

DGRAM: A Delay Guaranteed Routing and MAC Protocol for Wireless Sensor Networks

Chilukuri Shanti and Anirudha Sahoo
Department of Computer Science and Engineering
Indian Institute of Technology, Bombay, Powai, Mumbai, India 400076.
email: {shanti, sahoo}@cse.iitb.ac.in

Abstract

This paper presents an integrated MAC and routing protocol called Delay Guaranteed Routing and MAC (DGRAM) for delay sensitive wireless sensor network (WSN) applications. DGRAM is a TDMA-based protocol designed to provide deterministic delay guarantee in an energy efficient manner. The design is based on slot reuse to reduce the latency of a node in accessing the medium, while ensuring contention free medium access. The transmission and reception cycles of nodes are carefully computed so that data is transported from the source towards the sink while the nodes could sleep at the other times to conserve energy. Thus, routes of data packets are integrated into DGRAM. We provide a detailed design of time slot assignment and delay analysis of the protocol. One major advantage of DGRAM over other TDMA protocols is that the slot assignment is done in a fully distributed manner making the DGRAM network self-configuring. We have simulated DGRAM using ns2 simulator and compared the results with those of SMAC for a similar network. Simulation results show that the delay experienced by data packets is always less than the analytical delay bound for which the protocol is designed. As per simulation results, the average energy consumption does not change as the event rate changes, and is less than that of SMAC. This characteristic of DGRAM provides flexibility in choosing various operating parameters without having to worry about energy efficiency.

1. Introduction

Wireless Sensor Networks (WSNs) is an emerging technology with a wide range of potential applications such as environment monitoring, earthquake detection, patient monitoring systems etc. Sensor networks are also being deployed for many military applications, such as target tracking, surveillance and security management. WSNs typically consist of small, inexpensive, resource-constrained devices that communicate among each other using a multi-hop wireless network. Each node, called a sensor node, has one sensor, embedded

processors, limited memory and low-power radio and is normally battery operated. Each sensor node is responsible for sensing a desired event locally and for relaying a remote event sensed by other sensor nodes so that the event is reported to the end user. Sensors have limited energy resources and their functionality continues until their energy is drained. Therefore, applications and protocols for WSNs should be carefully designed in an energy-efficient manner so that the lifetime of sensors can be longer. The sensing element of a sensor probes the surrounding environment. If an *interesting* event is detected, after performing signal processing of the observed data, sensors communicate this data to the sink or base station using a radio based link. This communication happens in a single or multi-hop fashion depending on the location of the sensing node and the node has to access the medium and then transmit the data. Thus, in a distributed system like a WSN, medium access control (MAC) protocol plays an important role. As stated earlier, these MAC protocols should be energy efficient. In addition, if the sensor network is to be used for real time applications, the MAC protocol should provide QoS (e.g., delay guarantee).

In this paper, we propose a Time Division Multiple Access (TDMA) based energy efficient integrated MAC and routing protocol, called Delay Guaranteed Routing And MAC (DGRAM) protocol, which provides deterministic delay guarantee. Traditional TDMA MAC protocols suffer from high latency or centralized slot allocation. Most of them like [5] consider a centralized slot-allocation based on graph-coloring approach to reuse slots beyond two-hop neighbors. However, this approach is not scalable and requires slot-allocation messages to be passed by the base station, resulting in wastage of energy. On the other hand, DGRAM uses slot reutilization technique to reduce the latency between two successive channel accesses of a sensor node, with a completely distributed slot-allocation strategy which makes the deployment *self-configuring*. DGRAM only requires a sensor network to be deployed with uniform node density. Packets are then transmitted/received following an entirely logical topology. While traditional TDMA MACs require a routing protocol to run on top of them, DGRAM has the routing mechanism built

