ISSN: 2321-7782 (Online)
Volume 2, Issue 11, November 2014
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com
Special Issue: International Conference on Advanced Research Methodology Held by The International Association for Scientific Research & Engineering Technology, India

RPMAC: A Novel Receiver Pivotal MAC Protocol Event Driven Wireless Sensor Networks

Kancherla Vindhya[1]
M.Tech Student
Department of Computer Science and Engineering,
Siddharth Institute of Engineering and Technology

P.Ramesh Babu[2]
Associate Professor
Department of Computer Science and Engineering,
Siddharth Institute of Engineering and Technology

P. Nirupama[3]
Head of the Department
Department of Computer Science and Engineering,
Siddharth Institute of Engineering and Technology

***Abstract:**Wireless sensor networks (WSNs) have been deployed for many applications [1] [2] [3] [4]. For event-driven applications such as surveillance, intrusion detection and target tracking, sensor nodes normally operate under light traffic load. However, once an event of interest is detected, nodes may generate a large amount of data. A high-throughput channel is desired to deliver the bursty data in a timely fashion. Unfortunately, the severe channel contention due to nodes' simultaneous needs of data transmission together with the lossy nature of wireless communication makes it a challenging task to achieve high throughput and reliable data transmission in such an occasion. Event-driven wireless sensor networks (WSNs) usually operate under light traffic load. However, when an event is detected, a large number of packets may be generated. A MAC protocol designed for this kind of WSNs should be able to swiftly adapt to the two conditions. Most WSN MAC protocols are optimized for light traffic for the energy efficiency consideration.*

In this paper, we propose a novel receiver-pivotal MAC protocol called RP-MAC that seamlessly integrates duty cycling and receiver centric scheduling, providing high throughput without sacrificing the energy efficiency. To handle bursty traffic triggered by an event, RP-MAC takes advantage of the underlying data gathering tree structure of WSNs and multichannel technique supported by current IEEE 802.15.4 RF transceivers to assist scheduling of medium access. We design a scheduling pattern that ensures fairness among source nodes without sacrificing the throughput. We evaluate the performance of our RP-MAC through measurements of an implementation in extensive ns-2 simulations.

Keywords : RP-MAC, Event Driven WSN.

I. INTRODUCTION

In WSNs, energy efficiency is a fundamental criterion in the design of WSN protocols. Most designs let nodes work in the duty cycling mode where each node periodically alternates between active and sleep states. Because two nodes can communicate only when they are both active, early studies in WSN MAC protocols mainly focus on how to establish the common active periods between nodes efficiently. The contention resolution is left to the Carrier Sense Multiple Access with Wireless sensor networks (WSNs) have been deployed for many applications [1] [2] [3] [4]. For event-driven applications such as surveillance, intrusion detection, and target tracking, sensor nodes normally operate under light traffic load. However, once an event of interest is detected, nodes may generate a large amount of data. A high-throughput channel is desired to deliver the bursty data in a timely fashion. Unfortunately, the severe channel contention due to nodes' simultaneous needs of data transmission together with the lossy nature of wireless communication makes it a challenging task to achieve high throughput and reliable data transmission in such an occasion.

Collision Avoidance (CSMA/CA). When an event is detected, multiple sensor nodes may report the event. The contention nature of CSMA/CA may intensify channel collision when a lot of nodes in the same communication neighborhood contend for sending. Because nodes that detect the same event are close to each other, the spatial correlation between sources motivates us to introduce medium access scheduling in data delivery.

A characteristic of traffic in WSNs is that a data gathering tree is naturally formed when data are delivered from source nodes to the sink. On the tree, each intermediate node receives packets from multiple children and forwards packets to its own parent at a higher level. We define a basic parent-children set as one parent with its multiple children. The central position of the parent makes the parent an ideal candidate to manage medium access of its children, which can reduce contention from multiple children to a single node by only allowing one child to send at any time.

In addition, the fairness among source nodes is ensured by allocating more medium access opportunities to children that have higher bandwidth demands. We show that the network throughput and the fairness between source nodes are improved by shifting the scheduling function to the receiver side. The main contributions of this paper can be summarized as follows.

