A MAC Interaction Aware Routing Metric in Wireless Networks

Saquib Razak
Dept. of Computer Science,
Carnegie Mellon University,
Qatar.
srazak@cmu.edu

Vinay Kolar
Dept. of Computer Science,
Carnegie Mellon University,
Qatar.
vkolar@cmu.edu

Nael B. Abu-Ghazaleh
Dept. of Computer Science,
Carnegie Mellon University,
Qatar, and
Binghamton University, USA.
naelag@cmu.edu

ABSTRACT

Carrier Sense Multiple Access (CSMA) based MAC protocols induce several types of harmful interactions, such as hidden and exposed terminals, in wireless networks. Existing routing protocols do not consider the effect of these MAC interactions on route quality, leading to the selection of inefficient routes. We propose a MAC Interaction Aware Routing Metric (MIAR) that explicitly accounts for MAC interactions among the links forming a route. The protocol favors routes with links that have better interactions and avoids ones with detrimental interactions. We compare the performance of the proposed metric with an existing shortest-path routing protocol and show that our metric substantially improves the performance and efficiency of the network.

Categories and Subject Descriptors

C.2.2 [**Computer Communication Networks**]: Network protocols—*Routing protocols*

General Terms

Algorithms, Measurement, Performance

1. INTRODUCTION

In Multi-hop Wireless Networks (MHWNs), routing protocols find routes to connect communicating nodes. These routes are made of a *chain* of intermediate nodes that forward the traffic from the source to a destination. The protocol assigns a routing metric to the available routes, and uses them to select the route with the best metric to forward the traffic.

Several routing metrics have been proposed to determine the quality of a route. The first-generation of routing protocols minimized the number of hops in a route, thus using hop-count as a routing metric [3,5]. The next generation of routing protocols identified that hop-count does not account for the quality of the hops, and proposed link-quality based routing. For example, ETX is a metric that estimates interference by the estimated number of transmissions to transfer a packet [2].

Hop-count and link-quality based routing metrics fail to account for the detrimental interactions, such as hidden and exposed terminals [1], experienced at the MAC layer; thus not accounting for the effect of MAC layer on routing. Recent analysis have shown that two links in a CSMA network interact in more complex ways than hidden and exposed terminals [6]. The MAC interactions between a pair of links can be categorized into few discrete modes, and each interaction has a specific impact on the performance of the link. Our earlier work analyzed the effect of MAC interactions between the links in a route, and concluded that routes with harmful MAC interactions experience severe performance penalties, such as low throughput and inefficient channel utilization [4,7]. We briefly discuss these results in Section 2.

In this paper, we propose an MAC Interaction-Aware Routing (MIAR) metric that evaluates routes based on the types of interfering interactions that occur between links. MIAR metric selects routes that maximize the throughput of the network while minimizing collisions that decrease available network capacity. Specifically, the paper has two main contributions. First, we propose a metric that considers interactions between the links in the same route, i.e. we consider *self-interference* in a route. We discuss this metric in Section 3. We use this self-interference metric to choose between multiple available routes. This metric distinguishes between routes that are similar in throughput, but vary substantially in the amount of wasted transmissions required to achieve the same throughput. Hence, we show that the resulting routes utilize the channel more efficiently.

The second contribution is a hybrid routing metric that combines the effect of self-interference, and the interactions observed across links of different chains (cross-chain interference). We explain this metric in Section 4. We evaluate the performance of these metrics in routing decisions. Our results show that, although performance improvement with the first metric is slight, the metric that combines the two interactions produces improvement in more than 80% scenarios and provides an overall throughput improvement of more than 30%. Finally, we discuss the future work and conclude in Section 5.

2. BACKGROUND AND MOTIVATION

Carrier Sense Multiple Access (CSMA) based MAC pro-

MSWiM'10, October 17–21, 2010, Bodrum, Turkey.

