Simulating Cooperative Diversity Protocols for Multi-hop Wireless and Sensor Networks

Rana Azeem M. Khan and Holger Karl
University of Paderborn, Germany
Email: azeem@mail.upb.de, holger.karl@upb.de

***Abstract*—This paper discusses the simulation of Cooperative Diversity Protocols (CDP) which have emerged as a promising technique for improving reliability in a wireless environment. In CDP, neighboring node(s) cooperate with a transmitter to deliver multiple copies of a packet to a receiver, via independent fading channels. These multiple copies can be combined at the receiver to recover the original data. Researchers have proposed CDP at the physical and Medium Access Control (MAC) layers which exploit this behavior to improve reliability. This also improves the overall system energy efficiency by reducing the number of retransmissions required. However, the performance evaluation of such CDP has been limited to either analytical results or a simple three node scenario, comprising a single source-partner-destination setup. This is because CDP requires cooperative functions such as partner selection, cooperative addressing, packet combining, and three-way handshaking between source-partner-destination which are absent in conventional simulation and hardware platforms. Our previous work has shown that performance of CDP in such three node setups nodes might not reflect the actual behavior of the protocols because factors such as channel contention and collisions are simply not present. In this paper, we identify the functions needed to simulate cooperative protocols at different layers of the communication protocol stack. We use OMNet++ with MiXiM as a reference framework as it very closely resembles the TCP model. Challenges arising from simulating large scale networks, redundant packet storage, and energy utilization are also discussed.**

***Keywords*—Cooperative Diversity; Simulation; OMNet++; MiXiM; Media Access Control; Reliability.**

I. Introduction

Cooperative diversity has emerged as a promising technique for improving reliability in a wireless environment [1]. Conventional wireless systems are designed in such a way that any single transmission involves only two nodes, a transmitter and a receiver. On the other hand, in cooperative diversity, additional node(s) will cooperate with transmitter-receiver pair to deliver multiple copies of a packet to the receiver via independent fading channels.

As shown in Figure 1, the source broadcasts a message, represented by solid lines, to the destination in the first phase which is also overheard by the partner node(s). In the second phase, these neighboring partner node cooperates with source/destination by forwarding the message received in the first phase, to the destination. An interesting thing to note is that a transmission in the wireless channel is overheard by neighboring nodes anyway, but discarded in conventional systems. However, in cooperative communication, it is simply utilized by partner node for retransmission to destination .

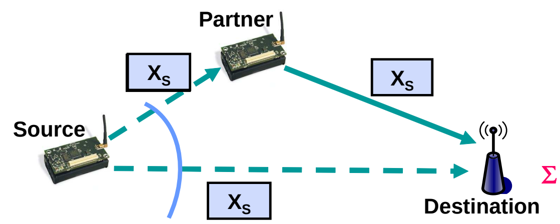

Fig. 1. Cooperative Communication Scenario

The destination can *combine* the packets received from the source and partner [2], [3], thereby exploiting spatial diversity to recover a packet which would unnecessarily be discarded in conventional communication. This reduces end-to-end propagation loss, provides robustness against channel variations due to fading, and improves coverage. These systems are also known as virtual multiple input multiple output (VMIMO) [3], [4] systems.

Development of cooperative diversity protocols has received significant attention from the research community during the last decade. Both cooperative physical and medium access control (MAC) schemes have been proposed with one survey estimating the number of proposed cooperative MAC protocols at more than 15 [5]. As researchers tend to focus their research on individual layers of the TCP model such as physical, link, or routing layer, it becomes challenging to evaluate cooperative protocols which require multi-layer support. Furthermore, lack of guidelines on simulating cooperative protocols along with the implementation complexity has limited their performance evaluation to either analytical results or a simple three node scenarios comprising a single source, partner, and destination nodes. In such setups, absence of multiple transmitters means that certain factors inherent to wireless medium such as channel contention, collisions and overhearing are simply not present. In practice, these factors significantly affect the performance of a contention-based protocol in a wireless environment.

This has motivated us to identify the cooperative functions needed for simulating or implementing cooperative protocols for large networks. We provide guidelines on how and where these functions can be implemented. To the best of our knowledge, currently no such guidelines exist. We categorize the functions into appropriate layers of the communication protocol stack to ensure compatibility with TCP/OSI model.

