

Improving Reactive Routing on Wireless Multi-rate Ad-hoc Networks

Rafael Paoliello Guimarães and Llorenç Cerdà
Polytechnic University of Catalonia
Computer Architecture Department
Jordi Girona 1-3, E-08034 Barcelona, Spain
e-mail: {rafael.guimaraes, llorenc}@ac.upc.edu

Abstract—Traditional ad-hoc routing protocols typically choose minimum hop paths for transmissions. In a multi-rate environment, where nodes may elect between several transmission rates in order to accommodate different channel conditions, these protocols usually choose paths that contain long range links, with low effective throughput and low reliability. In this paper, we propose a mechanism for reactive routing protocols that leads to the election of high throughput routes while not increasing significantly the signaling overhead. By allowing a node to have a complete knowledge of its one-hop neighborhood topology, we are able to increase significantly the performance of the network through the improvement of the route election process on multi-rate environments.

I. INTRODUCTION

Wireless communications have been spread all over the world over the last years. Most of the commercially available wireless devices are based on the IEEE 802.11 standards family. Most of them, such as 802.11b [1], 802.11a [2] and 802.11g [3], allow the use of different transmission rates.

The election of which transmission rate should be used depends on the wireless medium conditions. The worse the channel quality, the stronger the code that should be used and, consequently, the lower the achieved transmission rate. Since channel quality is directly related to distance between nodes, we may say that usually, the closer two nodes are from each other, the higher the transmission rate used between them.

In 802.11a (and also in 802.11g), for example, the set of possible data transmission rates are 6, 9, 12, 18, 24, 36, 48, 54 Mbps while 802.11b supports 1, 2, 5.5 and 11 Mbps. In order to exploit this capability, some Medium Access Control (MAC) mechanisms are required, the Auto Rate Fallback (ARF) [4] protocol was the first to deal with this issue. The sender increases (or decreases) the transmission rate to be used in future transmissions based on the successes (or failures) in the previous ones. In other mechanisms, like the Receiver Based Auto Rate (RBAR) [5] protocol, the receiver measures the quality of the channel when it receives a Request To Send (RTS) message and selects the appropriate rate to be used under these conditions. It then informs the sender the rate to be used for data transmission through the Clear To Send (CTS) message.

This work was supported by CAPES - Brazil, the Ministry of Education of Spain under grant CICYT TEC2004-06437-C05-05 and the European NoE EuroNGI.

However, in order to fully use the multi-rate capabilities on a wireless ad-hoc network, the routing protocol should also be aware of this information. There is no point on being able to transmit at so many different rates on the MAC layer, if at the end, the routing protocol always chooses routes based only on hop count. Traditional routing protocols, like the Ad-hoc On-demand Distance Vector (AODV) [6] or the Optimized Link State Routing Protocol (OLSR) [7], usually elect this kind of path, where the minimization in the number of hops causes the election of long range links over short range ones. If short range links were elected, although the number of hops would increase, higher transmission rates could be used, and the overall performance of the network could be significantly improved. However, although it seems to be very important to take multi-rate into account in the routing layer, there are not many publications that deals with this issue.

In proactive routing protocols (like OLSR), the solution for this problem is quite straightforward. Since each node knows the (almost) entire network topology, information about link rates would be enough to choose an efficient path. In [8], the authors propose a routing metric that is able to maximize the achievable throughput on chosen paths. However they only implement it on a proactive protocol and no further comments on how to do so on reactive protocols are made.

Reactive protocols (like AODV) do not have any previous information about the network topology, they choose their routes by flooding the network with Route Request messages trying to reach the destination node. This makes the problem much more complex, as we will discuss in further sections. Providing a simple and yet efficient solution is not trivial. Authors of [9] propose a solution for AODV that introduces too much signaling overhead, increasing heavily the number of broadcasted Route Request messages.

A totally different approach is proposed by [10]. It deals with the multi-rate issue completely in the Medium Access Control (MAC) layer. The MAC layer hides from upper layers the existence of low throughput links, by selectively filtering received frames. As a result, on the top one could use any standard routing protocol and high throughput path would always be elected. However, hiding topology information from the routing layer may not always be a good solution. Although solving multi-rate path election issues, it may degrade significantly the performance of mechanisms such as rapid route repair, or any other schemes which performance is directly

related to the amount of topology knowledge that a node has. The authors of [11] propose a similar method where an intermediate layer is created between the network and link layers to deal with multi-rate. Also in this proposal, the fact that the decision is not taken by the routing layer may have a negative impact on other mechanisms.

