A Cooperative MAC Providing Alternate Path for the Poor Link

Mohammad Helal Uddin Ahmed
Rim Haw
Choong Seon Hong

Department of Computer Engineering
Kyung Hee University
Korea
E-mail: cshong@khu.ac.kr

Md. Obaidur Rahman

Department of Computer Science and Engineering
Dhaka University of Engineering and Technlogy (DUET)
Gazipur-1700, Bangladesh.

***Abstract*—We exploit helper-to-helper *(h-2-h)* cooperation using the broadcast nature of wireless sensor networks. The potential helpers overhear the packet from the transmission of previous hop and the helper transmits the packet in the current hop on behalf of the assigned node (sender). We call it friendship relaying cooperative medium access control (FRC-MAC) protocol. This mechanism eliminates the sender transmission phase, improving the energy efficiency and end-2-end delay. Our innovative proposed MAC protocol provides alternate path when a link is poor temporarily, hence no need to find a new path frequently. In this protocol, each node manages a *helper queue* in additional to its data queue to store cooperative data so that the helper can transmit cooperative data with higher priority than its own data. We performed a simulation study to investigate network performance and compared with existing protocols. We found that FRC-MAC outperforms existing protocols in terms of end-to-end delay and energy efficiency.**

I. INTRODUCTION

[1] In traditional cooperative communication, a sender transmits a data packet to receiver and the neighbors overhear the packet in the first phase (1^{st} time slot). Subsequently, the helper retransmits the packet to the receiver in the second phase. In wireless sensor networks (WSNs), cooperative communication technology significantly increases the network performance [1] and provides alternate paths when the defined link fails [2]. However, one of the major challenges of cooperative communication is handling cooperation overhead. The cooperation overhead can be incurred in either helper selection or in helper retransmission. Cost of these overheads causes larger delay and higher energy consumption. Hence, if we can optimize these overheads then it will certainly improve the expected costs of cooperative transmission.

Current research in cooperative wireless networks [1], [3], [4], [5] and [6] has mostly focused on protocols that consider either energy efficiency to maximize the network lifetime or to increase network throughput. In [7] and [8], residual-energy aware transmission models were developed to increase the lifetime of the network. In contrast, a few paper pay attention in optimizing the cooperation overhead. Recently, LC-MAC [9] proposes single-phase cooperation concept where they try to remove one transmission phase. In their protocol, the helper overhears and decodes the packet from the transmission of previous hop and retransmits the packet in the current hop along with the sender. Since, helper and sender transmit at the same time, thus the additional second phase is no needed for LC MAC.

The LC-MAC improves the network performance significantly. However, it has several drawbacks such as, the sender and helper transmit at the same time using distributed space time coding (DSTC) technology. The DSTC technique uses complex signaling and processing method; it also requires extra signaling over dedicated channels [2] [10] which are not taken into account in estimating the LC-MAC performance. Furthermore, some extra cooperation overhead is introduced in LC-MAC, such as a group indication (GI) message, a member indication (MI) message and additional re-contention period that prolongs the relay selection time as well as affect the end-to-end delay.

In our scheme, we do not use DSTC, thereby avoiding extra signaling overhead. In our proposal, we consider the benefits of the broadcast nature of wireless communication, in which the potential helper(s) receives and decodes data packets successfully along with the receiver in the previous hop. Subsequently, the helper alone transmits the data packet on behalf of the sender in the current hop. The rationales behind this proposal are as follows:

- to avoid extra dedicated control channel
- to reduce relay retransmission phase of traditional cooperative protocol.
- to reduce transmission energy (only helper transmits the data on behalf of sender unlike LC-MAC)
- to avoid the re-establishment of a whole route when one link is temporarily dropped and it is the most important goal of FRC-MAC.

[1]"This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency)" NIPA-2012-(H0301-12-1004) *Dr. Choong Seon Hong is the corresponding author.

The proposed protocol has been designed in the context of IEEE 802.11 DCF. However, the model changes in cooperative communication [9] as it introduces new features to increase the network performance. We believe that our protocol is not restricted to WSNs; it can also be applied to other wireless systems such as wireless mesh networks and wireless broadband networks.

The rest of the paper is organized as follows. The network model is described in Section II. In Section III, we describe the helper selection criteria. Protocol description of FRC-MAC is given in Section IV. Simulation results are explained in Section V-D. Finally, we conclude this paper in Section VI.