_ By utilizing the unique tree structure that is naturally formed during data gathering in WSNs, we propose a receiver-centric scheduling method that addresses the collision problem in a basic parentchildren set.

» We propose a scheduling pattern that ensures fairness between source nodes without sacrificing the throughput by taking different bandwidth demands of nodes and packet process time of nodes into consideration.

» We propose a lightweight multichannel scheme to achieve parallel data gathering.

» When multiple parent-children sets share a channel, we propose a dynamical scheduling cycle length adjustment scheme to ensure fairness between sets.

II. Related Work

Corresponding to the important role of Medium Access Control (MAC), there are plenty of studies on WSN MAC protocols [5] [6] [7] [8] [9] [10] [11]. Most of them are optimized for light traffic for energy efficiency consideration. They focus on how to incorporate duty cycling to achieve low power operation.

To let a node know that there are data for it, the preamble sampling technique is introduced along with the Mica wireless platform [12]. In protocols that employ the technique [13] [5], a sender is required to prefix a packet with a preamble that is long enough to be detected by any intended receiver. A drawback is that the long preamble also wakes nodes that are not the intended receiver up and causes an overhearing problem that compromises the energy efficiency gained by duty cycling.

To address the issue, some protocols [14] [15] [6] divide a long preamble into a series of short preamble packets that contain useful information. If the intended receiver's address is included in each short preamble packet, the overhearing problem can be alleviated. The reason is that during the pauses between short preamble packets, the receiver can send an *early ACK* to stop preamble transmission, initiating data transmission immediately. As a result, a strobed preamble reduces the preamble length in average, but the preamble transmission still occupies the wireless medium and prevents neighboring nodes from transmission, which decreases the throughput and increases the delay.

To squeeze more room for data transmission, one solution is to use the active low power probing (LPP) [15] [16] [17] to replace the passive low power listening (LPL). In the LPP technique, a node does not need to send a long preamble prior to each packet. Instead, each node broadcasts a short beacon to indicate that it is ready to receive when it wakes up. In this way, a sender will not occupy the wireless medium until the communication can be established. As a result, LPPbased RI-MAC [7] demonstrates higher throughput and greater power efficiency than X-MAC. PW-MAC [18] further improves LPP-based protocols with prediction of the intended receiver's beacon so that the sender no longer needs to stay awake waiting for the beacon. However, under heavy traffic load, the throughput of LPP-based protocols is bounded by collisions between senders. The performance of RI-MAC is similar to that of X-MAC as they all leave contention handling to the CSMA/CA mechanism.

Prior researches seek efficient ways to find communication periods between nodes. The contention between nodes is left to the CSMA/CA. In MAC protocol designs, TDMA has been demonstrated to provide higher throughput than CSMA/CA when multiple nodes contend [19]. A main problem of TDMA is that the channel utilization is low when few nodes have data to send. Because the overhead of time slot assignment is usually high, it is hard to adaptively assign time slots only to nodes that are involved in current data gathering. Some designs [19] [20] [21] thus integrate CSMA/CA and TDMA to increase channel utilization. We show that by utilizing the single parent multiple children relationship formed in data gathering, the contention issue can be addressed in a more efficient way.

Further, RP-MAC boosts network throughput by trying to assign different channels to different parent children sets instead of different links. This requires fewer orthogonal channels for a non-overlapping channel assignment.

In multichannel schemes, dynamic assignment is more adaptable to traffic. However, dynamic channel hopping schemes like Y-MAC [22], MuChMAC [23], and EM-MAC [24] face a challenge that neighboring nodes may hop to the same channel. It has a minor impact on the performance if nodes jump to a channel for only one or two packets and then jump to the next channel. In RC-MAC, a parent-children set will stay in a channel for receiver-centric scheduling. To reduce scheduling interruptions, RP-MACtries to assign each parent-children set a unique channel.

III. Overview

Routing protocols in WSNs usually form a data gathering tree [1] [2]. RP-MAC aims at exploiting the tree structure to improve network performance. Fig. 1 shows that each intermediate node on a data gathering tree is both a parent of multiple directly connected nodes that are one level deeper down the tree and a child of a directly connected node that is one level higher on the tree. Because of the single parent multiple children relationship, a node is able to schedule its children's data transmissions by reusing the ACK.