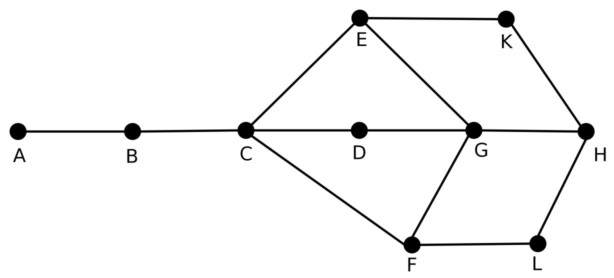


Figure 1: Example network with multiple routes from source A to destination H.

tocols, such as IEEE 802.11b, is widely used in MHWNs. CSMA is not effective in arbitrating the wireless medium between the nodes, and thus leads to several harmful effects (e.g., hidden and exposed terminals). Recent studies have identified that two links can interact only in a few discrete modes [6]. Predominantly, the links interact in four modes: No interaction (NI), Sender Connected (SC), Asymmetric Incomplete State (AIS), and Hidden Terminal with Capture (HTC). Two links have NI interaction when both the links can transmit concurrently without any packet collisions. SC interaction is observed when two links share the channel without causing any hidden terminal to the other. AIS occurs when data transmission from one link causes packet collision at another link. HTC interaction occurs due to capture effect [8]. In HTC, the receiver locks to the interferer, and fails to receive the packet from its sender.

In our previous work, we evaluated the effect of these interactions on chains [4, 7]. Links in a chain interact with other links in the same chain (self-interference), or with links in other chains (cross-chain interference). We considered a 4-hop chain since it is smallest chain with interesting interactions between the links; all interactions in smaller hop-count routes are generally SC.

We showed that – irrespective of the link interactions – all chains achieve the same throughput in a single chain. However, interactions affect the transmission efficiency of the chain; links with harmful interactions such as AIS and HTC use the channel inefficiently due to repeated re-transmissions. We also showed that routes with *good* MAC interactions, such as interactions without packet timeouts (e.g., NI, SC), are less susceptible to collision from links of other routes. Our analysis shows that MAC interactions have a significant effect on the stability and performance of a route, and this motivates us to formulate a MAC Interaction-Aware Routing Metric.

3. SELF-INTERFERENCE AWARE ROUTING METRIC

In this section we present a MAC Interaction Aware Routing (MIAR) metric that evaluates routes based on the interactions between the links in a single chain; we call the metric as *MIAR with Self-Interference* (MIAR-Self). The metric is based on two main observations [4, 7]:
(1) Type of interaction: Chains with harmful interactions, such as AIS and HTC, are inefficient.
(2) Location of interaction: Harmful interactions at the starting links of the chain have a higher impact on chain throughput than those towards the end of the chain.

Our routing metric assigns a link `AB` with a type- and location-cost (denoted by T_{AB} and L_{AB}, respectively). The type of interaction for a link is the most harmful interaction observed at the link. Based on the extent of negative impact, the interactions can be arranged in descending order as AIS-HTC-SC-NI; AIS being most harmful and NI being least harmful [6].

We learn the accurate cost of each type of interaction by simulating various 4-hop chains, and using standard linear data-fitting to compute the cost. We have observed that a type-cost (T_{AB}) of $0, 0, 1, 1.25$ to NI, SC, HTC and AIS interactions, respectively, is a good estimate to predict the end-throughput of the chain. Similarly, location-cost (L_{AB}) for n^{th} hop of a chain is the n^{th} element of the geometric sequence $1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots$. A geometric sequence is chosen because of two reasons: it is reasonably accurate, and it has an interesting decomposable property that helps us to build distributed algorithms (Section 3.2). In future, we plan to extend this approach using a more rigorous analytical method to compute the metric.

The final MIAR-Self metric of a route is the weighted summation of these costs for the constituent links. For example, MIAR-Self for a route `ABCD` in Figure 1 is given by

$$\text{MIAR-Self}(\texttt{ABCD}) = T_{AB}L_{AB} + T_{BC}L_{BC} + T_{CD}L_{CD}. \quad (1)$$

Finally, the routing protocol favors routes with lower MIAR-Self metric.

