combined with the Network Allocation Vector (NAV) to determine the busy/idle status of the medium. A mobile node must sense the medium before initiating the transmission of a packet. If the medium is sensed as being idle for a distributed inter-frame space (DIFS) period, the mobile node can transmit a packet. Otherwise, transmission is deferred, and a *backoff* procedure is started. In the *backoff* process, the mobile node computes a random value ranging from 0 up to the current CW size. A *backoff* period is computed as this random value multiplied by the duration of the slot time. This *backoff* interval is used to initialize the *backoff* timer, which is decreased only when the medium is sensed as idle. As soon as the *backoff* timer expires, the mobile node transmits the packet.

The MAC detects that its channel is free when the value of the NAV is less than the current time, receive state is idle and send state is idle. On the other hand, the MAC claims that the channel is busy when the NAV sets a new value receive and send states change from idle to any other state. A node estimates its B_{res} as the channel bandwidth times the ratio of free time to overall time. B_{res} is cross layered to the network layer to compute F_b. Hence, the local elementary function F_b is given by

$$F_b = \frac{B}{B_{\max} - (B_{res} + B)} \quad (1)$$

Where:

B: is the requested bandwidth;

B_{max}: is the maximum bandwidth supported by a link, e.g.: 2Mb/s, 11Mb/s, 54Mb/s;

B_{res}: is the residual bandwidth.

To admit bandwidth requirements B, the following inequality must be verified.

$$B_{res} + B \le B_{\max}$$

In the other hand, estimating end-to-end delay in mobile ad hoc networks is a crucial problem due to the unsynchroni-zed nature of the network. In ACOR, a "Hello" packet is used to estimate the delay to the next neighbor. When a node transmits a Hello packet, it starts a local timer called $d_{scheduled}$ to wait for the Hello's acknowledgement. Hence, the node the originator of the Hello packet records the sending time of its Hello, d_1. Upon receiving the acknowledgement, the originator records again the receiving time, d_2. The time difference between d_2 and d_1 (d_2 - d_1) is then the RTT (round trip time). However, to estimate an accurate delay we should consider an error probability factor D_e. Comparatively, D_e is a small value to estimated delay, represents the queuing delay at each relaying node, the packet transmission time (we assume all nodes with same data rate for each direction) and the propagation delay. Hence, the estimated delay at node i is D_i = RTT + D_e. At each node, the elementary function is given by

$$F_d = \frac{D}{D_{\max} - (\sum_{i=1}^{j} D_i + D)} \quad (2)$$

Where

D: is the estimated delay to a next hop;

D_{max}: is the upper bound of delay supported by a flow;

$\sum_{i=1}^{j} D_i$: is the accumulated delays from node i (e.g.: source) to node j.

Also, to admit a delay D, the following inequality must be verified.

$$D_{\max} \ge \sum_{i=1}^{j} D_i + D$$

Specifically, the local cost functions F_b and F_d are hyperbolic limited respectively by B_{max} and D_{max} which are parallel to the ordinate axis.

The end-to-end cost of a route is represented by a global cost function called F_g that results from the sum of local cost functions (F_b + F_d) evaluated at each node participating in the route discovery (described bellow in section 2.3). The value of F_g is accumulated from source where is set to cipher to destination. In particular, a high value of F_g represents an overloaded route.

$$F_g = \frac{B}{B_{\max} - (B_{res} + B)} + \frac{D}{D_{\max} - (\sum_{i=1}^{j} D_i + D)} \quad (3)$$

2.2 Neighborhood maintenance

Neighborhood information is very important in ACOR since it provides the mobility information, traffic and local topology. This information is critical for traffic measurement and route failure detection and recovery.

To maintain the neighborhood information, each node is required to periodically disseminate a "Hello" packet to announce its existence and traffic information to its neighbor set. The Hello packet contains the B_{res} of the originator. The Hello is sent at a default rate of once per second with time to live (TTL) set to 1. Every node in the network receives the Hello packet from its neighbors, maintains a neighbors list that contains all its neighbors with their corresponding traffic. If a node does not receive a Hello packet from a neighbor within P_{lost} (unity of time), the node assumes that the link to the neighbor is currently lost.