II. Network Model

The considered network scenario for the proposed work is shown in Fig.1. Assume that source S has a packet to be transmitted to the node C through a predefined path. Our protocol selects either direct communication (DRC) mode or cooperation mode based on certain conditions. In a suitable environment, direct transmission occurs in the hop SA. Further, we classify the cooperation mode into two types: traditional relaying cooperative (TRC) mode and friendship relaying cooperation (FRC) mode. TRC mode is triggered in the hop AB where sender A transmits to receiver B in the first phase, and this transmission is overheard by helper H_1 which subsequently H_1 retransmits to receiver B in the second phase. FRC mode is triggered in the hop BC where H_2 overhears and decodes the packet from the transmission of the previous hop AB and on behalf of node B transmits the packet in the current hop BC. This is an innovative single-phase cooperation mode that is introduced in this paper for the first time. In Fig. 1, H_2 overhears the packet from the transmission of either A or H_1 of the previous hop AB. Note that helper H_2 can decode the packet transmitted by A or H_1 if they lie in each other's transmission ranges.

In this protocol, the set of helpers (S_H) are divided into two subsets: (S_{FRC}) and (S_{TRC}) for FRC and TRC mode, respectively. The helper set S_H can be written as $S_H = S_{FRC} \cup S_{TRC}$, where $S_{FRC} \cap S_{TRC} = \{\phi\}$. If helper H_i can decode the packet from the previous hop, then $H_i \in S_{FRC}$; otherwise $H_i \in S_{TRC}$.

Our proposed protocol is adaptive and the transmission mode depends on the link quality and availability of helper subsets. Suppose as in Fig.1, node S has a data packet to be sent to C through forwarding nodes A and B. Here, link SA is better but the links AB and BC are vulnerable/poor in terms of link quality. Thus, direct transmission occurs in the hop SA and cooperative transmission occurs in the hop AB and BC. Since, $S_{FRC} = \{\phi\}$ for the hop AB thus node A transmits using TRC mode. On the other hand, $S_{FRC} \neq \{\phi\}$ for the hop BC, thus the node B uses FRC mode for the hop BC. The decision process to select the transmission mode of the sender is shown in Algorithm 1.

Uniform distribution is applied among the nodes in the network. We also propose a concept of a *helper queue* which nodes use to buffer others' data to be transmitted when acting as a helper. Therefore, each node manages two queues: a transmission queue and a *helper queue*. Furthermore, each node keeps track of the packet error rate of its neighborhoods. All the control packets are sent using 1 Mbps.

Algorithm 1 Algorithm for selection of transmission mode by sender

Initialization: Packet error rate between sender and receiver$= PER$;
if $PER < \gamma\%$ **then**
  transmit using direct mode;
else
  if $S_{FRC} \neq \{\phi\}$ **then**
    switch to FRC mode;
  else
    switch to TRC mode;
  end if
end if

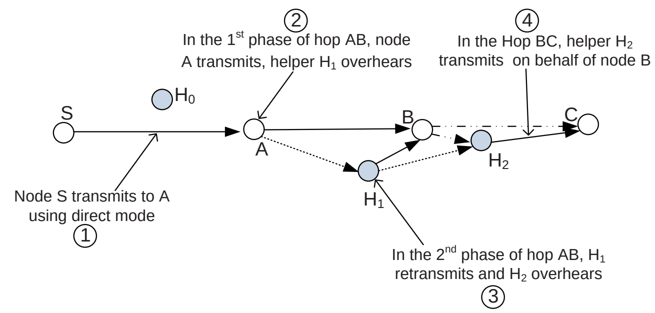

Fig. 1. Basic network model: DRC, TRC and FRC modes are used in the hop SA, AB and BC respectively

In brief, node S transmits to A using DRC mode for the better link SA, node A transmits to B using TRC mode for the poor link AB, and node B transmits to C using FRC mode for the poor link BC. Since, $S_{FRC} = \{\phi\}$ for the hop AB, thus the node A uses TRC mode. In contrast, $S_{FRC} \neq \{\phi\}$ for the hop BC, thus the node B uses FRC mode.