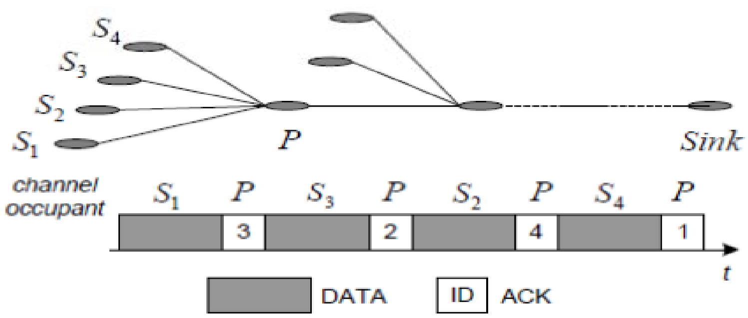


Fig 1. RP-MACschedules the next sender through overhearing.

Due to the broadcast nature, the ACK sent to one of the children can be overheard by all children. A parent can schedule children's data transmissions by piggybacking a scheduling message, which is the ID of the child that can send immediately, to an ACK. Overhearing an ACK containing a scheduled node ID, a child backs off if the scheduled node ID is not equal to itself, or starts to transmit immediately if the scheduled node ID matches its own ID. As a result, channel collisions are avoided because only the scheduled child starts to transmit while all other siblings postpone their attempts. RP-MACoperates as "pulling" data rather than the traditional pattern of "pushing" data. This simple yet important transition can improve the network performance significantly as demonstrated in the performance evaluation section. The transition relies on the many-to-one traffic in WSNs. It may be inappropriate for applications that use disjoint paths. The basic idea makes no computationally complex decisions and easily fits to a sensor node's limited memory and computing power. To make it practical, there exist several design challenges.

First, for energy efficiency consideration, we need to incorporate low power operation for light traffic load. We follow the trend of receiver-initiated LPP design and propose an offset-based method for predicting the receiver's probing time.

Second, due to the unbalanced tree structure, different nodes have different bandwidth demands and thus we need to assign medium access opportunities to them accordingly. Improving fairness faces a challenge that the channel utilization may be reduced if we keep scheduling the same node. A scheduling pattern is introduced in the paper to improve the fairness without sacrificing the throughput.

Third, although the parent-children relationship can help reduce collisions in basic parent-children set contentions between sets still exist in multi-hop scenarios. On the one hand, we do not want the scheduling of a set to be interrupted by contentions so that we can achieve high throughput. On the other hand, we cannot let one set occupy the channel exclusively due to fairness issue. Therefore, dynamic adjustment and collision resolving are highly desirable.

IV. DESIGN DETAILS

In this section, we describe each component in detail, addressing the aforementioned issues one by one.

a) Adapt to Low Duty Cycle

RP-MACis designed to be effective under heavy traffic load when events of interest are detected. However, in most of the time, an event-driven WSN generates very few or no data packets. In order to improve energy efficiency, RP-MACshould be able to adapt to the widely adopted duty cycling technique. Although receiver-initiated Low Power Probing (LPP) improves communication efficiency by avoiding occupying the medium for preamble transmission [15] [17] [7], all senders have to stay awake waiting for the receiver's beacon. To reduce idle listening at senders, PW-MAC [18] utilizes pseudo-random wakeup schedules to keep track of neighbors' beacons. However, the method requires a node to keep calculating the next beacon of all one-hop neighbors so that it does not lose track of the pseudo-random sequences. When nodes' beacons can be spread out in a beacon interval, we use an offset-based method to help a node estimate the proper time to wake up. The periodic beacons allows a node to perform the calculation only when it has data to the receiver.

When nodes are deployed, they first enter in the setup phase and stay awake during the phase. Each node randomly selects a time to broadcast a beacon and periodically broadcasts the beacon every beacon interval Ib. When a node i receives a beacon from node j, it calculates the offset between the neighbor's beacon time Bj and its own beacon time Bi (i.e., $Oij = Bj-Bi$). Each node constructs a beacon table that records its one-hop neighbors' IDs and their corresponding beacon offsets. When a node has data to a neighbor, it estimates the receiver's beacon time by inverting the offset calculation (i.e., $Bj = Bi + Oij$). If two nodes' beacon schedules are similar, they may interfere with each other when receiving data.