In this paper, we propose an efficient solution for the election of high throughput paths through the use of reactive routing protocols. More specifically, we propose that each node keeps track of its 1-hop neighborhood topology, using a proactive approach for choosing the route in the neighborhood, and a reactive approach for choosing the route towards distant nodes. We believe that reactive routing can provide better response to the constant changes in the topology of a mobile ad-hoc network, while monitoring the 1-hop neighborhood may improve routing decisions and should not be a problem even when mobility is not so low. Furthermore, the knowledge of the 1-hop neighborhood may also be useful for other mechanisms that may improve the network overall performance, such as efficient route repairing and controlled flooding [12, 13].

Although we focus our attention to the AODV protocol throughout the paper, the proposed mechanism can be applied to any reactive routing protocol (as long as it is based on the exchange of Route Request / Route Reply messages). Moreover, although we deal with choosing high throughput paths on multi-rate networks, our proposal is more general, in the sense that it deals with routing through minimum cost paths, no matter what the cost represents. That means that our proposal could also be used for searching paths that minimize different metrics (link rate, link delay, available bandwidth, link stability, loss probability etc).

The paper is organized in 4 additional sections. In the next section we discuss the problem of using traditional reactive protocols on multi-rate ad-hoc networks more deeply. We then propose in section III modifications on the routing election process in order to take transmission rates into account. In section IV we show through simulations the overall improvement that can be obtained when using our proposal under different scenarios. Finally, we present some conclusions in section V.

II. TRADITIONAL REACTIVE ROUTING

Wireless ad-hoc networks are usually composed by portable nodes – notebooks, palmtops or even mobile phones. This portability also brings an important issue: mobility. This is a key factor in ad-hoc networks. The mobility of the nodes causes the topology of the network to change constantly. Keeping track of this topology is not an easy task, and may consume too much resources in signaling. Reactive routing protocols were designed for these environments. They are based on the idea that there is no point on trying to have a picture of the entire network topology, since it will be constantly changing. Instead, whenever a node needs a route to a given destination, it initiates a route discovery process on the fly, for finding out a path.

This kind of protocols (which has AODV as its major example) is usually based on flooding the network with Route Request (RREQ) messages. The source node broadcasts a

RREQ message with a time-to-live equal to 1, i.e., a broadcast limited to its 1-hop neighborhood. Each RREQ is uniquely identified through a sequence number, so that the first copy of a RREQ received by a node is processed, while duplicate messages are discarded. When a node receives the first copy of a given RREQ, it records the address of the node that sent the message, establishing thus a reverse route. When the first RREQ reaches the desired destination, a Route Reply (RREP) message is generated and sent back to the source node through the recorded reverse path, confirming then a path from the source to the destination.

This kind of protocol is usually very effective on single-rate networks. It usually minimizes the number of hops of the chosen path. However, on multi-rate networks, the number of hops is not as important as the throughput that can be obtained on a given path.

In figure 1, for example, if node A wants to transmit to node E and a reactive protocol is used to find a path, the elected path would be A-C-E. Node A would broadcast a RREQ, which would be received by B and C. Node B would re-broadcast the RREQ, that would be discarded by C (since it has previously received a copy of this RREQ from A). Node C would broadcast the RREQ and it would reach E (as well as D). Node E would then reply with a RREP, that would cross node C and reach A. The path would then be established.

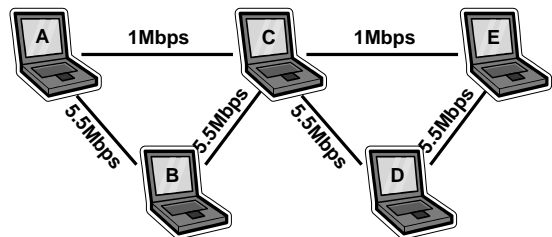


Fig. 1. An example of a multi-rate wireless ad-hoc network

It is not very difficult to notice that, in this case, the path A-B-C-D-E, although being longer, would have been a better choice. Data would be transmitted using a 5.5Mbps rate, instead of 1 Mbps. This simple example shows that traditional routing protocols do not cope with the multi-rate network requirements. We should, thus, take transmission rate into account when choosing the path towards a given destination, using it as a routing metric.

