A Multi-Channel DS-CDMA Media Access Control Protocol for Wireless Sensor Networks

Bao Hua Liu, Chun Tung Chou, Sanjay Jha

Network Research Laboratory
School of Computer Science and Engineering
The University of New South Wales, Sydney, Australia

Email: {mliu, ctchou, sjha}@cse.unsw.edu.au

UNSW-CSE-TR-0503

February 23, 2005

THE UNIVERSITY OF
NEW SOUTH WALES

SYDNEY · AUSTRALIA

Abstract

This paper proposes a novel multi-channel media access control (MAC) protocol for direct sequence code division multiple access (DS-CDMA) wireless sensor network. Our protocol design uses combination of DS-CDMA and frequency division to reduce the channel interference and consequently improves system capacity and network throughput. We provide theoretical characterization of the mean multiple access interference (MAI) at a given node in relation to the number of frequency channels. We show that by using only a small number of frequency channels, the mean MAI can be reduced significantly. Through discrete event simulation, we provide comparison of our proposed system to a pure DS-CDMA system as well as a contention based system. Simulation results reveal that our proposed system can achieve 15-20 times of system efficiency than a contention based system. When same number of packets are transmitted in the network, our system consumes only 10% of communication energy than the contention based system.

A distributed channel allocation protocol is also proposed for the network formation phase. We prove that our algorithm converges with correct channel assignments. Simulation results reveal that much smaller number of channels are required than theoretical value when nodes are uniformly randomly deployed.

1 Introduction

Many future applications will increasingly depend on embedded wireless sensor networks. A sensor network consists of numerous sensor/actuator devices. These devices consist of one or more integrated sensing units, embedded microprocessors, low-power communication radios, on-board energy, with location awareness and organized in an ad hoc multi-hop network. Sensors devices are normally untethered and powered by batteries. They communicate over short distance using wireless media.

1.1 Motivation

Since sensor network applications are generally expected to utilize low data rate (e.g., 1-100Kbps), have small data packet size (e.g., 50 bytes), and sensors normally have limited on-board energy, processing capability, buffer space, and other resources, the contention based protocol may not be a suitable choice for the MAC layer. Contention based protocols suffer from both low network throughput and long packet delay. Associating with each *small* data packet transmission, the RTS-CTS-DATA-ACK handshake sequence can constitute up to 40% overhead in sensor networks [18]. Although IEEE 802.11 standard specifies that RTS/CTS can be avoided with a small data packet transmission, this may not be a practical choice for sensor networks. Given the lower data rate (e.g., 20Kbps) in sensor networks, a *small* data packet will take much longer time to transmit than in an IEEE 802.11 network which has high data rate (e.g., 2Mbps). As a result, the collision probability in sensor networks is much higher. The consequence is that control packet exchange is necessary to avoid collisions. Moreover, some energy efficient algorithms proposed for contention based protocols require the information embedded in RTS/CTS packets. For example, SMAC [23] uses the transmission time embedded in RTS/CTS to turn off unintended receivers to avoid overhearing energy consumption. Furthermore, contention based protocols also suffer from the well documented hidden node and exposed node problems.

It is well known that energy consumption is the crucial factor in sensor network design. This may lead to sensor network MAC protocols which prioritize energy savings over network throughput and packet latency. However, we argue that both network throughput and packet latency are critical for many sensor network applications, such as battlefield surveillance, real-time monitoring seismic waves, machine operations, bush fires. Accurate and timely delivery of sensed data in these cases sometimes means the difference between life and death. When a sensor network is used to track an object (or group of objects), out-of-date information is of no use if the object being tracked is no longer in the vicinity when the information is received.

While a sensor network is expected to operate with low duty cycle (e.g., 1% on average) and may remain silent for a long time, a communication "hot region" can emerge quickly due to simultaneous events [28]. Such unpredictable traffic pattern requires highly adaptive protocols to achieve both real-time guarantees and energy efficiency.

As an alternative to traditional contention based MAC designs, direct sequence code division multiple access (DS-CDMA) based system has been proposed in literature [8][9][27]. It is well known that the *multiple access interference (MAI)* is the key factor in determining the network performance (e.g., throughput) in a DS-CDMA system. The inherent randomized distribution of sensor nodes and the lack of centralized base station creates significant challenge on how to effectively control the interference in a DS-CDMA sensor network. In this paper, we propose a frequency division based DS-CDMA system which can simultaneously achieve low energy consumption, high network throughput and low packet latency. These advantages are the result of applying frequency division to reduce the MAI in the system. By using an analytical model, we show that a small number of frequency channels can reduce MAI significantly.

1.2 Paper Contributions and Organization

Our protocol design differs significantly from previously proposed self-organizing MAC protocols for sensor networks, in that both network throughput and packet latency are emphasized as well as energy consumption. This paper makes the following contributions:

- Propose a novel multi-channel MAC protocol for wireless sensor networks.
- A mathematical model to calculate the expected value of MAI at a given node with a uniformly randomly distributed topology.
- A distributed channel allocation (for both frequencies and CDMA PN codes) protocol for the network formation phase.
- Simulation results demonstrating that the proposed system can achieve less channel contention, lower packet latency, higher network throughput, and lower energy consumption.

The rest of this paper is organized as follows. Section 2 presents background and limitation of using DS-CDMA system in sensor networks. Section 4 describes the proposed design architecture. Section 5 provides the analytical model. Section 6 details the network formation and channel allocation protocol. Section 7 provides simulation results and analysis. Section 8 describes the related work. We conclude our paper in section 9 with future directions.

2 Preliminaries

This section provides a brief review of DS-CDMA system and the limitations of using DS-CDMA system in an ad hoc wireless sensor network, specially, the multiple access interference (MAI) and resulted near-far problem.

2.1 DS-CDMA System

DS-CDMA based system uses *spread spectrum* (SS) modulation technique, in which the baseband signal is spread (e.g., multiplied by) using a *pseudo noise* (PN) code. A PN code is a sequence of chips valued -1 and 1 (polar) or 0 and 1 (non-polar). Such bit-sequences have noise-like properties like spectral flatness and low cross and auto correlation values, and thus complicate jamming or detection by non-target receivers [58]. The receiver uses cross-correlation to separate the appropriate signal from other interfering signals (those destined for other receivers), and uses auto-correlation to reject multi-path interference. In practice, the PN sequence selected must have both good cross-correlation and auto-correlation.

Spread spectrum has many properties that makes it particular attractive for wireless environment. The most important advantage is its inherent interference rejection capability. Not only can a particular spread spectrum signal be recovered from a number of other spread spectrum signals, it is also possible to completely recover a spread spectrum signal even when it is jammed by a narrow band interferer. Resistance to multi-path fading is another fundamental feature of spread spectrum modulation. Spread spectrum system is not very bandwidth efficient in a single user environment but becomes very bandwidth efficient in a multiple-user environment. Compare to TDMA, CDMA requires less synchronization overheads. There is no need for synchronization among users in a CDMA system except between a transmitter and a receiver.

The main parameter in a DS-CDMA system is the *processing gain*, which is defined as the ratio of spread data rate to the baseband data rate. Processing gain determines the number of simultaneous transmissions that can be accommodated in a DS-CDMA system.

3 The Multiple Access Interference of DS-CDMA SYSTEM

Although spread spectrum system provides superior characteristics, the performance of DS-CDMA system is primarily limited by *multiple access interference (MAI)*. Considering a DSSS/BPSK (Direct Sequence Spread Spectrum/Binary Phase Shift Keying) system, let P_0 denote the average received power of the desired signal at the detector. Further, assume that there are k interferers with received powers $P_1, P_2, \ldots, P_k$, the MAI is defined as the sum of all interference powers, e.g., $MAI = \sum_{i=1}^{k} P_i$. The *effective bit energy-to-noise ratio* at the detector is then given by [42]:

$$\mu \triangleq \frac{E_b}{N_{0\text{eff}}} = \left(\frac{2\sum_{i=1}^{k} P_i}{3LP_0} + \frac{1}{\mu_0} \right)^{-1} \tag{1}$$

where L is the processing gain, and $\mu_0 = E_b/N_0$ equals $E_b/N_{0\text{eff}}$ at the detector in the absence of interferers. The probability of bit error P_e with a given μ is then given by $P_e = \frac{1}{2}\text{erfc}(\sqrt{\mu})$ which is decreasing function of μ, where $\text{erfc}(\cdot)$ is the complementary error function.

In a cellular DS-CDMA network, the MAI can be controlled by the base station by limiting the number of active nodes, and requiring that all active nodes control their transmission powers so that the signals arrive at the base station with the same *lowest* power level. However, the same principle can not be applied to a practical sensor network. The difficulty of using DS-CDMA system in an ad hoc sensor network lies in the fact that a sensor network does not have a centralized base station which leads to the MAI being uncontrollable. The consequence is that the system performance is compromised severely especially in higher traffic scenario. In a practical sensor network, where nodes are normally randomly deployed, each transmitting node can transmit to multiple receivers and each receiving node can receive from multiple transmitters. It is infeasible to implement a power control scheme to guarantee that each receiver receives *all* signals (both desired and interferers) with *equal* power. In addition, each node may transmit to a random neighbor with random probability at any given time. The number of active nodes can not be controlled without using RTS/CTS in a potential interference vicinity. These unpredictable traffic loads make the MAI very difficult to be controlled. We will illustrate these problems using Figure 1, where sensors are randomly deployed and R_R represents the communication range. Each node has a number of neighbors situated at different distances. For example, A

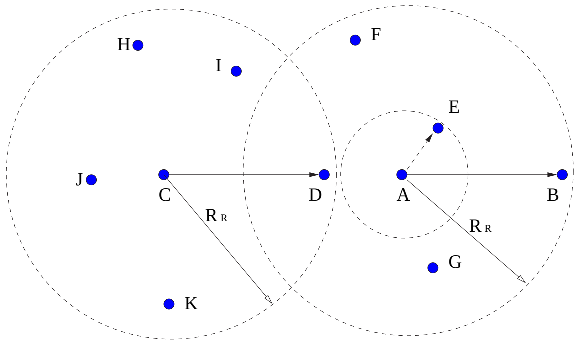


Figure 1: *Example showing the uncontrollable MAI is an ad hoc sensor network.*

has neighbors B, E, F, D, and G. Assume that each node uses the minimum required power to communicate with each other. When A is transmitting to a neighbor, the interference power caused by this transmission at other neighbors can have different values. Considering two simultaneous transmissions from $A \rightarrow B$ and $C \rightarrow D$, where distance $d_{AB} \gg d_{AD}$, the interference power at D caused by A is much higher than that of the desired power from C, and this makes the desired signal hard to be recovered. However, if the transmission is $A \rightarrow E$ instead of $A \rightarrow B$, the interference caused to D's reception is negligible. The problem caused by interference signal(s) drowning out desired signal at a receiver is a serious consequence of MAI. This example demonstrates that it is not possible to use power control to minimize the effect of MAI in a DS-CDMA sensor network. The MAI may cause significant degradation in network throughput and is considered the main problem prohibiting the usage of DS-CDMA system in sensor networks.

4 The Proposed Architecture

In last section, we discussed the limitations of using DS-CDMA system in sensor networks. The root cause of MAI lies in the fact that, unlike FDMA and TDMA channels, CDMA codes are not completely orthogonal.

Completely orthogonal codes (e.g., Walsh codes) are normally used in synchronous systems. However, in asynchronous systems, perfect orthogonal codes are sub-optimal and exhibit high cross-correlation.

Because MAI is caused by the non-perfect orthogonality of CDMA codes, the rationale of our design is to *orthogonalize the reception* in the vicinity of a sensor node. In this paper, we propose a frequency division based DS-CDMA system to reduce MAI. Because most sensor network applications are expected to utilize low data rate, it is possible to use a narrow band DS-CDMA system operating over multiple frequency channels. For example, assuming that the data rate is 20 Kbps, and we use 50 chip/bit PN code to spread the baseband signal. The resulting bandwidth is 1MHz. With 2.4GHz ISM band (2400MHz-2483.5MHz), we can have more than 80 frequency channels. Our simulation results (see section 7.2) reveal that much less number of channels are required than theoretical value when sensors are uniformly randomly distributed. We make the following assumptions in our design:

- Sensors are normally static nodes so we can execute our network set up process once at the beginning of deployment.
- A topology control protocol is available to limit the number of neighbors for each node.
- Multiuser detection receivers are available but may only be monitoring a limited number (e.g., number of neighbors) of PN codes.
- Each sensor can adjust its transmission power to reach different neighbors.
- Each sensor can estimate its location or relative location.

