CSMAC: A Novel DS-CDMA Based MAC Protocol for Wireless Sensor Networks

Bao Hua Liu, Nirupama Bulusu*, Huan Pham, Sanjay Jha
School of Computer Science and Engineering, The University of New South Wales
School of Science and Engineering, Oregon Health and Sciences University*
Email: {mliu, huanp, sjha}@cse.unsw.edu.au, nbulusu@cse.ogi.edu

***Abstract*— This paper proposes CSMAC (CDMA Sensor MAC), a novel self-organizing, location-aware media-access control (MAC) protocol for DS-CDMA based sensor networks for applications such as battlefield surveillance that feature higher traffic and stringent latency requirements. Previously proposed MAC protocols for sensor networks such as S-MAC [10] primarily prioritize energy efficiency over latency. Our protocol design balances the considerations of energy efficiency, latency, accuracy, and fault-tolerance in sensor networks.**

CSMAC uses a combination of DS-CDMA and frequency division in channel allocation to reduce channel interference and consequently the message latency in the network. It exploits location awareness of sensor nodes to enable efficient network formation for collaborative sensing applications using two algorithms – Turn Off Redundant Node (TORN) and Select Minimum Neighbor (SMN). Our simulation results show that CSMAC significantly reduces mean message latency and mean energy consumption per message in comparison to traditional sensor network MAC protocols.

I. Introduction

Future military applications will increasingly feature communication scenarios involving a data-gathering or intelligence-gathering wireless sensor network [15]. A wireless sensor network deployed for such military applications typically consists of numerous sensor/actuator devices with integrated sensing, embedded microprocessors, low-power communication radios and on-board energy; organized in an ad hoc multi-hop network. This paper considers the problem of media access control for such sensor network applications.

A. Motivation

Because the design of an effective media access control (MAC) protocol is one of the fundamental communication challenges in sensor networks, it has been previously addressed in [1], [2], [3], [10]. Previously proposed MAC protocols for sensor networks such as SMAC [10] primarily prioritize energy efficiency over reducing network latency.

When sensor data is being collected for scientific research, the network may be inherently delay-tolerant. Whereas in a battlefield application where sensor data may be used to detect land mines, alert soldiers of the detection of enemy convoy vehicles, movement of explosives, car bombs etc., accurate and timely delivery of sensed data may mean the difference between life and death. These applications have stringent latency requirements. Our goals for the design of the MAC protocol are:

- *Fault Tolerance* of individual sensor nodes.
- *Low Latency* to enable the observer to learn about the phenomena quickly.
- *Scalability* to a large number of sensor nodes.
- *Energy Efficiency* to maximize the lifetime of entire system.

B. Paper Contributions and Organization

The contribution of the paper is the design and evaluation of CSMAC, a self-organizing, location-aware MAC protocol for DS-CDMA based sensor networks. Where CSMAC differs significantly from previously proposed self-organizing MAC protocols for sensor networks, is in that the network formation is explicitly influenced to meet collaborative sensing objectives (and conserve energy) rather than purely networking objectives.

The rest of the paper is organized as follows. Section II discusses related work. Section III presents the protocol design. Section IV describes the channel allocation pattern. Section V provides simulations and analysis. We conclude our paper in Section VI with possible future research directions.

II. Related Work

Previously proposed MAC protocols for sensor networks can be broadly classified into *contentionless* and *contention-oriented*. Contentionless MAC protocols are normally based on FDMA or TDMA approaches. Contention-oriented MAC protocols are adapted from the IEEE 802.11 standard.

SMACS (Self-organizing MAC for Sensor Networks) [3] is a distributed protocol which enables a collection of nodes to discover their neighbors and establish transmission/reception schedules for communicating with them without the need for any local or global master nodes. Unlike our protocol, network formation in SMACS is not location-aware, so neighbors selected may not be nearest. Moreover, a node must wait for its turn to transmit even if the channel is idle. And this waiting time can accumulate along the multi-hop route from source to sink. Shih *et al.* [1] have investigated the impact of non-ideal physical layer electronics on MAC protocol design for sensor networks and proposed a centrally controlled MAC scheme. A hybrid TDMA/FDMA scheme optimizes the power consumption of the transceiver, and results in lowering the overall power consumption of the system. LEACH [9] provides a combined TDMA/CDMA based MAC approach. Each node communicates with a dynamically elected cluster head directly (no multi-hop) using TDMA scheme. Cluster heads communicate with a remote destination (sink) directly using a CDMA approach. For low power, short range sensors, direct communications are not always practical. Muqattash and Krunz [7] proposed a CDMA-based MAC protocol for wireless ad hoc networks where out-of-band RTS/CTS are used to dynamically bound the transmission power of a node in the vicinity of a receiver. In this approach, RTS/CTS packet sizes are enlarged to accommodate MAI (Multi-Access Interference) related information, which may not be a suitable approach due to the short data packet size for sensor networks.

