Exploiting Spatial Correlation at the Link Layer for Event-driven Sensor Networks

Chih-Yu Lin

Gogolook Inc,
Taipei, 107, Taiwan
E-mail: lincyu@cs.nctu.edu.tw

Yu-Chee Tseng

Department of Computer Science,
National Chiao-Tung University, Hsin-Chu, 300, Taiwan
E-mail: yctseng@cs.nctu.edu.tw

Ten H. Lai

Department of Computer Science and Engineering,
The Ohio State University, Columbus, Ohio 43210 USA
E-mail: lai@cse.ohio-state.edu

Abstract: Wireless sensor networks (WSNs) can generally be classified into two categories: time-driven and event-driven. In an event-driven WSN, sensors report their readings only when they detect events. In such behavior, sensors in the event area may suffer from higher contention. In this paper, we solve this problem by jointly considering two subissues. One is exploiting the spatial correlation of data reported by sensors in the event area and the other is designing a specific MAC protocol. We propose a novel hybrid TDMA/CSMA protocol with the following interesting features that differentiate itself from traditional TDMA-based protocols: (1) the TDMA part is based on very loose time synchronization and is triggered by the appearance of events. (2) the slot assignment strategy associated with the TDMA part takes the spatial correlation of sensor data into consideration and thus allows less strict slot allocation than conventional TDMA schemes. Interestingly, by intentionally allowing one-hop neighbors to share the same time slot, the number of slots required is significantly reduced. (3) by intentionally enlarging the slot size, packets will be separated sufficiently in distance to avoid interference after leaving the event area. In addition, we also discuss how to combine our protocol with the LPL (Low Power Listening) technique to achieve energy efficiency. Simulation results show the efficiency of the proposed protocol.

Keywords: wireless sensor network; medium access control; TDMA; CSMA; spatial correlation.

Reference to this paper should be made as follows: Lin, C.-Y., Tseng, Y.-C. and Lai, T.-H. (2012) ‘Exploiting Spatial Correlation at the Link Layer for Event-driven Sensor Networks’, *Int. J. of Sensor Networks*, Vol. 1, Nos. 3/4, pp.197–212.

Biographical notes: Chih-Yu Lin received the BS and MS degrees from the National Central University, Taiwan, in 1998 and 2000, respectively. He obtained his Ph.D. degree in Computer Science from the National Chiao-Tung University, Taiwan, in September of 2007. His research interests include mobile computing and wireless communication.

Yu-Chee Tseng obtained his Ph.D. in in Computer and Information Science from the Ohio State University in January 1994. He is currently a Chair Professor and the Dean of College of Computer Science, National Chiao Tung University. His research interests include mobile computing, wireless communication, network security and parallel and distributed computing. He is a member of ACM and IEEE Fellow.

Ten H. Lai is currently a Professor in the Department of Computer Science and Engineering, the Ohio State University. His research interests include distributed computing and wireless networks.

1 Introduction

Wireless sensor networks (WSNs) can generally be classified into two categories: time-driven and event-driven. In a time-driven WSN, sensors report their sensed data to the sink periodically. Such reporting behavior is typically predictable and involves all sensors in the WSN. Thus, such reporting behavior usually exhibits a unique *funneling* effect (Ahn et al., 2006), where sensors near the sink may suffer from higher contention. Several approaches have been developed to relieve such contention (Ahn et al., 2006; Wan et al., 2005). Some convergecast or scheduling schemes have also been proposed to provide collision-free transmission for this type of communications (Hohlt et al., 2004; Pan and Tseng, 2008). On the contrary, in an event-driven WSN, sensors report only when they detect events. The occurrence of events is unpredictable and spontaneous in terms of time and space. Thus, traffics near the sink may not be heavy, but sensors in the *event area* may suffer from higher contention, because they are likely to detect, and thus intend to report, events simultaneously.

Our goal in this work is to solve the contention problem of event reporting in event-driven WSNs. The proposed approach joints two subissues. One is exploiting the spatial correlation of data reported by sensors in the event area and the other is designing a specific MAC protocol. Due to the spatial correlation of sensor data, nearby sensors typically have similar values. Thus, it may not be necessary for every sensor to report its sensed data. For example, in Fig. 1(a), the cloud area denotes the event area, and each sensor x is associated with a correlation region, in which sensors' readings are highly correlated with x. We can see that it is sufficient to have sensors a, b, c, and d to report to cover the event area. In addition, we can also note that the selection of reporting sensors needs to be done carefully. For example, in Fig. 1(b), the five gray sensors are insufficient to cover the whole event area. Exploiting the spatial correlation of sensor data to reduce redundant reports has been studied by Pattem et al. (2004); Pradhan et al. (2002); Scaglione and Servetto (2005); Vuran and Akyildiz (2006); Yoon and Shahabi (2005) and so on.

The second subissue is to design a specific MAC protocol. After reporting sensors are selected, we need to reduce the contention and collision among these reporting packets and to minimize the latency in transmitting these packets. Contention and collision are likely because these sensors may be close to each other. In addition, because packets in WSNs are typically small, using the RTS/CTS mechanism to avoid the hidden terminal problem is not preferred. Thus, the collision problem becomes more severe without the aid of RTS/CTS. Besides, since these packets are likely to share common paths when moving toward the sink, we would like to see that packets are separated spatially (e.g., Fig. 1(a)), rather than close to each other (e.g., Fig. 1(b)), along these paths to avoid collision. Thus, designing a specific MAC protocol for event-driven WSNs is required.

Exploiting the spatial correlation of sensor data at the MAC layer has been discussed by Vuran and Akyildiz (2006), where the relation between the spatial positions of sensors and the event estimation reliability is investigated. Specifically, a distortion function is derived and a term *correlation radius* (R_{corr}) is introduced. Then, CC-MAC (spatial Correlation-based Collaborative Medium Access Control) is proposed. CC-MAC consists of two components: E-MAC (Event MAC) and N-MAC (Network MAC). E-MAC aims to filter out correlated reporting (i.e., determine which sensors can report). On the other hand, N-MAC is mainly used for sensors not in the event area to forward reporting packets. However, CC-MAC has the following drawbacks: (i) E-MAC is a purely contention-based protocol. Although some sensors may withdraw from reporting, those sensors that decide to report will still cause a lot of contention, because they will report simultaneously. (ii) The RTS/CTS mechanism is adopted, which causes high overheads when packet sizes are small. (iii) In CC-MAC, when a sensor x overhears a packet reported by another sensor y, x will judge whether the distance between itself and y is smaller than R_{corr}. If so, x will suspend its report. As to be shown later, this simple condition cannot completely remove redundant reports. (iv) The report reduction technique proposed in CC-MAC highly depends on overhearing; thus, redundancy may still exist when when one misses overhearing.

In this paper, motivated by Rhee et al. (2008) and Vuran and Akyildiz (2006), we propose a schedule-based approach to exploit the spatial correlation of sensor data at the link layer. Our goal is to solve the contention problem in the event areas. We do not modify the underlying MAC protocol directly. On the contrary, we develop a scheme for making report decision and propose a protocol for transmitting reporting packets at the link layer. By doing so, any CSMA-based protocol designed for WSNs could be adopted as the underlying MAC protocol. Thus, we can simply leave some issues (e.g., power saving) to the MAC protocol itself. In our approach, a node has two modes: *ES (Event-Source) mode* and *NES (Non-Event-Source) mode*. Initially, each sensor is in the NES mode. On detecting an event, a sensor will enter the ES mode and adopt a schedule-based protocol to transmit its packets. (The schedule-based protocol can be regarded as a loose TDMA-based protocol built on top of a CSMA-based protocol.) This schedule-based protocol has some characters that make it different from conventional TDMA-based protocol. First, the TDMA part is based on loose time synchronization and is triggered by the appearance of events. Second, the slot assignment strategy associated with the TDMA part takes the spatial correlation of sensor data into consideration and thus allows less strict slot allocation than conventional TDMA schemes. Interestingly, by intentionally allowing one-hop neighbors to share the same time slot, the number of slots required is significantly reduced. Third, by intentionally enlarging the slot size, packets will be separated sufficiently in distance to avoid interference after leaving the event area. In addition, a scheme is devised to exploit the spatial correlation of sensor data. Specifically, by exploiting TDMA's features, redundant reports can be further reduced with or without the aid of overhearing. Finally, we will also discuss how to achieve energy efficiency by combining our protocol with the LPL (Low Power Listening) technique proposed in B-MAC (Polastre et al., 2004).

The rest of this paper is organized as follows. Related work is discussed in Sec. 2. We make some observations to motivate our work in Sec. 3. Then, our proposed approach is presented in Sec. 4. Performance studies are conducted in Sec. 5. Finally, this paper concludes with Sec. 6.