4.1 System Architecture

A set up phase (see section 6) is required for our sensor network architecture where, during this phase, frequency channels and PN codes are assigned to enable the nodes to communicate during steady state. In this section, we primarily focus on the steady state operation.

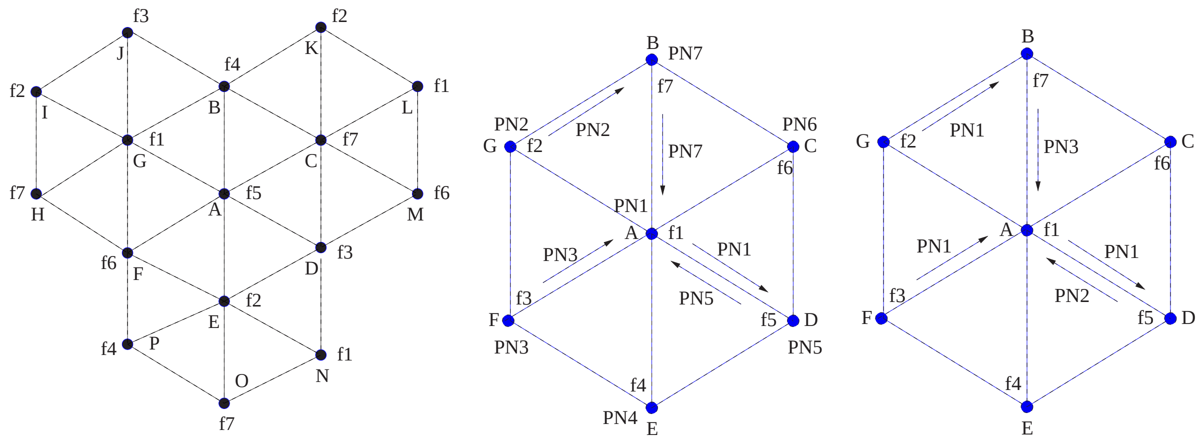
In our system, each node is assigned a unique *receiving* frequency that is different from its one hop and two hop neighbors. Without loss of generality, we use a regular graph to explain the concept. Figure 2 (a) illustrates the frequency channel allocation pattern with a regular triangular sensor network topology, where each sensor has exactly six neighbors and the *receiving* frequency of each node is shown next to the node. Note with a random deployment, the number of neighbors can be determined by a topology control protocol. There are two PN code assignment schemes in our system:

1. *Node based*, where each neighbor of a given node is assigned a unique PN code that is different from each other; or
2. *Link based*, where each *directed* link is assigned a unique PN code that is different from its adjacent links (two directed links are adjacent if they have a common end node).

Figure 2 (b) shows the node based (transmitter based) PN code assignment, where the code assigned to each node (for transmission) is shown next to the node. Figure 2 (c) shows the link based PN code assignment, where the code assigned to each *directed* link is shown along the link.

When a node wants to transmit a packet to a neighbor, it synthesizes its transmitter to the corresponding receiving frequency of the neighbor and uses the pre-determined PN code with the neighbor to spread baseband signal. We next use Figure 2 (c) to explain the communication paradigm between a node and its neighbors (Node based PN code assignment can achieve similar performance). Note that $PN1$ can be re-used assuming that two links are not adjacent in Figure 2 (c).

When A wants to transmit to D, it synthesizes its transmitter to $f5$ and uses $PN1$ to spread the data packet. Note this transmission does not cause interference to other neighbors such as C, B, G, F, E because their receivers are running on different receiving frequencies. Also note that B, F and D can transmit to A simultaneously because A's multiuser detection receiver can distinguish all its *neighbors'* transmissions



(a) (b) (c)

Figure 2: *System architecture (a) Frequency allocation pattern. (b) Node based (transmitter based) code assignment. (c) Link based code assignment.*

concurrently. Note that A does not need to monitor the whole set of PN codes but only the set of codes that are employed by its immediate neighbors. A's transmission to D will not destroy A's reception from B, F and D even if they are happening in parallel because they are operating on different frequencies. Furthermore, the transmission signal from G to B will not contribute to the noise floor at A because it is operating on $f7$. To this end, we notice that MAI only occurs at a given receiving node (e.g., A) when multiple neighbors (e.g., B, F, D) transmit to this node (e.g., A) simultaneously. When these simultaneous transmissions occur, they are actually desired signals as they are all addressed to this node (e.g., A). With proper power control, the resulted MAI at this node (e.g., A) can be controlled at the lowest level. Those uncontrollable MAI presented in a pure DS-CDMA system, which is caused by the transmission between an interference node (e.g., G) and its neighbor (e.g., B but not A), is significantly reduced due to the frequency division. Note, it is possible that some other nodes are transmitting with the same frequency in the network (as frequencies are reused spatially). But assuming that the channel allocation scheme (see section 6.2) is functioning, these transmissions are normally far enough and the resulting interference is negligible. Because a node does not have to consider the interference caused by its transmission on the *unintended* receivers, it is much easier for a node to control its transmission power to guarantee that the transmitted signal arrives at the *intended* receiver with a certain power level, e.g., the *lowest receiving threshold*.

By employing frequency division, our protocol design can achieve significant reduction in MAI and consequently less channel contention. Reduction in MAI makes it possible to employ shorter PN sequence and lower processing gain, which results in smaller spread signal bandwidth. Smaller signal bandwidth means more frequency channels are available if the system bandwidth is fixed, or smaller system bandwidth is required if the number of channels is fixed. Reduction in MAI also makes it possible to forego control packet (RTS/CTS/ACK) exchange which leads to lower protocol overhead, higher network throughput, lower packet latency and less energy consumption. Frequency division also decreases the energy consumption due to reduction of overhearing.

4.2 Broadcasting Traffic

One of the problems of using frequency division is how to deal with broadcasting traffic. Considering Figure 2 (c), node A and its neighbors are operating on different receiving frequencies, which means the usual method of broadcasting over a common frequency channel cannot be used.

Broadcasting forms an important part of routing protocols and service discovery in sensor networks. For example, in *directed diffusion* [17], a node uses flooding to diffuse an *interest* into the network. Flooding and broadcast are always costly and may cause serious redundant re-broadcasts, media contention and packet collisions. In this paper, we propose three different schemes to achieve broadcast at MAC layer with our proposed protocol. In two of these proposals, we require two transceivers in a sensor. Note that multiple transceivers design is popular in sensors. For example, both Mica Mote [59] and Pico Node [60] are equipped with dual transceivers. Our proposed schemes are described as below:

1. In first scheme, each node employs only one transceiver and transmits a broadcast packet to its immediate neighbors with multiple unicasts. For example, in Figure 2 (c), node A unicasts a broadcast packet multiple times to all its neighbors in turn.
2. In second scheme, each node employs a second transceiver. This second transceiver is dedicated to broadcast traffic (and possible other control purposes) and synthesizes to a default frequency channel. With this second scheme, each node uses a different transmitting PN code for broadcast. And each node monitors the broadcast PN codes of its immediate neighbors.
3. In third scheme, each node employs a second transceiver. The transmission scheme is the same as IEEE 802.11, where each node transmits broadcast packets with the same frequency and the same PN code. Note data spreading in this case is only for better channel performance but not for multiple access.

Detailed simulations and analysis for these three broadcast schemes are provided in section 7.3.

4.3 Idle Energy Consumption

As stated before, energy consumption is crucial to prolong the lifetime of the sensor network. To achieve significant energy savings, a node should turn off its radio when it does not participate in data forwarding or no event occurs. A simple way is to allow each node to sleep and wake up periodically as proposed in SMAC [23]. With proper coordination, nodes can sleep and wake up with the same duty cycle. Guo [27] *et al.* proposed a super low power radio called *wake-up radio* that can allow the normal radio to power down during idle listening time. The wake-up radio serves as a *small ear* and keeps monitoring the channel signal on a super low power. The monitoring power is around $1\mu W$ and the wake-up radio may induce 1 ms delay. Schurgers *et al.* [25] proposed a technique to efficiently wake up nodes from deep sleep state by separating the data and wake up with two radios. The wake up radio operates on low power listening mode and uses a periodic sleep-wakeup scheme similar to SMAC. The wake up radio does not assume any specific protocols at MAC layer so it can work with different MAC protocols.

Although our design is targeting high network throughput and low packet latency, we would like to emphasize that our design can also accommodate any one of these sleep-wakeup schemes. The combination of our design with one of these schemes can achieve both high network throughput and low energy consumption.

5 Multiple Access Interference Modeling

In section 2, we discussed the uncontrollablility of MAI in ad hoc sensor networks. In order to show how MAI can be reduced by employing frequency division, we derive an analytical model which shows how the number of frequency channels can influence the mean MAI at a given node. Following assumptions are used in this analysis:

- Sensors are uniformly randomly distributed.
- Each sensor transmits with an independent probability to a random neighbor.
- We assume a random frequency allocation pattern which represents the worst case comparing with the systematic frequency allocation pattern described in section 6.2. Note with this random approach, two (or more) neighbors may transmit over the same frequency channel.

We assume that the node whose mean MAI we are interested to compute is located at the origin with receiving frequency f_0.

5.1 System Model

We assume a sensor network with h nodes which are uniformly randomly deployed into a plane $\mathcal{R} \subseteq \mathbb{R}^2$. For convenience, we assume $\mathcal{R}$ to be a square $[-d/2, d/2]^2$, having area $\|\mathcal{R}\| = d^2$, and suppose d and h increase together in such a manner that $h/\|\mathcal{R}\| \to \lambda$ where $0 < \lambda < \infty$. Let $\mathcal{S}$ denote a bounded Borel subset of $\mathcal{R}$. For large d where $\|\mathcal{R}\| \gg \|\mathcal{S}\|$, and then the chance that $\mathcal{S}$ contains precisely k of the uniformly distributed nodes is given by [56]:

$$P[k \text{ in } \mathcal{S}] = \binom{n}{k}\left(\frac{\|\mathcal{S}\|}{\|\mathcal{R}\|}\right)^k \left(1 - \frac{\|\mathcal{S}\|}{\|\mathcal{R}\|}\right)^{h-k} \tag{2}$$

As $\mathcal{R}$ increases, the binomial distribution of equation 2 is well approximated by a Poisson process:

$$P[k \text{ in } \mathcal{S}] = \frac{(\lambda\|\mathcal{S}\|)^k}{k!} e^{-\lambda\|\mathcal{S}\|} \tag{3}$$

where λ equals the mean number of nodes per unit area of $\mathcal{R}$, or node density.

5.2 The Distribution of Interference Power

We first analyze the distribution of the interference power at the origin (where the node we are interested is located) which is caused by a single interference node. Assume that each node has a bounded normalized (the normalization is with respect to antenna gain, system loss, and wavelength) maximum transmission power P_T. Let P_R denote the lowest receiving threshold, then by using the propagation law, the maximum transmission range R_R is given by $R_R = \sqrt[n]{P_T/P_R}$, where n is the path loss exponent[1] and normally $2 < n < 6$. Let P_I be carrier sense threshold, then the interference range R_I is given by $R_I = \sqrt[n]{P_T/P_I}$, normally $R_I \approx 2R_R$.