Therefore, we define a minimum offset requirement τ that is long enough for transmitting a few packets. If a node detects an offset that is smaller than τ , the node adjusts its beacon schedule as follows. It sorts the offsets and searches for the gaps between two adjacent beacons. It divides the gaps into slots of size $\tau+g$ where g is a short guard interval and randomly chooses one as its new beacon schedule.

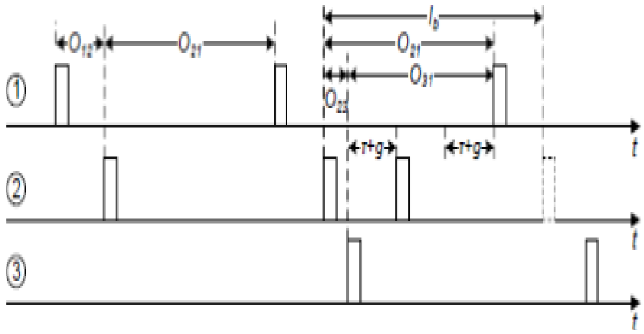


Fig 2. Probing Schedule adjustment

It is possible that several nodes adjust their schedules to the same slot. We let a node include its one-hop neighbors' beacon offsets in its beacons. If a node cannot find its offset in its neighbors' beacons, it learns that it has a schedule conflict with some nodes. The node shall choose another slot as its new beacon schedule. In addition, because a slot is long enough for multiple packets, a node broadcasts its beacon multiple times based on CSMA/CA. Each beacon carries the offset to the first beacon of the node in the slot so that neighbors understand that the minimum offset requirement is not violated.

In the medium access contention for beaconing in a slot, a node should stop contending if it detects a successful transmission or a collision. Suppose *k* nodes have selected the same slot and a slot contains *l* subslots for contention. The probability of a successful transmission in a subslot is

$$p_s = \binom{k}{1} p_0(1-p_0)^{k-1} = kp_0(1-p_0)^{k-1}$$

where *p*0 is the probability that a node transmits in a randomly selected subslot as analyzed in [25]. After *n* contentions, the probability of at least one successful transmission is

$$p = 1 - (1-p_s)^n.$$

Let *p* = 0.99 and *k* be the average number of neighbors. The value of *n* can be estimated from Eq. 1 and Eq. 2. If there is a schedule conflict, nodes will detect it with a probability of 99% in beacon intervals. Therefore, a node switches to the duty-cycling mode if there is no collision and no change of beacon offset in beacon intervals. Under light traffic load, nodes go to sleep mode whenever possible. Under heavy traffic load, high throughput and low delay are more critical than energy efficiency. Therefore, we let nodes work in the full active mode, searching for any chance of data transmission.

The total active period can be shortened if data are delivered faster. The mode is switched by bandwidth demand information attached to data packets. When a node detects an event, it switches to the full active mode and attaches bandwidth demand information to its data packets to inform its parent of the packet generation rate. When a node receives a packet with bandwidth demand information, it switches to the full active mode, preparing for the imminent heavy traffic. The parent also informs its own parent of the bandwidth demand when its parent wakes up. When a packet is no longer attached with bandwidth demand information, relevant nodes switch back to the duty cycling mode.

b) Medium Access Scheduling

In this section, we first introduce how nodes operate under scheduling and then discuss the scheduling pattern that guarantees fairness among source nodes without sacrificing throughput.

1. Operation under scheduling

As nodes initially operate in the duty cycling mode, a node wakes up slightly earlier than its parent when it has data to send. Once it receives a beacon from its parent, it starts to contend for sending. When a node receives a data packet with bandwidth demand, it disables the duty cycling and responds an ACK that contains the ID of the next sender. Because sources are spatially correlated in event-driven applications [28], a node assumes that all of its children have related data to send if a node receives an event report from one of its children. As discussed in Section 4.1, the underlying data gathering tree is implicitly constructed through periodic update of neighbors' beacon offsets.