III. TAKING LINK RATE INTO ACCOUNT

There are already some proposals that use the transmission rate between wireless nodes as a routing decision metric. The Medium Time Metric (MTM) [8], for example, establishes a link cost for each transmission rate, which is computed through the analysis of how much time it takes to transmit a 1500 bytes packet on 802.11. The link costs for several transmission rates are presented in table I.

The implementation of this metric on a proactive routing protocol is very straightforward. Since each node already knows the topology of the network, it should only add this metric to each link and compute the less costly route towards

TABLE I
MTM METRICS

Transmission rate	MTM link cost
11.0 Mbps	5
5.5 Mbps	7
2.0 Mbps	14
1.0 Mbps	25

the desired destination. Nevertheless, on reactive protocols, the problem becomes more complex.

The main problem with reactive protocols like AODV is the fact that nodes discard duplicate copies of received RREQ messages. In the example depicted by figure 1, the best path from node A to C would be through B, however, the RREQ sent by B only reaches C after the one sent by A, what causes it to be discarded.

A simple solution would be to accumulate the link cost on each retransmission of the RREQ and not to discard duplicate RREQs when its accumulated link cost is below the cost of all previously received RREQs. If this is the case, the RREQ would be re-broadcasted and the reverse path would be updated. The destination node would not reply the first received RREQ, but instead would wait for a certain period or for a given number of RREQs and then would reply the one with the lowest cost. This solution, presented in [9], although being simple, increases very much the number of RREQ messages on the network. Ad-hoc networks with a very dynamic behavior could suffer from performance degradation due to avalanches of RREQs. Notice that this increment in the number of broadcasted RREQs is concentrated in a very short period of time (during the route discovery procedure). The occurrence of these RREQ bursts would significantly increase the number of collisions among copies of the same RREQ. Since broadcast transmissions are not acknowledged in 802.11, many RREQs would be lost and the route discovery procedure would not perform well. Furthermore, as the number of nodes in the network increases, so will the number of duplicate RREQs that are transmitted.

Our proposal

In order to avoid an increase in the number of transmitted RREQs over the network, we propose that every node keeps track not only of their 1-hop neighborhood (what is already done by most of the existing ad-hoc routing protocols through the periodic exchange of HELLO messages), but also of the topology of these neighbors. That means that a node should know the links that exist between its neighbors. Notice that even on highly dynamic networks, this information is not difficult to be maintained, since a node is aware of any change on its 1 hop neighborhood very quickly.

Once a node is aware of the topology of its 1-hop neighborhood, the RREQ/RREP procedure can take place with minor changes. Whenever a node receives and processes a RREQ, it may compute the best path (it terms of throughput) towards the node that sent him the RREQ message, or towards any other node before in the path (if it is more efficient not to

pass through the previous node). After computing this part of the path, the complete path information is updated in the RREQ message and it is re-broadcasted. When the first RREQ reaches the destination, a RREP is sent to the source following the path recorded in the request.

A. Keeping track of the 1-hop neighborhood topology

In order to keep track of the topology of the 1-hop neighborhood, nodes should include a list of their 1-hop neighbors (nodes from which they receive HELLO messages) in the HELLO messages that they periodically broadcast together with the link cost towards each of the neighbors. This link cost is computed based on the link rate (see table I).

By receiving HELLO messages from every neighbor, a node is able not only to have a complete view of the 1-hop neighborhood topology, but also to know its 2-hop neighbors and their connectivity with the 1-hop neighborhood (in order to have a complete 2-hop topology, it would be necessary also to know the links among the 2-hop neighbors). Figure 2 shows an example of the topology map that can be built by node A, using the information we propose to be carried by HELLO messages.

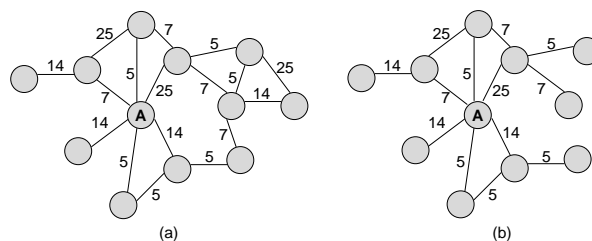


Fig. 2. (a) Complete ad-hoc network topology with link costs (b) Partial topology known by node A due to HELLO messages

B. The multi-rate route discovery procedure

Once the nodes know the complete 1-hop topology and a partial 2-hop topology, the route discovery procedure can be modified in order to retrieve not the minimum hop path, but the maximum throughput one.