We assume that each node has perfect power control so that the power level of the desired signal at the intended receiver equals to P_R. Assume that both the node at the origin O and node j use receiving frequency f_0. Now consider an interfering node i transmitting to a random neighbor node j as shown in Figure 3. The transmission power of node i is a random variable that is dependent on the distance between node i and node

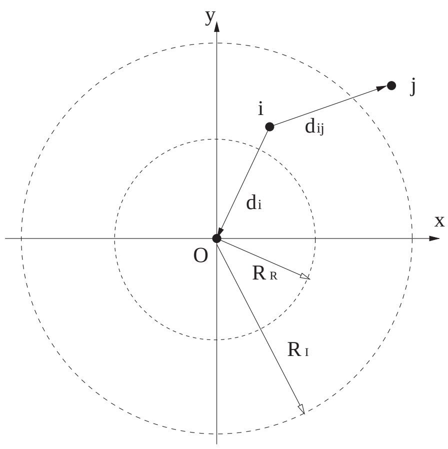


Figure 3: *Multiple access interference model.*

j. We define this random variable as $\mathbf{x}$, where

$$\mathbf{x_{ij}} = P_R \mathbf{d_{ij}^n} \quad d_{ij} \in (0, R_R] \tag{4}$$

Note $\mathbf{d_{ij}}$ is also a random variable representing the inter-nodal distance. Assume that both the node at the origin O and node j use receiving frequency f_0. Now consider the interference power at the origin that is

[1] To make the moment generating function of the interference power exist, we assume that $n \neq 2$.

caused by the transmission from node i to node j. We define this interference power as a random variable $\mathbf{z}$:

$$\mathbf{z} = \frac{\mathbf{x_{ij}}}{\mathbf{d_i^n}} = \frac{P_R \mathbf{d_{ij}^n}}{\mathbf{d_i^n}} \quad d_{ij} \in (0, R_R], d_i \in (0, R_I] \tag{5}$$

where random variable $\mathbf{d_i}$ denotes the distance of node i to the origin. We can prove (see [63]) that the density function of $\mathbf{z}$ is:

$$f_{\mathbf{z}}(z) = \begin{cases} \frac{\alpha}{2}\, a^{-\alpha} z^{\alpha-1} & 0 \le z < a \\ \frac{\alpha}{2}\, a^{\alpha} z^{-\alpha-1} & a \le z < b \end{cases} \tag{6}$$

where $a = P_R R_R^n / R_I^n$, $b = P_R R_R^n$, and $\alpha = 2/n$.

5.3 Mean MAI at A Given Node

Knowing the interference density from one interference node, we can compute the total interference from multiple nodes. Let E be a Poisson process in the plane with density λ. The probability law for E is determined by equation 3. We assume that the probability that a node is transmitting equals to p, then the set of transmitting nodes forms a Poisson process E_t with parameter $\lambda_t = \lambda p$. Further assume that there are M frequency channels. And each node selects a frequency channel for receiving with equal probability (note we assume a random selection scheme). The probability that a node is transmitting with a specific frequency (e.g., f_0) equals to $p' = p/M$. The set of transmitting nodes with a specific frequency also forms a Poisson process E_t' with parameter $\lambda_t' = \lambda p/M$. Now, with each sample function of E_t', we can associate the random variable

$$\Omega = \sum z_k \tag{7}$$

where the summation is over all points of the sample function of E_t' within the disk that is centered at origin with radius R_I, we denote the disk as $D(R_I)$. To find the expected value of MAI at the origin, we work with the moment generating function of Ω, denote as $\Phi_\Omega(s)$. The expected value can then be derived from the first derivative of $\Phi_\Omega(s)$ at $s = 0$. The moment generating function of Ω is related to Laplace transform as follows

$$\Phi_\Omega(s) = \int_0^\infty e^{s\omega} f_\Omega(\omega) d\omega = \mathbf{E}[e^{s\Omega}] \quad s \ge 0 \tag{8}$$

where $f_\Omega(\omega)$ is the density function of Ω. Using conditional expectations, $\Phi_\Omega(s)$ may be evaluated as

$$\Phi_\Omega(s) = \mathbf{E}[e^{s\Omega}] = \mathbf{E}[\mathbf{E}[e^{s\Omega} \mid k \text{ in } D(R_I)]] = \sum_{k=0}^{\infty} \frac{e^{-\lambda_t' \pi R_I^2} (\lambda_t' \pi R_I^2)^k}{k!} \mathbf{E}[e^{s\Omega} \mid k \text{ in } D(R_I)] \tag{9}$$

where 'k in $D(R_I)$' is the event that there are k transmitting nodes with the same frequency (f_0) in disk $D(R_I)$, and the expectation is over the random variable Ω.

Now, given that there are k transmitting nodes with a specific frequency (f_0) in disk $D(R_I)$, and since the moment generating function of the sum of a number of independent random variables is the product of the individual moment generating function, we have

$$\mathbf{E}[e^{s\Omega} \mid k \text{ in } D(R_I)] = \left(\int_0^\infty e^{sz} f_{\mathbf{z}}(z) dz \right)^k \tag{10}$$

Using equation 6, and after some simplification and substitution, we obtain

$$\Phi_\Omega(s) = \exp\left(\frac{\lambda p \pi R_I^2}{M} \left(\frac{\alpha a^{-\alpha}}{2} \int_0^a e^{sz} z^{\alpha-1} dz + \frac{\alpha a^{\alpha}}{2} \int_a^b e^{sz} z^{-\alpha-1} dz \; - \; 1 \right) \right) \tag{11}$$

We then obtain the expected value of Ω as below

$$\eta = \frac{d\Phi_\Omega(s)}{ds} \Big|_{s=0} = \frac{\alpha a \lambda p \pi R_I^2}{2M} \left(\frac{1}{\alpha+1} + \frac{1 - (a/b)^{\alpha-1}}{\alpha - 1} \right) \times \exp\left(- \frac{\lambda p \pi R_I^2}{2M} \left(\frac{a}{b} \right)^{\alpha} \right) \tag{12}$$

The above equation gives the expected value of the multiple access interference power at a given node in relation to the number of frequency channels. As an example, Figure 4 plots the mean MAI versus the number of frequency channels with following parameters: $\lambda = 0.01$, $P_R = -70dBm$, $R_R = 25m$, $R_I = 50m$, and

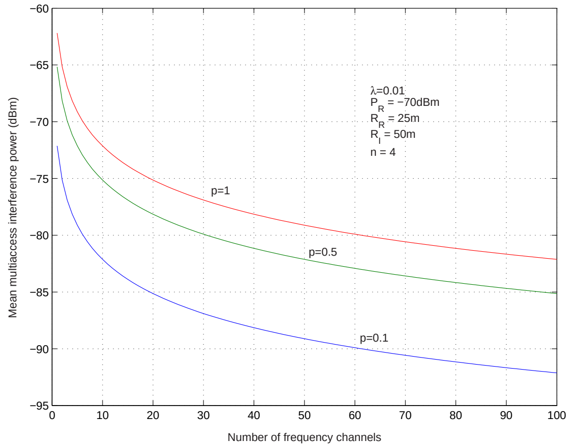


Figure 4: *Mean MAI versus number of frequency channels*

$n = 4$. We see that the mean MAI reduces sharply with the employment of a small number of frequency channels. For example, the mean MAI with $p = 1$ and one frequency is -62.2 dBm, the mean MAI reduces to -72.1 dBm with 10 frequency channels. This almost equals to 10 times reduction. Moreover, our simulation results (presented in section 7) show that the reduction of MAI leads to higher network throughput, lower packet latency, and less energy consumption.

6 Network Formation

In section 4.1, we described the system architecture for steady state operation. In this section, we elaborate the network formation in which frequency and PN code are assigned to each node.

The network formation in our protocol is based on nodes dynamically discovering themselves and selecting suitable communication neighbors to form a connected multi-hop network topology. This enables our protocol to be flexible and scalable to large scale networks. The network formation consists of two phases, with *topology formation* followed by *distributed channel allocation*. After the nodes have powered up after deployment, the *topology formation* begins with the nodes discovering their neighbors and then followed by a topology control step which limits the number of neighbors for a given node. The topology control step is important because it limits the number of frequency channels required by our system. The topology after topology control will be the input to the channel allocation phase. In particular, the nodes only have to know their one and two hop neighbors in this topology for channel allocation.

6.1 Topology Formation

Instead of relying on global topology information to achieve topology formation, our distributed protocol only requires that each node to obtain the node information of its one hop and two hop neighbors. The main idea is to use a default common control channel to exchange information for all nodes on the network. We assume that each node starts at similar time and can be loosely synchronized at the beginning of network deployment, the information of neighbors within two hops away can be obtained with following several steps:

1. Each node broadcasts its node information (e.g., node id) as well as location information to its radio range neighbors.

2. Each node selects k nearest neighbors base on a topology control algorithm, e.g., *K-Neigh*. Each node then set up its maximum transmission power to reach the furthermost immediate neighbor.

3. Each node broadcasts its neighbors' information to its one hop neighbors with its maximum transmission power.

4. Each node then calculate the set of *symmetric neighbor* based on the information received from its one hop neighbors. A symmetric neighbor b of node a means that node a considers node b as a neighbor if and only if node b also considers node a as a neighbor.

Because above steps employ random transmissions, it is possible for a packet sent by a node to collide with packets sent by some other nodes. Due to lack of acknowledgement in packet transmission, large contention windows and/or multiple retransmissions are need for a node to achieve the delivery of a packet to its one hop neighbors with high probability. Blough *et al.* [7] proved the crude lower bound that no contention occur in a wireless channel with the following lemma:

Lemma 1 *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$(-3h(h-1)/2m)$, where h denotes the number of nodes that are contending for the channel and m denotes the contention window size.*

If a packet can be transmitted multiple times, the probability to achieve at least one successful delivery can be expressed as follows:

$$p = 1 - \left(1 - \exp\left(-\frac{3h(h-1)}{2m}\right)\right)^n \quad (13)$$

where n is the number of times that a packet will be transmitted. Figure 5 plots the probability for a successful delivery with 20 and 30 contending nodes. We see that to achieve probability of successful delivery near 1,

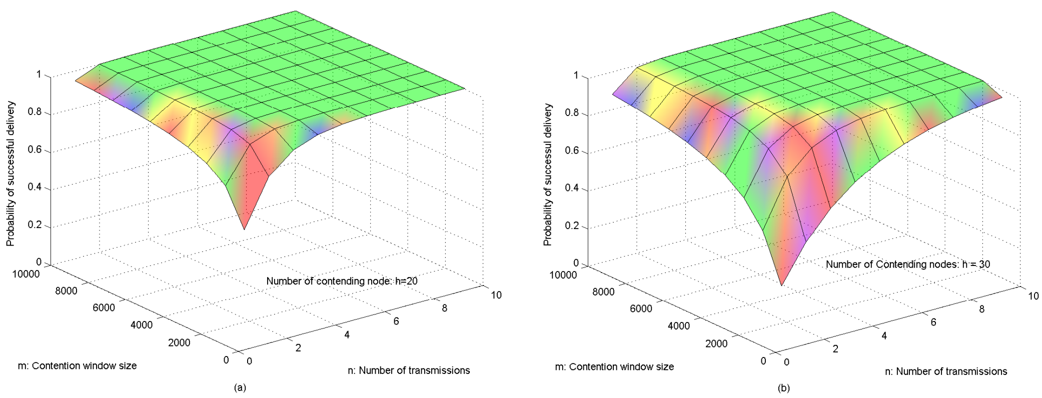


Figure 5: *The probability of a successful delivery with 20 and 30 contending nodes.*

we can either increase the contention window size or number of transmissions or both. For example, assume that there are 20 contending nodes, further assume that we use $m = 10000$ contention window size, Figure 5 shows that to achieve probability of a successful delivery near 1, we can set the number of transmissions to 2. With the transmission time of each packet is in the order of milliseconds, assume that $m = 10000$, the duration for each re-transmission period is about tens of seconds. Even we assume that each node transmits a packet several times, the total time period is within several minutes. Comparing to the lifetime of a sensor node, which is normally several months or years, the network setup time is almost negligible. Moreover, the assumption in Lemma 1 is that each node transmits directly based on its randomly selected contention windows without considering any other transmissions in its vicinity. If we assume that each node has carrier sense capability, much smaller contention window size can be used in practice.

6.2 The Distributed Channel Allocation Protocol

Figure 2 (a) illustrates the frequency channel allocation pattern during the steady state. We now describe the distributed channel allocation algorithm to achieve the frequency allocation pattern. We assume that sensors are asynchronous and the broadcast of each node can be successfully received by all of its one hop neighbors. We will relax this assumption at the end of this subsection.

Assume that a sensor network has n nodes and each node can be ordered by assigning a unique identifier $1, 2, \ldots, n$ according to any specified criterion (e.g., a unique node id, decreasing/increasing number of neighbors, etc.). Let $H1(i)$ be the set of *one* hop neighbors j of i such that $j < i, \forall i = 1, 2, \ldots, n$. Let $H2(i)$ be the set of *two* hop neighbors j of i such that $j < i, \forall i = 1, 2, \ldots, n$. We denote PACKET($i, k$) the control packet exchanged between a node and its neighbors in the algorithm which represents that channel k has been *definitively* assigned to node i. We also assume that each node maintains a *ChannelPool* which contains all available channels that can be assigned to this node. The distributed channel allocation algorithm is shown in Algorithm 1

```
Input   : H1(i), H2(i), ∀i = 1, 2, ..., n; ChannelPool: all available channels.
Output  : Each node is assigned a unique channel which is different from its one hop and two hop neighbors.
counter = | H1(i) | + | H2(i) | ;
while counter > 0 do
    if PACKET(j, k) is received from j then
        counter = counter - 1 ;
        remove k from ChannelPool;
        broadcast PACKET(j, k);
    else if PACKET(j, k) is received and j ∈ H2(i) then
        counter = counter - 1;
        remove k from ChannelPool;
    else
        discard PACKET(j, k)
    end
end
k' = random select a channel from ChannePool;
channel[i] = k' ;
broadcast PACKET(i, k');
```

Algorithm 1: Distributed Channel Allocation Algorithm.

