PMC: An Energy Efficient Event Transport Service For Wireless Sensor Network

Xiuchao Wu, A. L. Ananda and Mun Choon Chan
School of Computing, National University of Singapore
#3, Science Drive 2, Singapore 117543
Email:{wuxiucha, ananda, chanmc}@comp.nus.edu.sg

Abstract— **A wireless sensor network is normally a multi-hop wireless ad hoc network whose purpose is to transmit events sensed by sensors to the sink. Since sensor nodes are normally powered by non-rechargeable and non-replaceable battery, an energy efficient event transport service is crucial for extending the life time of a wireless sensor network.**

Due to the possibility of crossing multiple lossy links between the source and the sink, many events may be lost before they arrive at the sink. In addition, the overhead of low layer protocols is very large due to the short event data. These two factors can result in significant event loss and waste of energy. Many mechanisms have been proposed on other networks to improve reliability and reduce overhead. Due to scarce memory and poor computing ability of sensor nodes in a sensor network, these mechanisms may not be feasible to implement and they may not work efficiently with respect to energy.

In this paper, PMC (Pipelined Multiple Copies) is proposed for transmitting events in a wireless sensor network, with the goals of achieving low event loss rate and high energy efficiency. In PMC, multiple copies of an event are pipelined and transmitted in several consecutive frames according to Silent CSMA, a variant of CSMA/CA. Multiple copies of an event reduce end-to-end event loss rate and Silent CSMA reduces packet energy consumption significantly. Mathematical analysis shows that PMC achieves the best trade-off between packet energy consumption and event loss rate compare to the Silent CSMA and CSMA/CA/ARQ schemes. It also shows that PMC is really an energy efficient event transport service for resource-limited wireless sensor networks.

I. Introduction

There has been tremendous amount of work in the wireless sensor network (WSN) area in the past few years. A number of WSNs have been designed for various kinds of applications, such as Habit Monitoring [1], Interactive Surrounding [2], and Disaster Prevention & Relief [3], etc. In order to satisfy different application requirements, sensor nodes that have different capacity of energy, memory and computing abilities are adopted. These sensor nodes are deployed sparsely or densely and generate different traffic load with different pattern. These diversities in WSNs make it almost meaningless to design protocols without any assumption about the application domain.

In this paper, we focus on WSNs that expect long life, composed of simple sensor nodes with scarce memory and poor computing ability, and powered by non-rechargeable and non-replaceable battery. In addition, we focus on applications where one or more sensor nodes serving as data sources *continuously* send small size data packets to a data sink node. Within this kind of resource limited WSNs, energy efficient protocols are very important [4].

In addition, a WSN is typically a multi-hop wireless ad hoc network. Within such an infrastructure-less network, it is very hard to do synchronization required by TDMA. FDMA is also not suitable for low cost sensor nodes with very simple hardware. Thus, we assume a simple transceiver is installed on a sensor node and CSMA/CA is used at the MAC layer.

The characteristics of WSN bring many challenges to transmit events in an energy-efficient way. Firstly, a WSN is normally deployed in harsh environment. That means bad wireless channel and high packet loss rate. The possible collision due to broadcast wireless medium and the accumulated packet loss over multi-hop wireless links make this situation worse. Many events might have been lost even before they arrive at the sink, and thus the energy consumed in transmitting them is wasted. Secondly, a packet is generated and transmitted for each event within a WSN. The short event data makes the overhead of preamble, headers, and control message (ACK) relatively very large. Too much energy is consumed by these overheads.

In this paper, we focus on energy efficient event transport service that considers how packets should be sent in order to reduce end-to-end event loss rate and how transmission should be managed between neighboring nodes so as to increase energy efficiency. These functions typically belong to the transport and MAC layers respectively in classical networks. In particular, we propose PMC (Pipelined Multiple Copies), an energy-efficient event transport service. In PMC, multiple copies of an event are pipelined and transmitted in several consecutive frames according to Silent CSMA, a variant of CSMA/CA without ACK. Multiple copies of an event reduce the end-to-end event loss rate and Silent CSMA reduces energy consumption significantly. Mathematical analysis show that PMC is an energy efficient and reliable event transport service.