The first step toward achieving this objective is to extend the RREQ message by introducing a list of nodes and link costs that represents a maximum throughput path from the source node to the node that received the message. Every node that receives the RREQ completes this list using their 1-hop topology knowledge in order to create a complete path from the source to the destination.

The route discovery procedure works as follows:

- 1) The source node broadcasts a RREQ message to its 1-hop neighbors.
- 2) Each node that receives the RREQ message computes the maximum throughput (minimum cost) path to the last node through which the RREQ passed.
- 3) The node includes the maximum throughput path it computed in the RREQ message by introducing the IP address of the nodes between the current node and the previous one together with the link cost to go from one

node to another. In figure 3(a) for example, when node C receives a RREQ from node A, it computes that the maximum throughput path towards A is passing through node B. So it includes the IP address of A with 5 as the link cost (to reach A from node B), and then the IP address of B with 7 as the link cost (to reach node B from the current node C).

- 4) Finally, when the RREQ reaches the destination node, it replies with a RREP that should follow the path included in the RREQ, which represents the maximum throughput path from the source to the destination (see figure 3(b)).

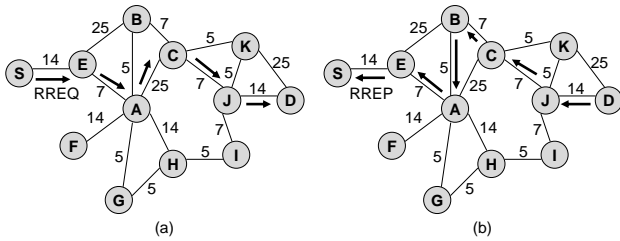


Fig. 3. (a) Path followed by the 1st RREQ to reach the destination (b) Path followed by the RREP

In the ad-hoc network depicted by figure 3, if node S wants to find a route towards node D, it broadcasts a RREQ message that, at each intermediate node, receives a list of nodes that represents a candidate path. In this example, at each intermediate node through which a given RREQ message passes, the list of nodes that it carries is updated as follows. Notice that we are only dealing with the RREQ that first reaches the destination node.

at E: $\underline{S \ 14}$

at A: $\underline{S \ 14, \ E \ 7}$

at C: $\underline{S \ 14, \ E \ 7, \ A \ 5, \ B \ 7}$

at J: $\underline{S \ 14, \ E \ 7, \ A \ 5, \ B \ 7, \ C \ 7}$

at D: $\underline{S \ 14, \ E \ 7, \ A \ 5, \ B \ 7, \ C \ 7, \ J \ 14}$

In fact, an intermediate node may not only insert new nodes in the candidate path included in the RREQ message, but it may also replace existing hops of the path by other hops that it considers more efficient. This may happen whenever the node is able to reach another node that is in the path included in the RREQ message with a lower cost than the one presented in the RREQ. This optimization is only possible due to the fact that the routing protocol is the entity responsible for the multi-rate routing decisions, having a complete (or almost complete) knowledge of the topology and link costs. If routing decisions like that were left to lower layers (like in [11] or [10]), such optimizations would not be possible.

Notice that the cost towards a given intermediate node can be obtained by summing the links costs from the last node in the list up to the the desired node. In the last example, the cost for node J to reach A is $7 + 7 + 5 = 19$. If this is the case, all the nodes included in the list after this intermediate node should be erased and substituted by the new path elected by the current node.

By doing this procedure, we can not guarantee that the minimum cost path is finally elected but, at least, we can guarantee that the chosen path, will perform better than the

minimum hop path. In order to guarantee the the election of the minimum cost path, we should allow the re-broadcast of RREQs (as proposed by [9]), however we think that the collateral effect of such solution (high increase on the number of RREQs) is a very high price to pay. We believe that our mechanism provides a better trade-off between performance and overhead.

IV. SIMULATION RESULTS

We have modified the AODV implementation provided by the network simulator ns-2 [14] version 2.30 and launched several simulations for validating our proposal and checking its overall performance. For all these simulations, we used the parameters listed in table II.