III. Helper Selection Criteria

The helper evaluation criteria greatly affects the performance of the cooperative protocol. We assume that nodes having residual energy below r% of their initial energy ($E_{res} < r\%$ of E_{ini}) cannot participate in cooperation. These nodes can only send their own data or forward others' data, but they do not act as helpers. Since, cooperation is a volunteer task to increase the network performance causing some extra energy consumption.

It is obvious that in Fig. 1 the signal-to-noise ratio of RTS (SNR_{rts}) reduces toward node C and the signal-to-noise ratio of CTS (SNR_{cts}) reduces toward the node B for the hop BC. Therefore, the closer a helper is to the sender B, the stronger the SNR_{rts} and poorer the SNR_{cts} are. If we select such a helper which overhears strong SNR_{rts} and poor SNR_{cts}, then it might increase the rate of packet error. To avoid this problem, each potential helper manages *cooperative SNR thresholds* SNR_{rts_th} and SNR_{cts_th} for RTS and CTS, respectively. Basically, both

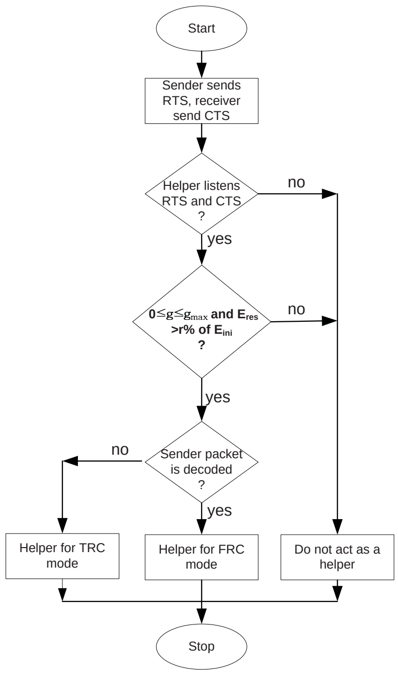

Fig. 2. Who the suitable helper is?

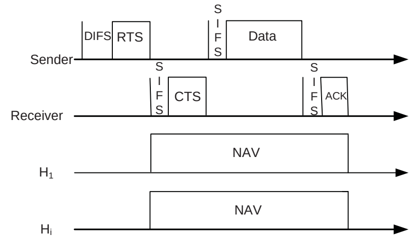

Fig. 3. Frame sequence for DRC mode

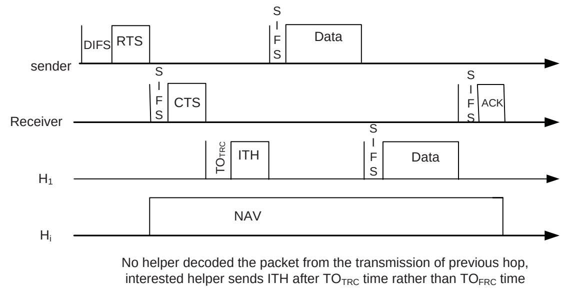

Fig. 4. Frame sequence for TRC mode

the threshold values represent the same point but the values are different.

If a potential helper overhears RTS and CTS, and satisfies $SNR_{rts} \leq SNR_{rts\_th}$ and $SNR_{cts} \geq SNR_{cts\_th}$, then it is allowed to participate in cooperation. The difference of the *cooperative SNR thresholds* is called cooperative range g, where g is given by $g = \mid SNR_{rts\_th} - SNR_{cts\_th} \mid$. The value of the cooperative range g is $0 \leq g \leq g_{max}$.

$$g = \begin{cases} g_{max} & \text{if } SNR_{rts} = SNR_{rts\_th} \text{ and } SNR_{cts} = SNR_{cts\_th} \\ 0 & \text{if } SNR_{rts} = SNR_{cts} \end{cases}$$

A helper with the value of $g = 0$ depicts that its position is about in the middle of the sender and receiver for a symmetric link. A helper which is in the middle of the sender and receiver shows better performance [10]. Therefore, the lower the value of g is, the better the helper is. When a helper listens to both RTS and CTS within the range g, then it checks its own helper queue, whether the packet with the given tag is decoded or not. If the packet is decoded, then it can participate as a helper in FRC mode; otherwise it can compete for TRC mode. The entire process is shown in Fig.2.

IV. Protocol description

Our protocol activates one of two modes: direct (DRC) or cooperative transmission. We additionally classify the cooperative mode into two modes: TRC and FRC. In this section, we focus our discussion on the hop BC in which our proposed FRC mode activates, as shown in Fig. 1. However, for better understanding of the readers, we explain the modes DRC, TRC, and FRC for the hops SA, AB, and BC respectively.

