PAPER

Reliable Broadcast Scheme for IEEE 802.15.5 Low-Rate WPAN Mesh Networks

Woongsoo NA[†], **Gunwoo LEE**[†], **Hyungchul BAE**[†], **Jungsuk YU**[†], *Nonmembers*, *and* **Sungrae CHO**[†], *Member*

SUMMARY The IEEE has recently released IEEE 802.15.5 standard [5] to provide multi-hop mesh functions for low-rate wireless personal area networks (WPANs). In this paper, we extensively describe a link-layer reliable broadcast protocol referred to as timer-based reliable broadcast (TRB) [5] in the IEEE 802.15.5 standard. The TRB scheme exploits (1) bitmap based implicit ACK to effectively reduce the unnecessary error control messages and (2) randomized timer for ACK transmission to substantially reduce the possibility of contentions. Performance evaluation shows that the TRB scheme achieves 100 % reliability compared with other schemes with expense of slightly increased energy consumption.

key words: *Personal area network, reliable broadcast, mesh network, timer, and acknowledgement.*

1. Introduction

Wireless mesh networks (WMNs) has been emerged with attractive features such as (1) extension of network coverage without increasing transmit power or receive sensitivity, (2) enhanced reliability via route redundancy, (3) easier network configuration, and (4) longer device battery life. Thanks to these capabilities, WMNs has provided a large number of potential applications. To keep pace with the rapid growth of WMNs, IEEE standard working groups have developed standards for WMNs based on 802.11, 802.15, and 802.16.

The IEEE 802.15 task group 5 has recently released the IEEE 802.15.5 standard [5]. The standard provides an architectural framework enabling WPAN devices to promote interoperable and stable wireless mesh topologies. Although the applications enable the IEEE 802.15.5 to utilize a fully distributed MAC without any central coordinator. Logical groups are formed around each device to facilitate contention-free exchanges while exploring medium reuse over different spatial regions. The membership of devices to these groups can vary in time due to changes of locations or the topology. The distributed MAC mechanism ensures a high performance and efficient relaying of a MAC frame from a source to a destination in the network, possibly over several multihop relay devices, forming wireless personal area mesh networks (WPAMNs). Although the applications enabled by WPAMNs are very attractive, there are many technical challenges to overcome in order to build well-functioning robust system. The identified challenges include (1) scalability, (2) reliability, and (3) energy-efficiency. The IEEE 802.15.5 has called for proposals for the above technical challenges, and we have proposed a reliable broadcast protocol [8] which has been accepted for standardization of the IEEE 802.15.5. This paper describes the proposed reliable broadcast protocol extensively and its performance merit against the other legacy techniques.

Reliable broadcast is necessary for WPAMNs and has many applications such as service discovery, device paging, routing information propagation, and even data transfer. The IEEE 802.15.5 community requires 100% reliability for those applications under the condition that nodes are reachable via mesh and their battery is operable.

Typically, reliable broadcast techniques have been based upon ACK, NAK, or both. In the ACK-based approach [11], a transmitter needs a positive acknowledgment (ACK) from all receivers to guarantee full reliability (100% reliability). However, since ACKs from the receivers are typically synchronized, it will cause significant contention in the wireless channel. This problem is exacerbated as the number of receivers increases (ACK implosion problem).

On the contrary, negative acknowledgments (NAKs) [10] are well established as an effective loss advertisement mechanism in multi-hop wireless networks, in particular, and group communication, in general, as long as the loss probabilities are not high. However, NAKs cannot handle the unique cases of all frames being lost at a particular node in the network. Since the node is not aware that a data frame is expected, it cannot possibly advertise a NAK to request retransmission. For the short message types like queries consisting of a few frames, the probability that a node does not receive any packet in a message is not negligible.

To tackle the above problems, hybrid scheme [9] (using both ACK and NAK) is proposed. In this scheme, a transmitter elects a broadcast group leader. To cope with the ACK implosion problem, non-leader receivers use the NAK-based scheme while the leader uses ACK-based scheme. However, this scheme still

[a]The authors are with School of Computer Science and Engineering, Chung-Ang University, 221 Heukseok, Dongjak, Seoul 156-756 South Korea; Email: srcho@cau.ac.kr; Tel: +82-2-820-5766; Fax: +82-2-824-1394.