During the channel assignment phase, each node monitors the common control channel and *only processes control packets coming from its one hop neighbors*. A node can not assign a channel to itself until its *counter* reaches zero. As an example, we show the channel assignment process for node G in Figure 2 (a). We assume that the ordering of nodes is in alphabetical order. It is easy to obtain that $H1(G) = A, B, F$ and $H2(G) = C, D, E$, so $counter = |H1(G)| + |H2(G)| = 6$ initially. When a PACKET(j, k) arrives at node G from a node h when G's counter is larger than zero, following scenarios may happen:

1. If $h \in H1(G)$ and $j = h$, thus $j < G$, e.g., PACKET($A, f5$) from node A, node G decreases *counter* by one, removes $f5$ from *ChannelPool*, and marks A as checked[2]. Node G then re-broadcasts PACKET($A, f5$) to *its* one hop neighbors.

2. If $h \in H1(G)$ and $j \in H2(G)$, and so $j < G$, e.g., PACKET($D, f3$) from node A, node G decreases *counter* by one, removes $f3$ from *ChannelPool*, and marks D as checked.

3. If $j \notin H1(G) \cup H2(G)$, node G discards PACKET(j, k).

[2]Further copies of PACKET(j, k) that is re-broadcasted from G's other one hop neighbors will be discarded.

As soon as PACKET(j, k) has been received from all $j \in H1(G) \cup H2(G)$, $counter = 0$, node G will self-assign a channel, say $f1$, through a random selection from the *ChannelPool*. Node G then broadcasts PACKET($G, f1$) to its one hop neighbors. Note that even after node G has assigned itself a channel, it must continue to re-broadcast PACKET(j, k) from its one hop neighbor h if $j = h$.

We next validate the correctness of our proposed algorithm. We represent a sensor network by a graph $G(V, E)$ after the *topology formation* phase, where V is the set of nodes, and E is the set of links. Further assume that the node degree of G is Δ, and there are n nodes in the network. The correctness of our algorithm is then given by following theorem.

Theorem 1 *Algorithm 1 converges (e.g., no deadlock) with correct channel assignments. The number of control packet needs to be sent is upper bounded by $O(\Delta)$ per node and $O(n\Delta)$ for the whole network.*

Proof: To validate the correctness of Algorithm 1, note that:

1. Each node waits for the control packets coming from its one hop and two hop neighbors that having smaller identifiers before it can assign a channel to itself.
2. There is at least one node having empty $H1$ and $H2$ set, which can immediately assign itself a channel and initiate the algorithm.
3. Each node only randomly selects its own channel from *ChannelPool* which excluded channels that have been assigned by its one hop and two hop neighbors with smaller identifiers.
4. Each node needs to broadcast its own control packet as well as control packets PACKET(j, k) from all its one hop neighbors h if $h = j$. In the worst case, each node needs to broadcast $\Delta + 1$ packets if we assume that all nodes are asynchronous. Thus the number of control packet needs to be sent is $O(\Delta)$ for each node. Since there are n nodes in the network, the overall packets that need to be sent is upper bounded by $O(n\Delta)$.

Above conditions together guarantee that no deadlock can occur and the distributed algorithm converges within a finite time.

Our NS-2 simulations (see 7.2) also demonstrate that the algorithm converges with correct channel assignments. In the worst case, a node may receive $|H1| + |H2| = \Delta^2$ control packets before it can self-assign a channel, so the run time complexity of Algorithm 1 is $O(\Delta^2)$.

At the beginning of this section, we assume that sensors are asynchronous and can communicate by exchanging control messages successfully, e.g., a broadcast packet from node i can be successfully received by all i's one hop neighbors. Because the channel assignment process is sequential (e.g., each node only assigns channel to itself after all its neighbors in $H1 \cup H2$ have done so) in Algorithm 1, the collision probability is actually much lower than normal broadcast traffic. However, our simulations reveal that potential collisions still exist.

To resolve this problem, we can employ large contention windows and/or allow each node transmit a packet multiple times. But because the run time complexity of Algorithm 1 is $O(\Delta^2)$, it may take a bit longer to achieve optimized channel allocation if the node degree is large. An alternative is to employ a random channel allocation scheme (as we used in mathematical modeling of MAI in section 5) at the beginning of network formation. For example, each node only selects its frequency and PN code randomly and broadcasts this information during the *topology formation* phase. Random assignment of channels does not disrupt communications but only degrades the performance. We will see later from our simulation results that a random assignment scheme also achieves reasonably high performance. With this random allocation scheme, the network can become operational in a short time. After this, Algorithm 1 can be used to gradually correct the channel assignment to improve the network performance.

6.3 Frequency Assignment and Code Assignment

In this subsection, we present the theoretical value of the number of required frequencies and PN codes. The number of required channels in practice can be found in simulation section 7.2.

The frequency assignment is a coloring problem in graph theory considering both one hop and two hop neighbors. This coloring problem can be expressed as: *no two nodes (vertices) receive the same color if they are either adjacent, or have a common neighbor*. In the worst case, a node can be assigned color $1 + \Delta + \Delta(\Delta - 1) = \Delta^2 + 1$. So the upper bound of the number of required colors (channels) k can be expressed as:

$$k = min\{\Delta^2 + 1, |V|\} \quad (14)$$

Thus any graph (network) can be colored with $O(\Delta^2)$ colors. In our protocol design, frequency assignment can be achieved by using Algorithm 1.

There are several schemes for the PN code assignment: *receiver-based, transmitter-based, or pairwise code assignment* [14][31]. Since frequency channel assignment is receiver-based, PN code assignment should use either transmitter-based, where each neighbor of a given node should have a different code for transmitting; or transmitter-receiver pair (link) based, where no two adjacent directed links have the same code. The receiver-based code assignment can not be used because two concurrent transmissions to a same node will be indistinguishable. The transmitter based code assignment is a slightly different coloring problem compared to the frequency assignment. It can be expressed as follows: *nodes (vertices) sharing a common neighbor can not have the same color*. The minimum number of required colors (codes) k is given by

$$k = min\{\Delta(\Delta - 1) + 1, |V|\} \quad (15)$$

The pairwise scheme is an edge coloring problem in graph theory and requires a smaller number of colors (codes) than transmitter based scheme, e.g.,

$$k = \Delta \quad (16)$$

Both schemes can be used in our protocol.

The transmitter based code assignment can be achieved by using Algorithm 1 with following minor modifications. When node i receives a PACKET(j, k) from node j, node i checks if it has a common neighbor with node j, if no, node i will *not* remove k from the *ChannelPool*. For pairwise code assignment, a much simpler algorithm can be used. A given node X may select monitoring PN codes for all of its one hop neighbors and then broadcast its selections to them. Each one hop neighbor of node X then uses the pre-determined PN code by X for transmitting to X.

7 Simulations and Analysis

A simulation study has been performed using the discrete event network simulator NS-2. Our simulations include four sections:

1. Evaluation of channel allocation protocol and number of channels required in practice.
2. Comparison of different broadcasting schemes proposed in section 4.2.
3. One hop performance comparisons.
4. Multiple hop performance comparisons.

From now onwards, we refer our protocol as CDMA sensor MAC (CSMAC). The name comes from our earlier work in [62].

7.1 General Simulation Setup

In our simulation, frequency division is implemented at the wireless physical layer in NS-2. If a packet is received within the allocated frequency band of the node, the packet will be passed to the upper layer (MAC) for further processing; otherwise the packet is discarded. Directed sequence spread spectrum (DSSS) is implemented 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.

All simulations are conducted based on the same network topology structure: where 100 nodes are uniformly randomly deployed in a $100m \times 100m$ square area. We adopted a simple topology control protocol *K-Neigh* [7] in our simulations.

One of the most important parameter in the simulation is the *MAI threshold*, which is defined as the maximum ratio between the total interference signal power and the desired signal power. We consider a simple DS-CDMA system, where BPSK modulation and a convolution code with rate $1/2$ are used. Further assume that the processing gain $L = 50$ and the required $E_b/N_{0\text{eff}}$ is 5 dB. Ignoring the thermal noise, the *MAI threshold* is calculated using equation 1:

$$\textit{MAI Threshold} = \frac{\sum_{i=1}^{k} P_i}{P_0} = 23.72 \qquad (17)$$

where this number represents that when the ratio between the interference signal power and the desired signal power is larger than 23.72, the packet will be destroyed due to the MAI.

We adopted a simple energy dissipation model that is used in LEACH [22]. The model is shown in Figure 6. This model separates the electronic energy consumption and power amplifier energy consumption

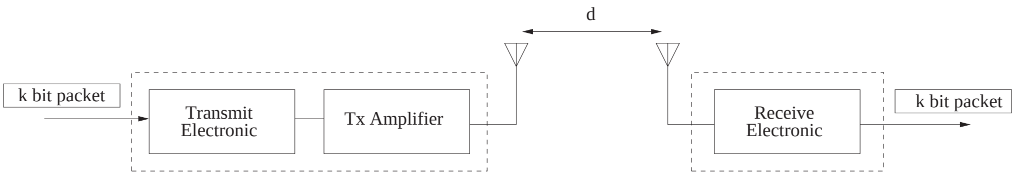


Figure 6: *Radio energy dissipation model*

at the transmitter side. We also assume that the transmission power can be adjusted with 1 dBm step. The parameters used in our simulation are shown in Table I.

Table 1: Parameters used in simulations.

Processing Gain	50	Antenna Gain	1
MAI Threshold	23.72	System Loss	1
Data Rate	20 Kbps	Rx Threshold	1e-10 W
Data Packet Size	50 Bytes	Carrier Sense Threshold	1e-11 W
CSMA Control Packet Size	10 Bytes	Tx Electronic Power	10e-3 W
CSMA Contention Window Size	31	Rx Electronic Power	10e-3 W
Capture Threshold	10 dB	Maximum Tx Power	5e-3 W
Radio Propagation Model	Log distance path loss	Tx Power Step	1 dBm
Path Loss Exponent	3	ISM Frequency Band	2.4-2.4835 GHz

7.2 Evaluation of Channel Allocation Protocol

Equation 14 gives the crude upper bound on the number of required channels for a *proper* channel assignment, which is proportional to the square of node degree. Fortunately, our simulation results reveal that the practical number of required channels are much smaller than the theoretical value given by equation 14 when nodes are uniformly randomly distributed.

To find out the practical value of the number of required channels for a *proper* channel assignment, we generated 10000 network topologies where each network topology has 100 nodes uniformly randomly distributed in a $100m \times 100m$ square area. We evaluate the number of channels used in each network topology after the channel allocation by using Algorithm 1 with node degree $\Delta = 6$ and $\Delta = 9$ in *K-Neigh*. The empirical distribution of the simulation results are shown in Figure 7. We see that when node degree $\Delta = 6$, the maximum number of channels required is only 13 compared to the theoretical value 37 given by Equation 14. When node degree $\Delta = 9$, the maximum number of channels required is only 18 compared to the theoretical value 82 given by Equation 14. Simulation results also reveal that our channel allocation algorithm converges with correct channel assignment.