Woo and Culler [2] propose a CSMA-based MAC protocol, designed specifically to support the periodic and highly correlated traffic of some sensor network applications. They propose an adaptive transmission rate control (ARC) scheme, whose main goal is to achieve media access fairness by balancing the rates of originating and route-through traffic. SMAC (Sensor-MAC) [10] is based on the IEEE 802.11 standard but improves upon its energy efficiency.

SMAC identified several major sources of energy waste including *collision, overhearing, control packet overhead, and idle listening*. SMAC uses IEEE 802.11 CSMA/CA approach to avoid collision and puts a node to sleep when a neighbor node is transmitting to avoid overhearing. A scheduled periodic sleep and listening pattern is used to decrease the idle energy consumption. The main drawback of SMAC is high message delivery latency as SMAC is specially designed to sacrifice latency for energy savings.

Contention based protocols suffer from both low network throughput and long packet delay. Associating each small data packet transmission with RTS/CTS control packets exchange produces significant overheads. For example, Woo and Culler [2] state that an RTS-CTS-DATA-ACK handshake series in transmitting a packet can constitute up to 40% overhead with small packet size in sensor network. Although 802.11 standard specified that RTS/CTS can be avoided with small packet transmission, this is not suitable for sensor networks. Due to the low data rate (e.g., 20 Kbps) in sensor networks, the transmission time, and consequently collision probability, of a *small* packet (e.g., 50 bytes) may be much longer than that of transmitting it with 802.11 high data rate (e.g., 2 Mbps). Moreover, some energy efficient algorithms proposed for contention based protocols require the information embedded in RTS/CTS packets. For example, SMAC [10] uses the transmission time embedded in RTS/CTS to turn off unintended receivers to avoid the energy consumption caused by overhearing. Besides, contention based protocols also suffer from the well documented *hidden node* and *exposed node* problems. By foregoing control packet exchange, our approach can achieve significant improvements in energy efficiency.

Numerous topology control protocols for ad hoc and sensor networks have been proposed in literature [6], [14]. We refer interested readers to a good survey on topology control in wireless ad hoc and sensor networks [13] by Santi for further information.

III. The Protocol Design

In this section, we describe CSMAC's network formation process, consisting of several phases illustrated in Figure 1. Our protocol

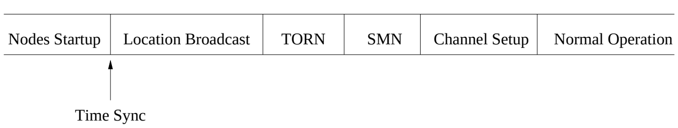

Fig. 1. *Network Formation Phases*

design assumes that each node: 1) starts up at approximately the same time; 2) can estimate its location; 3) is static during the network lifetime, so its location needs to be estimated only once (and the energy consumption of location estimation can be ignored).

A. Location Broadcast

In this phase, each node broadcasts its location information to its radio range neighbors. We assume that each node can estimate its location using GPS or alternate approaches [3],[5].

In CSMA/CA based protocols, RTS/CTS are normally not used for broadcast packets. To guarantee that each node can get an opportunity for a successful transmission, we employ large contention windows and allow each node to broadcast several times. Blough *et al.* [6] proved the crude lower bound that no contentions occur in a wireless channel with the following lemma: *Let $\bar{t}$ be the time necessary to transmit a packet. For $d = m\bar{t}$, the probability that no contention will occur in a wireless channel is strictly grater than $exp(-\frac{3h(h-1)}{2m})$, where h denotes the number of nodes that are contending for the channel*. An example was also given with 33 contending nodes, where d must be around $16000\bar{t}$ to achieve a probabilistic guarantee of no contention of at least 0.9. With $\bar{t}$ in the order of milliseconds, d is around tenth of seconds. In practice, small values can be used.

At the end of the location broadcast phase, each node should have a list with the locations of its radio range neighbors called *Redundant Neighbor List (RNL)*.