Primary version of this paper were presented and accepted to IEEE 802.15.5 standard [5, 8].

cannot correct the problem where the leader receives a frame successfully while that frame to the other receivers is lost.

As mentioned in the above, NAK or hybrid approaches do not guarantee full reliability. In this paper, we propose a link-layer[a] reliable broadcast scheme referred to as timer-based reliable broadcast (TRB) to overcome MAC/PHY layer impairments due to wireless channel errors and collisions. By the taxonomy provided the above, the proposed TRB scheme is an ACK-based reliable broadcast protocol. The TRB scheme achieves 100% reliability in the mesh devices while conserving unnecessary power consumption in the network. To tackle the ACK implosion problem, the receivers desynchronize ACK transmission times upon reception of a broadcast data, which will reduce collisions of ACKs, and thus save the energy.

Moreover, since separate transmissions of ACK frame and data frame is considered to be redundant, receivers that need to send ACK to the transmitter simply broadcast the received data frame without explicitly sending ACK. The received data frame anyhow needs to be forwarded to neighbors and it also can act as an ACK back to the transmitter. This approach is called implicit ACK [1]. The TRB uses the broadcasting version of the implicit ACK; however, broadcasting implicit ACK inherently has adversary effects. Therefore, we devise a bitmap-based approach to cope with the implicit ACK problem. The bitmap-based approach entails further energy saving in the network. Hence, the contribution of this paper is two-fold: (1) de-synchronization of ACKs and (2) bitmap-based implicit ACK. In the following, we will describe the TRB protocol and its performance.

The paper is organized as follows. In section 2, design guideline and features of the proposed TRB scheme are described, and detailed TRB protocol is given in section 3. In section 4, performance results and their consequences are analyzed, followed by conclusions in section 5.

2. TRB Protocol Design

In this section, we describe the design issues of the proposed TRB and present the components constituting the TRB protocol. The key idea that underpins the design of the TRB is randomization of the ACK transmission times as well as bitmap-based implicit ACK. We assume that frame loss in WPAMNs occurs because of transmission errors due to the poor quality of wireless link and collisions rather than traffic congestion. Hence, we rely upon any flow control mechanism provided in the transport layer protocol which is outside the scope of this paper. It has been often argued that

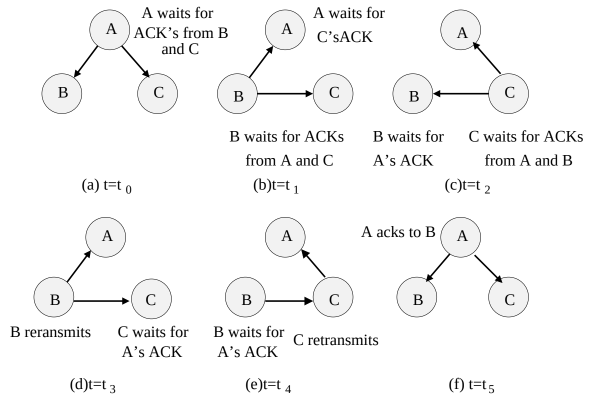

Fig. 1 Adversary effect of implicit ACK for reliable broadcast.

100% reliability or full reliability is not necessary in wireless sensor networks since sensor nodes are usually deployed densely. However, in WPAMNs, considering node dynamics due to mobility and energy deficiency of some nodes, it cannot be always guaranteed that every node is associated with multiple neighbors. Suppose that there is an only child from a parent to reach a large number of descendants. If the child node is not reachable from the parent, all the descendants cannot receive the broadcast frame from the parent. If the broadcast frame is mission-critical, the broadcast fails with disastrous effect. Hence, we argue that the TRB should tackle this type of problem. Of course, we do not consider reliability provision here for the node that is not reachable due to powered-off or out-of-range.