A. Protocol operation for DRC mode

In this subsection, we describe the protocol for DRC mode for the hop AB. When S has a data packet to send, it initiates the transmission process by sending RTS as per the legacy of 802.11 DCF. In response, node B sends CTS after SIFS time. Here, we consider that sender S satisfies the condition $PER < \gamma\%$ for the hop SA, thus S activates DRC mode. Therefore, sender S sends the data packet to node A after SIFS time. The frame transmission sequence for DRC mode is shown in Fig.3. Since, $PER \geq \gamma\%$ for the hop AB, thus A sent CTS in the hop SA with tag x for the data packet. The neighbors (helpers) who hear the CTS and who are interested in helping node A they decode and store the data packet along with the tag x.

B. Protocol operation in cooperative mode when there is no collision among the helpers

In this subsection, we describe the protocols for cooperative modes for simple scenario in which only one helper is interested in helping to the sender.

TRC mode (hop AB): Node A wants to forward the data packet that is tagged with x. Therefore, it sends RTS with tag x; in response, node B sends CTS. However, if the PER is $PER \geq \gamma\%$ for the link BC, then it includes a tag y in the CTS. The helper H_i who listens to both the RTS and CTS and satisfies the conditions $H_i \in S_{FRC}$, $0 \leq g \leq g_{max}$ and $E_{res,H_i} \geq r\%$ of E_{ini}, sends HTS message after TO_{FRC} time, where $TO_{FRC} = SIFS + slot$. Since, $S_{FRC} = \{\phi\}$ for the hop AB and no HTS is received after TO_{FRC} time, thus the node A switches to TRC mode. Therefore, H_1 sends ITH message after TO_{TRC} time, where $TO_{TRC} = SIFS + 2slot$ and $TO_{TRC} > TO_{FRC}$. In this case, node A transmits the data

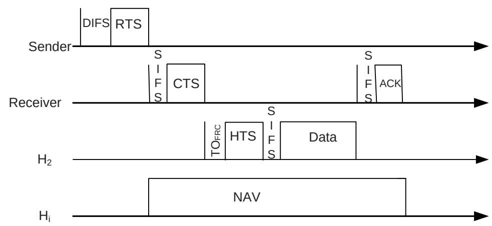

Fig. 5. Frame sequence for FRC mode

packet to node B in the first phase like traditional cooperation mode, where H_1 overhears the packet. Subsequently, H_1 retransmits the packet to the node B in the second phase. The frame transmission sequence for TRC mode is shown in Fig. 4. Note that H_2 overhears the transmission of either H_1 or A, because they are within each other's transmission range, and so they store the packet as a helper for the hop BC.

FRC mode (hop BC): Consider that node B wants to forward the data packet to node C that was tagged with y in the previous hop AB. If it is fact that B's PER is $PER \geq \gamma\%$ for the hop BC, then B sends RTS with tag y. In response, the node C sends CTS. A potential helper H_i who listens to both the RTS and CTS and that satisfies the conditions $H_i \in S_{FRC}$, $0 \leq g \leq g_{max}$ and $E_{res,H_i} \geq r\%$ of E_{ini} sends HTS message after TO_{FRC} time. Since, the helper H_2 has the y-tagged packet and decoded it from the transmission of previous hop AB and satisfies all other conditions; thus, it sends HTS message after TO_{FRC} time. Subsequently, after SIFS time of sending HTS, the helper H_2 transmits the packet on behalf of node B, and accordingly node C sends ACK. The frame sequence for FRC mode is shown in Fig.5. Note that if the PER of node C is $PER < \gamma\%$ for a link say for CD, then the tag field of C's CTS is empty; otherwise the CTS is tagged with z.