TABLE II
PARAMETERS USED IN SIMULATIONS

Parameter	Value
MAC Protocol	802.11 with multi-rate
Propagation Model	Two Ray Ground
Transmission Rates	1, 2, 5.5 and 11 Mbps
Transmission Ranges	115, 90, 70 and 50 meters ¹
Simulation Time	500 seconds
Simulation Area	Square of 500×500 meters

The first simulation used a simple topology composed of 25, 36, 49, 64, 81 and 100 nodes disposed in a regular matrix configuration (5×5, 6×6, 7×7, 8×8, 9×9 and 10×10 respectively). Then, an FTP connection was established between the node in upper left corner and the one in the bottom right corner. In figure 4 it is possible to see the topologies that were simulated. In all of them, we represented the transmission ranges using each available transmission rate.

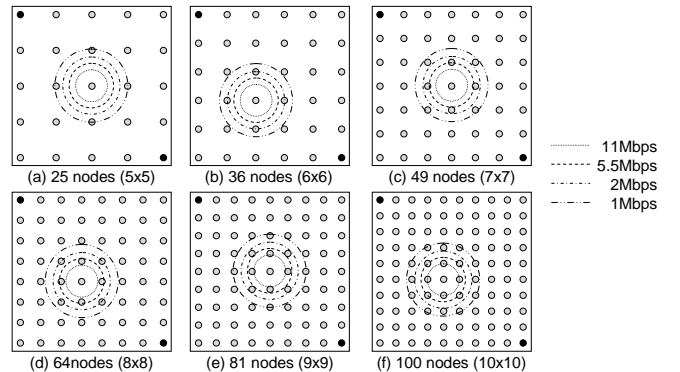


Fig. 4. Network topologies used in the first simulation. The black nodes are the source and destination of the FTP connection.

Figure 5 shows the throughput obtained using our proposal (which we called Multi-Rate AODV, or MR-AODV) and the standard AODV. As we may see, in the first three scenarios, when the density of nodes is low, the difference between the two mechanisms is almost inexistent.

Figures 6 and 7 show the amount of packets transmitted using each transmission rate in the 6 simulated scenarios, when

¹According to the ORiNOCO 802.11b PC card specification for a semi-open environment[15].

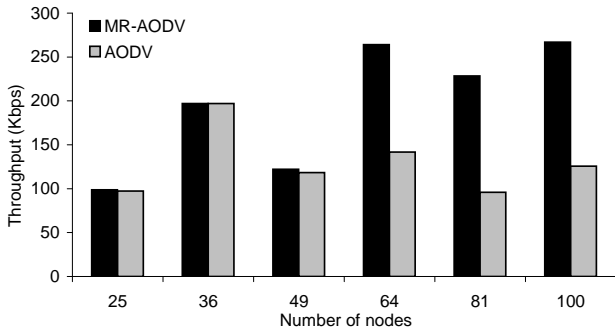


Fig. 5. Throughput against the number of nodes

using AODV and MR-AODV respectively. These graphs help us understand the throughput comparison provided by figure 5. In the first scenario (25 nodes), a node can only reach its neighbors by using the lowest transmission range (1Mbps), the impossibility of using another transmission rate causes that the use of our mechanism does not improve the performance of the network. The same happens in the second scenario (36 nodes), when a node can reach all its neighbors using 2Mbps.

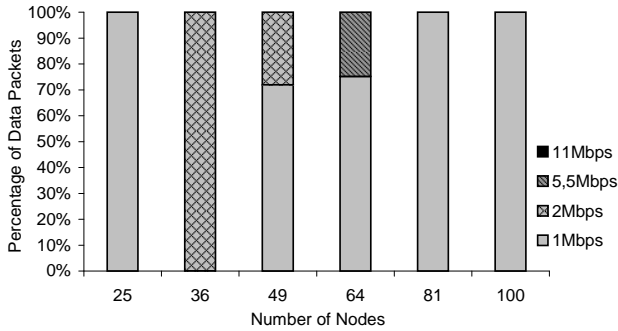


Fig. 6. Percentage of data packets transmitted with each transmission rate against the number of nodes when using AODV

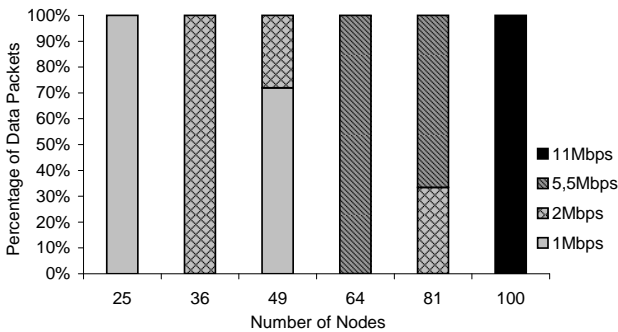


Fig. 7. Percentage of data packets transmitted with each transmission rate against the number of nodes when using MR-AODV