2.3 Route discovery and resource reservation

ACOR conforms to a pure on-demand routing protocol. It neither maintains any routing table, nor exchange routing information periodically. When a source node needs to establish a route to another node, with respect to a specific QoS requirement, it disseminates a route request (RouteRequest) packet that includes mainly, the requested bandwidth, delay and also the global cost function F_g which will be accumulated at every participating node to establish this route. Hence, each intermediate node, upon receiving the route request packet, tries to respond to QoS requirements by applying a bandwidth decision in reserving the requested bandwidth B, and appending the value of its local functions F_b and F_d to the received F_g. Afterwards, the node sets a route entry in its routing table and rebroadcasts the route request with the updated F_g to next hop neighbors.

To reduce the overhead generated by the control packets during the route discovery and contrary to other routing protocols [18, 19], ACOR adopts two efficient optimization mechanisms. One is applied on nodes that cannot support QoS requirements, thus ignore the route request packet. The other is for every intermediate node and based on the rebroadcast of the first received route request packet. In particular, when an intermediate node receives several route requests to one destination, they may contain different values of F_g. By applying the optimization mechanism (please see Fig. 1), the node records in its routing table the incoming values of F_g with the addresses of their nodes the originators, then, rebroadcasts the first received route request with the updated value of F_g.

Once, the destination node receives the route request packet, it responds by unicasting a route reply (RouteReply) packet which principally includes the source node's address and the end-to-end value of F_g to the source node along the reverse route. If one of the intermediate nodes has set in its routing table several sources with different values of F_g of a route request, then, must update the F_g value by deducting the received F_g in the route reply packet from the sent in the route request packet to extract the cost value which will be appended to the recorded F_g's of other neighbors, whose route request were not rebroadcast. Otherwise, it simply forwards the route request without modifying the value of F_g to its neighbor the originator of the route request.

2.4 Loop-free routing

The operation of ACOR is loop-free, and by avoiding the Bellman-Ford "counting to infinity" problem offers quick convergence. ACOR uses a destination sequence number to indicate the control packets freshness for each data flow.

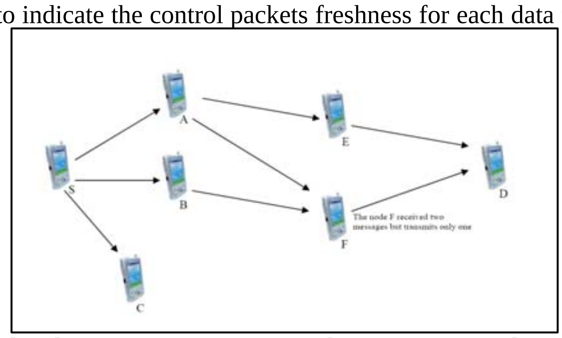


Fig. 1. ACOR's overhead optimization mechanism

The destination sequence number is maintained at each node for the IP address of the destination for which the route table entry is maintained. This sequence number is updated whenever a node receives new information about the sequence number from route request, route reply, or route error packets that may be received related to that destination. ACOR depends on each participating node in the network to own and maintain its destination sequence number to guarantee loop-freedom of all routes towards that node.

Using this technique, ACOR guarantees that along the discovered route no three intermediate nodes will be within the same neighborhood area of each other. This property reduces the control overhead by minimizing the transmission of unnecessary control packets.

2.5 QoS route recovery

ACOR adopts the common approach for route break detection used in most existing ad hoc routing protocols is by neighbor lost detection [18], i.e., the hello packet from a lost node does not arrive to its neighbor in time. A route error packet is sent back to the source of data to notify about the break. As the route error propagates towards the source node, each node invalidates routes to any unreachable destinations and releases the reserved resources. Then, the source node initiates the reroute process if it still has data to send. However, this approach may take several seconds which is inadequate to delay sensitive applications; also, it may engender excessive control overheads, and stales recourses. In essence, to efficiently recover QoS routes, we use a more efficient method based on the bandwidth reservation lifetime at the destination node to notify about the route break. Hence, if the destination node fails to receive data of reserved flow before its reservation lifetime, the destination node initiates the QoS route recovery. This technique aims to rebuild a route in the reverse way. The destination node broadcasts a reverse route (ReverseRoute) packet backward to the source of data. The reverse route packet is treated in the same manner as a route request. During the propagation of the reverse route packet towards the source node, the evaluation of the reverse route with local cost functions F_b and F_d, also the global F_g to represent the end-to-end quality of a route, in addition to the loop freedom to ensure route freshness are included. To perform accurate resource reservation, the destination node includes the known QoS requirements received previously in the route request. Once, the source of data receives the reverse route packet, it chooses the adequate route with respect to the value of F_g. However, due to frequent topology changes or packet loss, the reverse route packet may not arrive to the source node within a pre-defined tolerable time. The source of data may trigger a new route discovery by broadcasting a new route request for a route as described in 2.3, or send data on best effort via the reverse route of the route error packet.