B. Turning Off Redundant Node (TORN)

Sensor networks are expected to be densely deployed. In the TORN phase, nodes that are redundant in meeting the application's sensing objectives are turned off to conserve energy and reduce network interference. Let *Sensing Resolution (SR)* denote the sensing accuracy desired by an application. Each node ranks all its RNL neighbors from the location broadcast phase based on their distance relative to itself. If sensors are densely deployed, the probability of having neighbors within a radius *SR* is high. These are treated as redundant nodes. TORN forces these redundant nodes to turn off themselves[1]. The battery power of these redundant nodes are preserved for future use, prolonging the network lifetime.

Note our definition of *Sensing Resolution* is an application-specific criteria that is different from the *sensing range*. The later is widely used in studying the coverage process of sensor networks in literature. Sensing range is a subjective hard limit that sometimes makes no sense. Consider acoustic sensors that are used to monitoring sounds. Whether a sensor can detect the target is not only dependent on the distance (sensing range), but also dependent on the sound intensity generated by the target. When a sensor is used to monitoring the temperature, sensing range makes no sense. We believe our application-specific definition of sensing accuracy is more suitable for sensor networks.

Each node then uses a contention based approach to negotiate who should keep active. A random timer is set to avoid collision. The first node that gets the media to transmit can inform its redundant neighbor(s) to turn off by including the ID numbers of these nodes. A node turns itself off upon receiving such a request from a neighbor. It wakes up later to check the energy level of the active node and decides whether it should take over, providing fault tolerance. A TORN example is given in Figure 2. We assume that each node can reach all others within this small area. Suppose node B is within the sensing resolution (SR) range of both nodes A and C. They have the following RNL (the redundant nodes are in ***bold italic*** font):

- A RNL: ***B***, C, D, M, H, E, O, ...
- B RNL: ***A, C,*** D, H, G, E, M, ...
- C RNL: ***B***, H, A, E, G, F, D, ...

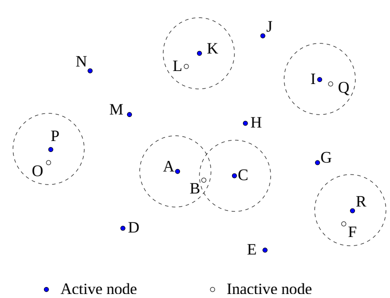

Fig. 2. *A TORN Example*

[1] Both the radio and sensing unit are turned off, only a very low power clock is running to wake up the node at sometime in the future

If node A grabs the media to transmit first, it will transmit a turn off message to node B. When node B receives this message, it turns itself into a backup node to node A and sends a broadcast that it will become an inactive node. When node C receives this turning off message from A to B or the broadcast message of B, C will remove node B from its RNL. If node B gets the media to transmit before A and C, it will transmit a turn off message including the ID numbers of both node A and node C. Other nodes receiving this message (D, H) remove both A and C from their RNL. To avoid a single node having too many backups, we can set the random timer of a node to be proportional to the number of redundant neighbors. For example,

```
Timer = (Random delay) + (Constant delay) *
        (Number of redundant neighbors - 1)
```

A node that has more redundant neighbors will less likely become an active node during TORN phase. We do this to provide fairness.

TORN generates an evenly deployed sensor network of reduced node density without diminishing the sensing accuracy requirement. At the end of TORN phase, only active nodes are left in the network. The resulting neighbor list in each active node is called *non-redundant neighbor list (NNL)*. This NNL will be used in the SMN process, which we discuss in the next subsection.

C. Select Minimum Neighbor (SMN)

In a wireless (sensor) network, the radio energy consumed for direct transmission between two nodes is directly proportional to d^k, where d is the separation distance between the two nodes and k is the path loss exponent that can vary from 2 to 6. Additionally, the static (distance-independent) power drawn by a transceiver, such as digital coding, modulation, and signal processing etc., can not be simply ignored. The difference between our algorithm and others (e.g., in [6], [8]) is that we also take the static power drawn into account.

In the SMN phase, two nodes are allowed to be direct network neighbors only if there is no alternate lower energy path between them. An optimized algorithm is used for a sensor node to select its neighbors from the *NNL* generated during the TORN phase. After SMN, each active node only has a near optimal set of neighbors, called *minimum neighbor list (MNL)*. This MNL will be used in the *Channel Setup* process wherein a peer-to-peer communication channel will be setup for each neighbor in MNL and this node. We denote the neighbor list of node i before and after SMN as $NNL(i)$ and $MNL(i)$. Further let P_{ij} the radio power between node i and j, PTX_{elec} the electronic power drawn of transmission, and PRX_{elec} the electronic power drawn of reception. The SMN algorithm is shown in Algorithm 1.