into the MAC, using coordinated sleep and wakeup cycles. Further, DGRAM allows sensors to go to sleep when they are not communicating (neither transmitting nor receiving) and hence it conserves energy. Since DGRAM can provide delay guarantee, it is suitable for realtime applications like detection of radioactive radiation, earthquake etc. We present the method by which time slots are assigned to sensor nodes and show how the slots are reused by nodes which are non-interfering. Then we present the delay analysis of DGRAM to show that the delay is bounded in DGRAM. Routing of data from source to sink or base station is integrated into DGRAM, as nodes broadcast during their transmission slots and receive data during their reception slots. This enables the flow of data from source towards the sink. Thus, a separate routing protocol is not required for DGRAM. This indirectly saves the energy which otherwise would have been expended in determining the route of packets from source to sink. We present our simulation results which show that our analytical delay bound is always guaranteed by the protocol and that there is no packet loss as long as the event rate is below the designed event rate. We compare performance of DGRAM with SMAC and show that DGRAM outperforms SMAC in terms of delay, energy consumption and number of packets missing deadline. One important simulation result we present is that DGRAM consumes almost constant energy independent of event rate as long as the event rate is low. Thus, DGRAM provides flexibility in designing the system without having to worry about energy consumption.

2. Related Work

Wireless sensor networks (WSNs) generally use MAC protocols which are either TDMA-based or are contention-based. In contention-based protocols, multiple nodes may access the medium simultaneously, resulting in collision. The MAC protocol then provides a mechanism to avoid collision. [3] gives an excellent overview of the various MAC protocols for sensor networks. IEEE 802.11 distributed coordination function (DCF) is one such protocol. Woo et al. [18] have studied different configurations of CSMA and proposed an adaptive rate control mechanism to achieve fair bandwidth allocation to all nodes. S-MAC [17] is a popular MAC protocol for WSNs that conserves energy by having listen and sleep cycles, so that idle-listening time is minimized. S-MAC conserves energy, but has high latency. Several extensions and modifications to S-MAC like WiseMAC [4] and DMAC [10] have been proposed to improve the protocol performance. [1] proposes an algorithm to determine the wake-up frequency of nodes depending on their proximity to the sink, so that the overall energy spent and delay are minimized. However, it does not give a schedule for the wake-up cycles.

TDMA based protocols are contention-free protocols in which sensor nodes transmit only in their assigned time slot. Sohrabi et al. proposed a self-organizing MAC for sensor

networks in which each node maintains a TDMA-like frame called the superframe [2]. Interference between adjacent links is avoided by using FDMA and CDMA in potentially interfering links. [16] proposes a TDMA protocol in which clusters are formed and each cluster elects a cluster head. All the nodes in the network are assumed to have enough radio power to communicate with the base station directly. Cluster heads communicate directly with the base station and other nodes communicate directly only with the cluster head. The slot assignment is randomized. [19] presents a self-reorganizing slot allocation mechanism for TDMA based MAC in multi-cluster sensor networks. The primary contribution of the paper is to demonstrate that, with adaptive slot allocation, it is possible to reduce inter-cluster interference under low load conditions. The second contribution is the design of a feedback-based adaptive allocation reorganization protocol that significantly reduces the inter-cluster interferences without relying on any global synchronization mechanism. In [15], the authors propose a new MAC protocol referred to as DEE-MAC which reduces energy consumption by making the idle nodes sleep to reduce idle listening using synchronization at cluster heads. Each cluster is dynamically formed and all nodes contend for the position of cluster-head. Nodes can join or leave the cluster any time they want to. RT-Link [5] focuses on reducing the communication delay between a node and the base station. It uses a centralized slot computing mechanism, using the distance-k node coloring approach (slot scheduling), to minimize the number of collisions along each transmission hop in a multi-hop wireless network. It also proposes a delay-sensitive schedule. Logical connectivity graph formation and slot assignment are done by the base station centrally. TDMA protocols need highly accurate time synchronization for correct operation. Out-of-band, hardware-based synchronization has been suggested for this purpose in [5]. Other mechanisms of time synchronization for TDMA protocols are presented in [7, 11, 12]. [9] uses a logical tier structure similar to that of DGRAM, but is contention-based and may not provide delay-guarantee when there is a burst of data.