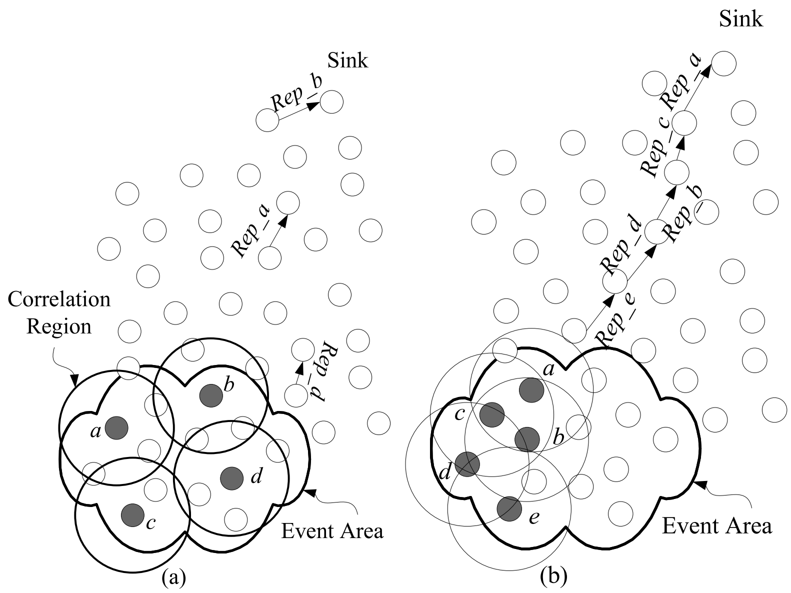

Figure 1 Two examples of event reporting in an event-driven WSN. The big circles denote the correlation regions of the gray sensors.

2 Related Work

A lot of research effort has been dedicated to the design of MAC protocols for WSNs (Ahn et al., 2006; Lee et al., 2006; Polastre et al., 2004; Rajendran et al., 2006; Rhee et al., 2008; Stone and Colagrosso, 2007; van Dam and Langendoen, 2003; van Hoesel and Havinga, 2004; Ye et al., 2004). The energy efficiency issue has been studied in S-MAC (Ye et al., 2004) and T-MAC (van Dam and Langendoen, 2003) by synchronizing sensors on a common wakeup/sleep schedule. In order to eliminate the synchronization overhead, B-MAC (Polastre et al., 2004) adopts a preamble sampling technique. These three protocols are contention-based ones. Lee et al. (2006) and Rajendran et al. (2006) have proposed TDMA-based MAC protocols to provide collision-free transmission for WSNs.

Several hybrid TDMA/CSMA MAC protocols have been proposed for WSNs. Ahn et al. (2006) have proposed a localized, sink-oriented MAC protocol, called Funneling-MAC, to relieve the contention problem occurred near the sink. However, in an event-driven WSN, contention may be more severe near the event areas than the sink area. Rhee et al. (2008) have proposed Z-MAC in which sensors in the low-contention areas will adopt a CSMA-based MAC protocol and those in the high-contention areas will adopt a TDMA-based MAC protocol. Thus, Z-MAC takes the advantages of low latency of CSMA and high channel utilization of TDMA. However, when applying Z-MAC to an event-driven WSN,sensors need time to switch their modes when they detect an event. In addition, Z-MAC does not take the spatial correlation of sensor data into consideration.

Some MAC protocols have been proposed for event-driven WSNs. Jamieson et al. (2006) have proposed a backoff probability distribution to minimize the time taken to send $R < N$ of reports without collision when N nodes detect an event. The drawback is that the topology is assumed to be single-hop. In addition, the issue of the quality of data (e.g., distortion or reconstruction performance) is not addressed. Dong et al. (2004) and Dong et al. (2006) have investigated the impact of the MAC design on the reconstruction performance of the signal. The authors show that the benefit from carefully scheduling transmission is substantial in the high SNR regime. On the contrary, in the low SNR regime, using the random access MAC is sufficient. This gives a hint that a hybrid TDMA/CSMA MAC protocol may be required. Unfortunately, no concrete MAC protocol is proposed by Dong et al. (2004) and Dong et al. (2006). Zhao et al. (2007a) have proposed a contention-based protocol called QS-SIFT in which sensors with high SNR have higher priority than those with low SNR when contending for the shared channel. Vuran and Akyildiz (2006) have proposed a MAC protocol called CC-MAC. CC-MAC is a contention-based MAC protocol and does not take SNR into consideration. We have introduced CC-MAC and described its drawbacks in Sec. 1.

Zhao et al. (2007b) have proposed a hybrid TDMA/CSMA MAC protocol that takes the spatial correlation into consideration. In the proposed scheme, CSMA is integrated with TDMA while filtering out the spatial correlation. The event area is divided into three sub-areas: outer area, middle area and inner area. Nodes in the outer area sleep most of the time. Nodes in the middle area will be assigned different priority based on their SNR. Nodes in the inner area will adopt a schedule-based scheme. However, this protocol has one major drawback. That is, positioning technique is required to help a node to identify the region that it belongs. In our scheme, only the distance information is required to exploit the spatial correlation of sensor data.

3 Some Observations

Due to small packet size, adopting RTS/CTS is not preferred in WSNs. In this section, we assume that a CSMA-based MAC protocol is adopted in which the RTS/CTS mechanism is disabled. In order to motivate our work, we make some observations from the interference and the spatial correlation aspects.

From the interference aspect, we raise two scenarios to show that the hidden terminal problem will be very serious in an event-driven WSN. First, as shown in Fig. 2(a), when two sensors that are two-hops apart detect an event at the same time, their reports may collide even though their receivers are different. Second, even for sensors not in the event area, collisions are inevitable as packets move toward the sink. Fig. 2(b) shows an example with a report tree. We assume that two reports are sent out. $Packet\_1$, $Packet\_3$ and $Packet\_5$ belong to the first report and $Packet\_2$, $Packet\_4$ and $Packet\_6$ belong to the second report. Without the aid of RTS/CTS, we can see that $Packet\_1$ could collide with $Packet\_2$ at sensor D, $Packet\_3$ and $Packet\_4$ could collide at sensor G, and $Packet\_5$ could collide with $Packet\_6$ at sensor I. Thus, the interference is serious for sensors in the event area, as well as those far away from the event area. We can see that designing a MAC protocol for event-driven WSNs without RTS/CTS is a challenging problem.

From the spatial correlation aspect, we argue that using inter-distance between sensors is insufficient to decide who shall report. A simple example is illustrated in Fig. 3(a), where sensor y is near the boundary of sensor x's correlation region. In CC-MAC (Vuran and Akyildiz, 2006), y has to report no matter whether it overhears x's report or not. We can see that the overlap area of x's and y's correlation regions is about 39% of one correlation region, which is high. A more sophisticated example is further shown in Fig. 3(b). Assuming that sensors a, b, c, and x have already reported, we consider two scenarios. First, if y does not overhear any of those reports (we can see that y is not in any of the transmission regions of a, b, c, and x), then y will report. However, y's report does not contribute any additional area to existing reports. Second, even if y can overhear x's report (this is possible when f forwards x's report), CC-MAC will enforce y to report, because the distance between x and y is larger than R_{corr}. Therefore, a more sophisticated report reduction scheme is desirable. In addition, because overhearing is opportunistic sometimes, the report reduction scheme should not highly rely on overhearing.

4 The Proposed Schedule-based Approach

4.1 Overview

We name our proposed schedule-based approach as SCMAC (a MAC protocol with Spatial Correlation consideration). We assume that a CSMA-based MAC protocol is adopted as the underlying MAC protocol. In order to solve the contention problem and the hidden terminal problem in the event areas, a schedule-based (or a TDMA-like) approach is proposed. However, for those sensors not in the event areas, because not all sensors need to help forward packets, assigning slots to those sensors that do not intend to transmit any packet is unnecessary and could increase delay. This means that the TDMA-like approach may not be suitable for sensors not in the event areas. Instead, they will take a CSMA-based approach. Therefore, SCMAC can be regarded as a hybrid TDMA/CSMA protocol.

In SCMAC, each node has two modes: event-source (ES) mode and non-event-source (NES) mode. Sensors in the NES mode will adopt the original CSMA-based protocol to reduce report latency. On the contrary, sensors in the ES mode will adopt a schedule-based approach to transmit packets. Issues involved in the ES mode include: (i) when to enter the ES mode, (ii) how to design a good slot assignment strategy based on the features of event-driven WSNs, (iii) how to determine the proper slot size, (iv) what will lead to synchronization error and how to conduct time synchronization, (v) how to exploit the spatial correlation of sensor data, and (vi) when to leave the ES mode.

To respond to these issues, SCMAC has the following important features: (i) The TDMA part will be started on-the-fly in an event-driven manner. Especially, no control message is required. (ii) In the TDMA part, a sensor can use the same slot with its one-hop neighbors but different slot from its two-hop neighbors. This actually violates the typical rule in TDMA slot assignment. (iii) The hidden terminal problem happened in the non-event area is relieved by prolonging the slot size in the TDMA part. (iv) We develop a simple synchronization scheme to overcome the synchronization error that may occur due to simultaneous detection of multiple events. The synchronization error caused by the event propagation delay will also be discussed. (v) A report reduction scheme is proposed in the TDMA part to address the spatial correlation of sensor data in an event area.