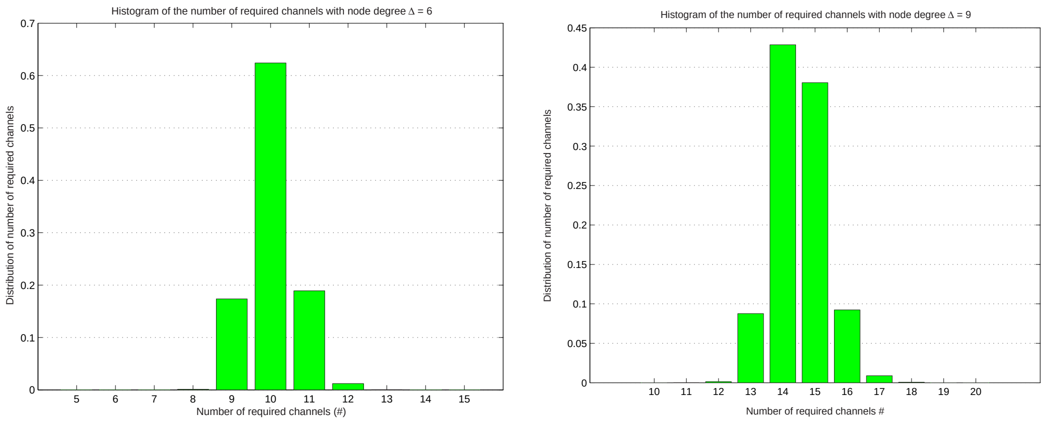


Figure 7: *Distribution of number of channels required with node degree* $\Delta = 6$ *and* $\Delta = 9$.

7.3 Comparison of Different Broadcasting Schemes

Three broadcast schemes have been proposed in section 4.2: 1) each node sends multiple unicast packets to its one hop neighbors; 2) each node broadcasts with different PN codes; 3) each node broadcasts with the same PN code. The first scheme seems to be the most inefficient approach, but interestingly, it achieves the least energy consumption. 100 network topologies are selected (from the 10000 network topologies generated in section 7.2) to conduct the simulations. Each network topology is selected such that when node degree $\Delta = 6$, the network is fully connected[3]. For each topology, a random node is selected as the source node which initiates the broadcast packet. This packet is then *flooded* to the whole network. Three critical parameters are measured in our simulations:

1. *Delivery ratio*: this parameter measures the ratio between the number of nodes that receive the packet in the network and the total number of nodes in the network.

2. *Delivery time*: this parameter measures the time for flooding to complete.

3. *Energy consumption*: this parameter measures the network energy consumption after flooding.

Two broadcast types are implemented in the simulation:

1. *Blind flooding*: where each node blindly transmits a broadcast packet to all of its one hop neighbors. With the first broadcasting scheme, each node unicasts the packet to all of its one hop neighbors in turn. With the second and third broadcasting schemes, each node sets the transmission power that can reach the furthermost one hop neighbor to transmit the packet.

2. *Optimized flooding*: where each node transmits a broadcast packet by using a simple optimized algorithm as follows. When a node receives a packet, it finds out its one hop neighbors that are *not* one hop neighbors of the sender (assume that each node gathered the neighbor information of its one hop neighbors during the *topology formation* phase) and only sends the packet to these neighbors. With the first

[3]Note that *K-Neigh* protocol does not guarantee the network connectivity. It only proved that when $\Delta = 9$, the network is connected with high probability (e.g., 0.95). We only select those network topologies that are fully connected when $\Delta = 6$.

broadcasting scheme, each node only unicasts the packet to those neighbors that are not one hop neighbors of the sender. With the second and third broadcasting schemes, each node sets its transmission power to reach the furthermost one hop neighbor that is not one hop neighbor of the sender.

Figure 8 plots our simulation results and Table II lists the average values. The notation used in the legend of figures and table are as follows: MUCAST (Multiple unicast) denotes the first broadcasting scheme, CDMA denotes the second, and CSMA denotes the third. We also use these notations in our following discussions.

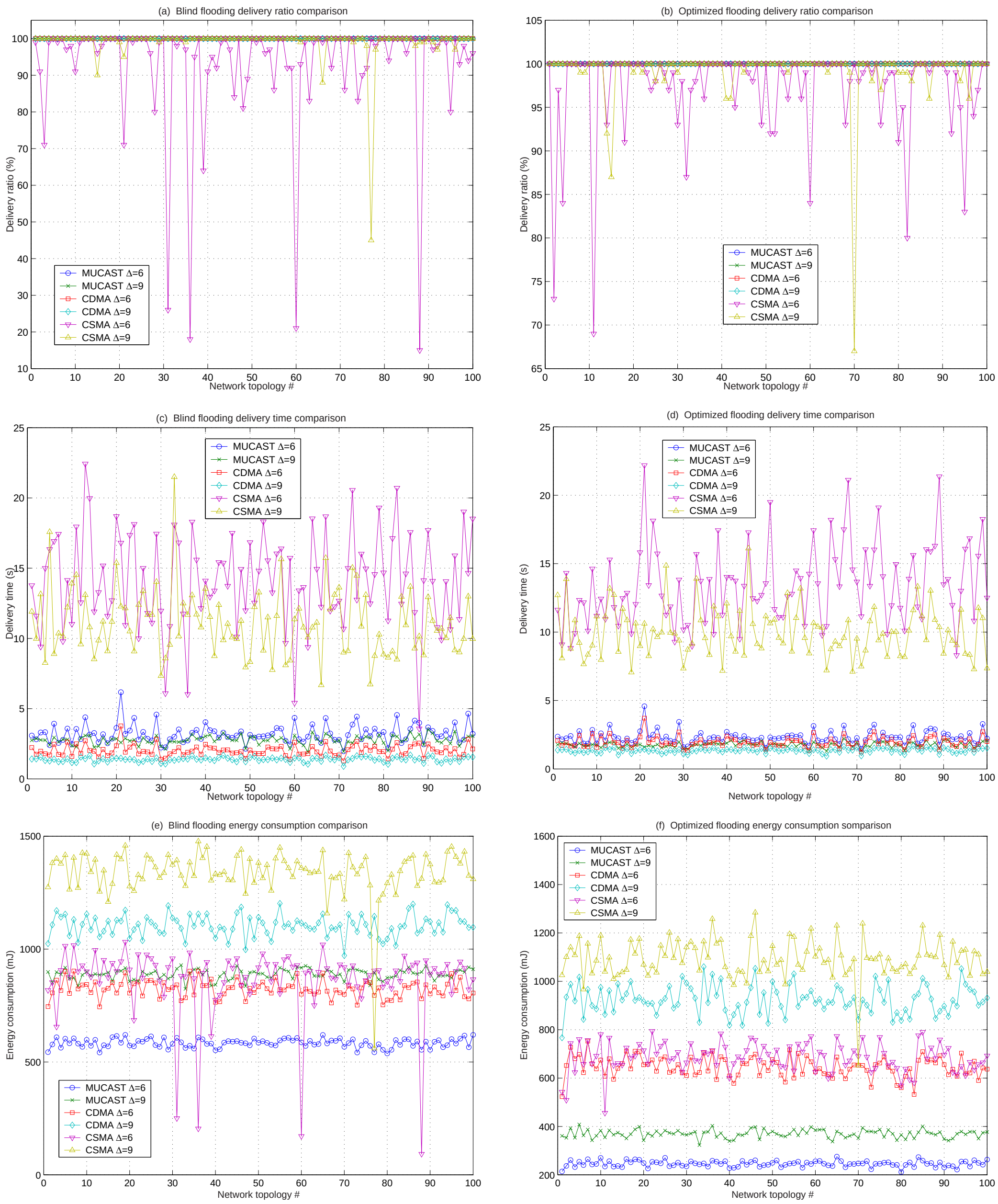


Figure 8: Comparisons of different broadcasting schemes.

From Figure 8 (a) (b) and Table II, we can see that both MUCAST and CDMA achieve 100% delivery ratio with node degrees $\Delta = 6$ and $\Delta = 9$, either blind flooding or optimized flooding. For CSMA with blind flooding, node degree $\Delta = 9$ achieves better average delivery ratio (98.97%) than $\Delta = 6$ (92.66%). Similar trend is exhibited in the optimized flooding. This is because when the node degree $\Delta = 9$, the average transmission range of each node is larger than when $\Delta = 6$. A packet is more likely to be successfully received by each node. We also note that CSMA with optimized flooding achieves better delivery ratio than

Table 2: Average values for different broadcasting schemes

Parameters	Blind flooding					
	MUCAST		CDMA		CSMA	
	$\Delta = 6$	$\Delta = 9$	$\Delta = 6$	$\Delta = 9$	$\Delta = 6$	$\Delta = 9$
Delivery ratio (%)	100	100	100	100	92.66	98.97
Delivery time (s)	3.213	2.765	2.025	1.388	14.107	11.101
Energy consumption (mJ)	583.74	888.51	824.00	1103.96	862.97	1340.73

Parameters	Optimized flooding					
	MUCAST		CDMA		CSMA	
	$\Delta = 6$	$\Delta = 9$	$\Delta = 6$	$\Delta = 9$	$\Delta = 6$	$\Delta = 9$
Delivery ratio (%)	100	100	100	100	97.07	99.05
Delivery time (s)	2.311	1.767	1.948	1.341	13.397	10.016
Energy consumption (mJ)	245.75	369.81	646.30	925.69	684.68	1093.59

blind flooding. The reason is that with optimized flooding, each node may reduce its maximum transmission power which leads to the reduction of interference to others. The consequence is that the probability for a successful receiving at each node is increased. Although CSMA achieves an average delivery ratio above 90%, its worst case performance can be very poor. In some simulations, the delivery ratio may be lower than 70% with optimized flooding (Figure 8 (b)) and even lower than 20% with blind flooding (Figure 8 (a)) due to packet collisions. Note that with CDMA, packets may also be dropped due to multiple access interference (MAI). But because of the code division approach, the probability for a successful receiving at each node is increased significantly compared to CSMA. For example, with CSMA, a packet colliding with other packet(s) at a receiver will be dropped unless the ratio between the intended signal and interference is over the *capture threshold*[4]. With CDMA, a packet colliding with other packet(s) may only be dropped if the ratio between the total interference signal power and the intended signal power is over the *MAI threshold*. With MUCAST, MAI only occurs when multiple nodes transmit to the same receiving node simultaneously. The MAI is actually reduced significantly than in CDMA due to frequency division.

Figure 8 (c) (d) illustrate the delivery time comparison. We can see that both MUCAST and CDMA achieve much better performance than CSMA. For example, with blind flooding and $\Delta = 6$, MUCAST uses 77% less mean delivery time and CDMA uses 85% less mean deliver time than CSMA (Table II). In CSMA, each node needs to contend the media by using distributed coordination function (DCF) with contention windows and back off its transmission if the media is busy. This approach induces large delays when many neighbor nodes attempt to transmit simultaneously. In CDMA, a node only delays its transmission when it is at receiving state. If a node is receiving, a transmission from the same node will drown out any received signals because the transmission power is too high comparing to those receiving signals. In MUCAST, each node does not delay a transmission (because of frequency division) but needs to transmit a broadcast packet multiple times to all (blind flooding) or correspondent (optimized flooding) one hop neighbors. The overall result is that MUCAST spends longer delivery time than CDMA. But MUCAST still achieves much better performance than CSMA.

Interestingly, Figure 8 (e) (f) and the Table II reveal that MUCAST achieves the least energy consumption compared to CDMA and CSMA although MUCAST seems to be the most energy inefficient scheme. For example, MUCAST achieves 29.1% and 32.3% average energy savings compares to CDMA and CSMA respectively with blind flooding when node degree $\Delta = 6$. Although MUCAST requires that each node transmits a broadcast packet multiple times, it saves the receiving energy consumption caused by multiple overhearing. Remember that the interference range (R_I, see section 5) of a transmission is much larger than the transmission range (R_R). The result is that both CDMA and CSMA consume more energy on receiving than MUCAST. The receiving energy consumption is especially important as sensors normally communicate with short range (e.g., tens of meters), where receiving energy may frequently exceed the energy of transmission.

[4]With spread spectrum modulation, it is possible for the strongest signal to successfully capture the intended receiver, even when many other users are transmitting. The capture threshold is set to 10 dB (default value in NS-2) in our simulation.

7.4 One Hop Performance Comparison

This section compares the one-hop performance of the following four MAC protocols.

1. Our protocol with the optimized channel allocation protocol proposed in section 6.2.
2. Our protocol with a random channel allocation algorithm which is used in our mathematical modeling in section 5. Both frequency and PN code are randomly selected by each node.
3. Pure CDMA approach without using frequency division.
4. A contention based MAC protocol: SMAC [23].

We denote these four protocols as CSMAC, RAND, CDMA, SMAC and use these notations in the legend of figures and discussions in this and next subsection. The node degree used in this section is $\Delta = 6$. The transmission range of SMAC is fixed and is calculated[5] according to $R_R = \sqrt{(\Delta+1)/(\pi\lambda)}$, where $\lambda = 0.01$ denotes the node density. In our simulation, we do not enable the synchronization flag in SMAC NS-2 implementation so there is no synchronization overhead. We assume that SMAC also employs spread spectrum (as IEEE 802.11) for better channel performance. Transmitter based PN code assignment is used for the other three protocols.