2.1 Bitmap Approach for Implicit ACK

Consider a network of three nodes A, B, and C as in Fig. 1, and assume that we use implicit ACK for broadcast frame and links among the nodes are reliable. We assume further that each node maintains a list of its neighbors (or *neighbor list*[b]). At $t = t_0$, node A broadcasts a data frame and waits for ACKs from B and C. At $t = t_1$, node B forwards the received data frame to A and C where node A considers the data frame as an implicit ACK. Thus, node A waits for C's ACK whereas node B waits for ACKs from A and C. At $t = t_2$, node C forwards the received frame to A and B. At this moment, node A finds its data frame is all acknowledged. Still, node B waits for A's ACK and node C waits for ACKs from A and B. At $t = t_3$, node B's timer expires and retransmits the frame, and thus node C only waits for A's ACK. At $t = t_4$, node C is timed out and retransmits the frame. At $t = t_5$, node B's retransmission (at $t = t_3$) triggers A's retransmission of the frame. Finally, all nodes receive the frame. As observed in this example, the implicit ACK for broadcast application poses unnecessary long chain of broadcast

[a]Since the IEEE 802 standard only deals with MAC/PHY, any multicast protocols above MAC are out of the scope of this paper.

[b]This assumption also holds for the TRB protocol by exchanging hello messages between 1-hop neighbors.

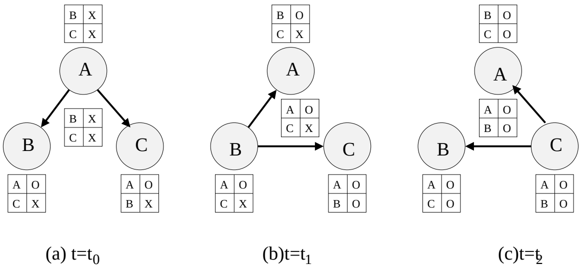

Fig. 2 Applying bitmap to tackle the problems of implicit broadcast ACK

transmissions even if we assume the link is reliable. The situation will become even worse if the link is not reliable.

To cure this adversary effect of the implicit broadcast ACK, we propose a bitmap approach where a bitmap is stored at each node and has information as follows:

- Addresses of neighbor nodes and
- Flag that indicates whether a broadcast frame of a specific neighbor node is received.

Since the acknowledgment from every neighbor node needs to be obtained in order for the transmission to be considered successful, each node has to maintain a bitmap of 1-hop neighbor's acknowledgment. The bitmap has the size (number of bits) equal to the number of one-hop neighbors and is initialized to 0. The bitmap is arranged in a way that the i^{th} bit in the bitmap corresponds to the i^{th} one-hop neighbor entry in the neighbor list. When a data frame or an implicit ACK (a data frame as an ACK) is received, the node sets the corresponding bit to 1. Only when all bitmap fields are set, the node transmits or relays the next broadcast data frame. Hence, the bitmap is maintained per each sequence number.

Fig. 2 explains the benefit of bitmap against the pure implicit broadcast ACK with the same example of Fig. 1. At $t = t_0$, node A initiates a data broadcast and initializes its bitmap (x denotes that the corresponding node is not acknowledged yet or never transmits a frame[a].). Then, node A transmits its broadcast data frame which includes A's bitmap as in Fig. 2(a). Upon receiving the frame, nodes B and C will try to forward the received frame. Suppose that at $t = t_1$, node B acknowledges the received broadcast data frame including B's bitmap. The bitmap of node B indicates that B has already received A's data frame, but B has not received C's frame. Upon receiving B's broadcast frame, node A updates its bitmap indicating that B has been acknowledged. Also, upon receiving B's frame, node C updates its bitmap by setting all bits for nodes A and B since node C has already received data frames from A and B. At $t = t_2$, node C acknowledges its received broadcast data frame including its bitmap. The bitmap of node C indicates that C has already received data frames of A and B. Upon receiving C's broadcast frame, node A updates its bitmap indicating that all nodes have been acknowledged. Also, upon receiving C's frame, node B updates its bitmap by setting all bits for nodes A and C.

[a]As in Fig. 2(a), the bitmap of A indicates that nodes B and C have not transmitted a broadcast frame yet. After receiving A's transmission, the bitmap of B is updated as A transmitted a frame successfully, but C has not transmitted a frame yet. Similarly, the bitmap of C indicates that A transmitted a frame successfully, but B has not transmitted a frame.

As observed in this example, we can reduce the number of transmissions compared with previous implicit ACK mechanism. If we have more neighbor nodes, this benefit significantly increases.