4.2 Operations in the ES Mode

4.2.1 Entering the ES Mode

To implement this TDMA-like protocol, we divide the time into slots at the link layer. When the network was deployed, a slot assignment algorithm will be run so that each sensor will be assigned a slot. Slots are numbered from 1 to *MAXSLOT*, and the first slot (i.e., slot 1) will be started on-the-fly in an event-driven manner.

Initially, each sensor is in the NES mode. When a sensor detects an event, it will enter the ES mode by starting slot 1 and wait the arrival of its own slot. (Note that the sensor does not need to advertise the mode change to other sensors. When multiple sensors detect an event simultaneously, they can synchronize with each other well.) Then, it will count slots until its slot arrives. If its slot arrives and it intends to transmit a packet, it will perform the access mechanisms defined in the underlying CSMA-based MAC protocol (such as the backoff and CCA mechanisms) as usual to access the channel. If a sensor cannot send its packet in its current slot, it will wait for its slot in the next cycle. Note that it is also

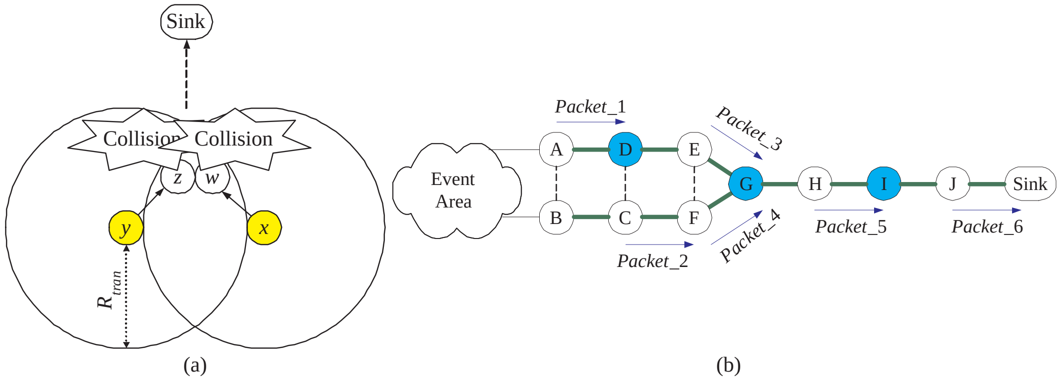

Figure 2 The hidden terminal problem, where R_{tran} denotes the transmission range of sensors.

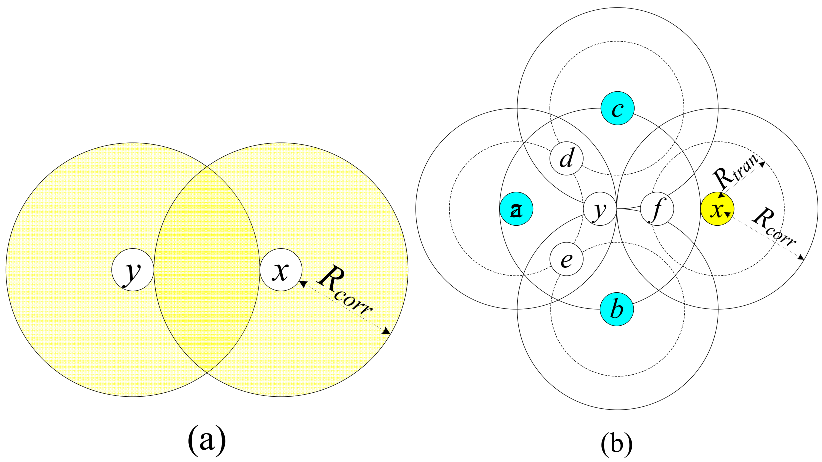

Figure 3 The redundancy problem.

possible that the sensor will suspend its packet and not retry again.

4.2.2 Slot Assignment Strategy

In order to reduce the value of *MAXSLOT*, we need a good slot assignment scheme rather than assigning slots randomly based on the IDs of nodes. Traditional slot assignment schemes will assign each node a slot different from those of its one-hop and two-hop neighbors to avoid interference (Rhee et al., 2009; Baoand and Garcia-Luna-Aceves, 2001). We argue that when the spatial correlation of sensor data is taken into consideration, such a strategy may not be efficient, because not every sensor needs to report. It is easy to see that if a node finally decides not to report, then the slot assigned to it will be wasted. Thus, we propose a new slot assignment strategy. Recall that our TDMA-like protocol is built on top of a CSMA-based protocol. Thus, assigning the same slot to neighboring nodes does not necessarily lead to collision. When neighboring nodes intend to report and share the same slot, the backoff mechanism can be used to determine who can report and the losers may suspend their reports due to the spatial correlation of sensor data when they overhear the packet sent by the winner. Therefore, we intentionally assign a slot used by a node's one-hop neighbors to that node, but the node still should be assigned a slot different from those used by its two-hop neighbors to avoid the hidden terminal problem (recall that we do not adopt the RTS/CST mechanism). Another advantage of our proposed slot assignment strategy is that the value of *MAXSLOT* required can be reduced significantly.

To verify the efficiency of the proposed slot assignment strategy, a simple simulation is conducted. 4096 sensors are randomly deployed in a 256×256 field with uniform distribution. As we can see in Fig. 4, when the transmission range of senors increases, the value of *MAXSLOT* increases from 45 to 301 when a node needs to have a slot different from those used by its one-hop and two-hop neighbors. (Note that our slot assignment algorithm that will be described later can be easily modified to support this strategy.) However, the value of *MAXSLOT* only increases from 14 to 23 when a sensor only needs to differentiate from its two-hop neighbors. Note that a lower *MAXSLOT* means a lower report latency.

We develop a simple distributed slot assignment algorithm to complement this slot assignment strategy. We make some assumptions. First, the network topology is static. Second, each sensor has a unique ID. Third, each sensor can correctly discover all its one-hop and two-hop neighbors. Finally, the number of two-hop neighbors of a node is finite.

Each sensor will maintain a *slot usage table* to record the slots used by its one-hop and two-hop neighbors. Each sensor that does not own a slot will select its slot in a distributed way. Thus, we only describe the behavior of a sensor x. First, x will

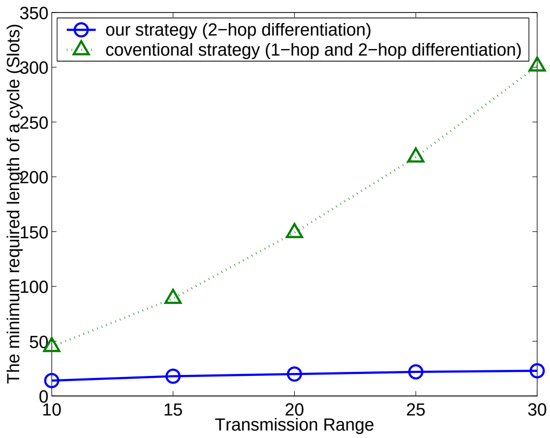

Figure 4 Comparison of slot assignment strategies.

send a $request$ to all of its two-hop neighbors. Any two-hop neighbor y of x receiving the $request$ will act as follows.

- If y does not own a slot yet and y.ID $< x$.ID, then y will reply a $grant$ to x. Because y does not own a slot yet, no slot information will be carried on the $grant$.
- Otherwise, y will do nothing.

Once x receives $grant$s from all of its two-hop neighbors, x will select a slot by the following rule. To begin with, x will check whether there exists a slot such that this slot has been assigned to x's one-hop neighbors but has not been assigned to x's two-hop neighbors. If such slots exist, x will pick up the most-used one among those slots. (Recall that we will intentionally assign a slot used by a node's one-hop neighbors to that node.) Otherwise, x will select the smallest slot that has not been used by its two-hop neighbors. (The reason is to minimize *MAXSLOT*.) After selecting its own slot, x will send a $grant$ with its selected slot to each of its two-hop neighbors. (Note that this $grant$ has two functions. The first one is to enforce x's one-hop and two-hop neighbors to modify their slot usage table. The second one is to make x's two-hop neighbors get the needed $grant$s.) Then, when another node z which may be x's one-hop or two-hop neighbor receives such a $grant$ on which a selected slot is carried, z will modify its slot usage table accordingly and check whether it has received all grants from all of its two-hop neighbors.

Finally, when a sensor determines its slot, it will notify the sink so that the sink can determine the value of *MAXSLOT*. Then the sink will announce *MAXSLOT* to all sensors. Note that when no packet loss occurs, the proposed slot assignment algorithm guarantees that each node will receive $grants$ from all of its two-hop neighbors. However, when packet loss cannot be avoided, a sensor may lose $grant$s and this will result in deadlock. In order to overcome the packet loss problem, a sensor can actively ask its two-hop neighbors to resend their $grant$s (if allowed) when it waits passively for a long period.

Note that the proposed slot assignment algorithm is run only once when sensors are first deployed. However, sometimes it will be rerun for some reasons such as topology change, load balance, and so on. We assume that the frequency of rerunning the slot assignment algorithm is low so that we can ignore the cost paid for slot assignment.