This paper is organized as follow. In section 2, we present and analyze several related work. In section 3, we present PMC (Pipelined Multiple Copies) over Silent CSMA (a variant of CSMA/CA), an energy efficient event transport service proposed for wireless sensor networks. In section 4, we model and compare PMC, Silent CSMA, and CSMA/CA/ARQ. Finally, we conclude with several future work in section 5.

1-4244-0355-3/06/$20.00 (c) 2006 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

II. Related Work

Many transport services have been proposed to exchange data among nodes within a network. Due to the characteristics of WSNs, these may not work or can't work energy-efficiently. In this section, we analyze these existing mechanisms within WSNs. Since high end-to-end event loss rate (the rate that an event is lost before it arrives at the sink) and large overheads are the main problems, we divided these mechanisms into two categories — mechanisms proposed for reducing event loss rate and mechanisms proposed for reducing overheads.

A. Mechanisms for reducing event loss rate

Since energy is wasted because of the loss of events, energy efficiency should be improved if event loss rate could be reduced. FEC (Forward Error Correction) and retransmission are normally used over lossy links to reduce packet loss rate. However these mechanisms improve event reliability at the expense of spending more resources.

Within FEC, redundant bits are sent in order to correct corrupted bits. But redundant bits waste resources when wireless link is good. Even worse, FEC algorithms are normally CPU intensive. They consumes energy and, in some cases, exceed the limited CPU processing capability of sensor node. A variation of FEC, which sends multiple copies of the same event within the same packet, does not cause much CPU overhead. But, it can't perform well on wireless links with bursty error. Under such an environment, one corrupted copy means high probability of other copies are also corrupted.

Retransmission or ARQ (Automatic Retransmission reQuest) can reduce event loss rate when link is bad and there is no redundancy when there is no loss. However, ARQ suffers from large delay and jitter. It also needs some methods to detect event loss. Below is our analysis for several variants that use retransmission to reduce event loss rate.

- End-to-End Loss Detection and Retransmission: End-to-End Retransmission, such as TCP [5], is a good mechanism to improve data reliability in wired networks. But it is not impractical for WSNs due to its high cost [6]. Firstly, end-to-end acknowledgements can't be piggybacked due to the uni-direction data flow, and they consume quite a lot of energy. Secondly, events and acknowledgements need to pass through multiple lossy wireless links. The event loss rate on the path will be so high that end-to-end retransmission can't reduce event loss rate much and more energy is wasted due to retransmission.
- Hop-by-Hop Loss Detection and Retransmission: Hop-by-hop loss detection and retransmission is a good solution for reducing event loss rate within WSNs [6]. In this mechanism, instead of retransmitting from the source, the events lost over one hop will be recovered locally. It is much more energy-efficient than end-to-end retransmission. There are several loss detection mechanisms.

  The first one is to let intermediate routers use sliding window mechanism and maintain a queue for each data flow. The router detects event loss according to sequence number gap. By this way, selective acknowledgement can be implemented to reduce the number of acknowledgement packets. But per-flow queue requires too much memory for sensor nodes with limited memory. In addition, the routers needs to decide when to send the acknowledgement, hence timers may also be required. Such a complex algorithm is not feasible to be implemented on resource-limited sensor nodes.

  The second one is CSMA/CA/ARQ (CSMA/CA with local retransmission), such as the MAC layer used by IEEE 802.11 [7]. The receiver feeds back a positive ACK for each successively received data frame and forwards the data. If there is no corresponding ACK received by the sender within a reasonable period, it will carry out exponential back-off procedure and retransmit the data frame. That means intermediate node need not maintain per-flow information and this mechanism is scalable on sensor nodes with limited memory. Except for event loss rate reduction due to local retransmission, the back-off procedure of CSMA/CA can reduce event loss rate further by avoiding collision. RTS/CTS is also proposed in IEEE 802.11 to reduce collision further. But it should not be used in WSNs due to the overhead of RTS and CTS frames.