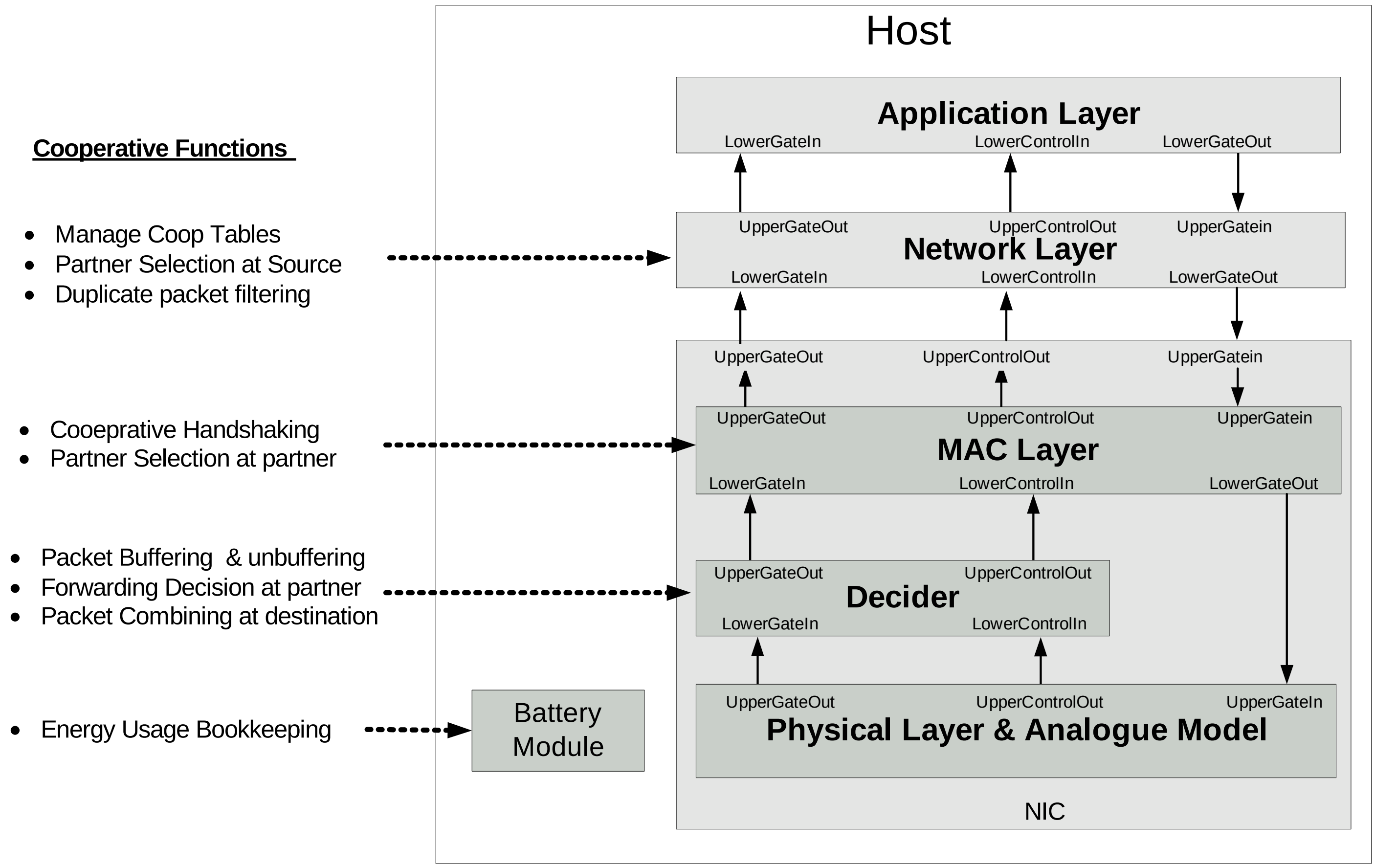

Fig. 2. Module Hierarchy in MiXiM Framework for OMNet++

We use OMNet++ with MiXiM as a reference model to elaborate the implementation details to the reader.