3. The Cross Layer Algorithm

The proposed scheme is based on the interaction between the 802.11e MAC and ACOR routing protocols. The current 802.11e draft basically defines four access categories; AC3 corresponds to the highest access priority, and AC0 to the lowest. Based on this traffic specification it is possible to differentiate ACOR's flow packets at the MAC layer. In this context we propose a marking algorithm that uses a reserved field in the header of ACOR's packets to map the traffic flows to a suitable traffic classes and thus allows for QoS continuity between the different OSI layers.

Thus, each packet arrives at the MAC layer along with a specific priority value (please see Fig. 2). According to the marking algorithm flow packets are encapsulated into a QoS data frame, where the traffic identifier (TID) field (in the MAC header) is used in order to differentiate between AC[i]'s frames. Here, the TID field is 4 bits, and can carry values between 0 and 15 represent traffic flows as specified in [10].

The marking algorithm

11 then QoS_TID = t_1 */*Insert the packet in AC3 queue*/*

10 then QoS_TID = t_2 */*Insert the packet in AC2 queue*/*

01 then QoS_TID = t_3 */*Insert the packet in AC1 queue*/*

$$t_1 < t_2 < t_3 \text{ and } t_1, t_2, t_3 \in [8,15].$$

The choice of AC is based on QoS metrics such as bandwidth and delay. Thus, the real time and delay sensitive applications are mapped to the highest priority access category (AC3). We argue this by the fact that a voice over IP (VoIP) flow is very sensitive either to packet latency and loss leads to voice quality degradation. Since video streams require a considerable bandwidth, a bounded delay and a minimum loss rate, we map them on AC2.

Finally, data does not require any QoS metrics is mapped to lower priority access category AC1.

In addition to the traffic access categories, we use the maximum retry limit. In fact, the 802.11e MAC layer uses a retry count variable, which is incremented after each transmission fails. Therefore, when the retry count exceeds the maximum retry limit, the failing frame is dropped. We use this parameter (maximum retry limit) to unequally protect the high priority information. One solution is to increase the retry of an important packet, at the expense of losing less important packets (as long as the receiver can accommodate this extra retry latency). However, we should keep in mind that a high retry limit's values decrease the frame drop rate, but may throttle the data rate and throughput because of longer *backoff* time, while a smaller retry limit value increases frame drop rate but shortens *backoff* time. Performing the precise analysis to find the appropriate values that satisfy these constraints is

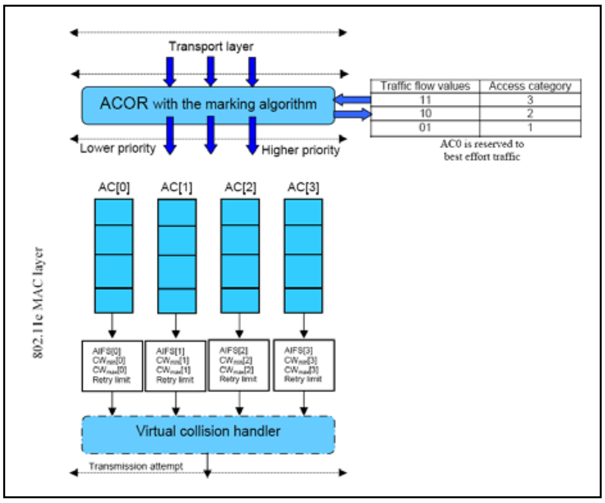


Fig. 2. Cross layer ACOR protocol

out of the scope of this paper, and we would like to leave it for future studies. In this paper we consider the following assumption to fix the retry limit for each AC:

- Since AC3's and AC2's CW range is small enough (small CW_{min} and CW_{max}), we use the maximum value allowed by the 802.11 MAC layer.
- For AC1 and AC0 we choose a smaller maximum retry count. This limits the MAC's retransmission packets of both AC1 and AC0, and hence discards packets that are too late to be sent (AC0 and AC1 packets wait longer in the queue than AC3 and AC2).

4. Simulation results

In order to evaluate the advantage of the proposed cross layer ACOR protocol, we have constructed simulations using ns2 [20]. The cross layer ACOR protocol is compared to the basic ACOR with 802.11b with different node's speeds and multiple traffic flows.