2.2 Randomization of Forwarding Times

If nodes B and C in Fig. 2 acknowledge at the same time (e.g., $t_1 = t_2$), collision will occur. This type of collision is inevitable since A's broadcast frame simultaneously triggers acknowledgments of B and C. This becomes even worse as the number of nodes increases.

To reduce the chance of collisions, the TRB scheme randomizes the transmission time of data frame from the receiver. The TRB exploits a uniformly-distributed random timer in a range of [0, *RBCastRxTimer*] at each node when each node needs to forward its data frame. Since the forwarded data frame can be lost due to error or collision, the transmitter also employs a transmit timer *RBCastTxTimer* where *RBCastRxTime* < *RBCastTxTime*.

3. TRB Framework

3.1 Transmission of Broadcast Data Frame

When sending a broadcast data frame, a node sets the destination address to the logic broadcast address, *BroadcastAddress*, and the source address to the address of the node, in the broadcast frame header. The sequence number for the frame is generated randomly, but incremented by 1 afterwards. Immediately after transmitting the broadcast data frame, the node starts the *RBCastTxTimer*.

A node which broadcasts or relays broadcast frames maintains a Broadcast Transaction Table. This table can be shared by both reliable and non-reliable broadcast transactions. For unreliable broadcast, only the originator's address and the sequence number need to be recorded to avoid duplicating processing of the broadcast data frames. For reliable broadcast, the acknowledgment (either by overhearing the re-broadcast of the same data frame or by receiving an acknowledgment frame for that data frame) from every neighbor node needs to be obtained in order for the transmission to be considered successful. Therefore, besides the originator's address and the sequence number, each node has to maintain a bitmap of 1-hop neighbor's acknowledgment. The bitmap has the size (number of bits) equal to the number of one-hop neighbors in the neighbor list and is initialized to 0. The bitmap is arranged in a way that the i^{th} bit in the bitmap corresponds to the i^{th} one-hop neighbor entry in the neighbor list (when the neighbor list also includes k-hop neighbors, make sure one-hop neighbors are listed first for easy indexing). When a data frame or a data frame as an ACK (with the same sequence number as the current broadcast data frame) is received, the node sets the corresponding bit to 1. Only when all bitmap fields are set, the node transmits or relays the next broadcast data frame.

3.2 Reception and Acknowledgment of the Broadcast Data Frame

Upon receiving the broadcast data frame successfully (the sequence number matches what the receiving node is expecting), the receiving node first needs to record the data frame in the broadcast transaction table, if it is the first received data frame. The payload will then be forwarded to the node's next higher layer for processing. If reliable delivery is required, the receiving node has to also set the corresponding bit in the bitmap to indicate the successful reception of this specific data frame from the transmitter. For unreliable broadcast, duplicate data frames will be silently discarded; for reliable broadcast, duplicate data frames will set their corresponding bits in the bitmap before being discarded.

The receiving node will then determine whether it should relay the broadcast frame. Nodes that find at least one of the bitmap fields remains 0 forward the frame to their one-hop neighbors. The frame forwarded back to the originator or transmitter is implicitly interpreted as an acknowledgment (ACK), which does not entail usage of an extra ACK frame. The time to forward the received frame is randomized using random timer to reduce the possibility of collision. The timer is generated in range of [0, *RBCastRxTimer*]. After forwarding the data frame, the node sets *RBCastTxTimer*.

Upon receiving the broadcast data frame successfully (with the same sequence number that the receiving node is expecting), the receiving node that finds all of the bitmap fields set 1 transmits a frame without payloads (but with the same sequence number) at random time. Randomization of transmission of a frame reduces the possibility of collision as the above.

3.3 Transmission and Reception of the Broadcast Data Frame

If any one of the bitmap fields remains 0 before expiration of *RBCastTxTimer*, the node rebroadcasts the data frame and waits for *RBCastTxTimer*. If the node tries MaxRBCastTrials times for rebroadcasting the data frame, the node reports to the next higher layer so that the next higher layer requests deleting the corresponding neighbor from the neighbor list.