Theorem 1: *The proposed slot assignment algorithm ensures that each sensor will eventually select a slot different from those used by its two-hop neighbors.*

Proof: For simplicity, we assume that packet loss will not occur. Because each sensor has the same behavior, we only consider a sensor, say x. We assume that x has n two-hop neighbors and the ID of x is the k-th largest one among these $n+1$ sensors, where $1 \leq k \leq n+1$. We will show that x can select a slot different from those used by all of its two-hop neighbors no matter what the value of k is.

We consider two cases. In the first case, we assume $k = 1$. In this case, when x sends a $request$ to all of its two-hop neighbors, each of x's two-hop neighbors will reply a $grant$ to x, because x has the largest ID. Thus, x will choose slot 1 (i.e., the smallest slot) to use. It is easy to see that all of x's two-hop neighbors cannot select their own slots at this time because they do not receive x's grant yet. Then, x will send a $grant$ with the slot number selected by x to all of its two-hop neighbors; thus all of x's two-hop neighbors will not select slot 1 to use. Finally, we can see that the $grant$ sent by x will make some x's two-hop neighbors whose IDs are the second largest among their two-hop neighbors get all required $grant$s and start to select their own slots.

Now we consider the second case. In this case, we assume $1 < k \leq n+1$. In this case, when x sends a $request$ to all of its two-hop neighbors, those x's two-hop neighbors whose IDs are smaller than $x.ID$ will send a $grant$ to x. However, those x's two-hop neighbors whose IDs are larger than $x.ID$ will send a $grant$ to x only when they have determined their slots. Thus, x will not use the same slot with them. (Note that it is possible that two of x's two-hop neighbors use the same slot if these two nodes are not two-hop neighbors with each other.) After x selects its slot, x will send a $grant$ with the slot number selected by x to all of its two-hop neighbors; thus those x's two-hop neighbors whose IDs are smaller than $x.ID$ will not select the same slot with x. Finally, we can see that the $grant$ sent by x may make some sensors get all required $grant$s and start to select their own slots. □

4.2.3 Slot Size

In conventional TDMA protocols, the slot size is usually set to the maximum one-way message delay denoted by d. (Note that in our approach, d should include the maximum backoff delay, the time to perform CCA, the transmission time, and so on.) Below, we will show that the hidden terminal problem could be alleviated by prolonging the slot size. Before that, we define a term *flow*.

Definition 4.1: Consider any sensor x that is in the ES mode and located at the event boundary. When x transmits a packet to a neighboring sensor that is in the NES mode, we say a *flow* is generated.

In the proposed approach, the slot size is set to $\ell \times d$, where ℓ is a real number larger than or equal to 1. Note that ℓ

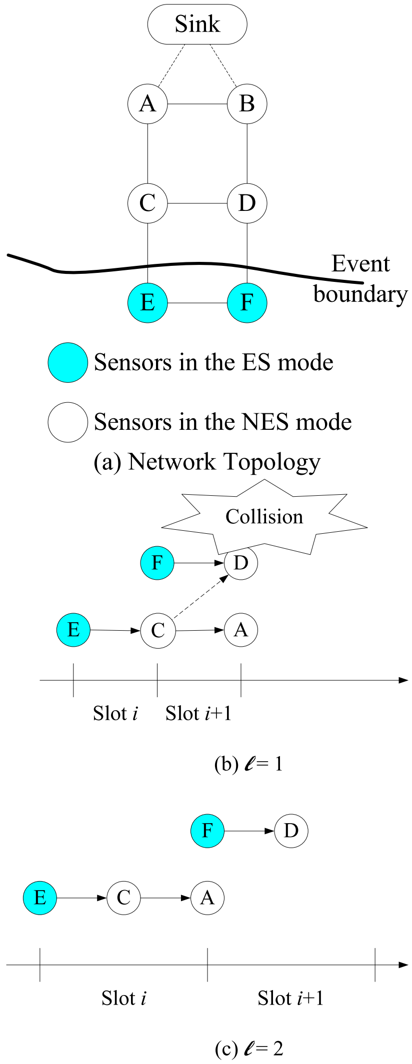

Figure 5 The impact of slot size beyond event areas.

is usually set to be 1 in most TDMA-based protocols. Fig. 5 shows an example of the advantage of $\ell > 1$. Fig. 5(a) shows the network topology used in this example, where E and F are in the ES mode and their assigned slots are i and $i+1$ respectively. Suppose that in some cycle E generates a flow in slot i and F generates a flow in slot $i+1$. Besides, we assume that the packet sent by E will flow along the path E-C-A-Sink and the the packet sent by F will flow along the path F-D-B-Sink. When C receives the packet sent by E, it will run a CSMA-based protocol immediately to forward the packet because it is in the NES mode. Fig. 5(b) shows the case of $\ell = 1$, where the transmission of F could easily collide with the transmission of C at D due to the hidden terminal problem. However, as Fig. 5(c) shows, if we set $\ell = 2$, the transmission of C will occur within slot i, thus avoiding the hidden terminal problem.

From the above example, we can see that the time interval between two neighboring sensors sending out their flows becomes larger when the slot size becomes larger. That means that prolonging the slot size can separate flows. The advantage can be illustrated by Fig. 2(b), where we assume that $Pacpet\_1$, $Packet\_3$, and $Packet\_5$ belong to flow 1 (i.e., they are the same packet), and $Packet\_2$, $Packet\_4$, and $Packet\_6$ belong to flow 2. When these two flows are sent out at different time, the hidden terminal problem in the non-event area can be avoided.

4.2.4 Synchronization

Time synchronization should be done in a strict way in conventional TDMA-based protocols. However, tight clock synchronization is not required in our protocol, because our TDMA-like protocol is built on top of a CSMA-based MAC protocol. This means that the backoff scheme and the CCA (Clear Channel Assessment) scheme can remove most of the collisions caused by synchronization error.

In our scheme, sensors are assumed to be synchronized by the occurrence of events, which trigger them to enter the ES mode. This scheme has two problems that may lead to synchronization error.

- Sensors may not enter ES-Mode simultaneously due to the event propagation delay.
- When multiple events occur close in time and space, some sensors may detect multiple events. In our scheme, when a sensor in the ES mode detects another event, we will allow it to continue its slot counting, instead of resetting to slot 1. On the contrary, some sensor may only detect one event and enter slot 1. This will also lead to synchronization error.

In our scheme, we do not consider the synchronization error caused by the event propagation delay, because the backoff and CCA mechanisms can alleviate the collision problem. For the multi-event problem, we propose a simple adjustment scheme. We assume that each sensor will count how many slots have passed after it entered the ES mode. This counter is denoted by s. (Note that s will be set to 0 whenever a node leaves the ES mode.) When a sensor x transmits a packet in the ES mode, the counter $x.s$ will be carried in the packet. Each of x's neighbor sensors, say y, that overhears the packet will react as follows. (Note that when y receives the packet, it can easily know the packet is transmitted by x in the $((x.s-1)(\text{mod } \mathit{MAXSLOT})+1)$-th slot of a cycle.)

- If $y.s >= x.s$, then y will do nothing.
- If $y.s < x.s$, then y will adjust its current slot to $((x.s-1)(\text{mod } \mathit{MAXSLOT})+1)$, and set $y.s$ to $x.s$.

With our adjustment scheme, when multiple events occur close in time and space, the sensors that detect the earliest event will dominate the clock in the ES mode. Although collisions could occur during the adjusting, the backoff and CCA mechanisms and the design of longer slot size can alleviate the collision problem. With this loose synchronization scheme, our scheme can handle the multi-event problem easily.

Finally, one should note that we cannot use the slot number assigned to sensors to correct the synchronization error. For example, in Fig. 6, when B overhears the packet transmitted by A, it will switch to slot 10. Later on, when B overhears the packet transmitted by C, it will switch back to

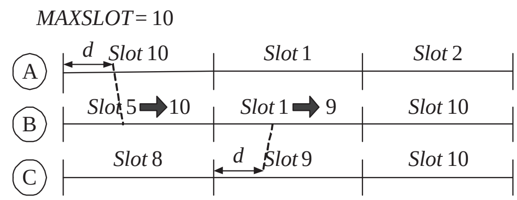

Figure 6 The synchronization problem.

slot 9. Thus, the counters rather than the slot number should be used in the adjustment scheme.

4.2.5 Exploiting the Spatial Correlation of Sensor Data

So far, we mainly focus on the medium access issue. Next, we discuss how to exploit the spatial correlation of sensor data. We assume that a correlation radius R_{corr} is given by applications and our goal is to minimize redundant reports under distortion constraints. We propose a report reduction scheme that provides two advantages. First, by exploiting TDMA's features, we adopt a probability to reduce redundant reports without the aid of overhearing. Second, compared to CC-MAC, when a sensor overhears a packet, the area of the overlap of correlation regions will be taken into consideration. (Note that a report reduction scheme is required, especially when $R_{corr} > R_{tran}$, because the number of sensors should be large enough to ensure network connectivity and most of them do not need to report.)