Upon receiving an ACK that contains the next-sender ID, the scheduled child transmits immediately if the channel is detected to be idle. On the contrary, neighboring nodes will refrain from sending for a random period of time and then start to contend if the specified child does not respond. Designing different contention window sizes for the random backoff can distinguish unscheduled children of the parent from nodes that are not the children of the parent. Upon receiving an ACK, scheduled child must respond within $T1$ and unscheduled siblings perform in-group contention with contention window (CW) $(T1, T2]$, non children nodes perform ongoing transmission contention with CW $(T2, T3]$, and nodes within the interference range use network allocation vector (NAV).

The extended interframe space (EIFS) used in NAV is set to be long enough for an ACK-DATA-ACK round. Here, $T1$ is a small value that accounts for the propagation delay and the ACK processing time. Since the parent knows the number of its children, $T2$ can be set optimally to handle contentions between children by analytical techniques [19], and $T3$ is the commonly used Binary Exponential Backoff (BEB) value. Fig. 3. illustrates an example of the scheduling and contention.

In the design, we cannot let one set occupy t channel exclusively. As shown in , if we do not terminate $P1$'s scheduling, $P1$ will keep scheduling its children for data transmission and no neighboring nodes including $P1$ itself can win a medium access opportunity. The length of a scheduling cycle is dynamically adjusted as described in Section 4.3.

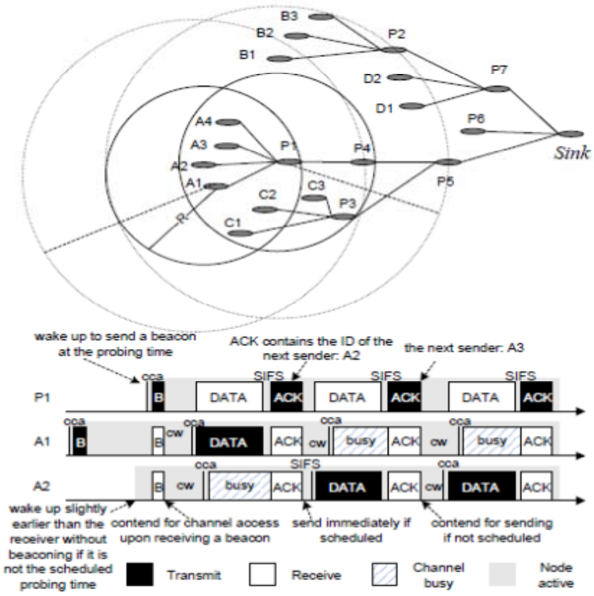


Fig 3. An example of Receiver-Pivotal medium access scheduling

2. Account for different bandwidth demands

Scheduling children in a simple round robin manner seems to be fair in a basic set, but it is unfair in a global view. As shown in Fig. 2, $P2$ should be given more medium access opportunities than $D1$ and $D2$ because in addition to its own data, it also gathers data from $B1$ to $B3$. If we use round robin scheduling, the fairness among source nodes cannot be guaranteed: the throughput of Bi $(i = 1, 2, 3)$ will be one third of that of Dj $(j = 1, 2)$.

Let bu denote the total bandwidth demand of node u, su represent the total bandwidth demand of node u's subtree, and du represent the data rate generated by node u. The total bandwidth demand of node u is equal to the total bandwidth demand of node u's subtree pΣlus the data rate of itself, which is $bu = su + du = ciu \sum Cu$ €ciu+ du where Cu is the children set of node u

and *ciu* is the *i*th child of node *u*. The probability that node *u* will be selected as the next sender by its parent *pu* is equal to *Lu* = *bu/spu* = *bu/Σcipu2Cpubcipu*. In this way, nodes of heavier traffic load obtain more medium access opportunities and the fairness among source nodes is improved.

c) Contention between Sets

1. Channel definition

We let each node keep track of three channels: the first one is the default common channel, the second one is its data gathering channel, and the third one is its data forwarding channel (which is same as its parent's data gathering channel). In the duty cycling mode, all nodes communicate in the default common channel. Although a node keeps track of three channels, it only needs to obtain a data gathering channel when it has data to receive.