4. Performance Evaluation

To evaluate the reliability performance of the TRB protocol, we developed a simulator based on IEEE 802.15.4 *ns-2* package [4]. On top of the package, we developed the IEEE 802.15.5 address assignment, association, and routing modules. The TRB algorithm is added into both package and modules. The simulator generates a random topology as shown in Fig. 3. We assume that the nodes have a fixed radio range of 10 meters and are placed in a square area randomly. Fig. 3 shows a typical network routing tree.

This tree is formed based on the proximity metric of each node, and each node keeps a neighbor list based on IEEE 802.15.5 address assignment algorithm. According to the address assignment algorithm, the root of the tree (sink) is randomly selected in the simulator. Beginning from the sink, nodes gradually join the network and a tree is formed. However, this tree is not a logic tree yet, since no node has been assigned an address. After a branch reaches its bottom, a bottom-up procedure is used to calculate the number of nodes along each branch. After the sink receives the information from all the branches, it begins to assign addresses. During address assigning stage, a top-down procedure is used. First, the sink assigns a block of consecutive addresses to each branch below it, taking into account the number of children and the number of requested addresses. This procedure continues until the bottom of the tree. After address assigning, a logic tree is formed and each node has populated a neighbor list for tracking branches below it.

When we vary the number of nodes, we vary the size of the area over which they are distributed so as to keep the density of nodes constant. For this simulation, we use the density as 0.01. For instance, 100 nodes are generated in area of $100m \times 100m$. All the simulation results are shown after averaging the metrics over 20 randomly generated topologies and calculating 95 percent confidence intervals. For wireless channel errors (not because of collisions), we choose a fixed frame error rate of 10 percent. We use the maximum capacity of 250 kbps (IEEE 802.15.4) and the frames are generated at the rate of 2 frames per second.

In the performance comparison, we consider four

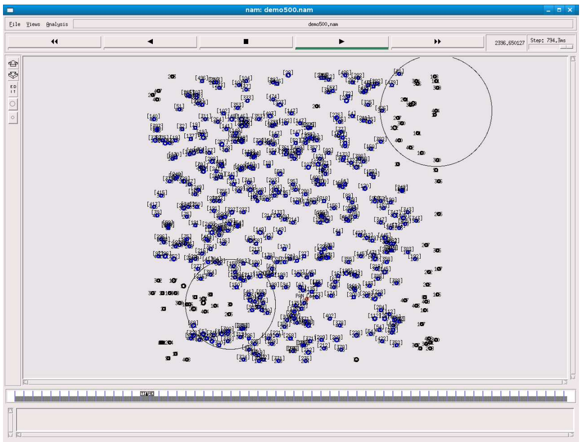

Fig. 3 A snapshot of ns-2 simulator of the TRB protocol where 500 nodes are randomly generated.

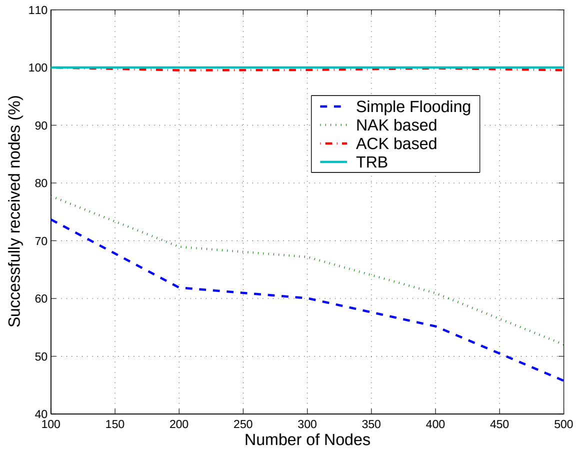

Fig. 4 The percent of successfully received nodes vs. number of nodes (frame error rate = 10%)

protocols: (1) simple flooding, (2) NAK-based scheme, (3) ACK-based technique, and (4) the TRB protocol. The simple flooding [6] starts with a transmitter broadcasting a frame to all neighbors. Each of those neighbors in turn forwards the frame to all its neighbors exactly one time and this continues until all reachable network nodes have received the frame. IETF proposed the use of this flooding for broadcasting and multicasting in ad hoc networks which are characterized by low node densities and/or high mobility.