Except FEC and retransmission, congestion control schemes [8][9] are also proposed in WSNs for reducing event loss due to congestion.

B. Mechanisms for reducing overhead

Within WSNs, the overhead caused by preamble, headers, and possible control message is very large due to the short event data. When large amount of energy is used to transmit the overhead, energy efficiency is low. Header compression algorithms, such as [10] used in the Internet, can't be used because the preamble and headers of a frame are used for synchronization-and-rate negotiation and can't be compressed. Hence, the only way for reducing overhead of preamble and headers is to increase the length of data. Data aggregation based on hierarchical topology [11][12] is such a method.

Data aggregation may be carried out by the source or intermediate routers. The source node can encapsulate events sensed by different sensors or several events sensed by the same sensor into one data frame. When data aggregation is carried out by an intermediate router, it collects events from its descendants and forwards them in one data frame.

Data aggregation increases the data length of a data frame and thus reduces the overhead of preamble-and-headers. But the frame loss rate also increases. This is not a big problem since local retransmission can be used to address it. Data aggregation with CSMA/CA/ARQ seems to be a perfect solution. However, data aggregation may not work in many cases. For example, when load is low, the nodes may not have enough data to be aggregated. If they persist in waiting for enough events, the delay will increase too much.

III. Pipelined Multiple Copies

According to previous analysis, while CSMA/CA/ARQ plus data aggregation can reduce event loss rate and overhead simultaneously but data aggregation can't work in many cases. In addition, while CSMA/CA/ARQ can effectively reduce event loss rate, the overhead of its ACK frame is quite large and it consumes energy even when wireless channel is very good. Finally, ACK frame may also be corrupted and causes unnecessary retransmission. Based on the above observations and that sensor network applications normally tolerate some amount of event loss rate [8], we propose PMC (Pipelined Multiple Copies) based on Silent CSMA (CSMA/CA without ACK), an energy-efficient event transport service, for WSNs.

Before describing how PMC works, it is important to note that PMC is designed for applications where one or more sensor nodes serving as data sources *continuously* send small size data packets to a data sink node.

A. Silent CSMA

Since ACK frame consumes energy and unnecessary retransmission may be caused by corrupted ACK, Silent CSMA, which keeps back-off procedure of CSMA/CA but without ACK frame, is proposed.

Within Silent CSMA, when a node transmits a data frame, it assumes that the receiver is silent and will not wait for acknowledgement. If the event is received correctly, the receiver will forward it, otherwise the received and corrupted event is ignored. In any case, the receiver will not send out any acknowledgement. Hence, *The sender does not carry out loss detection and retransmission*. It resembles UDP[13] of the Internet.

Without ACK, in Silent CSMA, the sender deduces the channel load without additional cost by monitoring how many transmissions occur when it is waiting for its time slot that is randomly selected within its contention window. Hence, it can adjust its contention window according to channel load and achieve collision avoidance. In addition, Silent CSMA could reduce collision probability further because collision area of Silent CSMA (without ACK) is smaller than CSMA/CA/ARQ (with ACK).

B. Pipelined Multiple Copies

Silent CSMA eliminates ACK overhead, reduces collision probability, and avoids unnecessary retransmission due to corrupted ACKs. However, without ARQ, packet loss can be high, and the wasted energy due to the events that are lost before they reach the sink, still persist. And the overhead of preamble and headers is also there. In this subsection, we shall present our PMC (Pipelined Multiple Copies) that works on Silent CSMA and analyze how it solves these problems.

The key additional feature of PMC is that when a source generates a packet, instead of only the current event, copies of several older events sensed by it earlier are also encapsulated into this packet. The number of events within one packet is an important parameter of PMC: m. Figure 1 demonstrates PMC mechanism when $m = 3$. When a packet is received successfully by the sink, these old events can still be valuable information for many applications. For example, in a target tracking application, the information within these consecutive events can be used to monitor and predict the movement of the target. Thus, the overhead of preamble and headers is effectively reduced though data redundancy is used to low event loss rate. As a result, PMC achieves a lower event loss rate compare to Silent CSMA with additional redundancy ($m > 1$).