It is desirable to assign different parent-children sets different channels to gather data simultaneously. Fig. 4 shows that when the sink is gathering data, its grandchildren (i.e., P2, P3, and P4) can gather data at the same time. To achieve the parallel data gathering, one challenge is to assign channels to parent-children sets in a distributed way. If each parent-children set can obtain a channel that is free of interference, each set can benefit from the receiver-centric scheduling to the most extent. Before introducing channel assignment, we first discuss how nodes switch from the common channel to the data gathering/forwarding channels.

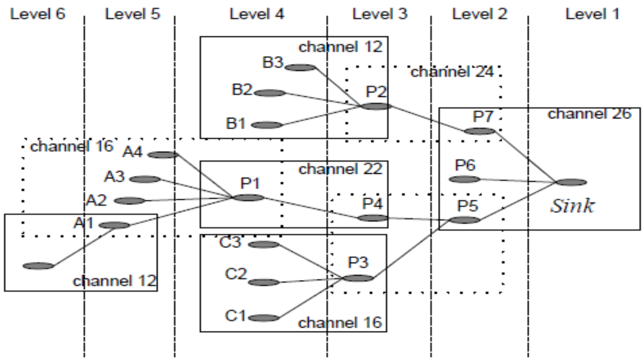


Fig 4. Receiver-Pivotal Chanel Assignemt

2. Channel switch

The sink notices that an event is detected, it enters in the scheduling mode. Once it finishes one cycle of data gathering, it backs off to give its children some time to gather data. To indicate the termination of a scheduling cycle, the sink sends an ACK that sets *PENALIZE* as the next sender and $\epsilon = 0$ to indicate the next data gathering cycle starts at $\epsilon + \tau$ where τ is predetermined to be long enough for gathering *M* packets.

When the children of the node receives the ACK, they start a channel switch timer. The timer indicates when they should switch back to the parent's data gathering channel for data forwarding (the sink simply uses the common channel for data gathering). In other words, the children of the sink must switch back to the sink's data gathering channel for data forwarding after a certain period of time τ .

3. Channel assignment

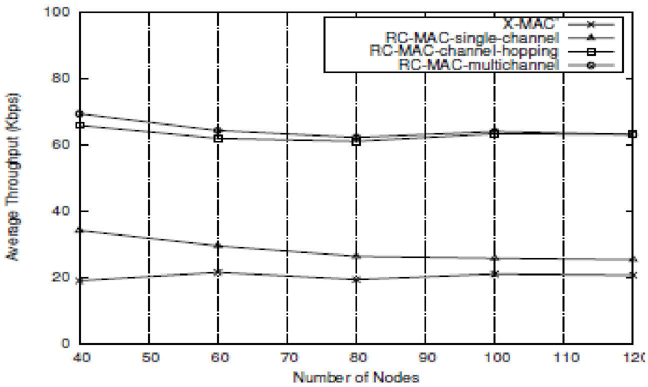
To assign different channels to parent-children sets that are interfering with each other, the first step is to obtain the interference list. In the duty cycling mode, a node learns its neighbors in its two-hop communication neighborhood via beacons. A node tries to avoid the channels that are used by these potential interference sources. In Fig. 4 node *P*4 should avoid channels used by *Sink*, *P*3, and *A*1 to *A*4. They are in the two-hop communication neighborhood of *P*4 and have data gathering conflicts with *P*4. If *P*1 and *B*1 are one hop away from each other, *P*4 and *P*2 may be close to each other. To remove possible interference, RP-MACalso assigns different channels to $P4$ and $P2$ even though they are three hops away. To accomplish this, a node attaches its parent's ID in its beacons.

V. SIMULATION EVALUATION

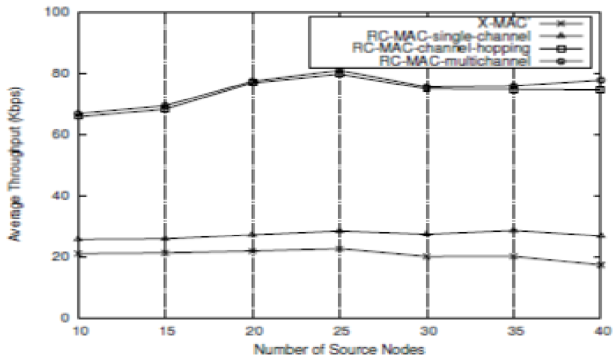
To create more parent-children sets that may interfere with each other, we conduct simulations in ns-2. The simulations show that the receiver-centric scheduling can benefit a lot from the multichannel support when there are many contending parent-children sets.