The NAK-based scheme [10] is a scheme where receivers only respond with negative acknowledgments. Since there are unnecessary additional features in schemes of [10], we only extract the NAK-based feedback mechanism from the scheme [10] to perform the comparison with our TRB algorithm. Both schemes are expected to be unreliable since the simple flooding does not guarantee the reliability and the NAK-based scheme does not handle loss of NAKs and all frames being lost as described in section II. The ACK-based scheme [11] is a more reliable technique where the transmitter waits for all ACKs from the receivers until it transmits the next frame. Because of this, ACK-based scheme is expected to consume more energy.

In our simulation, the performance is measured according to the following metrics:

- Successfully received nodes(%): to observe the reliability of different schemes, we collected the nodes that received the frame successfully per each frame. Then, the percent of successfully received nodes is calculated as the ratio of the number of successfully received nodes to the entire number of nodes
- Energy consumption: to measure how the protocol is energy-efficient, we take into account the average number of transmissions of a frame as an energy budget.

4.1 Reliability

In this section, we demonstrate the reliability performance of four algorithms for data broadcasting: Simple Flooding, NAK-based Scheme, ACK-based Scheme, and our TRB protocol. Fig. 4 shows the percent of successfully received nodes in the network versus the number of nodes. As can be seen, the reliability of NAK-based scheme and simple flooding decreases with the number of nodes. As more nodes are populated, there are more contentions for wireless channel, and NAKs and frames are more likely to be lost. Therefore, reliability of both schemes is very low with large number of nodes. At 500 nodes, both schemes could provide about 50% reliability of the network as expected, which is highly undesirable for delivering critical messages in the network.

Compared with the simple flooding and NAK-based schemes, the ACK-based and TRB algorithms provide much higher reliability. The TRB protocol achieves 100% reliability of the network while the reliability of ACK-based scheme is close to 100%.

4.2 Energy Efficiency

To observe how energy is dissipated for each different algorithm, we measure the average number of transmissions of a frame (or energy consumption) as in Fig. 5. The energy consumption of the ACK-based technique linearly increases and is not bounded while the other three schemes dissipate the energy rather constantly. The simple flooding achieves almost 1 transmission per frame since each node forwards the frame to all its neighbors exactly one time. The NAK-based technique consumes 1.5 frames while the TRB uses 3.5 frames per frame. In the following we will investigate more performance results on frame error rates and guideline to choose which protocols.

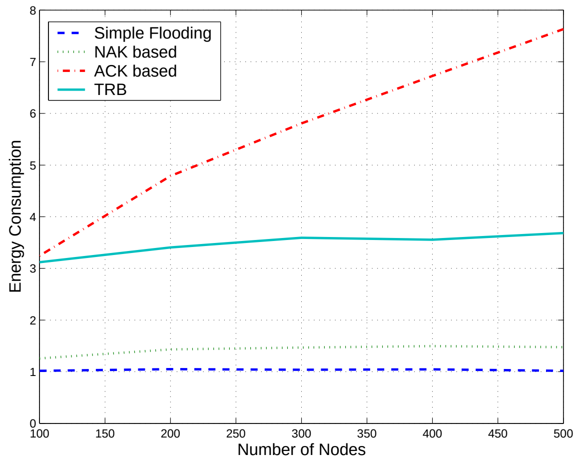

Fig. 5 The energy consumption vs. number of nodes (frame error rate = 10%)

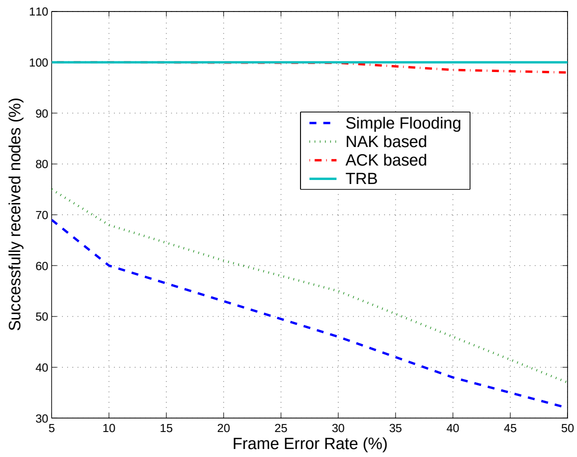

Fig. 6 percent of successfully received nodes vs. frame error rate (for 300 nodes)

4.3 Impact of Channel Error Rate

To investigate the impact of channel error rates on the performance of four schemes, we vary the frame error rates from 5% to 50%. Fig. 6 shows the percent of successfully received nodes in the network versus the frame error rates. We observe that the reliability of all three schemes except the proposed TRB protocol drops as the frame error rate increases but with different slope. Noticeably, the reliability of NAK-based and simple flooding is about mid 30's % at the frame error rate of 50%. On the contrary, the proposed TRB achieves 100% of reliability impendent with the frame error rate.