3. Deployment of the WSN

3.1. Topology

We assume that the sensing area has a base station or sink, which is wired and is not power-constrained. The sensor nodes (henceforth referred to as just *nodes*) sense the event of interest and transmit this data to the sink using a MAC described in this paper. The nodes are deployed with uniform density over the sensing area all around the sink with the sink at the center of the area. Thus, depending on the distance of a node from the sink and the transmission range of the nodes, data has to traverse single or multiple hops before being received by the sink.

- Our protocol is designed with the following assumptions,
- The nodes and the sink are stationary
 - Each node knows its location relative to the sink.

- The sink is roughly in the middle of the sensing area for better efficiency of the protocol.
- The clocks of sensor nodes are synchronized by using out-of-band time synchronization. A similar mechanism as used in [5] may be used for this purpose.
- The transmission and interference ranges of the nodes are generally not exactly circular. Hence, we consider the maximum interference radius and the minimum transmission radius when the transmission and reception cycles are calculated. This ensures that no packets are lost.

3.2. Location of the Sensors

DGRAM requires the sensors to be distributed with uniform density around the sink. This is the only assumption made by DGRAM regarding the deployment of the network. One of the main assumptions made above is that the nodes know their positions with respect to the sink. There are many distributed algorithms in literature to find the coordinates of the nodes in a sensor network. Many localization systems depend on having direct distance estimates to globally accessible beacons like the Global Positioning System (GPS). Recently, there has been some research in localizing a wireless sensor network where there are no globally accessible beacons. Savvides et al. [14] describe a distributed algorithm that recursively infers the positions of sensors with unknown positions from a set of sensors with known positions, using inter-sensor distance estimates. While algorithms like this need seed nodes that know their position, algorithms that do not need such seed nodes have also been proposed. [13] proposes a localization algorithm that does not need any anchor nodes. [8] describes a localization algorithm that requires a single anchor in the network. For our protocol, this single anchor can be the sink. Hence, the nodes can be localized without all of them having GPS sensors. For our protocol, we assume that each node already knows its position relative to the sink using one of the above algorithms. The position of a node is represented by its radial distance from the sink and the angle (in clockwise direction) relative to geographical North axis passing through the sink.

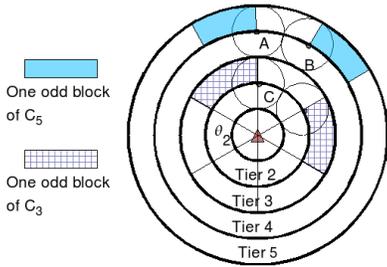


Figure 1. Depiction of Tiers and Blocks (for $\alpha = 1.0$)

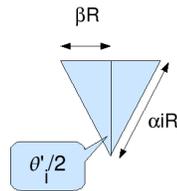


Figure 2. Calculation of θ'

Table 1. Explanation of Various Notations

| Notation | Definition |
|---------------|---|
| λ | Node density expressed in nodes per unit area |
| d_{worst} | worst case delay of an event from the time it occurred to the time it is delivered to the sink |
| n_i | number of slots in tier C_i |
| m_i | number of slots per block in tier C_i |
| R | Transmission range of a node |
| H | Total number of tiers in the sensing area |
| N | Number of subframes in the superframe |
| α | Radial width of a tier expressed as a fraction of the transmission range of a node ($\alpha \leq 1$) |
| I | Interference range of a node |
| β | Ratio of interference range to transmission range |
| C_i | ID of the i^{th} tier |
| Z_i | Number of blocks in the tier C_i |
| A_i | Area of the tier C_i |
| θ_i | Angle subtended(rads) by a block in tier C_i at the sink |
| B_j^i | ID of the j^{th} block in the tier C_i |
| S_i | Number of slots assigned to the i^{th} subframe of the superframe |
| S_i' | Size of the subframe for the i^{th} tier (as per calculation) |
| D_{ij}^k | Radial distance of node k belonging to tier C_i and block B_j^i from the sink |
| ϕ_{ij}^k | Angular distance of node k belonging to tier C_i block B_j^i from the sink with reference to geographical North passing through the sink (in clockwise direction) |
| T | Number of slots in the superframe |