C. Protocol operation in cooperative mode to avoid collision among the helpers

The most critical scenario appears when there are multiple helpers and collision occurs among them. If more than one suitable helper is available, then they send HTS messages simultaneously; this scenario is called helper collision, and the helpers are called *collided helpers*. Selection of a single helper from this candidate helper set is a critical issue for cooperative communication, because it increases the protocol overhead. In the proposed protocol, our efforts to optimize the overhead are validated by our performance evaluation results. In our protocol, several time slots (mini-slots) are allocated for selection of the most suitable helper. We apply an efficient algorithm named the *splitting algorithm* [11] [12] to select an optimal helper. The algorithm takes less than three mini-slots to find an optimal helper. Once a helper such as H_2 is selected, then it transmits the packet (on behalf of the sender B) to receiver C after SIFS time, as shown in Fig. 6. The node C sends ACK to the node B after another SIFS time. The same

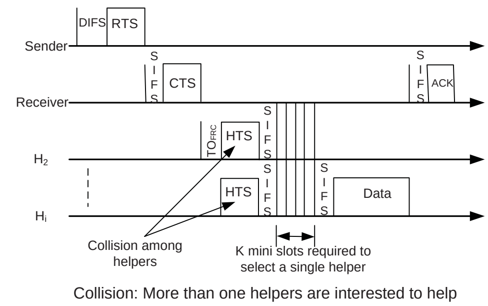

Fig. 6. Frame sequence for FRC mode when collision occurs

procedure is followed to resolve the collision when more than one helper sends an ITH message in TRC mode.

V. Performance Evaluation

A. Simulation Environment

We evaluate the performance metrics of end-to-end packet delay and per packet energy consumption for various node densities over radio transmission ranges up to 100 meters. We evaluate the performance using simulation experiments conducted on ns2 [13]. The simulation was done based on the legacy of 802.11 DCF. In our network setup, we placed 2000 nodes in a square region of side 2000 meters. Uniform node distribution was used in our network set-up. Initial energy of each node was 100 Joule and the sink was placed at the center of the network. For each data point in the graphs, we averaged the results of ten simulation runs.

B. Selection of Parameter Value

We assume that our protocol does not switch to cooperation mode when $PER < \gamma\%$. In this paper, we select a suitable value of $\gamma = 10\%$. The major motivation of such consideration is

- if the $PER < 10\%$ then the cooperative communication is not much benefited due to the cooperation overheads.

However, The value of γ is not restricted to 10 for designing a protocol rather it depends on the required QoS. One can set the value of γ as low as per the requirement of applications with sacrificing the cooperation benefits.

We also assume that nodes with residual energy $E_{res} \leq r\%$ of E_{ini} will not participate in cooperation. In this work, we select a suitable value of $r = 30\%$ which provides better balanced energy consumption [14] among the nodes. If the nodes having residual energy below 30% of initial energy participate in cooperation as helper then some nodes finish their energy earlier than the expected time. This fact may partition the network and results part of network may be non-functional.

C. Performance Metrics

1) Average end-2-end packet delay: End-2-end delay of a single packet is measured as the time difference between the packet is received at the sink and its generation time at the source node. Delays experienced by individual data packets are averaged over the total number of individual packets received

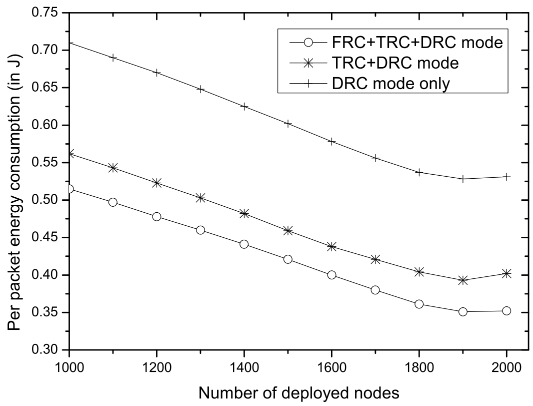

Fig. 7. Energy consumption vs. number of deployed nodes for different communication modes