The event loss rate is low even when burst error occurs over wireless channels if the duration over which m packets are sent is longer than the burst loss duration.

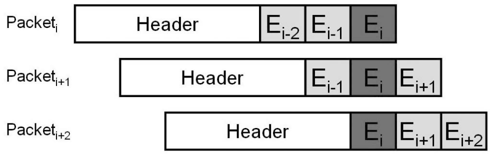

Fig. 1. Pipelined Multiple Copies

According to above analysis, PMC with Silent CSMA can reduce event loss rate and the overhead, and thus possibly can act as an energy efficient event transport service. In addition, PMC is also a highly scalable mechanism. First, PMC has a very low memory requirement, and as such an intermediate router need not maintain per-flow information and the source only needs a small buffer to hold several sensed events. Second, PMC has very low CPU requirement as the event processing procedures at the source and intermediate routers are very simple. This means low CPU overhead and less energy consumed by CPU of a sensor node.

Since only data transmission part is changed in PMC over Silent CSMA, the same idea can also be used in other CSMA based MAC protocols, such as S-MAC [14].

IV. Mathematical Analysis

In this section, we establish simple models for Silent CSMA itself, CSMA/CA/ARQ, and PMC over Silent CSMA. We assume that collision probability, contention window, and bit error rate are the same at all hops. Congestion loss is not modeled and uniform bit error model is adopted in this analysis. In addition, note that in all three protocols, there is no end-to-end retransmission as this is considered impractical in sensor networks.

In this analysis, we are mainly interested in two performance measures, P_{event} (the end-to-end event loss rate) and ξ_{event} (the energy spent for successively transmitting one event to the sink). Instead of modeling ξ_{event} directly, we first model ξ_{pkt} (packet energy consumption), the expected transmission time spent for sending a single packet independent of whether it is successfully transmitted end-to-end or not. The time spent in back-off is also included because CPU consumes energy during back-off procedure. Since one packet is generated for an event in all protocols, with P_{event} and ξ_{pkt}, we can approximate their ξ_{event} easily according to Equation 1. A smaller ξ_{event} indicates better event energy efficiency.

$$\xi_{event} = \frac{\xi_{pkt}}{1 - P_{event}} \tag{1}$$

A. Silent CSMA

Silent CSMA is the simplest mechanism. We first calculate T_{pkt_hop} (the time spent by a data frame at one hop) and P_{pkt_hop} (the probability that data frame is corrupted at one hop). With P_{pkt_hop} and hop number, P_{event} is very easily calculated. With $P_{pkt\_loss\_i}$ (the probability that a packet is lost at ith hop) and $C_{pkt\_loss\_i}$ (the time spent before it is lost), we can calculate ξ_{pkt} easily.

$$\begin{aligned}
T_{frm} &= T_h + T_e \\
T_{pkt\_hop} &= T_{ca} + T_{frm} \\
P_{frm} &= 1 - (1 - P_h) * (1 - P_e) \\
P_{pkt\_hop} &= 1 - (1 - P_c) * (1 - P_{frm}) \\
P_{pkt\_loss\_i} &= (1 - P_{pkt\_hop})^{i-1} * P_{pkt\_hop} \\
P_{pkt} &= 1 - (1 - P_{pkt\_hop})^n \\
P_{event} &= P_{pkt}
\end{aligned} \tag{2}$$

$$\begin{aligned}
C_{pkt\_loss\_i} &= i * T_{pkt\_hop} \\
\xi_{pkt} &= \sum_{i=1}^{n} (P_{pkt\_loss\_i} * C_{pkt\_loss\_i}) + \\
&\quad n * T_{pkt\_hop} * (1 - P_{pkt})
\end{aligned} \tag{3}$$

Here,

- n: the number of hops from a source to the sink
- T_h: time needed to transmit preamble and headers.
- T_e: time needed to transmit the data of an event.
- T_{ca}: time spent by Silent CSMA back-off procedure.
- T_{frm}: time needed to transmit a data frame.
- P_c: collision probability at one hop.
- P_h: preamble and headers' corruption rate at one hop.
- P_e: event corruption rate at one hop.
- P_{frm}: data frame loss rate at one hop.
- P_{pkt}: packet loss rate over the whole path.