Fig. 7 compares the energy consumption for each scheme with respect to the frame error rate. Similar to result of Fig. 5, the energy consumption of the ACK-based technique is not bounded while the other three schemes dissipate the energy rather manageably.

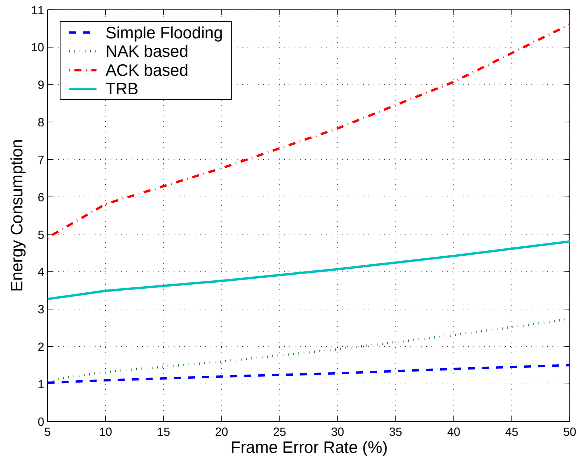

Fig. 7 The energy consumption vs. frame error rate (for 300 nodes)

4.4 Overall Comparison

In the preceding sections, we compare reliability and energy efficiency performance for 4 different protocols. We can summarize that the ACK-based and TRB protocols have more reliable than the NAK-based and simple flooding schemes for broadcasting while the NAK-based and simple flooding achieve better energy efficiency than the ACK-based and TRB protocols.

To investigate how many transmissions are required to achieve the same level of reliability of the TRB protocol, we perform the following scenario. We generate a single frame and observe whether all nodes successfully receive the frame. If any of the nodes fail to receive the frame successfully, we continue to generate the single frame until all the nodes successfully receive the frame. Fig. 8 and Fig. 9 shows the number of transmissions of a single frame with respect to the number of nodes and frame error rate, respectively.

As we observe from the figures, the proposed TRB requires the minimum number of transmissions and hence we conclude that the TRB achieves the maximum energy efficiency. More importantly, the TRB achieves decent degree of scalability compared with the other three schemes.

4.5 Usage Guideline

As discussed in the preceding sections, the TRB protocol can achieve 100% reliability significantly better than other schemes. However, it requires slightly increased energy consumption than NAK-based and simple flooding techniques. On the other hand, simple flooding achieves the minimum energy consumption among four schemes with expense of low reliability.

Therefore, we propose a single integrated broadcast protocol that can incorporate the TRB and simple

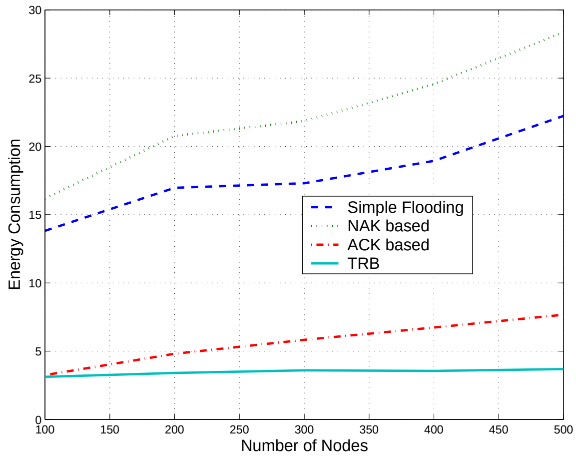

Fig. 8 The energy consumption vs. number of nodes (frame error rate = 10%)