In the third scenario, half of the neighbors can only be reached by using 1Mbps (neighbors in the diagonals) while the other half can also be reached by using 2Mbps (neighbors above, below, in the right and in the left). Although at first sight, it would be logical that AODV would choose the path that goes directly through the diagonal (the lowest hop path), we may see in figure 6 that AODV chooses paths that contain about 2/3 of 1Mbps links (diagonal links) and 1/3 of 2Mbps links (horizontal/vertical links). With MR-AODV, about the

same percentage of 1Mbps and 2Mbps links compose the chosen path. The explanation for such behavior in both AODV and MR-AODV is similar. As we may see in table I, 2 hops at 2Mbps is more costly than 1 hop at 1Mbps ($2 \times 14 > 25$). That makes the path composed of only diagonal links not only the shortest path but also the costless one (figure 8). Nevertheless, as we have already commented our mechanism not always chooses the best path, since, like AODV, once a node receives a RREQ, it discards its subsequent copies. In this case, the RREQ forwarded by node A is not the first RREQ received by node B, so it is discarded. That makes the link A-C (1 Mbps) to be chosen as part of the path (instead of A-B, that uses 2Mbps).

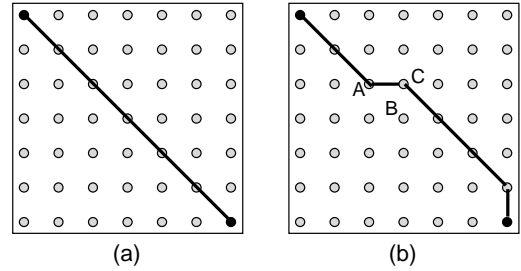


Fig. 8. (a) Shortest and also minimum cost path and (b) MR-AODV path for the 3rd scenario (49 nodes)

In fact, looking at figures 4(c), (d), (e) and (f), we can see that using 1Mbps links, a node may achieve neighbors that are farther away, what results in paths with a lower number of hops. That is what should happen with AODV in these scenarios. However, due to the just described AODV behavior, some links with higher rates (and also lower ranges) are chosen, causing some packets to be sent using these higher rates – 2Mbps in the scenario with 49 nodes and 5.5 Mbps in the scenario with 64 nodes (notice that in the latter, all neighbors that can be reached using 2Mbps can also be reached using 5.5Mbps).

When using MR-AODV, however, high rate links are preferred in most of the times. That makes it choose 5.5Mbps links in the scenario with 64 nodes (notice that in these scenarios, depicted by figures 4(d) and (e), a node can not reach any neighbor using 11Mbps). In the scenario with 81 nodes, some links that compose the chosen path use 5.5Mbps while others use 2Mbps. That happens due to the fact that according to the used metric (table I), 2 hops at 5.5Mbps have exactly the same cost as 1 hop at 2Mbps ($2 \times 7 = 14$). Finally, in the scenario with 81 nodes, the higher rate links were always chosen.

In a second simulation, we have fixed the number of nodes in 80 and randomly positioned them inside a $500m \times 500m$ region. We then varied the number of simultaneous FTP connections from 5 up to 25. For each number of FTP connections, we have launched 10 simulations with different random node positions.

Figure 9 shows the average throughput gain (and the respective standard deviation) obtained by using MR-AODV when compared to the standard AODV. As we may see, the average throughput gain we obtained when using our proposal stands between 20% and 40% in all scenarios. In a few

particular cases (6 out of the 50 launched scenarios) we had a performance decrease – of 0.36%, 0.75%, 1.37%, 1.53%, 2.94% and 13.89% – in comparison to AODV. We may see that due to this, the confidence interval crosses the unity line twice.