To compare receiver-centric scheduling with CSMA/CA, we compare RP-MACwith X-MAC' where the preamble is disabled under heavy traffic load. We first study the impact of node density. In a 1000×1000 $m2$ field, N nodes are uniformly distributed and 10 nodes are randomly selected as the source nodes. Each source node generate data packets at a rate of 32 Kb/s. Fig. 5(a) shows that the node density impacts the shape of the data gathering tree, which in turn impacts how data collection can benefit from the receiver-centric scheduling. When the number of nodes is small, the paths from source nodes to the sink usually merge at some nodes.

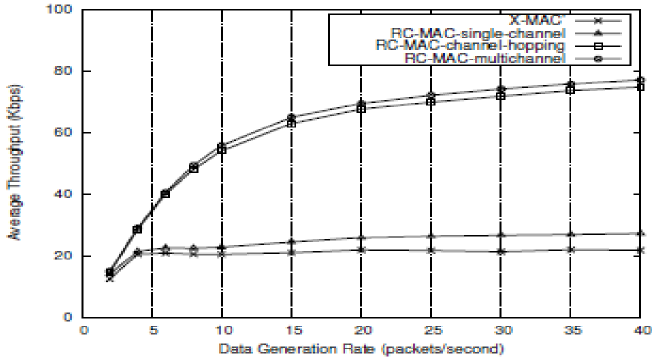
When a node has multiple children, the receiver-centric scheduling helps to improve the performance of data collection. When the node density is high, sources may find disjoint paths to the sink. The receiver-centric scheduling only helps data collection at the sink. The improvement of RP-MACover X-MAC' shrinks when the node density increases. However, if the number of source nodes is also increased, the difference between RP-MACand XMAC' becomes larger as shown in Fig. 5(b) where the total number of nodes is fixed to 100. This is because more active routes may merge at some nodes and the receiver-centric scheduling can help data collection at these intermediate nodes.



(a)The impact of node density in on Throughput



(b) The Impact of the number of source nodes on throughput



(C) the Impact of data generation rate on throughput

Fig 11. Simulation results in multihope networks

Fig. 5 shows that it is important to introduce multichannel support to protect the receiver-centric scheduling in each parent-children set. We also implement RC-MAC-channel-hopping where nodes hop to different data gathering channels in different data gathering cycles with the method introduced in EMMAC [24].

Fig. 5 shows that most parent-children sets can obtain an interference-free data gathering channel. Some parent-children sets may have to share a channel and the throughput is slightly decreased as shown in Fig. 5(a). If we let each set hop to a random channel for data gathering in different data gathering cycles, two neighboring sets may hop to the same channel and interfere with each other in that cycle.

Fig. 5(c) shows that the multichannel support can help RP-MACmaximize the effect of receiver-centric scheduling. With multichannel support, parent-children sets that interfere with each other can be allocated to different channels for data gathering.

VI. Conclusion

In this paper, we propose a novel receiver-pivotal MAC protocol called RP-MAC that seamlessly integrates duty cycling and receiver centric scheduling, providing high throughput without sacrificing the energy efficiency. To handle bursty traffic triggered by an event, RP-MAC takes advantage of the underlying data gathering tree structure of WSNs and multichannel technique supported by current IEEE 802.15.4 RF transceivers to assist scheduling of medium access. We design a scheduling pattern that ensures fairness among source nodes without sacrificing the throughput. We evaluate the performance of our RP-MAC through measurements of an implementation in extensive ns-2 simulations.

References

1 T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J. A. Stankovic, T. F. Abdelzaher, J. Hui, and B. Krogh, “VigilNet: An integrated sensor network system for energy-efficient surveillance,” ACM Trans. Sensor Networks (TOSN), vol. 2, no. 1, Feb 2006.

2 G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh, “Fidelity and yield in a volcano monitoring sensor network,” in Proc. OSDI, 2006.

3 G. Wittenburg, K. Terfloth, F. L. Villafuerte, T. Naumowicz, H. Ritter, and J. Schiller, “Fence monitoring – experimental evaluation of a use case for wireless sensor networks,” in Proc.EWSN, 2007.