B. CSMA/CA/ARQ

CSMA/CA/ARQ is a more complex mechanism and its parameter, r, gives the limitation of retransmission time. We first analyze its behaviors at one hop. $\overline{T_{pkt\_hop}}$ (the average time needed to transmit a packet over one hop) is calculated by considering $P_{succ\_i\_try}$ (the probability that a packet is transmitted successfully at the i^{th} transmission try) and $T_{succ\_i\_try}$(the time spent at one hop when a packet is transmitted successfully at the i_{th} transmission try).

$$\begin{aligned}
T_{frm} &= T_h + T_e\ ; \quad T_{ack} = T_h \\
T_{pkt\_try} &= T_{ca} + T_{frm} + T_{SIFS} + T_{ack} \\
P_{frm} &= 1 - (1 - P_h) * (1 - P_e) \\
P_{pkt\_try} &= 1 - (1 - P_c) * (1 - P_{frm}) * \\
&\quad (1 - P_{ack}),\ while\ P_{ack} = P_h \\
P_{pkt\_hop} &= (P_{pkt\_try})^r \\
P_{pkt} &= 1 - (1 - P_{pkt\_hop})^n \\
P_{succ\_i\_try} &= (1 - P_{pkt\_try}) * (P_{pkt\_try})^{i-1} \\
C_{pkt\_try\_f} &= \frac{(P_{frm} + P_c) * (T_{frm} + T_{ca})}{P_{pkt\_try}} + \\
&\quad \frac{P_{ack} * T_{pkt\_try}}{P_{pkt\_try}} \\
T_{succ\_i\_try} &= T_{pkt\_try} + (i - 1) * C_{pkt\_try\_f} \\
\overline{T_{pkt\_hop}} &= \frac{\sum_{i=1}^{r} P_{succ\_i\_try} * T_{succ\_i\_try}}{1 - P_{pkt\_hop}}
\end{aligned}$$

After that, we can analyze the issues of multiple hop transmission. P_{event} and ξ_{pkt} are deduced with the same methods used by Silent CSMA.

$$\begin{aligned}
P_{pkt\_loss\_i} &= P_{pkt\_hop} * (1 - P_{pkt\_hop})^{i-1} \\
C_{pkt\_loss\_i} &= (i - 1) * \overline{T_{pkt\_hop}} + r * C_{pkt\_try\_f} \\
C_{pkt\_succ} &= n * \overline{T_{pkt\_hop}} \\
\xi_{pkt} &= \sum_{i=1}^{n} P_{pkt\_loss\_i} * Cost_{pkt\_loss\_i} + \\
&\quad (1 - P_{pkt}) * C_{pkt\_succ}
\end{aligned} \tag{4}$$

$$P_{event} = P_{pkt} \tag{5}$$

In the above equations, except the symbols used by Silent CSMA. The following symbols are also introduced.

- T_{ack}: time needed to transmit an ACK frame.
- T_{SIFS}: Short Inter-Frame Space.
- P_{ack}: ACK frame loss rate at one hop.
- P_{pkt_try}: probability that a packet is lost during a transmission try.
- $C_{pkt\_try\_f}$: time wasted when a packet transmission try failed.
- C_{pkt_succ}: time spent when a packet is received successfully by the sink.