D. Channel Setup

At the end of the SMN phase, each node only has a small set of neighbors. The last step is to allow each node to setup connections to all it neighbors in the MNL. Each node estimates the transmission power required to reach its furthermost neighbor in its MNL. It uses this reduced power level for negotiation. Nodes far enough from this node can initiate another setup process simultaneously. A contention-based approach is used by the nodes to setup connections with each other. When a node grabs the media, it will hold the media until it finishes the channel allocation with *all* its neighbors in the MNL.

Two nodes (A and C) in Figure 2 are used to illustrate the process. At the beginning of the *Channel Setup* phase, each node sets a random timer and begins to count down. Each node can select

input : Non-redundant neighbor list $NNL(i)$
output : Minimum neighbor list $MNL(i)$
begin
  for $j \leftarrow n$ ***to*** 2 **do**
    isNeighbor = **true** ;
    foreach $k \in NNL(i)$ *and* $k \neq j$ **do**
      if $P_{ij} > P_{ik} + P_{kj} + PTX_{elec} + PRX_{elec}$ **then**
        isNeighbor = **false** ;
        break ;
      end
    end
    if *isNeighbor* = ***true*** **then**
      $MNL(i) \leftarrow MNL(i) \cup j$;
    end
    remove j from $NNL(i)$;
  end
end

Algorithm 1: Select minimum neighbor (SMN) algorithm.

random pseudo-noise codes (PN codes) and receiving frequency (Rx frequency) for communication with its neighbors (refer to section IV).

Figure 3 shows three different scenarios in the channel setup process and the packets exchanged in the process. After the channel

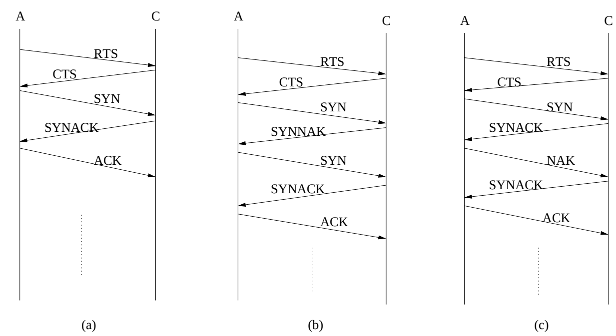

Fig. 3. *(a) Normal Operation. (b) C disagrees with the PN code or Receiving Frequency of A. (c) A disagrees with the PN code or Receiving Frequency of C.*

setup phase, each node should have information stored in its memory as shown in Table I and enter *Normal Operation* phase.

TABLE I

INFORMATION STORED IN A NODE'S MEMORY.

Node Information	Rx Frequencies – The receiving frequencies of this node (both unicast and broadcast).
	Broadcast PN code – The broadcast PN code of this node.
Neighbor Information	ID – The neighbor identity number.
	Broadcast PN code – Neighbor broadcast PN code.
	Rx PN code – From neighbor to this node.
	Tx PN code – From this node to neighbor.
	Tx frequency – The receiving frequency of neighbor.
	Tx power level – Transmission power level from this node to neighbor.

IV. Channel Allocation Pattern

DS-CDMA system uses Spread Spectrum (SS) modulation technique, in which the baseband signal is spread using a Pseudo Noise (PN) code. In this section, we describe the dynamic channel allocation pattern for PN code assignment. Figure 4 (a) illustrates A and B negotiating to use PN1 for communication from A to B. We can avoid A and B using PN1 again with their neighbors. But we can not avoid C and D using the same code. Even if we use stringent power control, the interference may still exist. There are two approaches to resolve this problem, namely *spatial division* and *frequency division*.

Spatial division separates nodes using the same PN code spatially. Each node negotiates with its neighbors using full transmission power so that each radio range neighbor can hear these messages. If nodes C and node D are within the radio range of nodes A or B, C and D can hear the negotiations between A and B. This way node C and D will not select PN1. When the network enters normal operation phase, each node will no longer transmit at full power but only communicate with its neighbors with much lower calibrated transmission power. The drawback of spatial division is the DS-CDMA *near-far* problem caused by *multiple access interference (MAI)*. A detailed discussion of MAI and near-far problem is out of the scope of this paper. We are currently working on extensions to provide detailed MAI mathematical mode and discussion for both spatial division and frequency division.