4 J. Koo, R. K. Panta, S. Bagchi, and L. Montestruque, “A tale of two synchronizing clocks,” in Proc. SenSys, 2009.

5 J. Polastre, J. Hill, and D. Culler, “Versatile low power media access for wireless sensor networks,” in Proc. SenSys, 2004, pp.95–107.

6 M. Buettner, G. V. Yee, E. Anderson, and R. Han, “X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks,” in Proc. SenSys, 2006, pp. 307–320.

7 Y. Sun, O. Gurewitz, and D. B. Johnson, “RI-MAC: A receiverinitiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks,” in Proc. SenSys, 2008,pp. 1–14.

8 W. Ye, J. Heidemann, and D. Estrin, “An energy-efficient MAC protocol for wireless sensor networks,” in Proc. INFOCOM, vol. 3, Jun. 2002, pp. 1567–1576.

9 T. van Dam and K. Langendoen, “An adaptive energy-efficient MAC protocol for wireless sensor networks,” in Proc. SenSys, Nov. 2003, pp. 171–180.

10 A. El-Hoiydi and J.-D. Decotignie, "Low power downlink MAC protocols for infrastructure wireless sensor networks," Mobile Networks and Applications, vol. 10, no. 5, pp. 675–690, Oct. 2005.

11 W. Ye, F. Silva, and J. Heidemann, "Ultra-low duty cycle MAC with scheduled channel polling," in Proc. SenSys, 2006, pp. 321–334.

12 J. L. Hill and D. E. Culler, "Mica: A wirelessplatform for deeply embedded networks," IEEE Micro, vol. 22, no. 6, pp. 12–24, November/December 2002.

13 A. El-Hoiydi, "Spatial TDMA and CSMA with preamble sampling for low power ad hoc wireless sensor networks," in Proc. of the Seventh International Symposium on Computers and Communications (ISCC'02), 2002, pp. 685–692.

14 S. Mahlknecht and M. Bock, "CSMA-MPS: a minimum preamble sampling MAC protocol for low power wireless sensor networks," in Proc. of IEEE International Workshop on Factory Communication Systems, September 2004, pp. 73 – 80.

15 E.-Y. Lin, J. Rabaey, and A. Wolisz, "Power-efficient rendez-vous schemes for dense wireless sensor networks," in Proc. of IEEE International Conference on Communications (ICC'04), vol. 7, June 2004, pp. 3769 – 3776.

16 A. El-Hoiydi and J.-D. Decotignie, "Low power downlink MAC protocols for infrastructure wireless sensor networks," Mob. Netw. Appl., vol. 10, pp. 675–690, October 2005.

17 R. Musaloiu-E., C.-J. M. Liang, and A. Terzis, "Koala: Ultra-low power data retrieval in wireless sensor networks," in Proc. IPSN, 2008, pp. 421–432.

18 L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "PW-MAC: An energy-efficient predictive-wakeup MAC protocol for wireless sensor networks," in INFOCOM, 2011, pp. 1305–1313.

19 I. Rhee, A. Warrier, M. Aia, and J. Min, "Z-MAC: A hybrid MAC for wireless sensor networks," in Proc. SenSys, 2005, pp. 90–101.

20 G.-S. Ahn, S. G. Hong, E. Miluzzo, A. T. Campbell, and F. Cuomo, "Funneling-MAC: a localized, sink-oriented MAC for boosting fidelity in sensor networks," in Proc. SenSys, 2006, pp. 293–306.

21 V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves, "Energy-efficient collision-free medium access control for wireless sensor networks," in Proc. SenSys, 2003, pp. 181–192.

22 Y. Kim, H. Shin, and H. Cha, "Y-MAC: An energy-efficient multi-channel MAC protocol for dense wireless sensor networks," in IPSN, 2008, pp. 53–63.

23 J. Borms, K. Steenhaut, and B. Lemmens, "Low-overhead dynamic multi-channel MAC for wireless sensor networks," in EWSN, 2010, pp. 81–96.

24 L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "Em-mac: A dynamic multichannel energy-efficient mac protocol for wireless sensor networks," in MobiHoc, 2011.