C. PMC

As for PMC, it is quite close to Silent CSMA except that its packet is longer since multiple events are encapsulated and an event is transmitted within multiple packets. The following equations give ξ_{pkt} and P_{event} of PMC.

$$\begin{aligned}
T_{frm} &= T_h + m * T_e \\
T_{pkt\_hop} &= T_{ca} + T_{frm} \\
P_{frm} &= 1 - (1 - P_h) * (1 - P_e)^m \\
P_{pkt\_hop} &= 1 - (1 - P_c) * (1 - P_{frm}) \\
P_{pkt\_loss\_i} &= (1 - P_{pkt\_hop})^{i-1} * P_{pkt\_hop}
\end{aligned}$$

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

$$\begin{aligned}
C_{pkt\_loss\_i} &= i * T_{pkt\_hop} \\
P_{pkt} &= 1 - (1 - P_{pkt\_hop})^n \\
\xi_{pkt} &= \sum_{i=1}^{n} (P_{pkt\_loss\_i} * C_{pkt\_loss\_i}) + \\
& \quad n * T_{pkt\_hop} * (1 - P_{pkt}) \qquad (6) \\
P_{event} &= (P_{pkt})^m \qquad (7)
\end{aligned}$$

D. Numerical Results

According to above models, we compare these mechanisms with numerical results under a typical scenario. Table I shows the symbol values used in this comparison. These values are selected based on IEEE 802.11 with 2Mbps, a small collision probability, and short event (5 bytes).

In this scenario, we investigate packet energy consumption, event loss rate, and event energy efficiency of Silent CSMA, CSMA/CA/ARQ (r=3), CSMA/CA/ARQ (r=7), PMC (m=3), and PMC (m=7) for sensor nodes with different hop numbers under different frame loss rates. We investigate hop number from 1 to 10 and set BER (bit error rate) so that the FLR (frame loss rate) of a data frame that encapsulates one event to be 5% and 15%.

TABLE I

SYMBOL VALUES

Symbol	Value	Comments
T_h	200μs	Time spent by preamble and headers
T_{ca}	130μs	T_{DIFS} and time spent by back-off procedure
T_e	20μs	Time for event data transmission
P_c	0.0001	Collision Probability
T_{SIFS}	10μs	Short Inter-Frame Space

Figure 2 shows that under most of situations, a packet in Silent CSMA consumes the smallest amount of energy, a packet in CSMA/CS/ARQ consumes the largest amount of energy, and PMC falls in between Silent CSMA and CSMA/CA/ARQ. As for end-to-end event loss rate, Figure 3 shows that Silent CSMA suffers the highest event loss rate and CSMA/CA/ARQ suffers the lowest event loss rate. Again, PMC falls in between Silent CSMA and CSMA/CA/ARQ. These facts imply that PMC achieves good trade-off between packet energy consumption and event loss rate. Figure 4 does show that PMC is really a highly energy efficient event transport service. We also find that when hop number and frame loss rate increase, PMC should increase m. Perhaps an adaptive m is valuable for PMC.

In addition, within Figure 3, we find that PMC can't reduce event loss rate much when hop number is large and wireless channels are very bad. We may keep PMC working well by increasing the number of sinks, i.e. reducing hop number. One better solution is to select PMC or CSMA/CA/ARQ according to application's expectation: event energy efficiency or event reliability.

Except the above results, we had investigated these protocols under 1%, 10%, and 20% FLR and the results are similar. We had also investigated these protocols within a ZigBee scenario (data rate: 250Kbps, event: 5bytes, headers: 25bytes) and the results are similar too.

V. CONCLUSION AND FUTURE WORKS

In this paper, PMC over Silent CSMA is proposed for transmitting events, with event energy efficiency as a goal, within wireless sensor networks by solving energy waste due to high event loss rate and large overhead. Protocol analysis and mathematical analysis show that PMC is a simple and energy efficient event transport service that can be implemented on resource-limited sensor nodes, especially when traffic load is low and data aggregation can't be used.