The same 100 network topologies selected in last section are used. Four parameters are measured: *delivery ratio, network throughput, one hop latency, and network energy consumption.* We measured the performance based on different packet generation rate, which scales from the lowest 0.01 packet/s to the highest 30 packet/s. Note with our parameter settings in Table I, the full transmission capacity of a node is around 23.25 packet/s.

In our simulations, each node randomly selects a neighbor and transmits 100 packets to this neighbor. CBR traffic is used but each packet's departure time is randomized according to a uniform distribution (refer to NS-2 manual). With this randomization, the packets generated from each node can be well approximated as a Poisson process. Different transmission patterns are randomly generated where each node can transmit to different neighbors. In each run, there are 10000 packets (100 nodes $\times$ 100 packet/node) transmitted in the whole network. The start time of each node's transmission is also randomized according to the packet transmission interval. This is to ensure that each node can start transmitting at similar time but not exactly the same time which may influence the performance of the contention based protocol. The idle energy consumption is not included in this study as it only adds a constant mean value to our results and it may hide the effect of communication energy consumption.

Figure 9 (a) shows the delivery ratio comparison with packet generation rate of 0.1 for all 100 network topologies and Figure 9 (b) shows the average delivery ratio versus different packet generation rates. Figure 9 (c) and (d) illustrate the correspondent network throughput comparison. The delivery ratio is defined as the ratio between the number of packets that have been successfully received and the total number of packets that have been transmitted. The throughput is then calculated as the product of delivery ratio and correspondent packet generation rate. We see that CSMAC achieves 100% delivery ratio with all packet generation rates and its throughput increases linearly with the increase of packet generation rate. RAND also achieves high delivery ratio with small degradation when the packet generation rate is over 1 packet/s. CDMA delivery ratio drops significantly when the packet generation rate is over 1 packet/s and drops below 80% when packet generation rate reaches the full transmission capacity (23.25 packet/s). The throughput of RAND and CDMA also increase linearly with small degradation in higher traffic scenario, where the MAI threshold is easily exceeded due to many concurrent transmissions. As a consequence more packets are damaged because of MAI. SMAC performs the worst with delivery ratio drops below 90% when the packet generation rate reaches 0.1 and drops to an unacceptable level (e.g., below 30%) when the packet generation rate reaches 0.5. Its throughput also reaches the bottleneck when the packet generation rate is over 0.1 packet/s. Because of the

[5] Derived from $\lambda = (\Delta+1)/(\pi R_R^2)$, where πR_R^2 represents the area which is covered by the transmission and $\Delta+1$ represents the number of nodes in the transmission area.

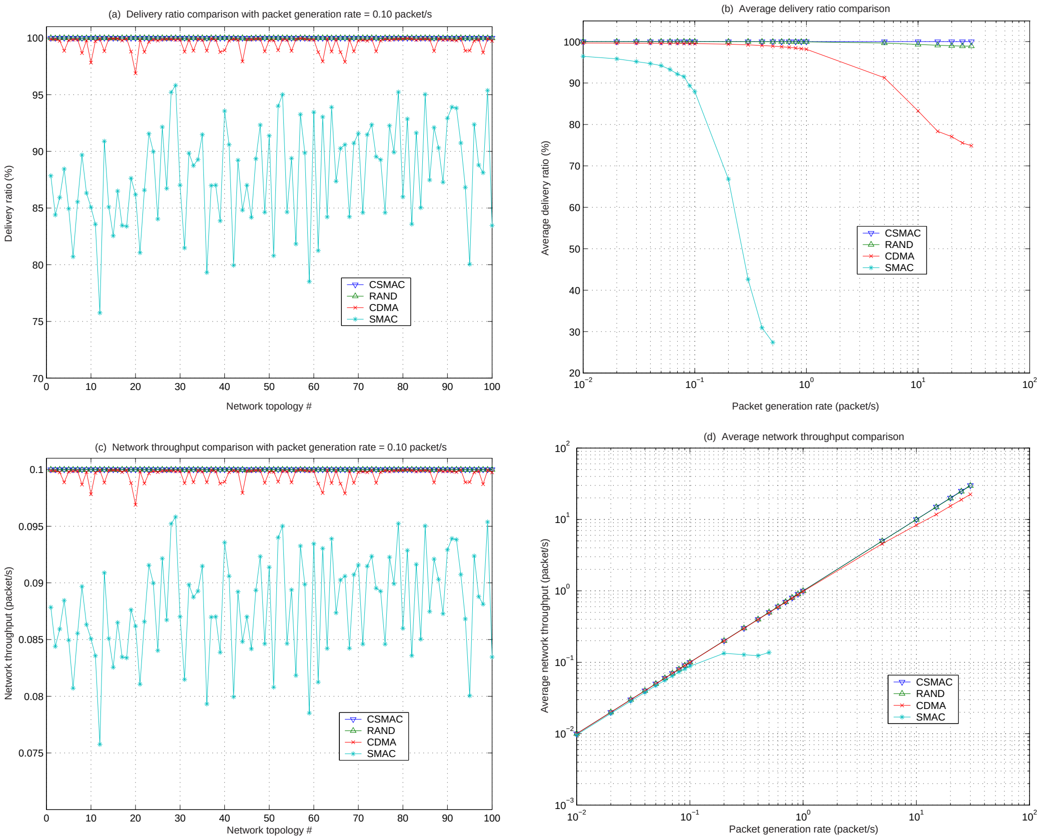


Figure 9: *Delivery ratio and network throughput comparison.*

contention based approach, many packets are dropped due to collisions and exceeding the number of retry limit for transmitting with the increase of packet generation rate.

We define the *effective packet rate* as the maximum packet generation rate that a system can achieve with an acceptable delivery ratio. If we set the acceptable delivery ratio to 90%, we see that our proposed system can accommodate several hundred times of *effective packet rate* (at 20-30 packet/s) than the contention based system (at 0.1 packet/s). Because CSMAC and RAND employ multiple channels (e.g., 13 with node degree $\Delta = 6$, see section 7.2) but CDMA and SMAC use only one channel, in order to make a fair comparison, we define the *system efficiency* as follows.

$$System\ Efficiency = \frac{Effective\ Packet\ Rate}{Number\ of\ Channels} \tag{18}$$

At 90% delivery ratio, the *system efficiency* of CSMAC and RAND is around 1.54-2.31 packet/s/channel, CDMA is around 5 packet/s/channel, and SMAC is around 0.1 packet/s/channel. It can be seen that CSMAC and RAND achieve 15-20 times improvement over SMAC. However, for a delivery ratio of 98%, CDMA *system efficiency* changes to 1 packet/s/channel while CSMAC and RAND are still 1.54-2.31 packet/s/channel. The relative reduction in system efficiency of CDMA in comparison with CSMAC can be explained as follows: At low packet generation rate (e.g., less than 1 packet/s), both CDMA and CSMAC achieve a delivery ratio above 98% but CSMAC is less efficient because it uses more channels. However, at high packet generation rate, CDMA has a high MAI which causes a reduction in the delivery ratio and throughput. By using only a small number of channels, CSMAC has a lower MAI and higher system efficiency. This shows that only a small number of frequency channels is required to achieve high system performance.

Figure 10 (a) shows the average one-hop latency comparison with packet generation rate 0.1 for all 100 network topologies. Each average value is calculated based on all packets that have been successfully received in each run on each network topology. Figure 10 (b) shows the average one hop latency comparison versus different packet generation rate. In this figure, each average value is calculated based on all 100 average one hop latencies over the 100 network topologies with a given packet generation rate. We see that CSMAC, RAND, CDMA achieve almost the same performance until the packet generation rate is over 1 packet/s. In our simulation implementation, a node may delay its transmission in a given frequency if it is at receiving state in the same frequency. Because the transmission signal power is too high comparing to the receiving signals that it may drown out all receiving signals. The latency of these three protocols increase sharply when the packet generation rate reaches the full transmission capacity where the queueing delay becomes dominant. It is to be expected that SMAC performs worse than the other three protocols because each node has to contend for the media to transmit. The graph shows that the one hop latency of SMAC increases steadily with the increase in traffic load. On the contrary, the one hop latency of CSMAC, RAND, CDMA is almost constant and is not influenced by the increases of traffic load. Normally they do not need to reserve media (by using control packet such as RTS/CTS) for a transmission. A node simply transmits a packet if it has one.

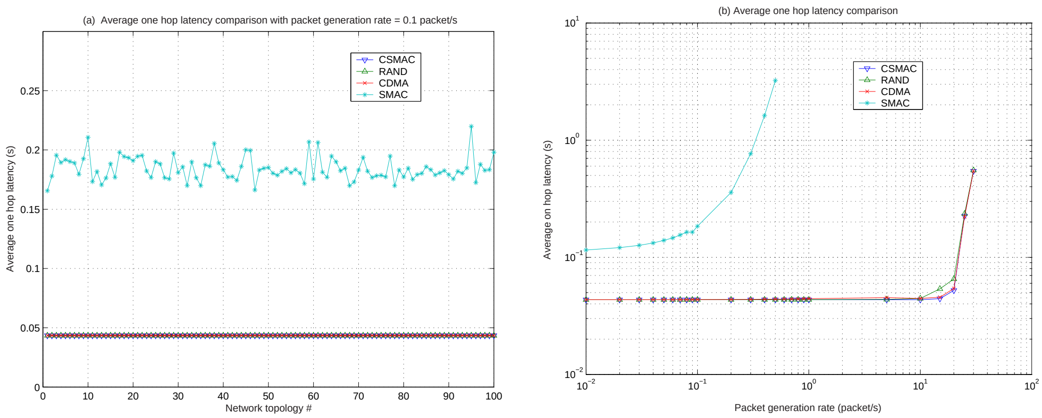


Figure 10: *One hop latency comparison.*

Figure 11 (a) shows the network energy consumption comparison with the lowest packet generation rate 0.01 for all 100 network topologies and Figure 11 (b) shows the network energy consumption comparison versus different packet generation rate. We see that SMAC consumes the largest amount of energy at the lowest packet generation rate compare to at other packet generation rates. The energy consumption drops steadily with the increase of traffic load. When the traffic load increases, the channel contention also increases. As a consequence more and more packets are dropped due to exceeding the number of retry limit, which leads to less energy are consumed in transmissions of data packets.

Figure 11 (b) reveals that CSMAC and RAND consume the lowest communication energy, with average around 12 J. CDMA consumes average energy around 60 J. SMAC consumes average 190 J. There are several reasons that SMAC consumes more energy than the other three protocols. First, control packet exchange consumes large amount of energy. With our parameter settings in Table I, the control packet contributes 37.5% overhead. Second, overhearing consumes energy. Actually, SMAC implements an *overhearing avoidance* scheme to turn off a node's transceiver when it overhears an RTS or CTS that is not destined to itself. The scheme is to avoid a node wasting energy by receiving subsequent data and acknowledgement packets. However, the problem is that the interference range of a transmission is much larger than the normal transmission range, e.g., $R_I \gg R_R$ ($R_I \approx 2R_R$, see section 5). A node may detect an RTS or CTS but can not correctly receive it. The result is that *overhearing avoidance* scheme does not function for those nodes located between R_R and R_I. Third, packet collision wastes energy. Although RTS/CTS are used, packet collision can not be fully avoided (including collisions of RTS/CTS). Collided packet need re-transmission and consume extra energy. Fourth, CSMA/CA back off scheme may cause additional control packet transmissions.

Figure 11 shows that CSMAC and RAND consume less than 10% communication energy as in SMAC. In our simulation, the Tx and Rx electronic power consumption for all of the four protocols are set to the same value. But a multiuser detection receiver may consume a bit more power than a single user detection receiver.