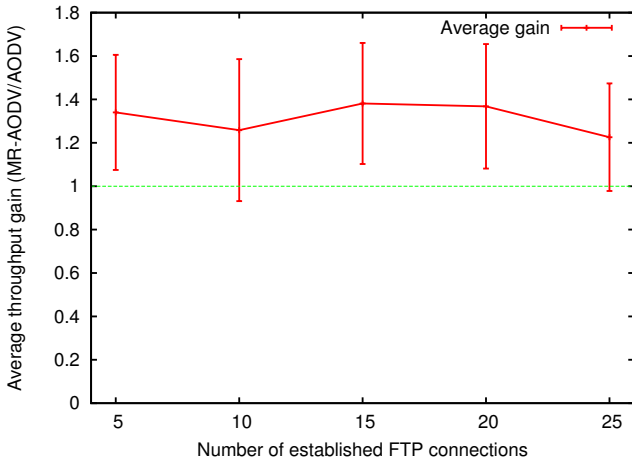


Fig. 9. Average throughput gain against the number of FTP connections

If we check the worst case (throughput decrease of 13.89%), we may see that the path chosen by MR-AODV for all 10 established FTP connections is less costly than the ones chosen by standard AODV (see tables III and IV.) We may see that 4 connections had a throughput decrease when MR-AODV was used (they are in bold in table IV), while the other 6 had their throughput increased.

TABLE III
PATHS CHOSEN BY AODV IN ONE OF THE SCENARIOS WITH 10 FTP CONNECTIONS

Path	Cost	Throughput
0 → 58 → 55 → 79	55	288Kbps
1 → 46 → 77 → 78	55	12.5Kbps
2 → 56 → 77	30	491Kbps
3 → 30 → 77 → 78 → 44 → 76	96	0.3Kbps
4 → 42 → 75	39	2.6Kbps
5 → 43 → 39 → 37 → 74	75	12.8Kbps
6 → 1 → 30 → 26 → 73	58	1Kbps
7 → 61 → 39 → 44 → 78 → 4 → 46 → 72	135	0.5Kbps
8 → 17 → 49 → 76 → 11 → 71	69	265Kbps
9 → 60 → 19 → 70	46	44.9Kbps

This decrease in the data throughput of some connections happened due to the fact that the used metric – the MTM metric – does not take into account the current state of the network. The MTM metric considers that a 5.5 Mbps link is always better than a 11Mbps, for example. However, that is not always true. A link with a lower throughput may be a better choice if it is less congested. That suggests that a metric that takes not only link rate but also network congestion into account would better distribute the load and, consequently, improve the overall network performance.

TABLE IV
PATHS CHOSEN BY MR-AODV IN ONE OF THE SCENARIOS WITH 10 FTP CONNECTIONS

Path	Cost	Throughput
0 → 18 → 79	32	38.5Kbps
1 → 20 → 4 → 62 → 78	24	29.8Kbps
2 → 56 → 9 → 59 → 77	22	32.3Kbps
3 → 30 → 20 → 4 → 62 → 40 → 66 → 31 → 5 → 76	67	14.5Kbps
4 → 62 → 75	14	43.2Kbps
5 → 44 → 29 → 54 → 74	38	363Kbps
6 → 19 → 1 → 20 → 4 → 26 → 73	43	405Kbps
7 → 71 → 61 → 39 → 29 → 54 → 75 → 77 → 59 → 52 → 72	72	8.5Kbps
8 → 13 → 5 → 44 → 39 → 61 → 71	50	48.7Kbps
9 → 59 → 46 → 60 → 19 → 6 → 70	36	14.5Kbps

Figures 10 and 11 show the percentage of packets sent using each of the available transmission rates. We may see that, when using MR-AODV, a greater percentage of data packets is transmitted using higher transmission rates, what lead to a performance increase in the huge majority of the cases, as shown by figure 9. More than 80% of all transmitted data packets used links with the two greater transmission rates with MR-AODV, while with AODV only about 50% of the packets used these rates.

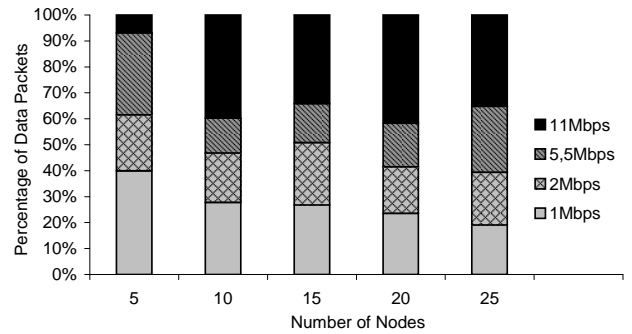


Fig. 10. Percentage of data packets transmitted with each transmission rate against the number of nodes when using AODV