In this paper, ξ_{event} is only a coarse estimation of event energy assumption because it does not consider the difference among energy consumption due to sensing channel, transmitting, and receiving events. Some accurate metric should be used in the future work. More importantly, only Silent CSMA, CSMA/CA/ARQ, and PMC are modeled and compared under a specific scenario. Simulation should be worthwhile to compare PMC, Silent CSMA, CSMA/CA/ARQ, and other event transport services under different topology, different routing protocols, different wireless environments, and different traffic load & pattern with the accurate metrics of event energy efficiency, event loss rate, delay, and jitter, etc.

REFERENCES

[1] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *ACM WSNA*, 2002.

[2] M. Srivastava, R. Muntz, and M. Potkonjak, "Smart kindergarten: sensor-based wireless networks for smart developmental problem-solving environments," in *ACM MOBICOM*, 2001.

[3] S. Ray, D. Starobinski, A. Trachtenberg, and R. Ungrangsi, "Robust location detection in emergency sensor networks," in *INFOCOM*, 2003.

[4] A. Ephremides, "Energy concerns in wireless networks," *IEEE Mag. Wireless Commun.*, vol. 9, 2002.

[5] J. Postel, "Transmission control protocol - darpa internet program protocol specification," RFC 793, DARPA, Sept. 1981.

[6] F. Stann and J. Heidemann, "Rmst: Reliable data transport in sensor networks," in *SNPA*, 2003.

[7] *Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, IEEE P802.11, IEEE Std., 1999.

[8] Y. Sankarasubramaniam, B. Akan, and I. F. Akyildiz, "Esrt: Event-to-sink reliable transport in wireless sensor networks," in *MobiHoc*, 2003.

[9] C.-Y. Wan, S. B. Eisenman, and A. T. Campbell, "Coda: Congestion detection and avoidance in sensor networks," in *SenSys*, 2003.

[10] M. Degermark, B. Nordgren, and S. Pink, "Ip header compression," RFC 2507, Feb. 1999.

[11] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *MOBICOM*, 2000.

[12] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," in *MOBICOM*, 2001.

[13] J. Postel, "User datagram protocol," RFC 768, Aug. 1980.

[14] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *IEEE INFOCOM*, 2002.

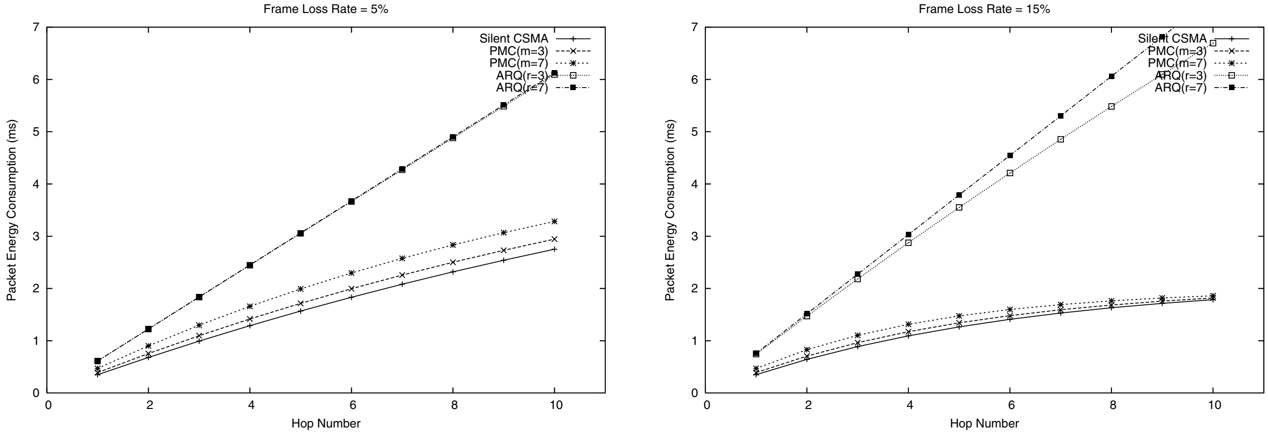

Fig. 2. Packet Energy Consumption of Silent CSMA, ARQ (CSMA/CA/ARQ) and PMC under 5% and 15% FLR