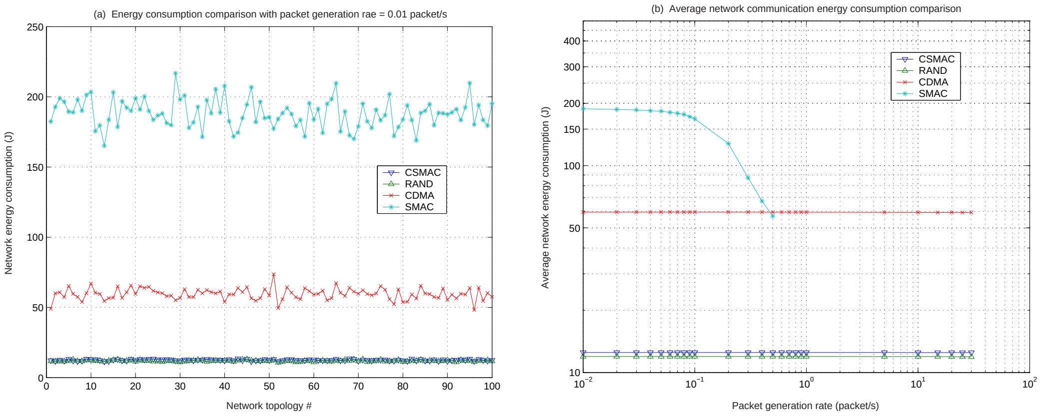


Figure 11: *Network communication energy consumption comparison.*

What we want to show here is the comparative result. If a multiuser detection receiver does not consume 10 times power than a single user detection receiver, it is sensible to use our proposed multi-channel protocol to achieve energy savings. In addition, our protocol achieves much better network throughput, system capacity, and one hop latency performance.

7.5 Multiple Hop Performance Comparison

In this section, we measure the performance of our proposed protocol with a well known routing protocol specially designed for sensor networks: *directed diffusion* [17]. The same 100 network topologies are used in this section. In each run, one randomly selected node acts as the sink and multiple randomly selected nodes act as sources. The sink node only broadcast its *interest* one time and each source node sends the *attribute-value* data packet back to the sink periodically. In our simulation, each source sends 50 packets back to the sink. The departure time of each packet from a source is also randomized according to a uniform distribution to approximate a Poisson process. The average packet generation rate from each source is 0.1 packet/s. The parameters measured in this section include: *delivery ratio*, *multiple hop latency*, and *energy consumption*.

Figure 12 (a) shows the delivery ratio comparison for all of the 100 network topologies with 15 sources and Figure 12 (b) shows the average delivery ratio versus different number of sources. The delivery ratio is

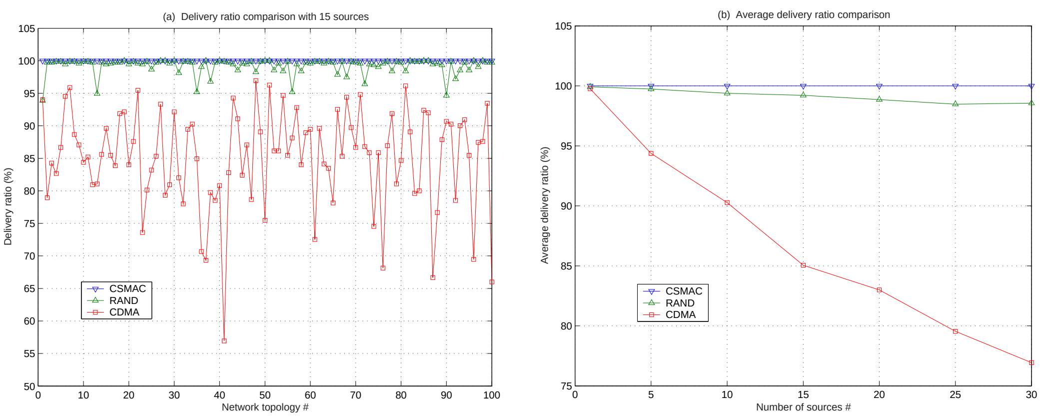


Figure 12: *Multiple hop delivery ratio comparison.*

defined as the ratio between the number of packets that have been successfully delivered back to the sink and the number of packets that have been sent from sources. We see that CSMAC achieves 100% delivery ratio and RAND also achieves reasonably high performance with 2% degradation when the number of sources is over 25. CDMA delivery ratio degrades steadily with the increase in the number of sources and drops below 80% when the number of sources is over 25 in the network. The reason for this performance degradation of CDMA is as follows. With the same packet generation rate for each source, the MAI increases with the increase of number of sources, which in turn causes more packets to be damaged by MAI. The small degradation of RAND is caused by similar reason because each node's frequency and PN code is randomly selected and there are possibilities that two (or more) neighbors select the same frequency channel or PN code.

Figure 13 (a) shows the average multiple hop latency for all the 100 network topologies with 15 sources and Figure 13 (b) shows the average latency comparison versus different number of sources. The average multiple hop latency for each network topology in Figure 13 (a) is calculated with following equation

$$\overline{L} = \frac{\sum_{i=1}^{n} \lambda_i \overline{L}_i}{\sum_{i=1}^{n} \lambda_i} \tag{19}$$

where $\overline{L}$ denotes the average multiple hop latency for a given network topology, n denotes the number of sources in the network, λ_i denotes the average packet generation rate for source i (in our case, all λ_i equals to 0.1 packet/s), $\overline{L}_i$ is the calculated average multiple hop latency based on all the packets generated from source i that have been successfully received by the sink. Figure 13 (b) shows the average of $\overline{L}$ over the 100 network topologies. It can be seen that for all three protocols, the average multiple hop latency increases with the increase in number of sources. There are two main reasons for this behavior. First, an intermediate routing node may receive multiple packets concurrently but needs to send them out sequentially. This introduces some queueing delay at the transmitter. When the traffic load increases, the queueing delay also increases. Second, a node may delay its transmission in a given frequency if it is in receiving state in the same frequency. When the traffic load increases, the probability for a given node to be at receiving state also increases which leads to longer delays. The second reason has more influence on CDMA but less influence on RAND and CSMAC (because of frequency division). We see that CDMA latency increases more quickly than CSMAC and RAND in Figure 13 (b) due to this second reason.

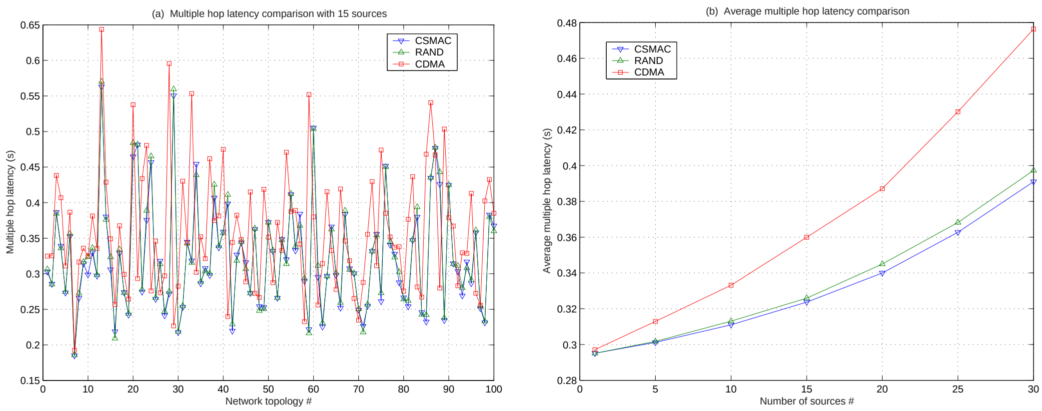


Figure 13: *Multiple hop latency comparison.*

Figure 14 (a) shows the energy consumption for all the 100 network topologies with 15 sources and Figure 14 (b) shows the average energy consumption comparison versus different number of sources. It is expected that the energy consumption of all three schemes increase steadily with the increasing number of sources. Both CSMAC and RAND achieve similar performance while CDMA consumes more energy. In general, CDMA consumes about 5 times of energy than CSMAC and RAND. The reason is that CDMA consumes more energy on overhearing than in CSMAC and RAND.

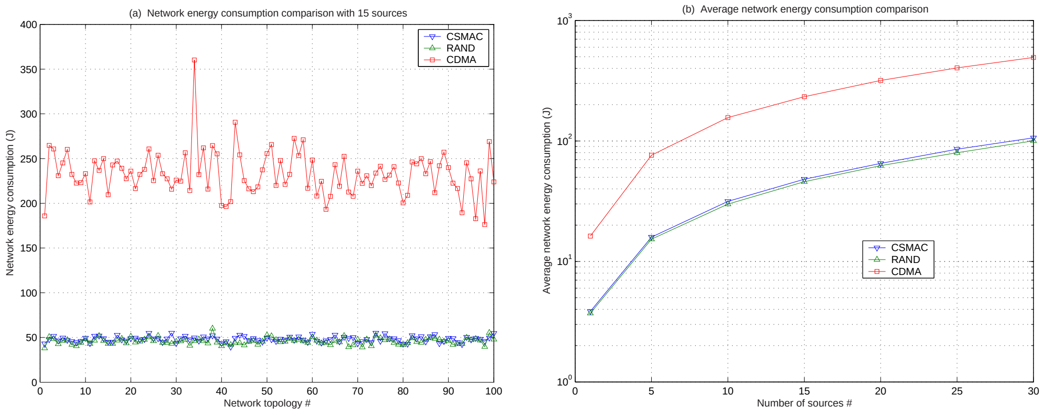


Figure 14: *Multiple hop energy consumption comparison.*

8 Related Work

DS-CDMA system and MAI related problems have been extensively studied in cellular networks which are infrastructure based. Mobile nodes communicate with base station directly and energy consumption is not a critical concern in either mobile nodes or base stations. However, the problem in sensor networks differs from the cellular framework in terms of limited resources (e.g. energy, processing) of sensor nodes, less mobility, and lack of centralized base stations. All these factors make the research problems in DS-CDMA based sensor networks different from the traditional cellular based DS-CDMA networks.

Spread-spectrum techniques have been employed in IEEE 802.11b standard. The primary issue addressed in IEEE 802.11b is to reduce the multi-path effects. Data spreading results in greater immunity to radio frequency interference as compared to narrow-band signaling. Because IEEE 802.11b is a contention based protocol, MAI is not an issue as concurrent transmissions are not allowed in the vicinity of potential interference range.

Muqattash and Krunz [8] 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. Both RTS and CTS are enlarged to accommodate MAI related information. However, our design goal is to remove control packets for energy savings. De *et al.* [9] characterized the MAI in wireless CDMA sensor networks and studied the tradeoff between interference and network connectivity. Their study revealed that high network connectivity can not be achieved without significantly increased MAI with random topology. To achieve high network connectivity and low MAI, nodes should be selectively activated such that the set of active nodes at any time lie on the vertexes of a regular polygon (e.g., square grid, hexagon, or equilateral triangle). However, this idea is not very practical for randomly deployed sensor networks. Guo *et al.* [27] proposed a set of low power MAC design principles targeting at multi-hop wireless sensor networks. Their system employs different CDMA PN codes but still requires control message exchange. Sousa [13] *et al.* characterized the optimum transmission range to maximize the throughput for a directed-sequence spread-spectrum multi-hop packet radio network. Their work assumes that each node has equal transmission power and no frequency division is used, both of these assumptions are different from ours.

Different MAC protocols have been proposed for sensor networks in literature. SMACS [6] 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. Each node maintains a TDMA frame, called super frame, in which the node schedules different time slots to communicate with its known neighbors. Network formation in SMACS is not location-aware, so neighbors selected may not be nearest. Moreover, a node must wait its turn to transmit with TDMA approach even if the channel is idle. And this waiting time can accumulate along the multi-hop route from source to sink. Interestingly, SMACS also employed frequency division but assumes a TDMA and narrow band system. Woo and Culler [18] 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 traffics. SMAC (Sensor-MAC) [23] is designed 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. TRAMA [24] assumes that time is slotted and uses a distributed election scheme based on information about the traffic at each node to determine which node can transmit at a particular time slot. It uses traffic-based scheduling to avoid wasting time slots when nodes do not have data to send and to switch nodes to a low-power standby radio mode when they are not intended receivers. Information about every two hop neighbor is used for time slot selection. The drawback of TRAMA is its high signaling overhead and high latency due to the time-slotted structure.