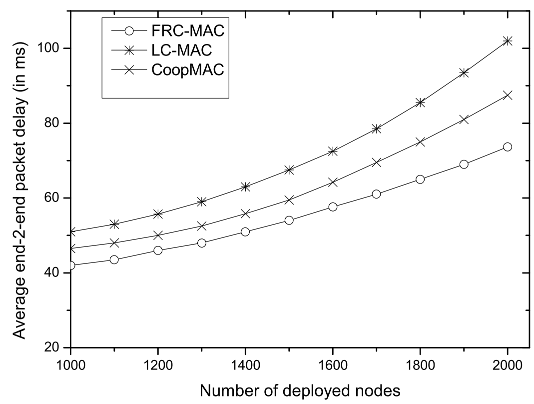

Fig. 9. end-to-end delay vs. number of deployed nodes for different protocols

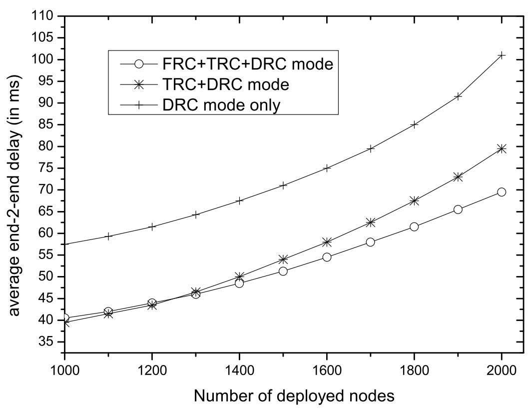

Fig. 8. Average end-to-end delay vs. number of deployed nodes for different communication modes

by the sink. The lower the value is, the better the performance is.

2) Per-packet energy consumption: Energy consumption per packet is measured as the ratio of the total amount of energy dissipated by all sources and forwarding nodes during the simulation period to the number of packets received by the sink: i.e, the average amount of energy expended for each successful packet reception.

D. Performance Analysis

In this subsection, we show the impacts of node density on the performance of the protocols. We compared the results of FRC mode with those of TRC and DRC modes for per packet energy cost and end-to-end delay. We have also compared the results of FRC-MAC with the state-of-the-arts LC-MAC [9] protocol and the very well-known CoopMAC [3] protocol.

The graphs of Fig. 7 show the per-packet energy consumption for DRC, TRC and FRC modes. The graphs show that our proposed FRC protocol is 12% more energy efficient over DRC and TRC modes at high node densities. This is because TRC mode requires extra transmission and reception phase over FRC mode. Our protocol is more energy efficient at higher node densities than that of at lower densities. This is because most of the times, senders activate either TRC mode or DRC mode at lower node densities due to the lack of suitable helper for FRC mode. Consequently, DRC mode requires more retransmission for successful transmission and hence required more transmission energy.

The graphs of Fig. 8 show that end-to-end delay improved significantly in the FRC mode compared to the TRC mode. Although end-to-end delay of the FRC mode is poor at lower node densities, it improves with node densities. The reason is that our protocol transmits in DRC mode or TRC mode at lower node densities due to the lack of a suitable helper in FRC mode. The number of suitable helpers increase with node densities in FRC mode. Thus, our protocol transmits more frequently in FRC mode than in TRC mode at higher node densities and hence end-to-end delay decreases. The probability of collision increases among the nodes as node density increases. Therefore, end-to-end delay deteriorates with node densities in all modes of operation.

The graphs of Fig. 9 show that the average end-to-end delay of FRC-MAC is significantly improved over LC-MAC and CoopMAC. The reason is that our FRC-MAC reduces sender transmission phase, thus requires lower cooperation overhead than the traditional cooperative protocol CoopMAC. LC-MAC also uses single phase cooperation; however, it requires higher cooperation overhead due to its different group indication (GI) message, member indication (MI) message and additional contention period in its helper selection mechanism. The end-2-end delay increases for all protocols because collisions increase with node densities.

Figure 10 shows that FRC-MAC is more energy efficient than LC-MAC and CoopMAC. This is because the number of successful packets in the simulation period was higher

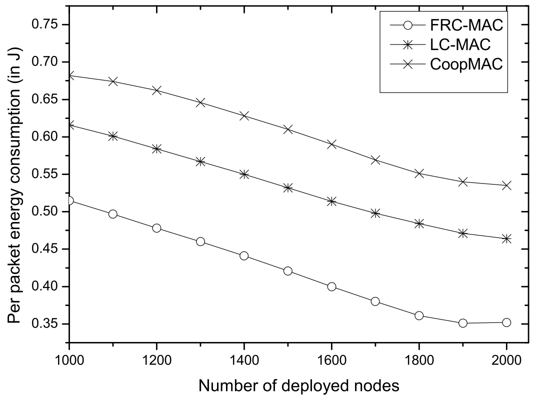

Fig. 10. Energy efficiency vs. number of deployed nodes for different protocols

in FRC-MAC. It is lower in LC-MAC and CoopMAC due to their higher collision rates during the simulation period. Therefore, per-packet energy consumption is lower in FRC-MAC than that of LC-MAC and CoopMAC. Another reason is that the sender and helper transmit same packet in LC-MAC and CoopMAC whereas, in FRC-MAC, only the helper transmits on behalf of the sender. Thus, it requires less average transmission energy than LC-MAC and CoopMAC. Energy efficiency increases with node densities because cooperative diversity increases. However, in all the protocols, energy efficiency decreases after certain node density because collisions increase with the node densities.