II. Implementation Details

A. MiXiM for OMNet++

OMNeT++ discrete event simulator, for network protocols, has been publicly available since 1997 with version 4.2 being the current release [6], [7]. It provides the base components from which further frameworks can be derived. For simulating wireless networks in OMNeT++, several frameworks have evolved such as INETMANET, Castalia, and Mobility Framework, which has now been merged into MIXIM [8]. These models follow their own development cycles, independent of OMNet++. Each of these framework comes with its own strengths and advantages.

MiXiM is an OMNeT++ modeling framework created for simulating mobile and fixed wireless networks, sensor networks, body area networks, ad-hoc networks, vehicular networks, etc [8]. The core framework provides detailed models of radio wave propagation, interference estimation, radio transceiver power consumption and wireless MAC protocols (e.g. Zigbee) which can be used to create simulations. These modules can be further inherited to incorporate additional functionality. MiXiM uses a layered protocol stack shown in Figure 2. *Physical Layer and Analogue Model* models the sending and receiving of analog signals and simulates features such as path loss, shadowing, and fading. The *Decider* is used on the receiver side to determine whether a frame was correctly received or not. The *MacLayer* is where the state machine for MAC, responsible for packet scheduling and switching the radio between sleep and awake state, is implemented. *Network layer* is responsible for storing and managing routing information. The *Application Layer* is responsible for generating packets for transmission as well as receiving packets from other hosts. Additionally, it provides a detailed radio model, several mobility models, address resolution protocols (ARP) models, dynamic connection management, and several other components [8].

B. Cooperative Modules

In this section, we identify various modules needed for cooperative diversity enabled protocols along with implementation details for OMNet++/MiXiM. Cooperative diversity protocols usually address one or more of the following challenges for enabling cooperation.

1) Cooperative Handshaking
2) Partner Selection
3) Forwarding Decision
4) Combining Decision

Figure 3 shows which node (source, partner, or destination) execute these modules and when would they be executed.

1) Cooperative Handshaking: Contention based MAC protocols, where nodes have to contend for a shared medium, attempt to minimize collisions by allowing sender and receiver to handshake i.e., negotiate medium availability prior to transmission. Such a handshake might not be needed in reservation/assignment based protocols where transmission schedules are assigned by a coordinator or an access point.

As additional nodes are involved in cooperation, contention based cooperative MAC protocol can benefit from a three-way handshake where a source, partner, and destination inform each other of their willingness to transmit and receive. Such a handshake could be source-initiated or destination-initiated[1]. Many cooperative schemes use a modified version of request-to-send (RTS) packet, known from the IEEE802.11 protocol, for this purpose. The modified RTS packet can contain destination address and some partner selection information depending on the underlying partner selection scheme. Cooperation can proceed once the partner and destination have acknowledge the RTS. Figure 3 shows an example of such a cooperative handshake. The source sends a source-ready-to-send denoted by S-RTS. The packet is acknowledged by partner which transmits relay-RTS (R-RTS). The destination responds by sending a destination-clear-to-send (D-CTS). Following this, cooperative data transmission can commence. Similar schemes can be found in [9]–[11]. While simulating such schemes, protocols providing option for direct transmission, in case of cooperative handshake failure, could be useful for maintaning overall system performance.

As shown in Figure 2, *MAC layer* in the MiXiM hierarchy serves as the appropriate place for implementing such a handshaking strategy. MAC layer implementations for 802.11 MAC layer are already available. These can further be extended for cooperation along with appropriate changes to frame formats. Wireless Sensor Networks (WSN) MAC layers such as B-MAC and Low-power-listening (LPL) have also been implemented. These MAC implementations use *radioState* to simulate radio switching between sleep, awake, and transmit states.

While overhearing is desirable in cooperative protocols, for large multi-hop networks where multiple nodes are competing for the medium and a handshake takes longer to complete, we observed that overhearing can easily interrupt an ongoing handshake. The significance of this affect will likely depend on protocol design and implementation. For this reason, we suggest that the design of a cooperative MAC protocol should include an effective overhearing strategy which utilizes overhearing for data packet and prevent an ongoing handshake from being disrupted.