Channel allocation problem for ad hoc network has been studied in literature for different purposes. Most of the proposed algorithms and protocols are based on CDMA code assignment. Bertossi and Bonuccelli [34] proposed a centralized as well as a distributed code (CDMA orthogonal PN codes) assignment algorithm in multi-hop packet radio networks to eliminate the hidden terminal interference. Only two hop neighbors are considered in their algorithm because hidden terminal problem only occurs when two nodes *can not* hear each other *but* have a common neighbor. The objective is to minimize the number of codes used in the assignment. Ramanathan [37] proposed a unified framework for FDMA/TDMA/CDMA channel assignment in wireless networks, called UxDMA algorithm. UxDMA identifies eleven atomic constraints (e.g., two vertices cannot receive the same color if they are either adjacent, or have a common neighbor, etc.) underlying most current and potential assignment problems as a combination of these constraints. Based on the global topology information, UxDMA computes the node or edge colorings, which correspond to channel assignments to nodes or links in frequency, time, or code domain.

9 Concluding Remarks and Future Research

In this paper, we propose a novel multi-channel (FDMA/CDMA based) media access control protocol for wireless ad hoc sensor networks. Traditional contention based MAC protocols suffer from both low network throughput and long packet latency. Control packet (RTS/CTS/ACK) exchange produce significant overhead due to short data packet size in sensor networks. We propose to use frequency division to reduce multiple access interference (MAI) in a DS-CDMA based sensor network to achieve less channel contention, lower packet latency, higher network throughput, and less energy consumption. By employing frequency division, the uncontrollable MAI encountered in a pure DS-CDMA system is effectively reduced and great improvement in network throughput and system capacity can be achieved. We characterize analytically the expected value of MAI at a given node in relation to the number of frequency channels and show that a limited number of frequency channels can reduce the MAI significantly. Through discrete event simulation (using NS-2), we provide comparisons of our proposed system to a pure DS-CDMA system and a contention based system. Simulation results reveal that our proposed system can achieve 15-20 times of system efficiency than a contention based system. When same number of packets are transmitted in the network, our system consumes only 10% of communication energy than the contention based system.

We also propose a distributed channel allocation algorithm for the network setup phase. We prove that our algorithm is correct and converge from both analytical analysis and simulations. Our simulation results reveal that much less number of channels are required than theoretical values when nodes are uniformly randomly distributed.

Future work includes performance comparison with different topology control protocols. There are enormous topology control protocols proposed for ad hoc and sensor networks in literature and we only adopted a simplest *K-Neigh* in this paper. Further development of the mathematical model for network throughput and system capacity based on current results is also under consideration. Our current design is based on a homogeneous system architecture, where all sensors are assumed to have the same capability. How to improve our design to be compatible with a heterogeneous system architecture, where sensors may have different capabilities (e.g., high-end sensors can have more power and processing capabilities than low-end sensors), is an interesting topic. We are also seeking hardware implementation of our proposed system for further verifications. This is our next big challenge for future work.

10 Acknowledgement

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

References

[1] I. F. Akyildiz, *et al.*, "A Survey on Sensor Networks", *IEEE Communications Magazine*, August 2002, pp. 102-114.

[2] J. M. Rabaey, *et al.*, "PicoRadio Supports Ad Hoc Ultra-Low Power Wireless Networking", IEEE 2000.

[3] D. Estrin, *et al.*, "Next Century Challenges: Scalable Coordination in Sensor Networks", *Proc. ACM Mobicom*, Seattle, WA, Aug. 1999, pp. 263-270.

[4] G. J. Pottie and W. J. Kaiser, "Wireless Integrated Network Sensors," *Commun. ACM*, vol. 43, no. 5, May 2000, pp. 551-58.

[5] 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.

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

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

[8] A. Muqattash, M. Krunz, "CDMA-Based MAC Protocol for Wireless Ad Hoc Networks", *Proc. of IEEE MobiHoc* 2003.

[9] S. De, *et al.*, "An Integrated Cross-Layer Study of Wireless CDMA Sensor Networks", *IEEE Journal on Selected Areas in Communications*, Vol. 22, No. 7, Sep. 2004, pp. 1271-1285.

[10] B. Hull, *et al.*, "Mitigating Congestion in Wireless Sensor Networks", *SenSys'04*, Nov. 2004.

[11] O. Dousse, *et al.* "Impact of Interference on Connectivity in Ad Hoc Networks", IEEE Inforcom 2003

[12] C. H. Liu, H. H. Asada, "A Source Coding and Modulation Method for Power Saving and Interference Reduction in DS-CDMA Sensor Networks", *Proc. American Control Conf.*, May 2002.

[13] E. S. Sousa, *et al.*, "Optimum Transmission Ranges in a Direct-Sequence Spread-Spectrum Multihop Packet Radio Network" *IEEE Journal on Selected Areas in Communications*, Vol. 8, No 5, 1990

[14] E. S. Sousa, J.A. Silvester, "Spreading code Protocols for Distributed Spread-Spectrum Packet Radio Networks", *IEEE Trans. on Comms.* Vol.36, No.3, March, 1988, pp.272-281.

[15] K. Sohrabi *et al.*, "Near-Ground Wideband Channel Measurement," IEEE Proc. VTC, New York, 1999

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

[17] C. Intanagonwiwat *et al.*, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", *Proc. of the 6th MobiCom*, Aug. 2000, Boston, Massachusetts.

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

[19] C. Chien, I. Elgorriaga, and C. McConaghy, "Low-Power Direct-Sequence Spread-Spectrum Modem Architecture For Distributed Wireless Sensor Networks," ISLPED 01, Huntington Beach, CA, Aug. 2001.

[20] S. Singh, C.S. Raghavendra, "PAMAS: Power Aware Multi-Access Protocol with Signalling for Ad Hoc Networks", ACM Computer Communication Review, Vol. 28, No. 3, pp. 5-26. July 1998.

[21] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," IEEE Proc. Hawaii Int'l. Conf. Sys. Sci., Jan. 2000, pp. 1-10.

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

[23] W. Ye, *et al.*, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks", *IEEE Proc. Infocom*, June 2002, pp.1567-1576.

[24] V. Rajendran, *et al.*, "Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks", *Proc. ACM SenSys* 2003, Nov. 2003.

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

[26] S. Tilak, N. B. Abu-Ghazaleh, W. Heinzelman, "A Taxonomy of Wireless Micro-Sensor Network Models", Mobile Computing and Communications Review, Vol.1, No. 2.

[27] C. Guo, *et al.*, "Low Power Distributed MAC for Ad Hoc Sensor Radio Networks", *IEEE Proc. GlobeCom* 2001, San Antonio, November 25-29, 2001

[28] J. A. Stankovic, *et al.*, "Real-Time Communication and Coordination in Embedded Sensor Networks", *Proc. of the IEEE*, VOL.91, NO7, July 2003, pp. 1002-1022.

[29] L. Kleinrock, J.A. Silvester, "Optimum Transmission Radii for Packet Radio Networks or Why Six Is A Magic Number", Proc. of IEEE Nat. Telecommunication Conf. (December, 1978) pp.4.3.1-4.3.5

[30] L. Hu, "Topology Control for Multihop Packet Radio Networks", *IEEE Trans. on Communications*, Vol.41, NO.10, pp. 1474-1481, October 1993.

[31] L. Hu, "Distributed Code Assignments for CDMA Packet Radio Networks", *IEEE/ACM Trans. on Networking*, Vol.1, No.6, Dec. 1993, pp. 668-677.

[32] R. Wattenhofer, *et al.*, "Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad Hoc Networks", *Proc. IEEE Infocom* 2001, pp. 1388-1397, April 2001.

[33] H. Zhang, J. C. Hou, "Maintaining Sensing Coverage and Connectivity in Large Sensor Networks", Technical Report UIUC, UIUCDCS-R-2003-2351, June 2003.

[34] A. A. Bertossi, M. A. Bonuccelli, "Code Assignment for Hidden Terminal Interference Avoidance in Multihop Packet Radio Networks", *IEEE/ACM Trans. on Networking*, Vol.3, No.4, Aug. 1995, pp. 441-449.

[35] R. Battiti, *et al.*, "Assigning Codes in Wireless Networks: Bouinds and Scaling Properties".

[36] R. Battiti, *et al.*, "Distributed Code Assignment in Multihop Radio Networks: Object-Oriented Software Simulations", Proc. of SoftCOM, 2000.

[37] S. Ramanathan, "A Unified Framework and Algorithm for Channel Assignment in Wireless Networks", *Wireless Networks* 5 (1999) pp. 81-94.

[38] J.J. Garcia-Luna-Aceves, J. Raju, "Distributed Assignment of Codes for Multihop Packet Radio Networks".

[39] L. Bao, J.J. Garcia-Luna-Aceves "Hybrid Channel Access Scheduling in Ad Hoc Networks",

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

[41] J.G. Proakis, M. Salehi, G. Bauch, "Contemporary Communication Systems using MATLAB and Simulink" 2nd ed. Southbank, Vic. : Thomson/Brooks/Cole, 2003,

[42] M. B. Pursley, "Performance Evaluation for Phase-coded Spread-Spectrum Multiple-Access Communications-Part I: System Analysis", *IEEE Trans. Commun.*, Vol. COM-25 pp. 795-799, Aug. 1977.

[43] S. Y. Ni, *et al.*, "The Broadcast Storm Problem in a Mobile Ad Hoc Network", *Proc. of 5th ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 151-162. ACM Press, 1999.

[44] N. Li, *et al.*, "Design and Analysis of an MST-Based Topology Control Algorithm", *Proc. of IEEE Inforcom* 2003.

[45] B. Williams, T. Camp, "Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks", *Proc. of Mobihoc*, 2002.

[46] J. Lipman, *et al.*, "Neighbor Aware Adaptive Power Flooding in Mobile Ad Hoc Networks", *International Journal of Foundations of Computer Science*, Vol. 14, No. 2, pp. 237-252.

[47] G. Toussaint, “The Relative Neighborhood Graph of Finite Planar Set”, *Pattern Recognition*, Vol. 12, No. 4, pp.261-268, 1980.

[48] G.N Karystinos and D.A. Pados, “Noew Bounds on the Total Squared Correlation and Optimum Design of DS-CDMA Binary Signature Sets”, IEEE Trans. Commun., Vol. 51, pp. 48-51, Jan. 2003.

[49] William C. Y. Lee, “Mobile Communications Design Fundamentals”, Second Edition, John Wiley and Sons, Inc, 1993

[50] R. Peterson, R. Ziemer, D. Borth “Introduction to Spread Spectrum Communication”, Prentice Hall, 1995

[51] A. J. Viterbi “CDMA Principles of Spread Spectrum Communication”, Addison Wesley, 1995

[52] T. S. Rappaport “Wireless Communications, Principles and Practice”, *Second Edition, Prentice Hall*, 2002

[53] J.A. Bondy and U.S.R. Murty, “Graph Theory with Applications”. New York: MacMillan, 1976.

[54] C. Berrou, A. Glavieux, P. Thitimajshima “Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes”, IEEE International Communication Conference (ICC), May, 1993, pp1064-1070.

[55] R.B. Cooper, “Introduction to Queueing Theory”, Third Edition, CEEPress, 1990

[56] P. Hall, “Introduction to the Theory of Coverage Process”, *John Wiley & Sons*, 1988

[57] M.N.S. Swamy “Graphs, Networks, and Algorithms”, John Wiley & Sons, Inc, 1981.

[58] J. P. F. Glas “Non-Cellular Wireless Communication Systems,”, *PhD Thesis*, Electrical Engineering, Delft University of Technology, 1996.

[59] ”MICA motes at Crossbow”, http://www.xbow.com/Products/Wireless_sensor_networks.htm, May 2002.

[60] J. M. Rabaey, “Ultra-low Power Computation and Communication enables Ambient Intelligence”, Presentation slides, Berkeley Wireless Research Center, 2003.

[61] B. H. Liu, N. Bulusu, H. Pham, S. Jha, “A Self-Organizing, Location-Aware Media Access Control Protocol for DS-CDMA Sensor Networks”, *First IEEE International Conference on Mobile Ad Hoc and Sensor System (MASS)*, short paper, Oct. 2004, pp. 528-530.

[62] B. H. Liu, N. Bulusu, H. Pham, S. Jha, “CSMAC: A Novel DS-CDMA Based MAC Protocol for Wireless Sensor Networks”, *IEEE GLOBECOM Wireless Ad hoc and Sensor Networks*, Nov. 29 - Dec. 3, 2004, Dallas Texas, USA.

[63] B. H. Liu, C. T. Chou, J. Lipman, S. Jha, “Using Frequency Division to Reduce MAI in DS-CDMA Wireless Sensor Networks”, *IEEE Wireless Communications and Networking Conference (WCNC)*, Mar. 13-17, New Orleans, LA, 2005, To appear.