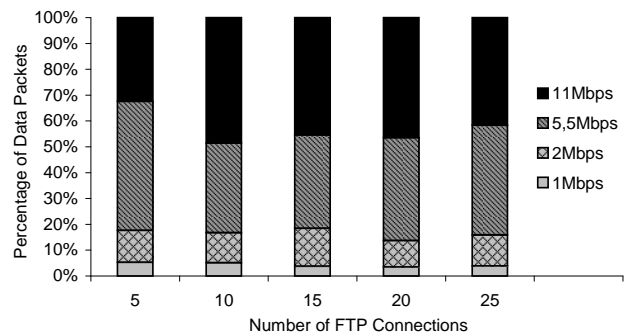


Fig. 11. Percentage of data packets transmitted with each transmission rate against the number of nodes when using MR-AODV

V. FINAL REMARKS

In this paper we have proposed a modification in the behavior of traditional reactive protocols in order to work better on multi-rate wireless ad-hoc networks. Using our proposal, reactive protocols are able to use the transmission rate as a routing metric. By doing that we were able to elect high throughput paths without increasing the signaling significantly.

We have conducted some simulations that shows the effectiveness of our proposal when applied to a particular reactive routing protocol (AODV). Through these simulations we could see that our proposal outperforms the traditional routing protocols, by choosing paths that increase significantly the overall throughput of data packets.

Finally, we may stress that although in this paper we used transmission rates as the routing metric, our mechanism could also work with any other metric, such as mean delay, link stability or available bandwidth. In fact, the simulation results for some specific cases suggested that a metric that not only take into account link rates but also network congestion could improve the overall performance even more.

REFERENCES

- [1] "IEEE Std 802.11b, Supplement to Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-speed Physical Layer Extension in the 2.4 GHz Band," 1999.
- [2] "IEEE Std 802.11a, Supplement to Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-speed Physical Layer in the 5 GHz Band," 1999.
- [3] "IEEE Std 802.11g/D8.2, Draft Supplement to Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band," 2003.
- [4] A. Kamerman and L. Monteban, "WaveLAN II: A high-performance wireless LAN for the unlicensed band," *Bell Labs Technical Journal*, pp. 118–133, Summer 1997.
- [5] G. Holland, N. Vaidya, and P. Bahl, "A Rate-adaptive MAC Protocol for Multi-hop Wireless Networks," in *Proceedings of the 7th ACM Annual International Conference on Mobile Computing and Networking (MobiCom)*, Sep 2001, pp. 236–251.
- [6] C. Perkins and S. D. E. Belding-Royer, "RFC 3561: Ad hoc On-Demand Distance Vector Routing," Jul 2003.
- [7] P. J. T. Clausen, "RFC 3626: Optimized Link State Routing Protocol (OLSR)," Oct 2003.
- [8] B. Awerbuch, D. Holmer, and H. Rubens, "The Medium Time Metric: High Throughput Route Selection in Multirate Ad Hoc Wireless Networks," *Technical Report: Department of Computer Science, Johns Hopkins University*, Oct 2004.
- [9] Z. Fan, "High throughput reactive routing in multi-rate ad hoc networks," *Electronic Letters*, vol. 40, no. 25, pp. 1591–1592, Dic 2004.
- [10] S. Zou, S. Cheng, and Y. Lin, "Multi-rate Aware Topology Control in Multi-hop Ad Hoc Networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, vol. 4, Mar 2005, pp. 2207–2212.
- [11] Y. Seok, J. Park, and Y. Choi, "Multi-rate Aware Routing Protocol for Mobile Ad Hoc Networks," in *Proceedings of the IEEE Vehicular Technology Conference*, vol. 3, Spring 2003, pp. 1749–1752.
- [12] I. Joe and S. G. Betsell, "MPR-based Hybrid Routing for Mobile Ad-Hoc Networks," in *Proceedings of the 27th Annual IEEE Conference on Local Computer Networks*, Nov 2002, pp. 7–12.
- [13] Y. Yi, M. Gerla, and T. J. Kwon, "Efficient Flooding in Ad hoc Networks: a Comparative Performance Study," in *Proceedings of the IEEE International Conference on Communications*, vol. 2, May 2003, pp. 1059–1063.
- [14] "The Network Simulator ns-2." [Online]. Available: <http://www.isi.edu/nsnam/ns>
- [15] "Datashet for Orinoco 11b Client PC Card." [Online]. Available: http://www.proxim.com/learn/library/datasheets/11bpccard_A4.pdf