Priority Levels in a HIPERLAN Based Forwarding Mechanism for Intermittent Connectivity

Constantine Coutras
Department of Computer Science
Montclair State University
Montclair, NJ 07043
Email: coutrasc@montclair.edu

***Abstract*—As the proliferation of wireless devices continues to outpace the necessary infrastructure to support them everywhere, intermittent connectivity is becoming common in certain wireless access networks. Today's existing network protocols are not resilient to disruption of communication links and communication often fails when faced with sporadic connectivity. Thus, there is a growing interest in Intermittently Connected Mobile Networks (also know as Mobile Opportunistic Networks) and a number of routing protocols and frameworks have been proposed in the literature to address the problem. Different priority classes, each containing different priority levels are introduced. The paper concludes with numerical results from simulation, which takes into account bit errors, hidden nodes and capture.**

***Keywords* - Wireless Networks; Intermitent Connectivity; Forwarding Mechanism.**

I. INTRODUCTION

The ongoing evolution of wireless systems has created a world demanding wireless connectivity everywhere. Most wireless access today is achieved through wireless LANs, operating mostly in infrastructure mode and to a lesser degree in Ad Hoc, smartphones and tablets. But there are still situations where such connectivity is not possible. But wireless access networks worldwide fail to fulfill the promise of continuous, high-bandwidth, and affordable service everywhere.

Intermittently connected mobile networks (ICMNs) are networks where most of the time, there does not exist a complete end-to-end path from the source to the destination. Even if such a path exists, it may be highly unstable because of topology changes due to mobility. Thus, traditional wireless technologies that require an end to end path are not suitable for these networks. There is a growing interest in these networks and many routing protocols have been proposed in the literature to address the problem. Publications presenting aspects of these networks include [1][2][3]. Extremely Opportunistic Routing (ExOR), a unicast routing technique for multihop wireless networks is presented in [4][5]. A middleware design called the Self Limiting Epidemic Forwarding (SLEF) [6], automatically adapts its behavior from single hop MAC layer broadcast to epidemic forwarding when the environment changes from being extremely dense to sparse, sporadically connected. Prioritized Epidemic Routing (PREP) is described in [7]. And lately, FansyRoute is introduced in [8], and a probability-based "Spray and Wait protocol" in [9].

A thorough analysis of different techniques used for routing (direct routing, epidemic routing, randomized flooding and spraying techniques) that also includes contention for the wireless channel in the analysis can be found in [10][11][12]. Finally, the problem of accurately modeling mobility and using realistic traces of mobility in these networks is analyzed in [13]–[16].

The High PErformance Radio Local Area Network (HIPERLAN) has been developed by the European Telecommunications Standards Institute (ETSI). The HIPERLAN protocol functionality is presented in [17]. Other sources of information and analysis of the HIPERLAN protocol can be found in [18]–[24]. A study of the protocol's performance for asynchronous traffic that takes into account the phenomena of hidden nodes and capture is presented in [25] and a study of the protocol's performance for real-time traffic that takes into account the phenomena of hidden nodes and capture is presented[26].

As an alternative to the approaches mentioned above for dealing with intermittent connectivity, we wish to provide a solution for intermittent connectivity based on modifying an existing wireless protocol (HIPERLAN) so that it operates in the same exact way either the wireless environment in one of intermittent connectivity or not. This modification stays at the medium access layer and thus does not involve a routing algorithm which would be found at a higher layer. Thus, this approach differs from the ones referenced above and a one to one comparison is not saught.

In this paper we extend our previous work, found in [27] on modifying the HIPERLAN Channel Access Control (CAC) Layer protocol to accommodate for loss of connectivity, by introducing two classes of service, each containing different priority levels. In Section 2, an overview of the HIPERLAN CAC Layer Protocol is given in which channel access in the presence of hidden nodes is also taken into account. In Section 3, our modification of the HIPERLAN CAC Layer for intermittently connected mobile networks is presented. We then discuss packet transmission in the presence of hidden nodes, capture and bit errors. In Section 4, classes of service and priority levels are introduced. In Section 5, numerical results from simulation study are presented. And in Section 6, the conclusion is given with directions for future work.

II. OVERVIEW OF THE HIPERLAN CAC LAYER PROTOCOL

The CAC layer is actually the “lower sublayer” of the MAC layer that basically deals with the mechanism of accessing the channel (EY-NPMA mechanism).