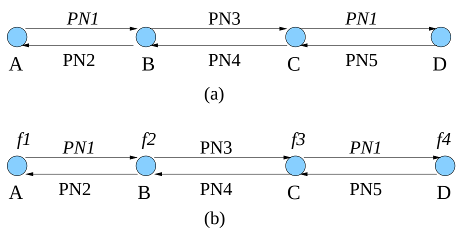

Fig. 4. *Channel Allocation Patterns*

In *frequency division*, each node uses a different frequency to *receive* signals as shown in Figure 4 (b). By using frequency division, both Tx and Rx can happen simultaneously. The *multiple access interference (MAI)* caused by competing transmissions at a specific receiving node are reduced significantly.

We adopted frequency division in our protocol design. The problem with this approach is that the transmitter is required to synthesize to different frequencies for transmission to different neighbors and thus this approach is not suitable for broadcast traffic. Broadcast is always expensive either in contentionless or contention-based protocols. We implement broadcast as follows: each node uses a different PN code but a common frequency to send broadcast packets. To implement this, we can employ two receivers in a sensor node, one dedicated to unicast, the other to broadcast. This approach is reliable and resistant to interference[2].

V. Simulations and Analysis

CSMAC has been implemented in the Network Simulator (NS-2). DSSS (Directed Sequence Spread Spectrum) is simulated as a PN code attribute in packet header. When a packet is received, its PN code is checked against the PN codes monitored by the receiver. If a match is found, the packet is passed to the next step for further processing. If no match is found, the packet is discarded. This procedure is used to simulate the de-spreading process.

Our simulations focus on the data transmission efficiency. As a comparison, we also measured the performance of SMAC [10], a well-known MAC protocol for sensor network but which can be used on the top of Directed-Sequence Spread Spectrum sensor networks. The parameters used in our simulations are shown in Table II:

[2]Multiple transceivers design is popular in sensors. For example, Mica mote and Pico Node are all equipped with two transceivers.

TABLE II

Parameters used in simulations.

Data Rate	10Kbps
Propagation Model	Log distance path loss
Reference Distance	1 meter
Path Loss Exponent	3.5
Antenna Gain	1
System Loss	1
Rx Threshold	1e-10 W
Carrier Sense Threshold	1e-11 W
Rx Elec Power	2 mW
Tx Elec Power	2 mW
Max Radio Power	10 mW
Power Amplifier Efficiency	33.33%
ISM Frequency Band	2.4-2.4835 GHz

A. *Measurement with Two-Hop Network Topology*

A two-hop topology is shown in Figure 5. We tested the energy and latency performance with two pairs of sources and sinks. The

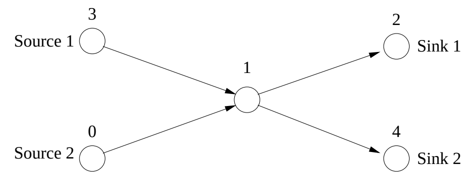

Fig. 5. *Two-hop network with two pairs of sources and sinks*

routing protocol used in this simulation is DSDV (Destination Sequence Distance Vector) with CBR (Constant Bit Rate) traffic. Each source sends 100 packets to the sink and the interval of the packet is set to 5s. The distances between nodes are deliberately set to make sure nodes 0, 1, 3 can hear each other and nodes 2, 1, 4 can hear each other.

Figure 6 compares the mean node energy consumption of CS-MAC and SMAC. We can see that CSMAC consumes 44.7% less mean energy per node compared to SMAC, because CSMAC does not use control packets exchange, avoids packet collisions, and uses calibrated power level for transmissions to different neighbors. Figure 6 also shows that SMAC consumes more energy with unicast