4.1 Scenario description

The simulated scenarios consist of 50 located in uniform distribution within a square area of 670m forming a multi-hop network. These scenarios are generated by the random mobility model [21]. Our simulation uses different types of traffics to evaluate service differentiation. Four queues are used in each active node. The queue implementing AC3 (highest priority) generates packets with packet size of 160 bytes and packet interval is 20ms, which corresponds to 64 Kbit/s PCM audio flow (high sensitive-delay application). The AC2 and AC1 traffic queues generate packets of size equal to 1280 bytes each 10ms which corresponds to an overall sending rate of 1024Kbit/s. Finally, the queue without QoS requirements AC0 in each active node generates packets with 256Kbit/s sending rate using a 200 bytes packet size. Note that the number of source nodes is 20. Moreover, the nominal bit rate is 2 Mb/s.

4.2 Performance metrics

In order to evaluate the performance of the different schemes, the following metrics are used:

- *End-to-end delay.* The latency incurred by the packets between their generation time and their arrival time at the destination. This metric indicates the performance of the admitted flows.
- *Routing control overhead.* The routing control overhead is the number of control packets propagated by every node in the network divided by the number of data packets received by the destination nodes.
- *End-to-end bandwidth evaluation*. We define the end-to-end bandwidth evaluation as the average available bandwidth on the established route. This scenario is novel and may offer precise information about the capability of QoS routing protocol to offer the requested bandwidth.
- *Pause time*. Affects the relative speeds of the mobile nodes, is varied. When the pause time increases, nodes tend to remain static.

4.3 Results and analysis

We present in this subsection the performance of the basic ACOR based on the 802.11b and the cross layer ACOR based on 802.11e for the various metrics presented above.

Fig. 3. depicts the average end-to-end delay of 20 source nodes aiming to transmit delay-sensitive audio flows to 20 destinations. We can observe that the cross layer ACOR outperforms the basic ACOR which has the longest delay. We argue this by the absence of traffic differentiation, and these results are due to the good affect of including AC traffic classes in the 802.11e MAC layer.

Fig. 4. illustrates the routing control overhead generated by the routing control packets. The cross layer ACOR has fewer number of control packets because active nodes participating in the end-to-end route apply the service differentiation algorithm and forward rapidly delay sensitive packets contrary to the basic ACOR which does not apply the service differentiation algorithm. However, when a route breaks both protocols adopt efficient mechanisms to reduce the control packet overhead. These mechanisms are, in one hand, the re-broadcast of the first received route request packet by an intermediate node, and in the other, the drop of the route request when the node does not support the QoS requirements.

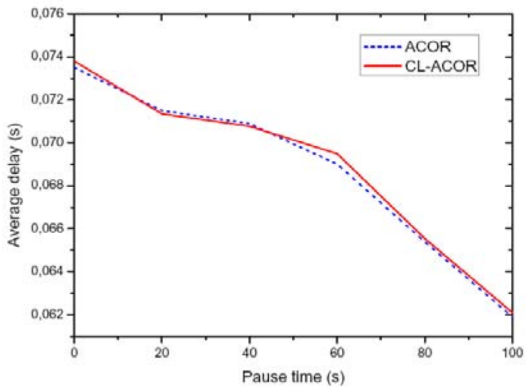


Fig. 3. Average end-to-end delay.

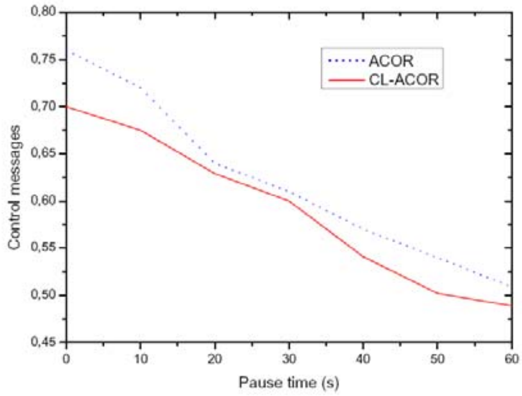


Fig. 4. Routing control overhead.

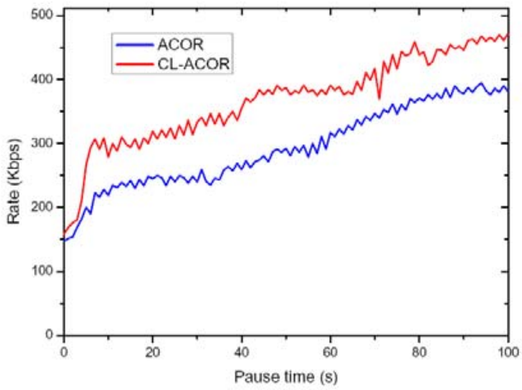


Fig. 5. End-to-end bandwidth evaluation

Fig. 5. presents results from a novel type of scenarios in ad hoc networks. This latter consists of evaluating the available bandwidth along the end-to-end route.