The three phases of the EY-NPMA mechanism constitute the contention phase of the *Synchronized Channel Access Cycle*.

In Figure 1 we see a renewal interval, its components, and their components as well. Transmission is denoted by black color, while its absence is denoted by white color, and a different shade filling is used for the synchronization slot.

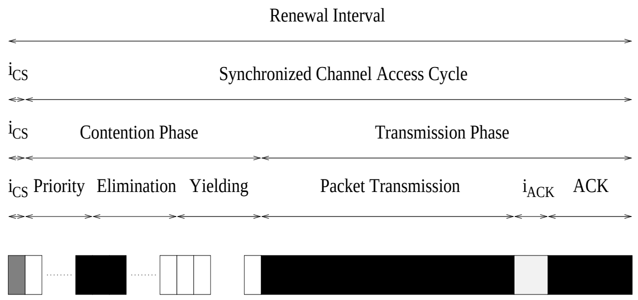

Fig. 1. The EY-NPMA mechanism.

In the prioritization phase, an active node of priority n must signal its intention to access the channel by transmitting a burst during the nth time slot, provided that no active nodes of higher priority have already signaled their intention to access the channel. A total of m_{CP} priority levels, from 0 to $m_{\mathrm{CP}}-1$ is assumed.

In the elimination phase, each active node bursts a signal for a random number of time slots and then listens to the channel; if another active node is still bursting this active node is eliminated, otherwise it may continue into the yielding phase. In the elimination phase we have a maximum of m_{ES} elimination slots. The probability of bursting in an elimination slot is p_{E} . The maximum burst allowed is $m_{\mathrm{ES}}-1$ time slots.

In the yielding phase, each active node listens for a random number of time slots and then, if the channel is still free, starts a packet transmission. In the yielding phase we have a maximum of m_{YS} yielding slots. The probability of yielding in a yielding slot is p_{Y} . An active node can listen for a maximum of $m_{\mathrm{YS}}-1$ time slots.

A. Channel Access in the Presence of Hidden Nodes

Due to the high channel speed used in HIPERLAN we can assume relatively limited *node mobility*. This means that if two nodes are hidden from each other in the beginning of a renewal interval they will remain hidden through out that renewal interval.

During the prioritization phase, for an active node of lower priority to falsely determine that it may continue into the next phase, it will have to be hidden from all active nodes of higher priorities.

From [25][26] we have that an active node might falsely determine that it has survived the elimination phase if it is hidden from all active nodes that burst for more time slots that it does; while in the yielding phase, a node might falsely determine channel access.

After the elimination phase is over, it is possible that not all nodes, which a node i “sees” winning the elimination phase, will continue into the next phase. This is due to the fact that some of them could have been eliminated by other -non hidden from them- nodes, that are hidden from node i. It is also possible that during the yielding phase, even more nodes are eliminated before i wins channel access due to the fact that they hear transmission from other nodes not hidden from them but -again- hidden from i. A detailed calculation of the probability of channel access for a node can be found in [25].

III. MODIFYING THE HIPERLAN CAC LAYER FOR INTERMITTENTLY CONNECTED MOBILE NETWORKS

Since a complete and stable end-to-end path between source and destination does not exist in intermittently connected mobile networks (ICMNs), packets need to be stored and then forwarded on an evolving path from source to destination. As shown in Figure 1, after the EY-NPMA mechanism determines the node(s) that can use the channel to transmit a packet, the packet and acknowledgment are expected to be directly exchanged between source and destination. In order to allow for the packet to be forwarded by other nodes to its destination, these other nodes should be allowed to acknowledge receipt of the packet after they have determined that the destination has not successfully received it while they have. These nodes will then take it upon themselves to forward the packet towards its destination. A key decision that has to be made is which nodes with direct access to the transmitter should forward the packet. Allowing too many of them would increase the offered load on the channel and decrease performance. On the other hand if not enough nodes take on the task of forwarding the packet, its total time to the destination is going to increase on average and the chances of the packet following an optimal path will also decrease. We propose that only the nodes that were actively competing for channel access and lost it should listen for the packet and acknowledge it if they have successfully received it. They should then add it to the list of packets that they need to forward. This would limit the number of forwarding nodes to the ones already competing, thus not introducing any new nodes into channel competition. When one of these nodes will eventually win channel access, it will transmit its own packet and then the packets that it has taken upon to forward. In addition, if a node receives for forwarding a packet that it has already forwarded, it will not forward it, but discard it as a duplicate. Each packet should also have a ”time to live” TTL, so that packets that are taking a long path towards the destination are eliminated (some packets might never reach destination).