2) Partner Selection: Selection of an appropriate partner for cooperation is vital for achieving high diversity gain and improving the link reliability. A partner selection scheme can be source, partner, or destination oriented.

For source oriented schemes, a partner can possibly be selected prior to initiating transmission. The source would need some local neighborhood information collected from previous transmissions, neighborhood discovery phase, or some routing layer information. This information can be stored in the *Network Layer* and represented using Coop Tables in Figure 2.

[1]Destination initiated handshake can be useful in protocols where nodes need to pull or request data from other node(s)

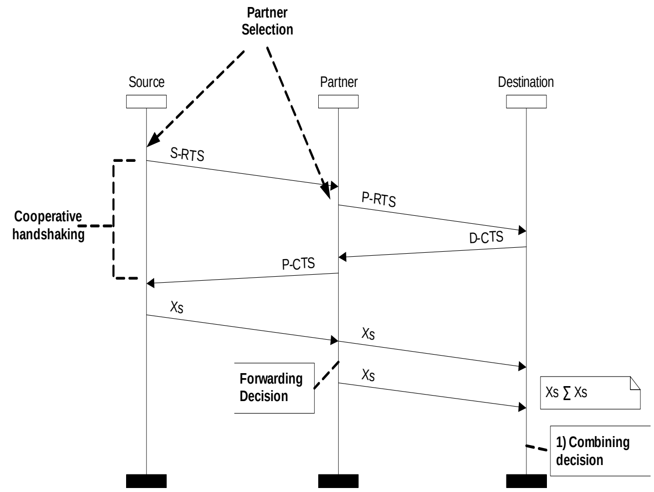

Fig. 3. Cooperative Transmission

Although a protocol for partner selection can be implemented in both *Network layer* and *MAC Layer*, implementation in *network layer* would provide better access to routing information and less intra-layer communication. In this case, the *Network layer* would be responsible for signaling the *MAC layer* to initiate a cooperative handshake, once an appropriate partner has been selected.

For a partner/destination oriented scheme, nodes can decide among themselves on a possible partner during the handshaking or in a special selection phase. The nodes can use channel quality estimates, signaling information, or some other distributed algorithm. Such information can be gathered using S-RTS/R-RTS/D-CTS packets, exchanged during handshaking, or using others control messages. Schemes using such channel quality estimates or control packets can directly be integrated into the *MAC layer*, as an alternative to *Network Layer*, where this information is readily available or can be retrieved from *Decider*. The amount of information needed to make decision, control packet overhead, and channel contention can affect the efficiency of such a scheme. To mitigate contention in such schemes, a number of protocols use an intermediate slot-assignment method where nodes are assigned time slots only during which they can send their willingness to participate [12], [13].

3) Forwarding Decision: As shown in Figure 3, the node selected as partner would be responsible for the forwarding decision. After a partner node has received a packet from source, it must decide if and how it wants to forward the packet. A partner may choose to retransmit an amplified or a compressed version of the signal. A partner can also choose to discard a packet if received erroneously. Details on forwarding schemes can be found in [2] and [3].

As signal-amplification and error-detection are related to physical and link layer, we suggest implementation of this module at the *decider* module. The conventional *decider* is responsible for determining if a packet has been received correctly. For this, *SNREval* compares the SNR value of the received packet with a pre-defined SNR threshold and correctly received packets are passed to the MAC layer while erroneous are discarded.

For a cooperative protocol, correctly received packets at the *Decider* will be passed to the MAC layer. However, for incorrectly received packets, *Decider* would need to take a decision based on its role defined during handshake. At destination, *Decider* can buffer incorrectly received packets for later combining. At partner, *Decider* can either discard erroneous packets and request retransmission from source, or buffer the packet if it expects to receive redundancy from any other node.

4) Combining Decision: The destination node is responsible for combining packets received from source and partner. Traditionally, only the correctly received packets are passed on to the MAC layer. This is to avoid unnecessary processing or scheduling of erroneous packets. To simulate this behavior in cooperative protocols, this function can be implemented in the *decider* module so only correctly decoded packets are passed on to the MAC and higher layers. This means that *decider* would need to buffer all erroneous packets for later combining. For this, a frame buffer would need to be implemented in the *Decider*. To avoid buffer overflow, *decider* can do a periodic buffer cleanup using self-timers.