In our report reduction scheme, we assume that when an event is detected, the sensor that detects the event will generate a packet (report) in its queue. However, this report may not be sent eventually. The report reduction scheme consists of three steps. The first step is executed when a sensor detects an event. The sensor will use a probability to determine whether it should report or not (note that if the sensor decides not to report, it still needs to enter the ES mode). Then, it will enter the second step, during which it will try to overhear others' packets before its slot arrives. With overhearing, some reports can be further discarded in this step. Finally, when the sensor's slot arrives, it will enter the third step in which it will transmit one of the packets in its buffer, if any. The details of this scheme are described as follows.

Step 1: Because overhearing is opportunistic or even impossible sometimes, it is hard for a sensor to collect enough information to judge whether it should report or not. Thus, a probability is adopted to help sensors to decide whether they should report or not. In our scheme, when a sensor detects an event, it will report this event with a probability α^{S-1}, where $0 < \alpha \leq 1$ and S is the slot number assigned to that sensor. This means that a sensor with larger slot number may tend to not report. To motivate this design, let's reconsider the example shown in Fig. 3(b), where we assume that sensors a, b, c, and x have already reported, but sensor y does not report yet, because y's slot number is larger than the others. Recall that we have shown that y's report is redundant. Based on this observation, sensors with larger slot number should tend to not report, because their neighbors may have reported. Note that this step can reduce redundant reports without the aid of overhearing. Note that a sensor with larger slot number needs to help forward reports generated by others; thus, the unbalance problem in terms of energy consumption will not be serious.

An issue is determining a proper α, which is affected by many factors, for example, the ratio of R_{corr} to R_{tran} and network density. It is a challenge to determine an optimal α. To simplify this problem, we suggest that α should be an adjustable parameter so that the sink can adjust the value of α based on the received reports. Given a distortion constraint, when the sink receives many redundant reports, it can decrease the value of α and announce the new value of α to sensors. (Note that although the sink cannot know the exact boundary of event area, it can compute the overlap area of correlation regions to judge the redundancy level based on R_{corr} and the locations of reporting sensors.) Besides, when the distortion is the major concern, the sink can just set $\alpha = 1$.

Step 2: Then, we describe the second step. To begin with, we define the *reporter* of a packet. The reporter of a packet is the sensor that first initiates this report packet by detecting an event. (Note that the sender of a packet may not be the reporter of that packet.)

The procedure in the second step works as follows. A sensor x will try to overhear others' packets. When x overhears a packet (whose reporter is denoted by $r_{received}$), x will check all of the packets in its buffer. We assume the reporters of the packets in x's buffer are denotes by $\{r_1, r_2, \ldots, r_k\}$, where k is the number of packets in x's buffer. According to the distance between $r_{received}$ and r_i, where $i = 1, \ldots, k$, three cases are considered separately:

- If the distance is smaller than R_{corr}, then the packet reported by r_i will be removed from x's buffer.
- From Fig. 3(a), we can observe that when the distance between two reports is smaller than $2 \times R_{corr}$, their correlation regions will overlap. Based on this observation, when the distance between $r_{received}$ and r_i is larger than or equal to R_{corr} but smaller than $2 \times R_{corr}$, x will determine whether the packet reported by r_i should be removed or not with a probability $INTC(d)/\pi(R_{corr})^2$, where d denotes the distance between $r_{received}$ and r_i and $INTC(d)$

is the intersection area of the two circles centered at $r_{received}$ and r_i. $INTC(d)$ can further be represented by $4\int_{d/2}^{R_{corr}} \sqrt{(R_{corr})^2 - x^2}dx$. (Note that $INTC(d)$ can be calculated in advance and only needs to be calculated once.)

- If the distance is larger than or equal to $2 \times R_{corr}$, nothing will be done.

Two notes regarding the second step should be addressed. First, the procedure in step 2 will also be run on the sensors that are in the NES mode. More precisely, the procedure in step 2 will be executed whenever a sensor overhears a packet. Second, compared to CC-MAC, we argue that our TDMA-based design can increase the opportunity of overhearing inherently, because a sensor has to wait until its own slot arrives. During the waiting period, the sensor could overhear a packet from other sensors and suspend its report.

Step 3: When the sensor's slot arrives, it will enter the third step in which it will transmit one of the packets in its buffer, if any.

Finally, one should note that different correlation models may have different design of exploiting spatial correlation. The objective of this section is presenting the idea of exploiting TDMA's features to reduce redundant reports.

4.2.6 Leaving the ES Mode

The final issue is how long a sensor should stay in the ES mode. We suggest that an ES mode sensor can return to the NES mode when it does not have any packet in its buffer and does not receive/overhear any packet during a cycle.

4.3 Operations in the NES Mode

Sensors in the NES mode will adopt a CSMA-based protocol to reduce the report latency. Note that collision and congestion problems for sensors in the NES mode have been relieved by separating flows in distance.

4.4 Extension for Achieving Energy Efficiency

Due to the power constraint of sensor nodes, energy efficiency is also an important issue for WSNs. As mentioned above, most CSMA-based MAC protocols can be adopted as the underlying MAC protocol. Thus, the energy efficiency can be done by the underlying MAC protocol. Below, we will use B-MAC (Polastre et al., 2004) as our choice and show how to utilize its LPL (Low Power Listening) technique to achieve energy efficiency.

In LPL, a sensor normally stays in the sleep state and wakes up periodically. When a sensor wakes up, it will turn on its radio for a very short duration and check for any activity. If a preamble is detected, the sensor will stay awake to capture the incoming packet. Since nodes are not synchronized and thus wake up at different times, the preambles of data packets should be longer than the check interval of sleeping nodes to ensure that sleeping nodes will not miss incoming packets. More details of LPL can be found in B-MAC (Polastre et al., 2004).

Fig. 7 shows an example of how to combine SCMAC with LPL technique. Three notes should be addressed. First, both the TDMA-like protocol and the periodical wake-up scheme need timers. These two timers should be run independently. Second, the maximum one-way message delay (i.e., d) should include the preamble length. Third, recall that CCA needs to be run before any transmission. If the CCA outlier algorithm observes that the channel is not clear, the sensor should switch to the receive mode instead of going back to sleep. The reason is that the step 2 of the proposed report reduction scheme depends on overhearing for inhibiting reporting and increasing data fusion opportunity.

5 Simulation Results

We have developed a simulator to demonstrate the efficiency of SCMAC. A sensing field with size 256×256 units where 4096 sensors are deployed randomly with uniform distribution is simulated. The sensor with ID 0 is selected to be the sink. An event generation model used to simulate the events arising in the network is proposed. In order to verify the proposed loose synchronization scheme, we assume that the event could spread and multiple events could occur simultaneously. In this model, we use four parameters to control the generation of events:

- ***MAX_INTERVAL***: This parameter defines the maximum time interval between two events.
- ***WIDTH*** and ***MAX_LEVEL***: In our model, an event area is represented by multiple concentric circles. The number of concentric circles is determined by ***MAX_LEVEL***. The first circle is the one with radius ***WIDTH***, the second circle is the one with radius $2 \times$ ***WIDTH***, and so on.
- ***PROPAGATION_DELAY***: This parameter is used to simulate the event propagation delay. When an event occurs, if sensors in the i-th annulus of the event area detect this event at t_i, then sensors in the $i+1$-th annulus will detect this event at $t_i +$ ***PROPAGATION_DELAY***.

Now, we describe the procedure of this event generation model. The first event will be triggered at the beginning of simulation. As we mentioned above, an event area is represented by multiple concentric circles. Therefore, a point in the sending field will be selected randomly as the center of those circles. Sensors in the first circle will detect this event first. Then, after ***PROPAGATION_DELAY***, sensors in the second annulus will also detect this event. This detection procedure will continue until sensors in the ***MAX_LEVEL***-th annulus detect this event. Finally, when an event e_j arises initially, the next event (i.e., event e_{j+1}) will also be triggered after t, where $0 \leq t \leq$ ***MAX_INTERVAL*** and t is determined randomly with uniform distribution. Note that when ***PROPAGATION_DELAY*** is equal to zero, this means that the event will not spread, and when t is zero, this means that two events will occur at the same time.

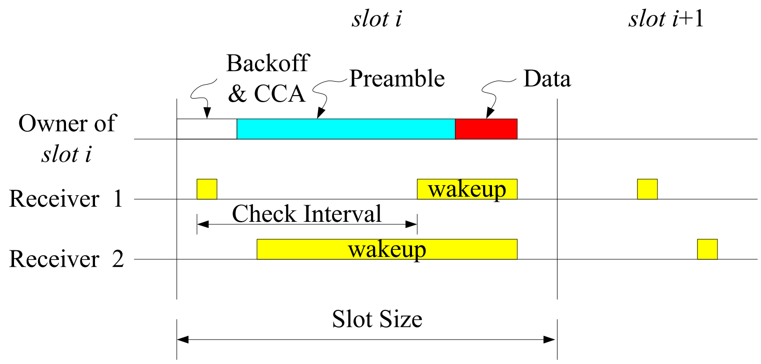

Figure 7 Combining SCMAC with LPL. In this example, $\ell = 1$.