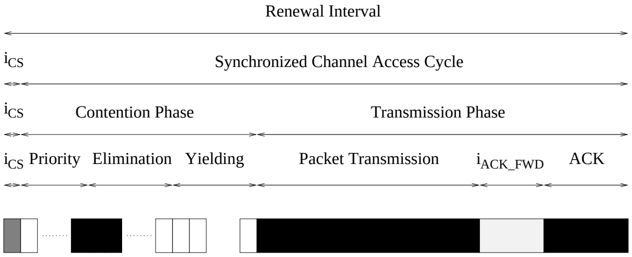

Fig. 2. Allowing acknowledgments from forwarding nodes.

Another key decision that has to be made, is the buffer size that each node will allocate for forwarding packets. With a larger buffer space more packets can be buffered for forwarding, but at the risk of increasing the number of extra copies of each packet that needs to be forwarded (increase in total traffic).

The proposed modifications which are necessary to implement this forwarding mechanism are the following:

- All nodes that unsuccessfully contended for channel access should listen for the transmission of a packet by the winning node. If they determine it to be successful (by hearing an acknowledgment after a time interval of i_{ACK}) they need to do nothing else. But If they hear no acknowledgment after a time interval of i_{ACK} they should acknowledge the packet at a time interval of $i_{\mathrm{ACK\_FWD}}$ from its transmission, where $i_{\mathrm{ACK\_FWD}}$ should obviously be longer than i_{ACK}. Figure 2 shows such an acknowledgment sent by a forwarding node.
- If no acknowledgment is received after the time interval of $i_{\mathrm{ACK\_FWD}}$, the transmission has failed and the renewal interval is completed.
- If the transmitter node receives acknowledgments in a time interval of $i_{\mathrm{ACK\_FWD}}$ it determines that its transmission is successful and its packet will be forwarded. This should be true even if multiple overlapping acknowledgments are received.
- A node that successfully transmits a packet of its own can continue in the same renewal interval to transmit packets it has taken upon to forward, up to a maximum number of ”forwarding transmissions” of FT_{MAX}. This modified transmission phase is shown in Figure 3, where $i_{\mathrm{ACK}*}$ could either be i_{ACK} or $i_{\mathrm{ACK\_FWD}}$ and i_{NT} (which is less than i_{CS}) is the time interval between transmissions.
- As soon as all forwarding transmissions are done or one of the extra forwarding transmissions fails the renewal interval is over.

We also propose that the total priority levels are only two: high and low. All nodes start out as low priority nodes. It is expected that traffic that will be allowed to switch to the higher priority will have better quality of service. Alternative methods for elevating a nodes priority are presented in the next section.

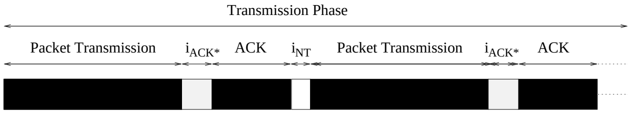

Fig. 3. Modified Transmission Phase.

A. Packet Transmission in the Presence of Hidden Nodes, Capture and Bit Errors

We start our analysis of packet transmission in the presence of hidden nodes, capture and bit errors by examining the unmodified HIPERLAN CAC Layer operation.

Let d_{ij} denote the distance between two nodes i and j. A transmission by station i will be “captured” by station j if no other node in a circle of radius αd_{ij}, $\alpha \geq 1$, around j is transmitting simultaneously. The parameter α will be referred to as the *capture parameter* and the area surrounding node j of radius αd_{ij} will be referred to as $j's$ *capture area* for node i. Let’s further denote $A_{ji}(\alpha)$ as the set of nodes that are in $j's$ capture area for i, and $\bar{A}_{ji}(\alpha)$ the set of nodes that are out of $j's$ capture area for i.

The assumption of relatively limited node mobility is extended, so that if a node is inside or outside of a specific capture area it will remain so throughout the renewal interval.