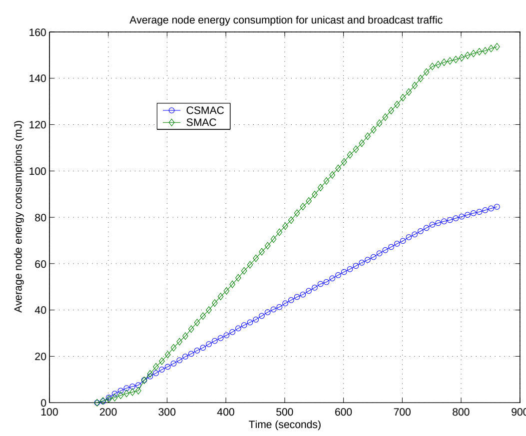

Fig. 6. *Mean per node energy consumption for both unicast and broadcast traffic*

traffic (E.g, between 250s-750s) due to the usage of control packets (RTS, CTS, ACK, SYNC, etc.) exchange. For broadcast traffic (E.g., between 750s-850s), energy consumption of SMAC and CSMAC is similar. In addition, CSMAC achieves a 62% lower mean packet latency as shown in Figure 7 (node 3 to node 4 has similar performance). In most cases, the latency using CSMAC is simply the

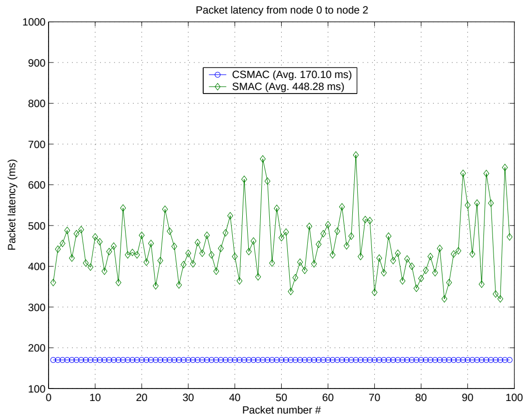

Fig. 7. *Packet latency from node 0 to node 2*

accumulation of the multi-hop transmission time. While in SMAC, the latency includes control packets (RTS/CST/ACK) exchange, carrier sense time, backoff delay and data transmission time.

B. Measurement with Ten-Hop Network Topology

We next tested the energy consumption and latency performance with a linear ten-hop network topology shown in Figure 8. We tested

Fig. 8. *Ten-hop network with one source and one sink*

by using directed diffusion protocol and ping application with one sink and one source. Figure 9 shows that CSMAC consumes 65%

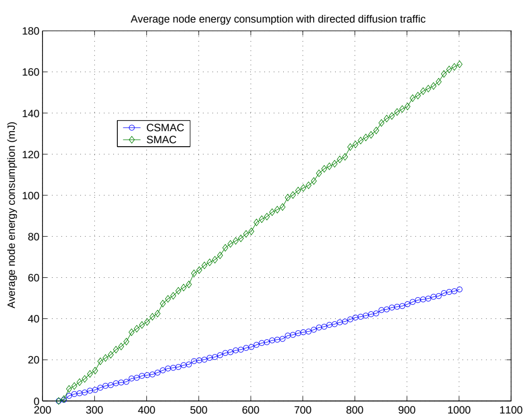

Fig. 9. *Mean node energy consumption for 10 hops*

lower mean energy per node compared to SMAC. CSMAC achieves better performance than for the two-hop topology, because broadcast traffic is not large in this linear topology. Besides this, CSMAC achieves 69% lower mean latency as shown in Figure 10.

C. SMN with Random Topology

We next tested our SMN algorithm with 20 randomly deployed sensor nodes in a $50m \times 50m$ area. The initial radio signal power is set to 10mW, which equals the full radio signal power, to ensure that each node can reach all its neighbors within its radio range.

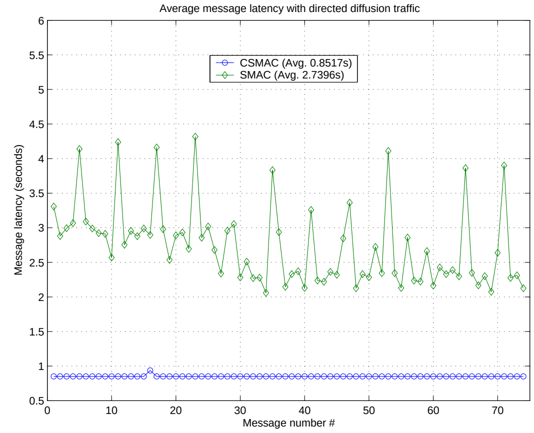

Fig. 10. *Mean message latency for 10 hops*

Figures 11 and 12 show the node connection pattern before SMN and after SMN respectively. From the diagrams, we can see that after SMN, the connection pattern is much simpler than before SMN. Each node removed some neighbors to conserve energy. For example, node 1 has 13 neighbors before SMN but has only 4 neighbors after SMN, node 17 has 12 neighbors before SMN but has only 5 neighbors after SMN.