Table 1 The default values of parameters used in the simulation.

Buffer Size	10
The length of DATA	30 Bytes
Bit rate	250 kb/s
Simulation Time	1 hour
R_{tran}	10 units
MAX_INTERVAL	10 seconds
WIDTH	10 units
PROPAGATION_DELAY	5 milliseconds
MAX_LEVEL	5
ℓ	1.5

Three metrics are used to evaluate the performance of medium access schemes. We count *the number of packets transmitted*. Usually, fewer packets means that sensors can stay in sleep mode longer. Thus, less energy is consumed. We also measure *the success rate of packet transmission* defined as the ratio of the number of packets received by the intended receiver to the number of packets transmitted by the sender. Success rate can be used to evaluate the efficiency of a MAC protocol. Higher success rate means less collision. *Average delay* is defined as the average delay of packets sent by ES mode sensors and received by the sink. Besides, two metrics are used to evaluate the performance of report reduction schemes. *Coverage* is defined as $A_{corr\_reg\_union}/A_{event\_area}$, where $A_{corr\_reg\_union}$ denotes the area of the field united by the correlation regions of reporters whose reports are received by the sink, and A_{event_area} is the area of event area. Higher coverage means that the sink has more accurate information regarding events. Finally, for an unit area in the event area, if it is covered by n sensors' correlation regions, where $n > 1$, then we define that the *redundancy* of that unit area is $(n - 1) \times 100\%$.

In Sec. 5.1, we compare SCMAC with a CSMA-based protocol. Also, two report reduction schemes and two slot assignment schemes will be applied separately. The detail will be described later. In Sec. 5.2, we further investigate the impact of two parameters used in SCMAC, that is, α and ℓ. The related parameters and their default values used in the simulation are shown in Table 1.

5.1 *Evaluation of SCMAC*

In this section, we compare SCMAC with several schemes. In the CSMA scheme, a CSMA-based MAC protocol without any spatial correlation consideration is adopted. In the CSMA-SSC (a CSMA-based protocol with Simple Spatial Correlation consideration) scheme, a CSMA-based MAC protocol with a simple report reduction scheme is adopted. This report reduction scheme works as that used in CC-MAC does. More precisely, when a node, say x, overhears a packet whose reporter is y, x will judge whether the distance between itself and y is smaller than correlation radius or not. If the answer is affirmative, x will suspend its report. Otherwise, x will continue its report. In the SCMAC-SSC-TSA scheme, the aforementioned report reduction scheme and our proposed schedule-based approach will be adopted; however, the Traditional Slot Assignment strategy (i.e., a node needs to own a slot different from those used by all of its one-hop and two-hop neighbors) is used. (Note that the values of *MAXSLOT* are 14 and 45 in the proposed slot assignment scheme and the TSA scheme respectively.) In the SCMAC-TSA scheme, our proposed schedule-based approach with the traditional slot assignment strategy is adopted. In the SCMAC-SSC, our proposed schedule-based approach with the simple report reduction scheme is adopted. Finally, one should note that the RTS/CTS mechanism is not used in all schemes, and the acknowledgement scheme is also disabled in the simulation.

Fig. 8 shows the results of the case where $R_{corr} > R_{tran}$. To begin with, we focus on the CSMA and CSMA-SSC schemes. Although they have the best performance in terms of average delay, they have the worst performance in terms of coverage, especially when the event area becomes larger. The reason is high contention and collision that make the sink receive fewer reports than expected. This can be further verified by Fig. 8(b). We can see that the success rate is low when the CSMA and the CSMA-SSC schemes are adopted.

In order to show the efficiency of our proposed report reduction scheme, we then focus on the SCMAC-SSC-TSA and SCMAC-SSC schemes. In Fig. 8(a), we can see that both of them transmit many packets in the network. However, in Fig. 8(d), we can observe that the SCMAC scheme can achieve the same coverage performance as them by transmitting fewer packets. The reason can be explained by Fig. 8(e). We can see that redundant reports are too much when these two schemes are applied. This means that the simple report reduction scheme does not perform well enough. In addition, because many reports have to be sent, these two schemes also do not perform well in terms of average delay.

Finally, in order to show the efficiency of our proposed slot assignment strategy, we focus on the SCMAC-TSA and SCMAC schemes. In fact, it is hard to compare these two schemes fairly. Although both of them set α to be 0.5, different slot assignment strategies may result in different performances. (From Fig. 8(c), it can be seen that in the SCMAC-TSA scheme, α should be higher to achieve better coverage performance because the average value of S is larger in the SCMAC-TSA scheme.) However, it is not hard to see that the SCMAC scheme has better performance in terms of average delay. This demonstrates that the proposed slot assignment strategy can reduce the report latency (this is because the value of *MAXSLOT* is reduced). Note that when SCMAC is combined with the LPL technique, the slot size will become larger and the improvement of report latency will become more notable. Briefly, our proposed SCMAC scheme has the best performance. It provides high success rate, high coverage, low redundancy, reasonable average delay, and reasonable amount of packets.

Fig. 9 shows the results of the case where $R_{corr} < R_{tran}$. In this case, we observe that the coverage will be low when α is set to be smaller than 1.0. Thus, α is set to be 1.0. We can see that the SCMAC scheme is still the best one. In addition, we can further note that the advantage of our proposed slot assignment strategy is revealed thoroughly in this experiment. In Fig. 9(c), we can see that our proposed slot assignment strategy can reduce the average delay. This also influences the coverage. The reason will be explained below. In Fig. 9(d), we can see that the coverage will become lower when the event area becomes larger. One reason is buffer overflow (note that we assume that each sensor's sending buffer is limited such that for a sensor, if there are too many packets to be sent simultaneously, some of packets will be discarded), because more report packets have to be sent when the event area becomes larger. Long delay will worsen the buffer overflow problem, because packets will be queued in a sensor for a long time. Thus, the performance of the SCMAC and SCMAC-SSC schemes is better than that of the SCMAC-SSC-TSA and SCMAC-TSA schemes.

To conclude, the results and advantages of our proposed SCMAC scheme can be summarized as follows:

- By comparing SCMAC with CSMA-SSC, the results show that our proposed medium access scheme can relieve the collision problem without the aid of RTS/CTS mechanism. This can be verified by high success rate.
- By comparing SCMAC with SCMAC-SSC, the results show that our proposed report reduction scheme can reduce more redundant reports than the simple report reduction scheme does.
- By comparing SCMAC with SCMAC-TSA, the results show that our proposed slot assignment strategy can shorten the average delay. This also relieves the buffer overflow problem.

5.2 Evaluation of Parameters in SCMAC

In this section, we further explore the impact of two parameters used in our proposed approach, that is, α and ℓ. α is introduced in Sec. 4.2.5 and used to control the redundancy level via setting appropriate reporting probabilities. ℓ is introduced in Sec. 4.2.3 and used to alleviate the contention problem via prolonging the slot size. To begin with, α is inspected. From Sec. 5.1, we can draw two conclusions regarding α. First, the optimal value of α depends on the slot distribution. For example, when the traditional slot assignment strategy is adopted, the slot numbers owned by nodes may be large. Thus, higher α should be used to achieve reasonable coverage. Second, when $R_{corr} < R_{tran}$, α should be set to be 1 in order to achieve reasonable coverage, because α is mainly used in the situation where redundant reports cannot be removed by overhearing.

In this section, we further investigate the impact of the ratio of R_{corr} to R_{tran} on the value of α. Fig. 10 shows the results where $N1/N2$ denotes that R_{corr} is $N1$ units and R_{tran} is $N2$ units. From Fig. 10(a), we can see that when the ratio of R_{corr} to R_{tran} increases from 1.0 to 2.0, we can use lower α to achieve enough coverage. The reason is that when R_{corr}/R_{tran} becomes larger, the probability that two sensors that are within each other's correlation region and cannot overhear each other's packets directly becomes larger. Thus, the value of α should be small in order to reduce redundant packets. Then, we can note that when R_{corr}/R_{tran} is fixed, larger R_{tran} can lead to better coverage, because more sensors can overhear the reports. This also implies that a smaller value of α should be used when the network density is large. However, from Fig. 10(b), we observe that when the R_{tran} becomes larger, the redundancy is high even a small value of α is used. Thus, we suggest that R_{tran} should not be too large and we can increase the value of α to achieve required coverage. As we mentioned in Sec. 4.2.5, we suggest that α should be an adjustable parameter so that the sink can adjust the value of α based on the received reports. These simulation results can give the system administrator idea how to adjust the value of α.

Next, ℓ is investigated. In Fig. 11(a), we can see that prolonging the slot size can increase the success rate, because the hidden terminal problem is alleviated. Although the increment is small, the improvement will be large when the amount of packets transmitted becomes large. In addition, in Fig. 11(b), it can be seen that prolonging the slot size can also enhance coverage, because more packets are transmitted to the sink successfully. However, in Fig. 11, we can see that prolonging the slot size will be penalized by long delay. Therefore, determining a proper value of ℓ may depend on application requirement.