In our modified HIPERLAN CAC Layer, we allow in between nodes to receive a packet, and if the destination node has not successfully received it, to store it for forwarding, and to acknowledge it. Thus, the conditions for a node to successfully transmit a packet and successfully receive its acknowledgment are now the following:

- The transmitter and destination nodes are not hidden from each other or the transmitter and a potential forwarding node are not hidden from each other.
- After the contention phase is over, the transmitter node determines channel access, while the destination node, if active and not hidden from the transmitter node and with a direct link to it, does not.
- If nodes other than the transmitter node have survived the contention phase, they have to be either hidden from the destination node or a potential forwarding node or otherwise they have to be outside of the destination’s capture area for the transmitter or of a potential forwarding node’s capture area for the transmitter, so that the transmitter’s packet can be successfully received.
- No node that is not hidden from the transmitter or that is not outside of the transmitter’s capture area for the destination, receives successfully a packet if a direct link between transmitter and destination exists.
- If a direct link between transmitter and destination node does not exist no node that is not hidden from the transmitter or that is not outside of the transmitter’s capture area for a receiving forwarding node, receives successfully a packet.
- The destination or a forwarding node receives a bit error free packet.

IV. CLASSES OF SERVICE AND PRIORITY LEVELS

In ICMNs it is expected that nodes that will not experience the same quality of service that they experience in ordinary fully connected WLANs. Thus, giving priority to some traffic over other will increase the quality of service it experiences.

Different ways of controlling priority elevation can be used. If no classes of service exist and all traffic from all nodes is treated equal, a simple mechanism can be in place to elevate the priority of packets that have been experiencing delays trying to get through to their destination. If a node has failed to gain channel access for a specific number of consecutive attempts (denoted by LS_{MAX}) and it has packets in addition to its own to forward, then its priority increases to the high priority. Once channel access is granted it returns to the low priority after a certain number of packets have been transmitted.

We further propose to allow for different classes of service. Nodes that have subscribed to a higher class of service have access to higher priority levels. Each class of service should further have a ”high” and a ”low” priority level. Packets can then transition from the lower to the higher priority for their class based on the mechanism described above. This allows for a total of four priorities, where the top two are only accessed by higher class of service nodes and the bottom two are only accessed by the lower class of service.

V. SIMULATION STUDY AND NUMERICAL RESULTS

The simulation of the proposed forwarding algorithm takes into account realistic conditions for wireless intermittent connectivity (contention, mobility, cluster formations, bit errors, hidden nodes and capture).

To evaluate the proposed forwarding algorithm we start with 25 wireless nodes that need to communicate with each other as in an ad-hoc wireless network, but each node can have direct transmission only to nodes in a limited area surrounding it, thus the nodes in that area (or cluster) that successfully receive a transmission from that node will have to forward it to other nodes outside this area. Then mobility is added that will allow for these clusters to change over time and communication to be realized over a dynamically evolving path. The mobility model for each node is that of a *random walk*. The maximum velocity allowed is 100km/h, to account for a wireless device in a vehicle. The simulation is run for various cluster area sizes. We also assume that the probability that two nodes will be hidden from each other for reasons other that being out of range from each other (in different clusters) to be a typical $p_h = 0.02$ and we also take into account capture but with a capture parameter of $\alpha = 10$, which introduces a very limited capture benefit. The bit error rate is $BER = 10^{-6}$, which is typical for wireless transmissions. The initial coordinates of the nodes were chosen randomly and are normalized to belonging inside a circle of radius 1. These nodes are then allowed to move totally randomly in a square defined by the points (-1,1), (1,1), (1,-1) and (-1,-1). The initial coordinates at the beginning of the simulation are chosen randomly.

It is assumed for simplicity that all sources generate packets following the Poisson distribution (inter-arrival times between packets follow the exponential distribution). And all nodes are transmitting the same amount of load and equally to all other nodes. It is also assumed that all nodes initially are of the same low priority level for the class of service they belong to. The new protocol parameter used is: $i_{\text{ACK\_FWD}} = 384$ bits (while $i_{\text{ACK}} = 256$ bits). The channel bandwidth for HIPERLAN/1 is 23529 Kbits/sec.

Various values were chosen for FT_{MAX}, LS_{MAX}, and the maximum buffer size allocated for receiving packets to forward FB_{MAX}. AS with previous work, we will present results for $FT_{\text{MAX}} = 8$, and $FB_{\text{MAX}} = 20$.

Finally, each packet has a time to live of $TTL = 10$. This value is smaller than the typical value found in other protocols, but given that a great number of nodes will be forwarding a packet as load increases in the network, a smaller TTL value will help in keeping network load from increasing to a point of severe congestion.