3.1 Centralized Routing

In this section, we present a simple centralized algorithm that chooses a route based on the global knowledge of routes and interactions. Available routes in a topology are computed based on the breadth-first search algorithm at a centralized node. However, in a realistic network, these routes can be discovered through route requests or other topology discovery protocols.

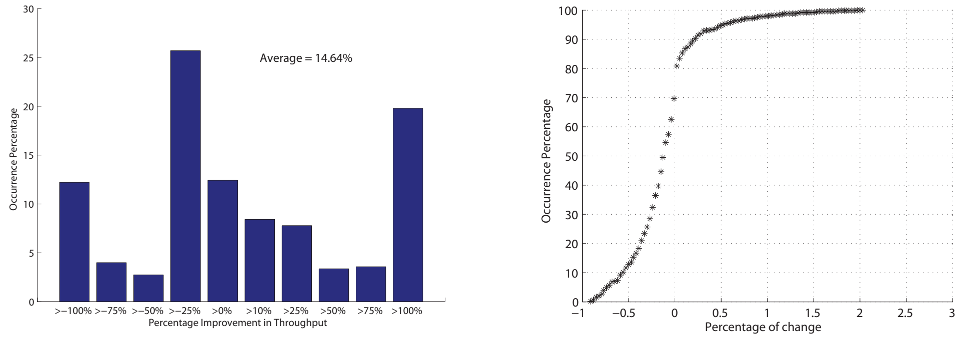
Our algorithm picks the route with the least MIAR-Self metric. This approach can be also be used in conjunction to other routing metrics. For example, if multiple routes with same hop-count or link-qualities are available, then the route with lower MIAR-Self metric can be favored. Existing routing metrics rate all routes as equal, and hence do not consider the effect of MAC interaction.

3.2 Distributed Routing

The above centralized algorithm can be easily extended to distributed routing algorithm. We modify DSR [3], a source routing protocol, for this purpose to store each route in the routing table with the computed MIAR-Self. The MIAR metric proposed in Equation 1 has an interesting property that assists a node to directly compute MIAR-Self of a new route based on the MIAR-Self of parts of route segments observed. Recall that the location-cost follow a geometric sequence, the MIAR-Self of route `ABCD` in Equation 1 can be decomposed as:

$$\begin{aligned} \text{MIAR-Self}(\texttt{ABCD}) &= T_{AB} + \frac{T_{BC}}{2} + \frac{T_{CD}}{4} \\ &= T_{AB} + \frac{T_{BC}}{2} + \frac{\text{MIAR-Self}(\texttt{CD})}{2} \\ &= T_{AB} + \frac{\text{MIAR-Self}(\texttt{BCD})}{2}. \quad (2) \end{aligned}$$

Hence, for a route request from A to destination D, each



(a) Throughput.

(b) Cumulative distribution function.

Figure 2: Throughput and transmission efficiency of MIAR-Self vs. standard DSR.

Node B		Node A	
Route	MIAR-Self	Route	MIAR-Self
BCEKH	1.25	ABCEKH	2.0
BCEGH	0.0	ABCEGH	1.25
BCDGH	0.0	ABCDGH	0.0
BCFGH	0.0	ABCFGH	1.0
BCFLH	1.25	ABCFLH	1.75

Table 1: Route metric at nodes B and A.

intermediate node C propagates the partial route till destination D (say, CD) by Route Reply (RREP) message. Each subsequent node B calculates its own interaction type-cost T_{BC}. The MIAR-Self metric for the new route BCD is directly computed by Equation 2.

For example, consider the route calculation from A to H in network shown in Figure 1. Let the pair of links (AB, EK), (AB,EG), (BC,KH) and (BC,LH) have AIS interactions, and pairs (AB, FL) and (AB,FG) have HTC. Let other links have SC or NI interactions. Intermediate node B has five routes to H, and conveys these routes to A through RREP. Table 1 shows the routes to H and the MIAR-Self computed at nodes A and B. Our algorithm picks the route ABCDGH, and all links have SC or NI interactions with each other.