VI. Conclusion

In this paper, we present a new adaptive relaying cooperative MAC (FRC-MAC) protocol for wireless sensor networks. Our protocol switches to cooperative transmission mode from DRC mode if the link is poor. In this protocol, helper overhears and decodes a data packet from the previous hop. The helper transmits the packet on behalf of the sender in the current hop if the link is poor to transmit. Thus, our protocol provides alternate path for the poor link instead of finding a new route frequently. Since, FRC-MAC reduces sender transmission phase, thus our protocol also minimizes the cooperation overheads. We performed a simulation experiment using ns-2 to analyze the performance. Simulation results show that FRC-MAC outperforms the interesting protocol LC-MAC and CoopMAC in terms of energy efficiency, and end-to-end delay.

References

[1] H. Shan, W. Zhuang, and Z. Wang. Distributed Cooperative MAC for Multihop Wireless Networks. *Communications Magazine*, 47:126–133, February 2009.

[2] B. Escrig, Splitting Algorithm for DMT Optimal Cooperative MAC Protocols in Wireless Mesh Networks, *Physical Communication, Elsevier*,vol. 4, no. 3, p. (on line), Septembre 2011.

[3] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar. CoopMAC: A Cooperative MAC for Wireless LANs. *IEEE Journal on Selected Areas in Communications*, 25(2):340 -354, 2007.

[4] Y. Chen and Q. Zhao. Maximizing the Lifetime of Sensor Network Using Local Information on Channel State and Residual Energy, *Conference on Information Sciences and Systems*, 2005.

[5] X. Liang, I. Balasingham, and V. Leung. Cooperative Communications with Relay Selection for QoS Provisioning in Wireless Sensor Networks. *Global Telecommunications Conference, GLOBECOM IEEE*, 2009.

[6] H. Adam, W. Elmenreich, C. Bettstetter, and S.-M. Senouci. CoRe-MAC: A MAC-Protocol for Cooperative Relaying in Wireless Networks. *Global Telecommunications Conference, GLOBECOM*, pages 1–6, 2009.

[7] E. Liu, Q. Zhang, and K. Leung. Residual Energy-Aware Cooperative Transmission (REACT) in Wireless Networks. In *Wireless and Optical Communications Conference (WOCC)*, , pages 1 -6, May 2010.

[8] A. Banerjee, C. H. Foh, C. Yeo, and B. S. Lee. A Network Lifetime Aware Cooperative MAC Scheme for 802.11b Wireless Networks. *Consumer Communications and Networking Conference (CCNC), IEEE*, pp 1 –5, 2010.

[9] Y. Zhou, J. Liu, L. Zheng, C. Zhai, and H. Chen. Link-Utility-Based Cooperative MAC Protocol for Wireless Multi-Hop Networks. *IEEE Transactions on Wireless Communications*, 10(3):995 -1005, March 2011.

[10] Zhengguo Sheng and Leung, K.K. and Zhiguo Ding, Cooperative Wireless Networks: from Radio to Network Protocol Designs *Communications Magazine, IEEE*, pp. 64 -69 vol. 49, May 2011.

[11] X. Qin and R. Berry, “Opportunistic Splitting Algorithms for Wireless Networks,” in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 3, 2004, pp. 1662 –1672 vol.3.

[12] V. Shah, N. Mehta, and R. Yim, Splitting Algorithms for Fast Relay Selection: Generalizations, Analysis, and a Unified View, *Wireless Communications, IEEE Transactions on*, vol. 9, no. 4, pp. 1525 –1535, 2010.

[13] ns2, *“The Network Simulator”*, http://www.isi.edu/nsnam/ns/.

[14] C. Shu, S. Jintao, J. Zhuqing, Q. Ningning, and X. Baoguo. A Novel Energy Consumption Balance Strategy for Wireless Sensor Network. *International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, , volume 1, pages 226 –229, 2010.