To simulate packet combining, we have used *selection combining* and *maximal-ratio combining* (MRC) schemes, known from MIMO systems. More combining techniques can be found in [14]. In selection combining, the strongest signal from the N received signals is selected. We simulate this by selecting the packet with the highest SNR. In MRC, SNR values of received signals are summed up. As SNR values for packets are already available at *Decider*, the SNR of original and repeated packets is added up and then and then compared to the SNR threshold value. Successfully combined packets are passed from the decider to MAC layer where a partner node can either decide to forward them or a destination node can pass them to higher layer for processing.

5) Other Functions: In cooperative protocols, situations may arise where a packet arrives at the destination after a delay or multiple partner nodes forward multiple copies to the destination. In such cases, It could be useful to implement duplicate packet filtering in *Network Layer* layer for cooperative protocols. The size of the filtering table must be decided after considering the traffic load on the network and physical memory constraints of real world sensor nodes.

Additionally *Battery Module* module can be used to keep track of energy usage by transceiver. This could be useful in comparing energy consumption of conventional protocols with cooperative protocols. The appropriate energy usage values of the transciever along with other parameters, can be obtained from data sheet of the hardware, which the *Physical Layer & Analog Model* are simulating.

III. Multi-hop Performance

To elaborate the importance of evaluating CDP in a large scale environment, we show results from our previous work, in which we have analyzed the performance of a cooperative MAC protocol, namely CPS-MAC, in both three-node and multi-hop scenario. The multi-hop scenario consists of 17 nodes placed around a sink for data gathering and all nodes periodically generate data packets. All other parameters are same for the two configurations. The performance of CPS-MAC is compared with a traditional relaying MAC protocol, where a relay node forwards packet from source to destination and does not uses spatial diversity. For CPS-MAC, we show results for two different kind of combining techniques, namely *selection combining* and *maximal-ratio combining*, discussed in Section II-B4.

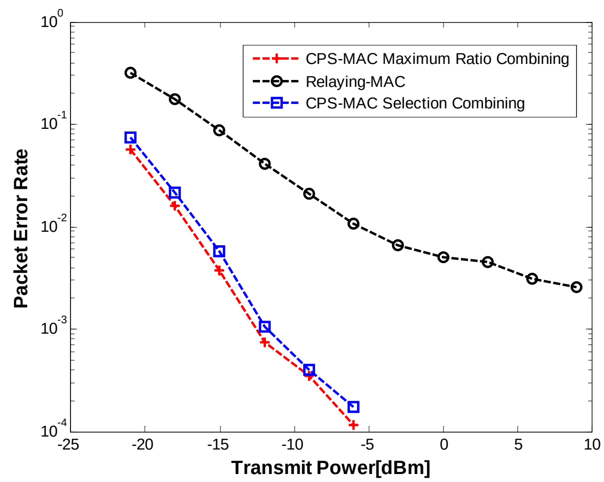

Fig. 4. Packet Error Rate for a 3 Node Network

Figure 4 and 5 show the difference in packet error rate (PER) for these two cases. Our results show that, in multi-hop network, overall PER of both cooperative and relaying protocol is higher as compared to three node scenario. This

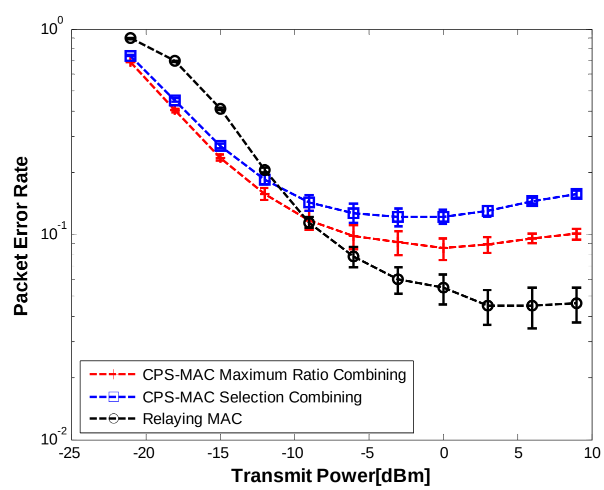

Fig. 5. Packet Error Rate for a Multihop Network

is because, an increase in network size introduces channel contention and collisions which play a significant role in increasing the PER, especially at higher transmission powers. Energy efficiency results also show a similar behavior where cooperative protocols have a higher energy consumption in the large network configuration as compared to a three node scenario. Details of protocol design, network hierarchy, and simulation parameters can be found in [15], [16].