We aim by this scenario to efficiency of the proposed protocol vs. mobility with different speeds of mobile nodes and when our protocol could offer a sufficient amount of bandwidth. In fig. 4 we can observe that the bandwidth is offered better when mobility decreases. This result is reasonable and we can notice that the cross layer ACOR outperforms the basic one. This is due to the service differentiation algorithm which enhances the process of packets with high priority.

5. Conclusion and future works

In this paper, presented a new QoS routing protocol called Cross Layer ACOR that ensures robust QoS support for QoS applications over IEEE 802.11e. The proposed protocol is based on a main top-down cross layer interaction that allows the network layer to express the QoS exigencies to an EDCA-based IEEE 802.11e MAC layer.

Experimental results show that the proposed protocol achieves better performances in terms of end-to-end delay and bandwidth. Through these improvements, the cross layer ACOR considerably increases the perceived real time applications.

Several aspects clearly require further investigation, and we proceed to name a few major ones. One important aspect is identifying a cross layer technique to obtain a security-aware ACOR. Another aspect is the incorporation of network optimization criteria within the path selection process, e.g., network utilization, carried load, number of flows successfully routed, etc.

References

[1] Q. Xue and A. Ganz, Ad hoc QoS on-demand routing (AQOR) in mobile ad hoc networks, Journal of Parallel and Distributed Computing, vol. 63, Issue 2, pp. 154-165, Feb. 2003.

[2] L. Chen and W. Heinzelman, QoS-aware Routing Based on Bandwidth Estimation for Mobile Ad Hoc Networks, IEEE Journal on Selected Areas of Communication, Special Issue on Wireless Ad Hoc Networks, Vol. 23, No. 3, March 2005.

[3] S. Chen, K. Nahrstedt, Distributed Quality of Service routing in Ad hoc Networks, IEEE Journal On Selected Areas In Communications 17 (8) (1999) 1-18.

[4] QoS-enhanced OLSR protocol for mobile Ad Hoc networks A. Munaretto, H. Badis, K. Al Agha and G. Pujolle ANWIRE 1st International Workshop, Glasgow, Scotland, April 2003

[5] A. Boukerche, Simulation based comparative study of ad hoc routing protocols, proceedings of the 34th Annual Simulation Symposium, Seattle, WA, April 2001, pp 85-92.

[6] S. R. DAS, et al., Performance Comparison of two On-demand Routing Protocols for Ad hoc Networks, INFOCOM 2000, Tel Aviv, Israel, vol. 1, IEEE press, 2000, pp. 3-12.

[7] J. Boch, et al., A performance Comparison of Multi-hop Wireless Ad hoc Network Routing Protocols, In Proceedings of the MOBICOM 98, Dallas, TX, October 1998.

[8] R. Leung, J. Liu, E. Poon, C. Chan and B. Li. MP-DSR: A QoS-aware Multipath Dynamic Source Routing Protocol for Wireless Ad-hoc Networks, in Proceedings of the 26th IEEE Annual Conference on Local Computer Networks (LCN 2001), pp. 132-141, Tampa, Florida, November 15-16, 2001.

[9] A. Goldsmith and S.B. Wicker. Design challenges for energy -constrained ad hoc wireless networks. IEEE Wireless Communications, 9(4):8–27, August 2002.

[10] S. Mangold, S.choi, P. May, O. Klein, G. Hiertz and L. Stibot, IEEE 802.11e Wireless LAN for Quality of Service, In Proc. Of European Wireless (EW 2002), Florence, Italy, Feb. 2002.

[11] IEEE 802.11e WG. Draft supplement to part 11: Wireless Medium Access Control (MAC) and Physical Layer (Phy) Specifications: MAC Enhancements for Quality of Service (QoS), IEEE std, 802.11e/D9.0, August 2004.

[12] L. Romdhani, Q. Ni and T. Turletti, Adaptive EDCF: Enhanced Service Differentiation for IEEE Wireless Ad hoc Networks. In Proc. of the IEEE Communications and Networking Conference, New Orleans, Louisiana, March 16-20, 2003.