When the network load is high, prolonging the slot size may not be a good idea, because long delay will worsen the buffer overflow problem. As we can see in Fig. 12, although the success rate becomes higher when ℓ becomes larger, the coverage becomes lower, because many packets are dropped due to buffer overflow. Thus, the network load also should be taken into consideration when we need to decide the proper value of ℓ. (Note that the success rate of packet transmission

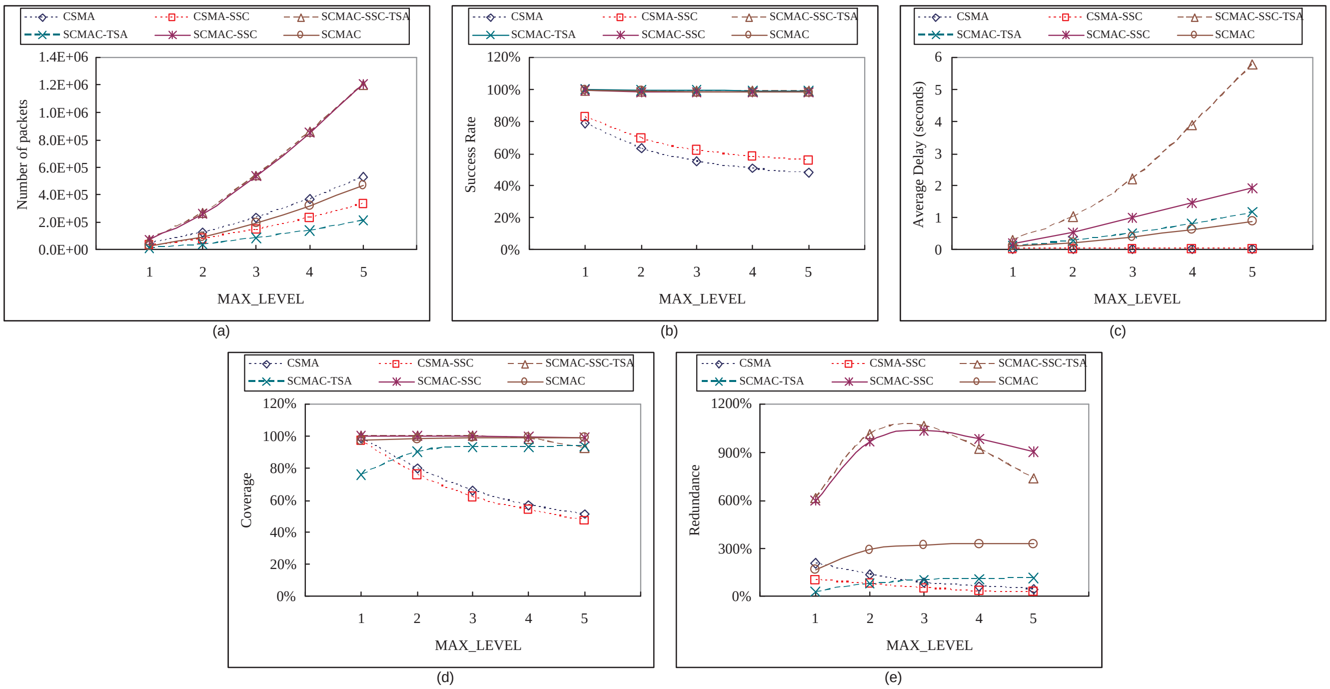

Figure 8 Comparison of different schemes, where R_{corr} is 15 units and R_{tran} is 10 units. In the SCMAC-TSA and the SCMAC schemes, α is set to be 0.5.

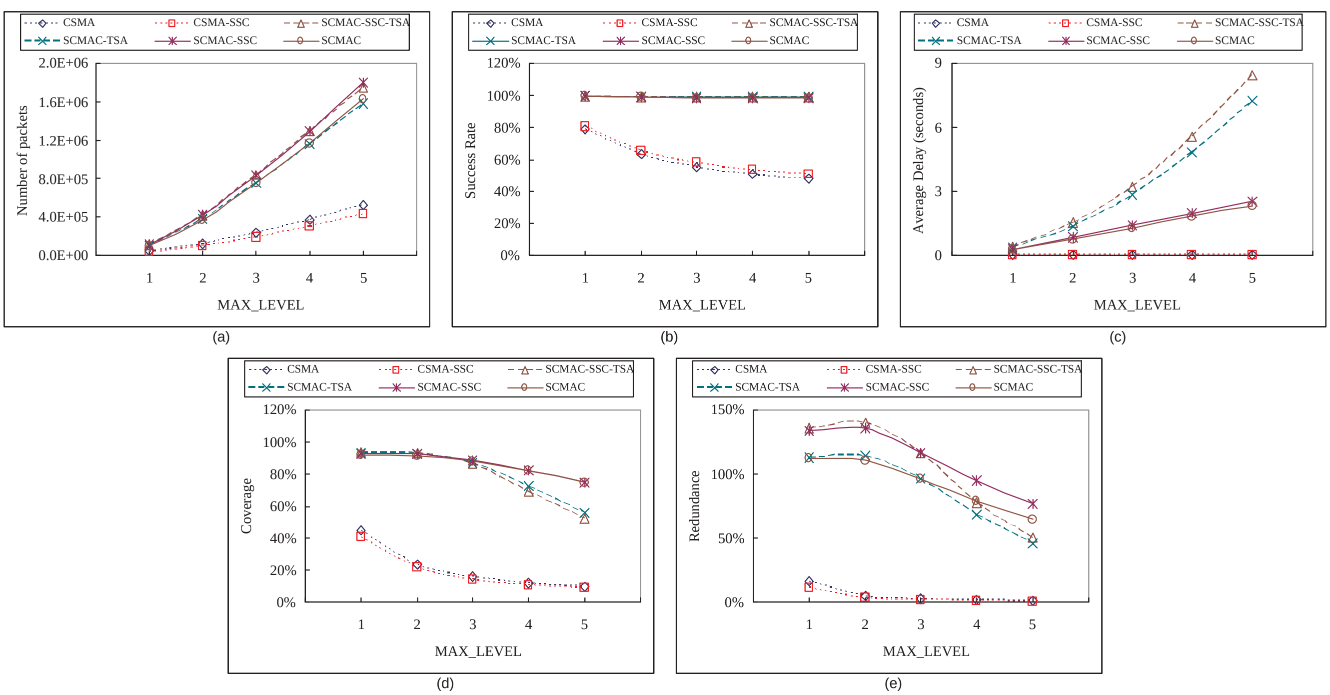

Figure 9 Comparison of different schemes, where R_{corr} is 5 units and R_{tran} is 10 units. In the SCMAC-TSA and the SCMAC schemes, α is set to be 1.0.

is defined as the ratio of the number of packets received by the intended receiver to the number of packets transmitted by the sender, and does not take the dropped packets into consideration. Thus, the success rate shown in Fig. 12(a) is still high.)

6 Conclusions

We have shown how to exploit the spatial correlation of sensor data at the link layer for event-driven WSNs. A hybrid TDMA/CSMA protocol is proposed. The protocol has three major features.

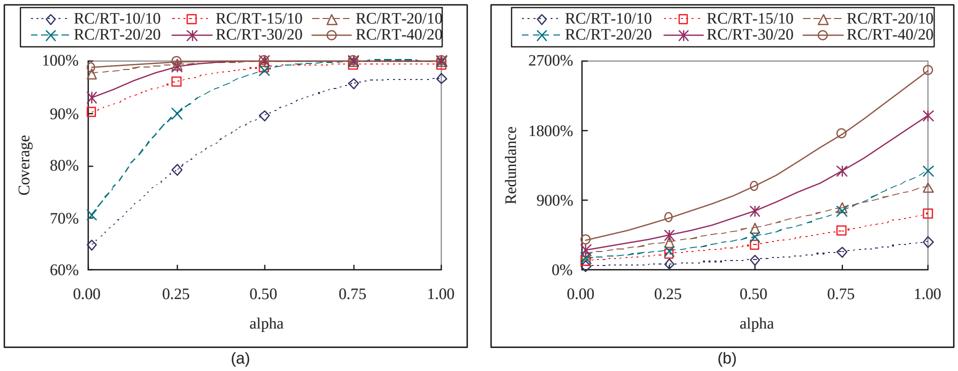

Figure 10 The impact of α under different ratios of R_{corr} to R_{tran}. The set of values of α is $\{0.01, 0.25, 0.50, 0.75, 1.00\}$.