A. Numerical Results

We are interested in the expected delay between initial transmission by the source and the final acknowledgement by the destination. For various values of load and for various values of the cluster area we plot the probability of overflow. The probability of overflow is the probability of a packet exceeding a maximum time delay value, and is the main focus of this simulation study.

In our previous work it was shown that the expected delays with no classes of service were acceptable and in some cases good enough for real time traffic. In this simulation study we have two different classes of service and each class of service has two different priority levels so that in each class of service fairness can be achieved and similar performance achieved.

The data for the lower class of service show that this class of service experiences increased delays in order for the higher class of service to achieve better performance. Thus, the lower class of service is not suitable for real time traffic. In this section we will concentrate in presenting data for the high priority class. All the data presented is for a simulation of 5 nodes being of higher class of service while the other 20 remain at the lower class of service. Other configurations were also explored, but as the number of nodes of higher priority increased the benefit to the higher priority deminished. Thus, the number of nodes of higher priority must be limited.

In Figure 4 one can see the plot of the probability of overflow for various loads. These loads are between 10% and 20% of the load capacity of the channel. All loads are for a cluster area = 9.61% Better performance is achieved at higher loads as more nodes are forwarding packets at higher loads. This is something that we also saw with the simulation of only one class of service.

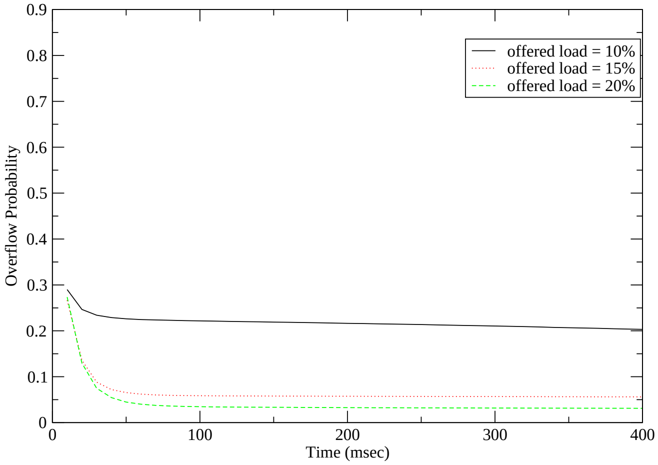

Fig. 4. Probability of overflow for cluster area = 9.61%

In Figure 5 one can see the plot of the probability of overflow for various loads, again between 10% and 20% of the load capacity of the channel. The cluster area has increased now to 12.56% and the performance is about the same. One interesting difference is that with the increase of cluster area we see that performnce is worse for a load of 20% compared to a load of 15%. Thus, the increase in load while initially increased performance, after a certain load increase, started to decrease performance in this cluster size.

In Figure 6 one can see another plot of the probability of overflow for various loads, again between 10% and 20% of the load capacity of the channel. The cluster area has increased now to 15.90% and the performance has slightly dropped as delays have slightly gone up. This can be attributed to the fact that with a bigger cluster area more competition exists to forward a packet to the next cluster area.

And in Figure 7 one can see another plot of the probability of overflow for the various loads, again between 10% and 20% of the load capacity of the channel. The cluster area has now