[13] E. Carlson et al. Performance Comparison of QoS Approaches for Ad hoc Networks: 802.11e vs Distributed Resource Allocation, European Wireless, Nicosia, Cyprus, April 2005.

[14] N. Kettaf et al. ”Admission Control enables Routing Protocol -ACOR-” Internet draft, IETF, (draft-kettaf-manet-acor-00.txt), July 2006, (Work in progress).

[15] F. A. Kuipers, P. F. A. Van Mieghem, Conditions that impact the complexity of QoS routing, IEEE/ACM Transactions on Networking (TON), Vol. 13 , Issue 4, pp 717-730, August 2005.

[16] Xiao, H., Seah, K.G., Lo, A. and Chua, K.C., A Flexible Quality of Service Model for Mobile Ad-hoc Network, Proceedings of IEEE VTC2000 - Spring, Tokyo, May 2000.

[17] K.M. Konwar, I.I. Mandoiu, A.C. Russell, and A.A. Shvartsman., Improved Algorithms for Multiplex PCR Primer Set Selection with Amplification Length Constraints, Proc. 3rd Asia-Pacific Bio-informatics Conference (APBC), pp. 41-50, 2005.

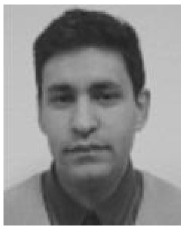
[18] D. B. Johnson and D. A. Maltz, Dynamic Source Routing in Ad-Hoc Wireless Networks. In T. Imielinski and H. Korth, editors, Mobile Computing, pp. 153-181. Kluwer, 1996.

[19] Y. Yang and R. Kravets. Contention-aware admission control for ad hoc networks, Technical Repport 2003-2337, University of Illinois, April 2003.

[20] Network Simulator 2 available at: http://www.isi.edu/nsnam/ns/

[21] M. V. Jean-Yves Le Boudec, Perfect simulation and stationarity of aclass of mobility models, INFOCOM 2005, pp. 834-843, 2005.

Noureddine Kettaf is a Ph.D. candidate in Computer Science at the University of Haute Alsace. He received his Diplom-Engineer from the Institute of Computer Science, Sétif, in 2003 and M.S. degree from the University of Franche-Comté at Besançon in 2004, both in Computer Science. His research interests include computer networking and mobile computing, QoS routing, random graph theory, distributed algorithms and network architectures.



Hafid Abouaissa is an Associate Professor at the University of Haut Alsace, in Colmar France. He received the BS degree from Technical University of Wroclaw, Poland, in 1995, and the MS degree from Franche-Comté University of Besançon, France, in 1996. He obtained the PhD at Technical University of Belfort, France in January 2000. His interests include multimedia synchronization, group communication systems, QoS routing in Ad-Hoc, MPLS, DiffServ, and QoS management.

Thang Vuduong is a senior network architect at France Telecom R&D, working on NGN, IP-based network and IP TV since 2000. Beforehand, he received the B.Sc, M.Sc and Ph.D degrees in Computer Science and Telecommunications from respectively Hanoi University of Technology, The Franco-phone Institute for Computer Science and ENSEEIHT (a French 'Grande Ecole' specializing in technology). He also contributes to the definition of NGN architecture for standardization (ITU-T NGN Focus Group, ETSI's TISPAN). His main research interests are NGN's architecture, QoS-based routing, IP TV, Fixe-Mobile convergence for multi-play services in telecommunications networks.

Pascal Lorenz received a PhD degree from the University of Nancy, France. Between 1990 and 1995 he was a research engineer at WorldFIP Europe and at Alcatel-Alsthom. He is a professor at the University of Haute Alsace and responsible of the Network and Telecommunication Research Group. His research interests include QoS, wireless networks and high-speed networks. He was the Program and Organizing Chair of the IEEE ICATM'98, ICATM'99, ECUMN'00, ICN'01, ECUMN'02 and ICT'03, ICN’04 conferences and co-program chair of ICC’04. Since 2000, he is a Technical Editor of the IEEE Communications Magazine Editorial Board. He is the secretary of the IEEE ComSoc Communications Systems Integration and Modeling Technical Committee. He is senior member of the IEEE, member of many international program committees and he has served as a guest editor for a number of journals including Telecommunications Systems, IEEE Communications Magazine and LNCS. He has organized and chaired several technical sessions and gave tutorials at major international conferences. He is the author of 3 books and 135 international publications in journals and conferences.