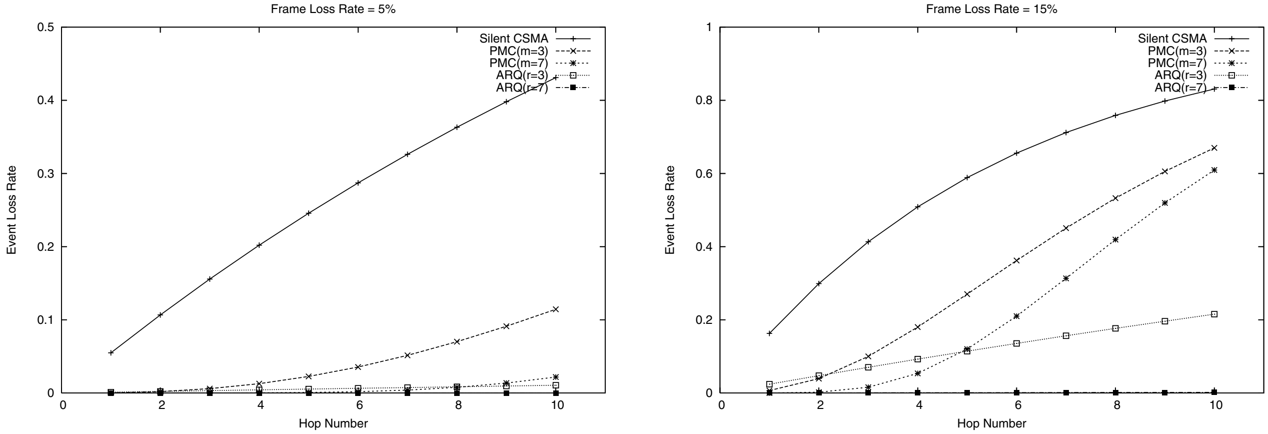

Fig. 3. Event Loss Rate of Silent CSMA, ARQ (CSMA/CA/ARQ) and PMC under 5% and 15% FLR

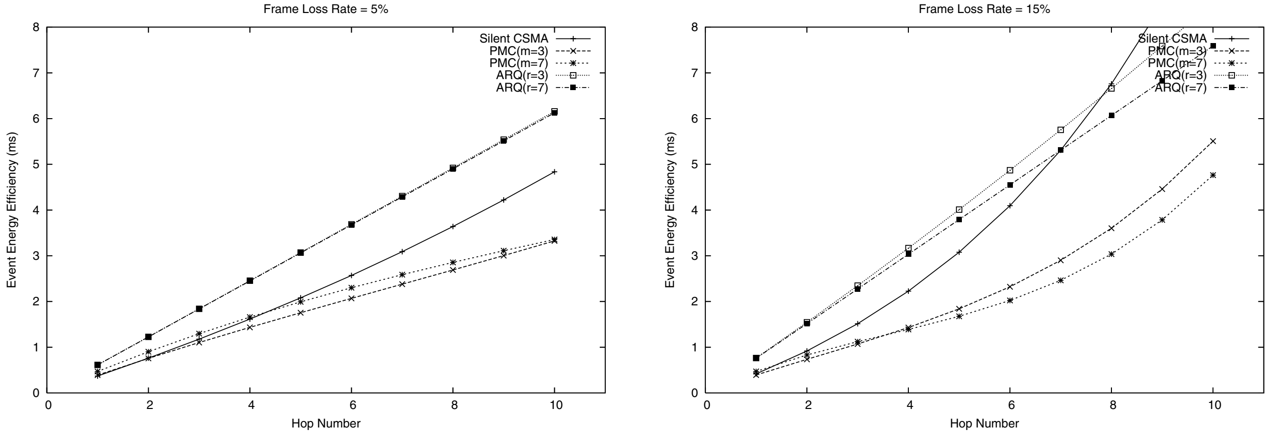

Fig. 4. Event Energy Efficiency of Silent CSMA, ARQ (CSMA/CA/ARQ) and PMC under 5% and 15% FLR