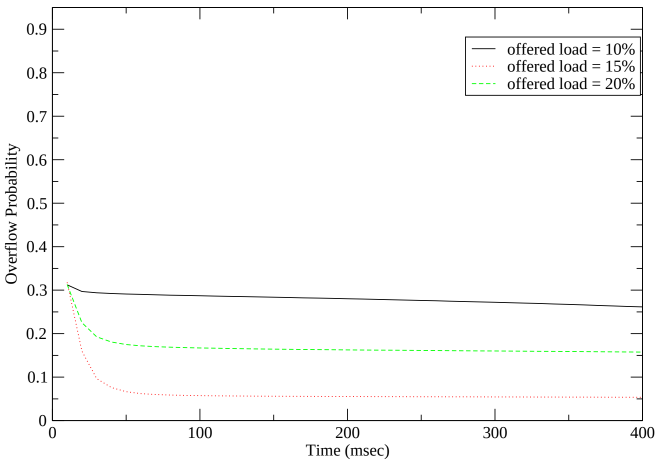

Fig. 5. Probability of overflow for cluster area = 12.56%

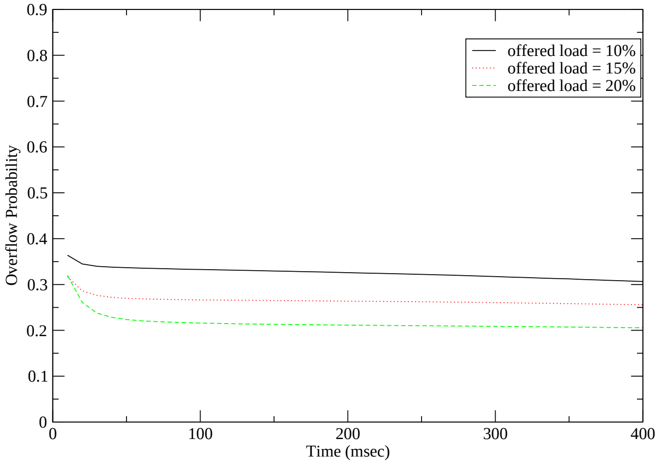

Fig. 6. Probability of overflow for cluster area = 15.90%

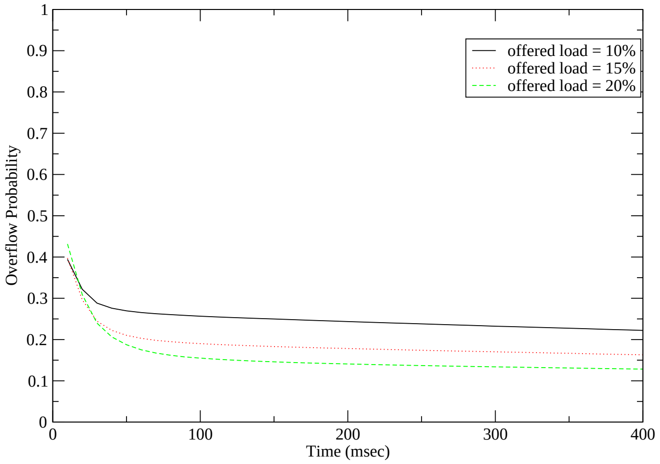

Fig. 7. Probability of overflow for cluster area = 23.74%

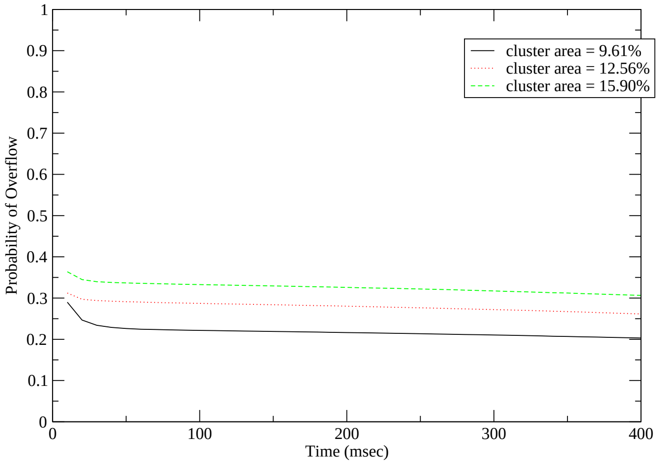

Fig. 8. Probability of overflow for load = 10%

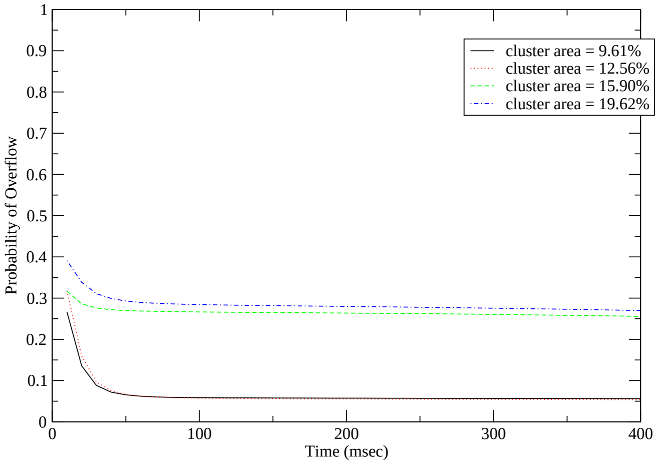

Fig. 9. Probability of overflow for load = 15%

increased to 23.74% and the performance has not decreased. We also see that performance differences between loads have come smaller.