Currently, the detection of interactions between links is computed through a centralized model [6]. In future, we plan to develop a distributed topology discovery algorithm that uses the model insights to detect interactions.

3.3 Evaluation

In this section, we evaluate MIAR-Self metric. We generate a network of size $1500 \times 1500\,\text{m}^2$, with 200 nodes placed at random locations, using NS2 simulator. Two random link-pairs are selected in each topology, and 500 such random topologies are evaluated. We compare the performance of DSR and MIAR-Self routing metric. To focus on the efficiency of the route and ignore the routing overhead, we eliminate the routing overhead by first picking the routes used by both the protocols, and then using these routes as static-routes in the simulation.

Figure 2(a) shows the histogram of the throughput improvement percentages. Negative values indicate that standard DSR routing metric performs better than MIAR-Self. MIAR-Self metric provides an average improvement of 15%.

Next, we compare the transmission efficiency percentage, which is the percentage of difference between the number of transmissions sent by DSR and MIAR-Self in Figure 2(b). It can be observed that MIAR-Self is more efficient; it sends lesser packets in around 80% of the scenarios.

The negative result is that the throughput of MIAR-Self is worse than DSR in around 45% scenarios (Figure 2(a)). We plan to investigate the reasons in our future work.

4. CROSS-CHAIN INTERACTION AWARE ROUTING METRIC

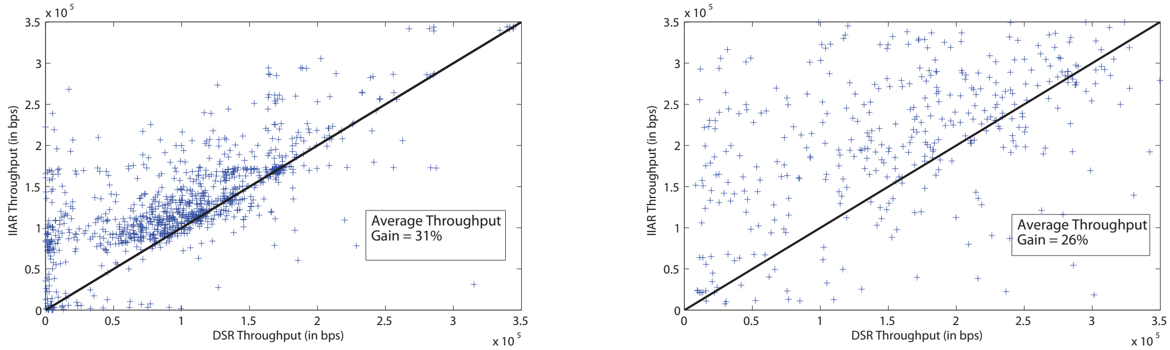
Analysis of interference across links of different chains have shown that cross-chain MAC interactions significantly affects the quality of the routes [4]. In this section, we present a MIAR-Cross; a cross-chain interaction aware routing metric.

4.1 Metric Formulation

As discussed in Section 3, the type and location of interaction between the links affect the performance of the routes. For example, an AIS between the first links of two chains has a much larger impact than between the last two hops.

MIAR-Cross assigns weights by considering the type and location of interactions for each link in a route, and assigning a cumulative metric to a route based on the individual metric. Each type and location of interaction observed at a link is considered as one variable, and is denoted by the tuple $(type, location)$. Complete characterization of route based on the tuples is complex. For example, we have to evaluate 10^{16} combination of tuples in a simple two 4-hop chain scenario (16 combination of link-pairs based on location, and each with 10 possible types of interactions [6]). Hence, a rigorous analytical evaluation of this problem is not feasible.