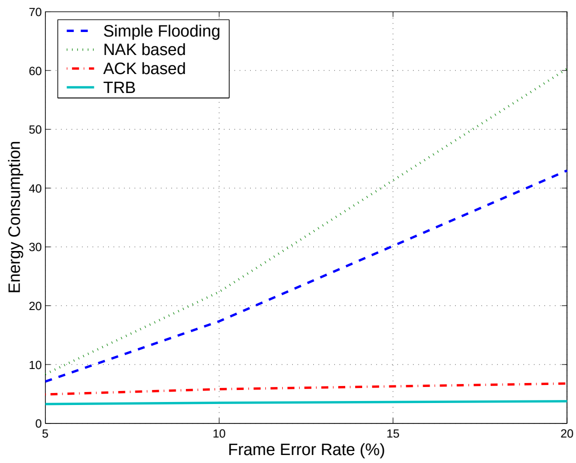

Fig. 9 The energy consumption vs. frame error rate (for 300 nodes)

flooding algorithms to cope with different level of reliability. In other words, if the message requires 100% reliability, the integrated broadcast protocol exploits the TRB algorithm while it uses the simple flooding for unreliable broadcasting service.

5. Conclusions

In this paper, we propose a link-layer reliable broadcast protocol referred to as timer-based reliable broadcast (TRB) for WPAMNs. The proposed TRB scheme exploits (1) bitmap based implicit ACK to effectively reduce the unnecessary error control messages and (2) randomized timer for ACK transmission to substantially reduce the possibility of contentions.

Performance evaluation shows that the TRB scheme achieves 100% reliability significantly better than other schemes with expense of slightly increased energy consumption. Integrated with simple flooding, the TRB scheme can be a useful tool for broadcasting services in WPAMNs.

References

[1] S. Chen and N. Yang, "Congestion Avoidance Based on Lightweight Buffer Management in Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems,* vol. 17, no. 9, pp. 934–946, September 2006.

[2] S. Cho, "Proactive Data Filtering Algorithm for Aggregation in Wireless Sensor Network," *IEICE Transactions on Communications,* vol. E91-B, no. 3, March 2008.

[3] S. Cho and I. F. Akyildiz, "A Poker-Game-Based Feedback Suppression Algorithm for Satellite Reliable Multicast," *in Proc. Of IEEE GLOBECOM,* November 2002.

[4] IEEE 802.15.4 Ns2 Package, "http://ees2cy.engr.ccny.cuny.edu/zheng/pub/".

[5] *IEEE 802.15.5: Mesh Topology Capability in Wireless Personal Area Networks (WPANs),* IEEE Std. 802.15.5, 2009.

[6] J. Jetcheva, Y. Hu, D. Maltz, and D. Johnson, "A Simple Flooding Protocol for Multicast and Broadcast in Mobile Ad Hoc Networks," *Internet Draft: draft-ietf-manet-simple-mbcast-01.txt,* July 2001.

[7] M. Lee, L. Zhang, J. Zheng, G. Ahn, C. Zhu, T. Park, S. Cho, C. Shin, and J. Ryu, "IEEE 802.15.5 WPAN Mesh Standard-Low Rate Part: Meshing the Wireless Sensor Networks," *IEEE Journal on Selected Areas in Communications,* vol. 28, no. 7, September 2010.

[8] I. Lee, S. Cho, S. Choi, L. Park. B. Kim, "Timer-based Reliable Broadcasting Algorithm for WPAN Mesh Networks," *IEEE P802.15.5 Draft*, March 2007.

[9] W. Lim, D. Kim, and Y. Suh, "Design of Efficient Multicast Protocol for IEEE 802.11n WLANs and Cross-Layer Optimization for Scalable Video Streaming, *IEEE TMC,* vol. 27, No. 8, October 2009.

[10] M. Huang, G. Feng, and Y. Zhang, Cooperative Loss Recovery for "Reliable Multicast in Ad Hoc Networks, *Internatioanl Journal of Communications, Network, and System Science,* pp. 72–78, 2010.

[11] V. Srinivas and L. Ruan, "An Efficient Reliable Multicast Protocol for 802.11-based Wireless LANs, *in Proc. of IEEE WOWMOM,* June 2009.

[12] B. Williams and T. Camp. "Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks," *in Proc. Of ACM MOBIHOC,* pp. 194–205, June 2002.