In Figure 8 one can see the plot of the probability of overflow for various cluster areas. The cluster areas are between 9.61% and 15.90%. The load is at 10% of the channel capacity. In this figure we see similar performance for all three cluster areas, with better performance at the larger cluster area.

In Figure 9 the load has increased to 15% of the channel capacity and we again look at the plot of the probability of overflow for various cluster areas. We see that the higher load has increased performance at the lower cluster areas. Similar observation was made in our previous studies with no classes of service.

In Figure 10 one can see again the plot of the probability of overflow for various cluster areas. This time he load has increased to 20% and performance remains strong.

If we compare the above results with the results of the analysis of of different techniques used for routing (direct routing, epidemic routing, randomized flooding and spraying techniques) found in [10][11][12], we see that we obtain similar or almost as good results, although in those studies focus mainly on the expected delay rather than on the probabilty of exceeding a maximum delay. And once again this is done by only modifying the wireless access mechanism rather than introducing a routing algorithm. Thus, operating without the computational overhead a routing algorithm would introduce.

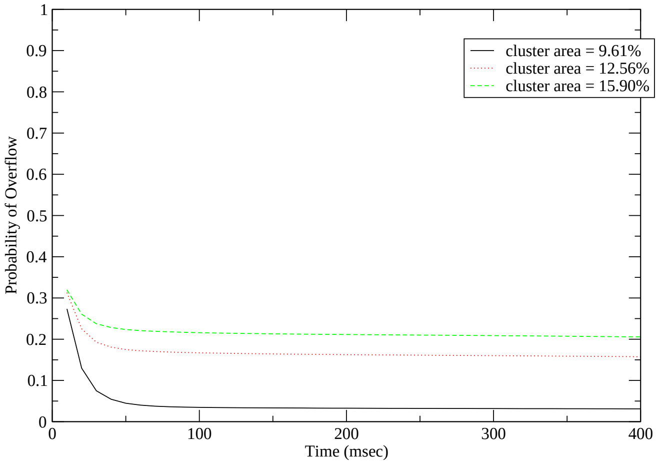

Fig. 10. Probability of overflow for load = 20%

VI. CONCLUSION

A brief outline of the HIPERLAN CAC layer protocol is presented. Then a modification to its operation that includes different classes of service is proposed and that would allow it to operate in ICMNs. The influence of the phenomena of hidden nodes, capture and bit errors is discussed and the conditions for successful packet transmission are presented for the modified forwarding mechanism. From the numerical data presented it is shown that the higher class of service can enjoy good enough performance to allow for real time loss tolerant applications. As the data for the lower class of service showed performance that would allow only non-real time traffic with substantial delays in some cases, it is not presented. This research is now progressing into deeper understanding of the performance data collected through simulation. One direction aims at further understanding the relationship between cluster areas and loads so a connection admission control mechanism can be created to ensure some minimum quality of service for the nodes of higher class service. A second direction is looking into how loss tolerant applications can further communicate their minimum needs of performance and affect the overall network performance.

REFERENCES

[1] K. S. Phanse and J. Nykvist, "Opportunistic wireless access networks," in AcessNets '06: Proceedings of the 1st international conference on Access networks. New York, NY, USA: ACM, 2006, p. 11.

[2] D. Zheng, W. Ge, and J. Zhang, "Distributed opportunistic scheduling for ad-hoc communications: an optimal stopping approach," in MobiHoc '07: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing. New York, NY, USA: ACM, 2007, pp. 1–10.

[3] J. Kim and S. Bohacek, "A comparison of opportunistic and deterministic forwarding in mobile multihop wireless networks," in MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking. New York, NY, USA: ACM, 2007, pp. 9–16.

[4] S. Biswas and R. Morris, "Opportunistic routing in multi-hop wireless networks," SIGCOMM Comput. Commun. Rev., vol. 34, no. 1, 2004, pp. 69–74.

[5] ——, "Exor: opportunistic multi-hop routing for wireless networks," in SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications. New York, NY, USA: ACM, 2005, pp. 133–144.

[6] A. El Fawal, J.-Y. Le Boudec, and K. Salamatian, "Multi-hop broadcast from theory to reality: practical design for ad hoc networks," in Autonomics '07: Proceedings of the 1st international conference on Autonomic computing and communication systems. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007, pp. 1–10.

[7] R. Ramanathan, R. Hansen, P. Basu, R. Rosales-Hain, and R. Krishnan, "Prioritized epidemic routing for opportunistic networks," in MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking. New York, NY, USA: ACM, 2007, pp. 62–66.