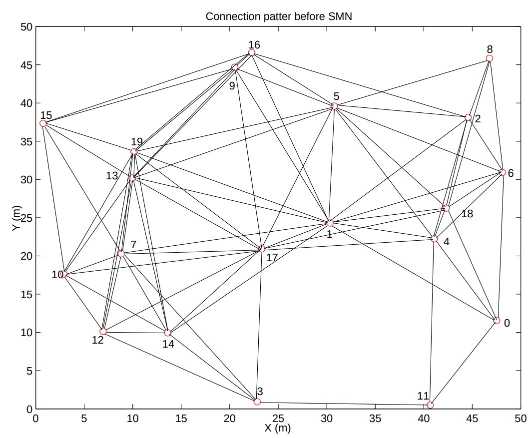

Fig. 11. *Node connection pattern before using SMN*

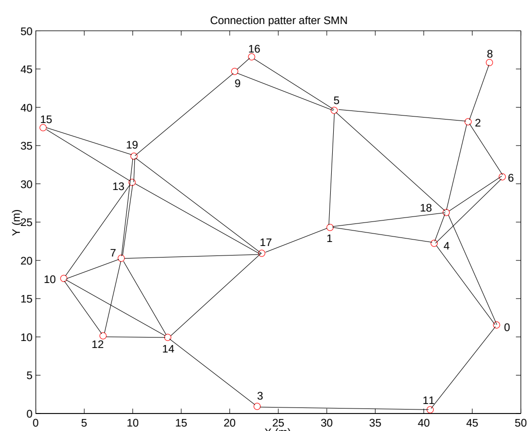

Fig. 12. *Node connection pattern after using SMN*

We tested the energy consumption and latency by using two pairs of sources and sinks with directed diffusion [4] ping application. One pair of source and sink is put on node 15 and node 11, the other is put on node 8 and node 12. The simulation plots the time period between 333s to 1001s. The ping message interval is 10 seconds. In all, 66 ping messages were received at each sink. NS-2 trace files reveal that the routes of the two pairs of source and sink are as follows:

```
Without SMN:
Source(8)  -> Sink(12): 8->6->0->11->3->12
Source(15) -> Sink(11): 15->7->3->11

With SMN
Source(8)  -> Sink(12): 8->2->18->1->17->14->12
Source(15) -> Sink(11): 15->13->7->14->3->11
```

By using SMN, the original routing path is altered. Figure 13 shows the mean node energy consumption, with and without SMN. Overall, 61% less mean energy is consumed with SMN.

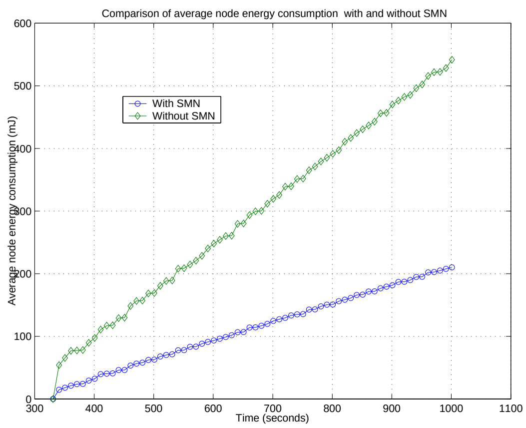

Fig. 13. *Mean energy consumption, with and without SMN*

Figures 14 plots the message latency for 15 $\rightarrow$ 11 source sink pair (the other has similar performance and is omitted to save space), with and without SMN. The figure shows that SMN increases the mean latency by 40%.

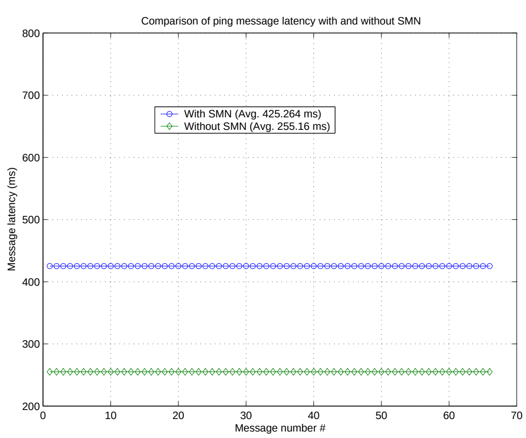

Fig. 14. *Comparison of message latency with and without SMN from node 15 to node 11*

VI. Concluding Remarks and Future Research

This paper proposed a novel self-organizing, location-aware MAC protocol design for DS-CDMA sensor networks; suitable for application scenarios with (i) high traffic (ii) stringent latency and (iii) fault tolerance requirements.