We use an empirical learning approach to solve this problem. We simulate 1000 random scenarios of two 4-hop chains, and record throughput of each chain. Next, we detect interactions between all pairs of links in all the scenarios, and record the $(type, location)$ tuples. We represent these tuples as variables and assign weights to all the variables. For each chain, the variables of the constituent links are mapped to its recorded throughput. This approach results in a system of equations. We use a non-linear simultaneous equation



(a) 2 concurrent flows

(b) 4 concurrent flows

Figure 3: A scatter plot that shows throughput comparison of MIAR and DSR.

solver `fsolve` from matlab to solve these equations. The weight of each tuple $(type, location)$ is obtained from this solver, and this weight is used as a metric for a given link. MIAR-Cross metric for a route is defined as the sum of metrics of constituent links.

4.2 Evaluation

We evaluate the effectiveness of the MIAR-Cross by comparing it with DSR using the similar simulation setting as described in Section 3.3. We analyze the performance of 2 and 4 simultaneous flows in a scenario.

For selecting routes based on MIAR-Cross metric, we use the following methodology. We start with assigning random minimum hop route to each end-connection between a source and destination. For a given end-connection, we compute the available routes between source and destinations. Then we compute its MIAR-Cross metric, assuming that routes of all other end-connections are fixed. This results in a best-route for one end-connection, given that other end-connections have fixed routes. Now we keep this best-route fixed, pick a different end-connection, and iterate through all of its available routes while keeping the routes of other end-connections constant. We repeat the process for all end-connections until we reach a combination of routes that converge.

Figures 3(a) and 3(b) show the comparison of the overall network throughput under MIAR-Cross and DSR in a network with two and four flows, respectively. MIAR-Cross increases the throughput of network in 80% of the cases and there is substantial improvement in the average throughput over all the runs.

5. CONCLUSION AND FUTURE WORK

Most routing protocols consider only physical aspects of topology (such as hop-count) or PHY layer parameters (such as link-quality) to evaluate the strength of route. In this paper, we presented a routing metric that considers the interactions at the MAC layer to construct efficient routes. We proposed and evaluated two types of MAC Interaction Aware Routing (MIAR) metrics: a metric that is cognizant of interactions with-in a route, and a metric that is aware of cross-chain interactions. The routing metric minimizes the interactions that have a negative impact on network performance. The paper explains the challenges involved in designing these metrics and our initial results.

Formulating a routing metric that considers key aspects of multi-hop wireless is a complex problem. In our immediate future work, we plan to analyze and refine the routing metric in more complex scenarios. The MIAR metric can also be used in conjunction with link-quality metrics, such as ETX; this is a part of our future work.

6. ACKNOWLEDGMENTS

This work was partially supported by National Science Foundation grants CNS-0958501, CNS-0751161, and Qatar National Research Fund grant number NPRP 08-562-1-095.

7. REFERENCES

[1] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang. MACAW: a media access protocol for wireless LAN's. In *SIGCOMM*, pages 212–225, 1994.

[2] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. In *ACM MobiCom '03*, pages 134–146, New York, NY, USA, 2003. ACM.

[3] D. B. Johnson, D. A. Maltz, and Y.-C. Hu. The Dynamic Source Routing protocol for mobile ad hocnetworks (DSR). In *IETF Internet draft*, 2003.

[4] V. Kolar, S. Razak, N. B. Abu-Ghazaleh, P. Mähönen, and K. A. Harras. Interference across multi-hop wireless chains. *IEEE WiMob*, pages 288–294, 2009.

[5] C. E. Perkins and E. M. Royer. Ad-hoc on-demand distance vector routing. In *IEEE workshop WMCSA '99*, page 90, Washington, DC, USA, 1999. IEEE Computer Society.

[6] S. Razak, N. B. Abu-Ghazaleh, and V. Kolar. Modeling of two-flow interactions under sinr model in multi-hop wireless networks. In *Proc. LCN*, pages 297–304, 2008.

[7] S. Razak, V. Kolar, N. B. Abu-Ghazaleh, and K. A. Harras. How do wireless chains behave?: The impact of MAC interactions. In *ACM MSWiM '09*, pages 212–220, New York, NY, USA, 2009. ACM.

[8] K. Whitehouse, A. Woo, F. Jiang, J. Polastre, and D. Culler. Exploiting the capture effect for collision detection and recovery. In *EmNets*, pages 45–52, 2005.