[8] S. Dabideen and R. Ramanathan, "Fansyroute: Adaptive fan-out for variably intermittent challenged networks," SIGMOBILE Mob. Comput. Commun. Rev., vol. 18, no. 1, Feb. 2014, pp. 37–45. [Online]. Available: http://doi.acm.org/10.1145/2581555.2581561

[9] E.-H. Kim, J.-C. Nam, J.-I. Choi, and Y.-Z. Cho, "Probability-based spray and wait protocol in delay tolerant networks," in 2014 International Conference on Information Networking (ICOIN), Feb 2014.

[10] A. Jindal and K. Psounis, "Contention-aware analysis of routing schemes for mobile opportunistic networks," in MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking. New York, NY, USA: ACM, 2007, pp. 1–8.

[11] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the multiple-copy case," IEEE/ACM Trans. Netw., vol. 16, no. 1, 2008, pp. 77–90.

[12] A. Jindal and K. Psounis, "Contention-aware performance analysis of mobility-assisted routing," Mobile Computing, IEEE Transactions on, vol. 8, no. 2, feb. 2009, pp. 145 –161.

[13] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Performance analysis of mobility-assisted routing," in MobiHoc '06: Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing. New York, NY, USA: ACM, 2006, pp. 49–60.

[14] L. Song and D. F. Kotz, "Evaluating opportunistic routing protocols with large realistic contact traces," in CHANTS '07: Proceedings of the second ACM workshop on Challenged networks. New York, NY, USA: ACM, 2007, pp. 35–42.

[15] M. Abdulla and R. Simon, "Characteristics of common mobility models for opportunistic networks," in PM2HW2N '07: Proceedings of the 2nd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks. New York, NY, USA: ACM, 2007, pp. 105–109.

[16] V. Erramilli and M. Crovella, "Forwarding in opportunistic networks with resource constraints," in CHANTS '08: Proceedings of the third ACM workshop on Challenged networks. New York, NY, USA: ACM, 2008, pp. 41–48.

[17] HIgh PErformance Radio Local Area Network Type 1: functional specification, pr ets 300 652 ed., European Telecommunication Standards Institute, 650 Route des Lucioles, Sophia Antipolis, Valbonne, France, May 1996.

[18] G. Anastasi, L. Lenzini, and E. Mingozzi, "Stability and performance analysis of hiperlan," Proceedings of the IEEE Infocom, 1998.

[19] S. Chevrel, A. Aghvami, H. Lach, and L. Taylor, "Analysis and optimasation of the hiperlan channel access contention scheme," Wireless Personal Communications, vol. 4, no. May, 1996, pp. 27–39.

[20] K. Fu, Y. Guo, and S. Barton, "Performance of the ey-npma protocol," Wireless Personal Communications, vol. 4, no. May, 1996, pp. 41–50.

[21] P. Jacquet, P. Minet, P. Muhlethaler, and N. Rivierre, "Data transfer for hiperlan," Wireless Personal Communications, vol. 4, no. May, 1996, pp. 65–80.

[22] ——, "Priority and collision detection with active signaling - the channel access mechanism of hiperlan," Wireless Personal Communications, vol. 4, no. May, 1996, pp. 11–25.

[23] W. M. Moh, D. Yao, and K. Makki, "Wireless lan: Study of hidden-terminal effect and multimedia support," Proceedings of the IEEE Infocom, 1998, pp. 422–431.

[24] R. O. LaMaire, A. Krishna, P. Bhagwat, and J. Panian, "Wireless lans and mobile networking: Standards and future directions," IEEE Communications Magazine, no. August, 1996, pp. 86–94.

[25] C. Coutras and P.-J. Wan, "Evaluating performance of the HIPERLAN CAC layer protocol for asynchronous traffic," in Proceedings of the 24th Annual IEEE Conference on Local Computer Networks, 1999.

[26] C. Coutras, O. Frieder, and P.-J. Wan, "Evaluating performance of the HIPERLAN CAC layer protocol for real-time traffic," in Proceedings of the 25th Annual IEEE Conference on Local Computer Networks, 2000.

[27] C. Coutras, "Modifying the hiperlan/1 cac layer protocol for intermittent connectivity," in Proceedings of the International Congress on Ultra Modern Telecommunications and Control Systems, as part of the 2nd International Workshop on Mobile Computing and Networking Technologies, 2010.