Previously proposed MAC protocols for sensor networks have prioritized energy efficiency foremost, ignoring other requirements. By exploiting CDMA-based techniques, self-organization and location-awareness in network formation (through TORN and SMN), our protocol design balanced performance requirements of sensor networks such as energy efficiency, low latency, sensing accuracy, and fault tolerance.

Our simulation results suggest that a combination of (a) location-awareness at MAC layer to improve energy efficiency and (b) DS-CDMA based techniques to improve network capacity may actually provide greater energy savings as well as much better latency performance in a multi-hop network. Analysis of TORN and SMN shows that in densely deployed networks, they reduce the operational network density and consequently interference, improving energy efficiency and network capacity.

We are currently working on a detailed mathematical model of MAI and the potential influence of frequency division approach. A more efficient and robust channel allocation protocol is also under consideration. Because we are targeting applications that have high traffic and stringent latency requirements, we have not incorporated a sleep and wakeup algorithm for idle energy savings at this stage.

The combination of our design and previously proposed sleep and wakeup schemes [10], [11], [12] could achieve greater energy savings and increased system capacity. Another approach to save idle energy is to allow sensors to sleep during non-duty cycles based on opportunistic application dependent criteria (e.g., no monitoring during night time) rather than simply turning sensors on and off based on redundant density criteria. We are pursuing these areas in our future work.

Acknowledgment

This work has been partially supported through Smart Internet Technology CRC (Cooperative Research Center), Australia.

References

[1] E. Shih *et al.*, "Physical Layer Driven Protocol and Algorithm Design for energy Efficient Wireless Sensor Networks," Proc. ACM MobiCom '01, Rome, Italy, July 2001, pp. 272-86.

[2] A. Woo, and D. Culler, "A Transmission Control Scheme for Media Access in Sensor Networks," Proc. ACM MobiCom 2001, pp.221-35.

[3] K. Sohrabi *et al.*, "Protocols for Self-Organization of a Wireless Sensor Network," IEEE Pers. Commun., Oct. 2000, pp. 16-27.

[4] C. Intanagonwiwat *et al.*, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks (MobiCOM 2000), August 2000, Boston, Massachusetts.

[5] N. Bulusu, "Self-Configuration Localization Systems," PhD Dissertation, UCLA, 2002

[6] D.M. Blough, *et al.* "The k-Neigh Protocol for Symmetric Topology Control in Ad Hoc Networks", Proc. of IEEE MobiHoc 2003, pp.141-152.

[7] A. Muqattash, M. Krunz, "CDMA-Based MAC Protocol for Wireless Ad Hoc Networks", Proceedings of MobiHoc'03, pp. 153-164.

[8] A Muqattash, *et al.*, "A Distributed Transmission Power Control Protocol for Mobile Ad Hoc Networks", IEEE Trans. on Mobile Computing, pp.113- 128 Vol,3, Issue:2, April 2004.

[9] W. R. Heinzelman "Application-Specific Protocol Architecture for Wireless Networks", PhD dissertation, MIT, June, 2000.

[10] W. Ye, J. Heidemann, D. Estrin, "An energy Efficient MAC Protocol for Wireless Sensor Networks", IEEE Proc. Infocom, June 2002, pp.1567-1576.

[11] C. Guo, L.C. Zhong, J.M. Rabaey. "Low Power Distributed MAC for Ad Hoc Sensor Radio Networks", IEEE Proc. GlobeCom 2001, pp. 2944-2948.

[12] C. Schurgers "Optimizing Sensor Networks in the Energy-Latency-Density Design Space", IEEE Trans. on Mobile Computing, Vol.1, No.1, Jan.-Mar. 2002, pp. 70-80.

[13] P. Santi, "Topology Control in Wireless Ad Hoc and Sensor Networks", submitted to ACM Comp. Surveys.

[14] R. Wattenhofer, L. Li, P.Bahl, Y.M.Wang, "Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad Hoc Networks", Proc. IEEE Infocom 2001, pp. 1388-1397.

[15] B. Ames, "Electronics are central to 21st century warfare tactics", Military and Aerospace Electronics, Jan 2004.