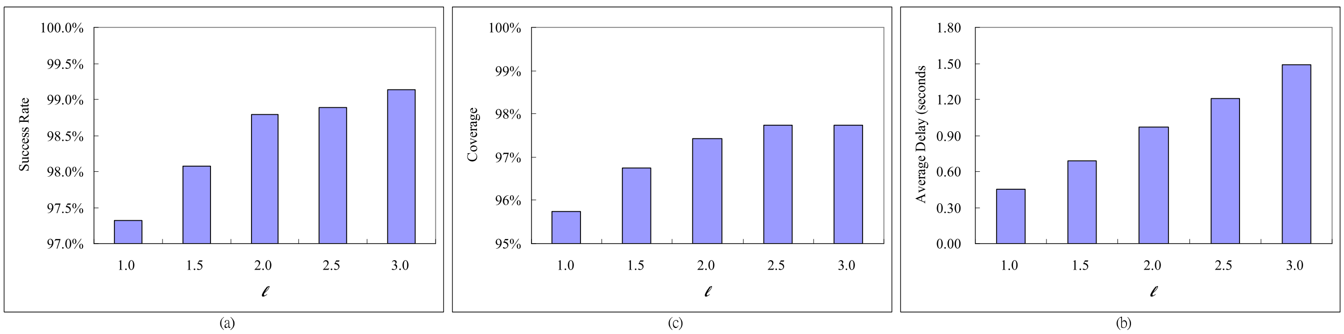

Figure 11 The impact of ℓ. R_{corr} is 15 units, R_{tran} is 10 units, and the value of α is 0.3.

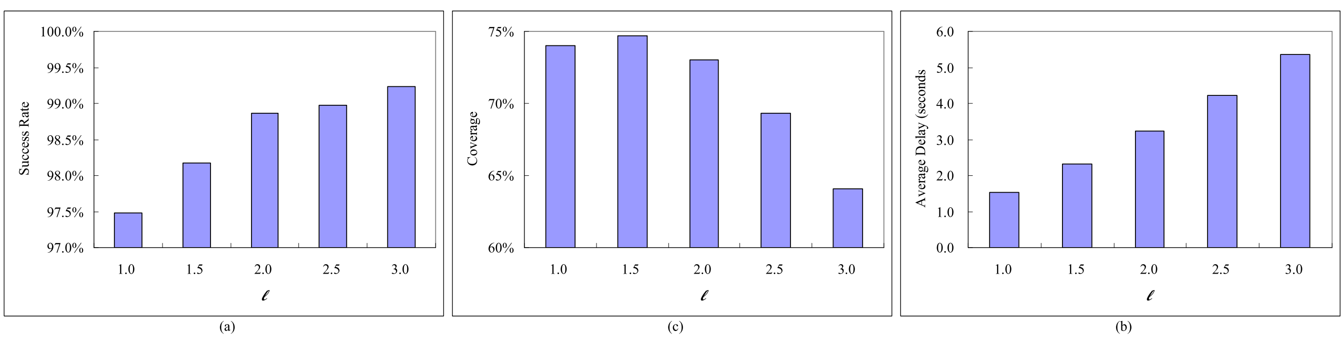

Figure 12 The impact of ℓ. R_{corr} is 5 units, R_{tran} is 10 units, and the value of α is 1.0.

- First, the TDMA part is triggered only when sensors detect an event. By doing so, the protocol enjoys the benefits of collision-free transmission of TDMA and low latency transmission of CSMA.
- Second, the slot assignment strategy associated with the TDMA part takes the spatial correlation of sensor data into consideration. By intentionally allowing one-hop neighbors to share the same time slot, the number of slots required is significantly reduced. Thus, the transmission latency is also reduced.
- Third, by intentionally enlarging the slot size, packets are separated spatially and thus the interference problem in the non-event area is alleviated.

Previous work has shown that different MAC schemes should be used in event-source and non-event-source areas. The major contribution of this paper is proposing a hybrid TDMA/CSMA link layer protocol for event-driven WSNs. In addition, the proposed solution can adapt to large and small event areas. Simulation results also demonstrated the efficiency of our scheme. However, our proposed solution does not take SNR into consideration. In the future, we will investigate how to enhance our protocol by considering SNR.

References

Ahn, G.-S., Miluzzo, E., Campbell, A. T., Hong, S. G., and Cuomo, F. (2006). Funneling-MAC: A localized, sink-oriented MAC for boosting fidelity in sensor networks. In *Proc. of SenSys'06*, pages 293–306, Boulder, Colorado, USA. ACM, New York, NY, USA.

Baoand, L. and Garcia-Luna-Aceves, J. J. (2001). A new approach to channel access scheduling for ad hoc networks. In *Proc. of MobiCom'01*, pages 210–221, Rome, Italy. ACM, New York, NY, USA.

Dong, M., Tong, L., and Sadler, B. M. (2004). Effect of MAC design on source estimation in dense sensor networks. In *Proc. of ICASSP'04*, pages 853–856, Montreal, Canada. IEEE.

Dong, M., Tong, L., and Sadler, B. M. (2006). Impact of data retrieval pattern on homogeneous signal field reconstruction in dense sensor networks. *IEEE Transactions on Signal Processing*, 54(11):4352–4364.

Hohlt, B., Doherty, L., and Brewer, E. (2004). Flexible power scheduling for sensor networks. In *Proc. of IPSN'04*, pages 205–214, Berkeley, California, USA. ACM, New York, NY, USA.

Jamieson, K., Balakrishnan, H., and Tay, Y. (2006). Sift: A MAC protocol for event-driven wireless sensor networks. In *Proc. of EWSN'06*, pages 260–275, Zurich, Switzerland. Springer.

Lee, W. L., Datta, A., and Cardell-Oliver, R. (2006). FlexiMAC: A flexible TDMA-based MAC protocol for fault-tolerant and energy-efficient wireless sensor networks. In *Proc. of ICON'06*, pages 1–6, Singapore.

Pan, M.-S. and Tseng, Y.-C. (2008). Quick convergecast in ZigBee/IEEE 802.15.4 tree-based wireless sensor networks. *Computer Communications*, 31(5):999–1011.

Pattem, S., Krishnmachari, B., and Govindan, R. (2004). The impact of spatial correlation on routing with compression in wireless sensor networks. In *Proc. of IPSN'04*, pages 28–35, Berkeley, California, USA.

Polastre, J., Hill, J., and Culler, D. (2004). Versatile low power media access for wireless sensor networks. In *Proc. of SenSys'04*, pages 95–107, Baltimore, MD, USA. ACM, New York, NY, USA.

Pradhan, S. S., Kusuma, J., and Ramchandran, K. (2002). Distributed Compression in a Dense Microsensor Network. *IEEE Signal Processing Magazine*, 19(2):51–60.

Rajendran, V., Obraczka, K., and Garcia-Luna-Aceves, J. J. (2006). Energy-efficient, collision-free medium access control for wireless sensor networks. *Wireless Networks*, 12(1):63–78.

Rhee, I., Warrier, A., Aia, M., Min, J., and Sichitiu, M. (2008). Z-MAC: A hybrid MAC for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 16(3):511–524.

Rhee, I., Warrier, A., Min, J., and Xu, L. (2009). DRAND: distributed randomized TDMA Scheduling for wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 8(10):1384–1396.

Scaglione, A. and Servetto, S. (2005). On the interdependence of routing and data compression in multi-hop sensor networks. *Wireless Networks*, 11(1-2):149–160.

Stone, K. and Colagrosso, M. (2007). Efficient duty cycling through prediction and sampling in wireless sensor networks. *Wireless Communications and Mobile Computing*, 7(9):1087–1102.

van Dam, T. and Langendoen, K. (2003). An adaptive energy-efficient MAC Protocol for wireless sensor networks. In *Proc. of SenSys'03*, pages 171–180, Los Angeles, CA, USA. ACM, New York, NY, USA.

van Hoesel, L. and Havinga, P. (2004). A lightweight medium access protocol (LMAC) for wireless sensor networks. In *Proc. of INSS'04*, pages 205–208, Tokyo, Japan. Society of Instrument and Control Engineers (SICE).

Vuran, M. C. and Akyildiz, I. F. (2006). Spatial correlation-based collaborative medium access control in wireless sensor networks. *IEEE/ACM Transactions on Networking*, 14(2):316–329.

Wan, C.-Y., Eisenman, S. B., Campbell, A. T., and Crowcroft, J. (2005). Siphon: Overload Traffic Management using Multi-Radio Virtual Sinks. In *Proc. of SenSys'05*, pages 116–129, San Diego, California, USA. ACM, New York, NY, USA.

Ye, W., Heidemann, J., and Estrin, D. (2004). Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12(3):493–506.

Yoon, S. and Shahabi, C. (2005). Exploiting spatial correlation towards an energy efficient clustered aggregation technique (CAG). In *Proc. of ICC'05*, volume 5, pages 3307–3313, Seoul, Korea.

Zhao, M., Chen, Z., and Ge, Z. (2007a). QS-Sift: QoS and spatial correlation-based medium access control in wireless sensor networks. *International Journal of Sensor Networks*, 2(3/4):228–234.

Zhao, M., Chen, Z., Zhang, L., and Ge, Z. (2007b). HS-Sift: hybrid spatial correlation-based medium access control for event-driven sensor networks. *IET Communications*, 1(6):1126–1132.