IV. Conclusion

In this paper, we have identified the various functions for modeling cooperative MAC protocols in OMNet++/MiXiM. The implementation details of these functions in various layer of MiXiM framework is meant to benefit developers interested in simulating cooperative protocols. We stress the importance of evaluating performance in a realistic network configuration by showing results from our own simulation experience. Factors such as packet overhearing, redundant packet storage, and energy usage and how they can be effectively implemented in MiXiM are also discussed.

References

[1] J. Laneman, D. Tse, and G. Wornell, “Cooperative Diversity in Wireless Networks: Efficient Protocols and Outage Behavior,” *IEEE Transactions on Information Theory*, vol. 50, pp. 3062–3080, Dec. 2004.
[2] S. Valentin, H. S. Lichte, H. Karl, G. Vivier, J. Vidal, and A. Agustin, “Cooperative wireless networking beyond store-and-forward : Perspectives for PHY and MAC design,” *Design*, pp. 1–14, 2006.
[3] a. Nosratinia, T. Hunter, and a. Hedayat, “Cooperative communication in wireless networks,” *IEEE Communications Magazine*, vol. 42, pp. 74–80, Oct. 2004.
[4] A. H. Dohler, M., Lefranc, E., Aghvami, “Virtual Antenna Arrays for Future Wireless Mobile Communication Systems,” in *ICT*, 2002.
[5] F. Gomez-Cuba and R. Asorey-Cacheda, “A Survey on Cooperative Diversity for Wireless Networks,” *IEEE Communications Surveys and Tutorials*, pp. 1–14, 2011.
[6] A. Varga, “An overview of the OMNeT++ simulation environment,” *of the 1st international conference on Simulation*, 2008.
[7] A. Verga, “OMNeT++Discrete Event Simulation System.” http://www.omnetpp.org.
[8] A. Köpke, M. Swigulski, K. Wessel, and E. al., “Simulating wireless and mobile networks in OMNeT++ the MiXiM vision,” *OMNet++ Workshop*, 2008.
[9] V. H. Rabarijaona and S. Shimamoto, “Partnership-Based Cooperative MAC Protocol,” *2010 7th IEEE Consumer Communications and Networking Conference*, pp. 1–3, Jan. 2010.
[10] M. Khalid, Y. Wang, I. Ra, and R. Sankar, “Two-Relay based Cooperative MAC Protocol for Wireless Ad hoc Networks,” *IEEE Transactions on Vehicular Technology*, vol. 60, no. 7, pp. 3361–3373, 2011.
[11] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. Panwar, “CoopMAC: A Cooperative MAC for Wireless LANs,” *IEEE Journal on Selected Areas in Communications*, vol. 25, pp. 340–354, Feb. 2007.
[12] Y. Zhou, J. Liu, C. Zhai, and L. Zheng, “Two-Transmitter Two-Receiver Cooperative MAC Protocol: Cross-Layer Design and Performance Analysis,” *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 8, pp. 4116–4127, 2010.
[13] A. Azgin and Y. Altunbasak, “Cooperative MAC and routing protocols for wireless ad hoc networks,” *, 2005. GLOBECOM*, pp. 6 pp.–2859, 2005.
[14] J. D. Gibson, *The communications handbook*. CRC Press, 2002.
[15] R. Khan and H. Karl, “Multihop Performance of Cooperative Preamble Sampling MAC (CPS-MAC) in Wireless Sensor Networks,” *Ad-hoc, Mobile, and Wireless Networks*, no. Cc, pp. 145–149, 2011.
[16] R. Khan and H. Karl, “Cooperative Communication to Improve Reliability and Efficient Neighborhood Wakeup in Wireless Sensor Networks,” in *UBICOMM 2010, The Fourth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, no. CC, pp. 334–341, 2010.