4. Time Slot Assignment

4.1. Logical Organization of Nodes

To facilitate multihop communication and time slot reuse, the sensing area is divided into tiers and blocks. The sensing area is organized into tiers, based on the radial distance of different nodes from the sink. Then each tier is divided into blocks based on the angular distance of nodes (measured in clockwise direction) from the sink with respect to the geographical North axis passing through the sink. This is depicted in Figure 1. Each tier is of radial width αR . The parameter α determines the minimum node density in the area for the network to remain connected. The tier closest to the sink is the first tier and is identified as C_1 . There are H number of tiers in the sensing area. Thus, the tier farthest from the sink is identified as C_H . The division of the network into tiers helps in multi-hop transmission of data towards the sink and the division into blocks aids in reusing time slots. Based on the position of the node and the sink, the radial distance of the node from the sink is calculated by each node. Based on this distance and α , each node calculates the tier C_i to which it belongs. If the radial distance is between $(i - 1)\alpha R$ and $i\alpha R$ ($i > 0$), the node determines itself to be in the i^{th} tier C_i . Data from any node in this tier flows radially inward, hopping from outer tier to inner tier and reaches the sink in i hops.

Once the tier to which the node belongs is calculated, the node needs to find the block to which it belongs within its tier. A block is an angular area covered by the interference range of a node in a given tier (see Figure 1). The division of a tier into blocks helps in slot reuse within the tier. A node in the extreme right of block B_j^i can interfere with a transmission by a node in block B_{j+1}^i , but not with a node even to the extreme

left of the block B_{j+2}^i . Thus, nodes in alternate blocks can transmit in the same slot without interfering with each other. For example, nodes A and B shown in Figure 1 can transmit simultaneously. This calculation is done based on the angle θ_i that each block in tier C_i subtends at the center or sink. To keep the number of nodes per block approximately the same, the angle θ_i is made smaller for tiers farther away from the sink. As shown in Figure 2, this angle for any tier C_i is calculated by the node as follows.

$$\sin(\theta_i/2) = \frac{I}{\alpha i R}$$

Solving the above equation for θ , we get

$$\theta_i' = \sin^{-1} \frac{2\beta\sqrt{(\alpha^2 i^2 - \beta^2)}}{\alpha^2 i^2} \quad (1)$$

Since we allow alternate blocks to use the same half of a subframe (explained in Section 4.2), the number of blocks should be even. However, the number of blocks in the tier C_i depends on the angle θ_i of that tier and may not always be an even number. To ensure that the number of blocks Z_i in the tier C_i is an even number, blocks are assigned in the half circle region i.e. the angle for each block is assigned from π . Thus,

$$Z_i = 2 \lfloor \frac{\pi}{\theta_i'} \rfloor \quad (2)$$

Note that we have used the floor function in the above equation so that the area of adjacent blocks will be greater than or equal to that required to make them non-interfering. Keeping the above constraints in view and using (1) and (2), the angle subtended by a block at the sink is modified as follows

$$\theta_i = \theta_i' + \frac{\frac{\pi}{\theta_i'} - \lfloor \frac{\pi}{\theta_i'} \rfloor}{Z_i/2} \quad (3)$$

Now, knowing the value of θ_i and the angular distance of the node from the sink, the node can find out the block to which it belongs. If the angular distance is between $p\theta_i$ and $((p+1)\theta_i)$ ($p \geq 0$), then the node belongs to j^{th} block, B_j^i , where $j = p+1$. Thus, B_1^i is the first block in tier C_i which is between 0 and θ_i (in clockwise direction) from geographical North passing through the sink.

4.2. Slot Assignment at the Tier and Block Level

A node in DGRAM can be in one of the two states - *active* or *idle*. A node is in active mode in its allotted transmission slot if it has pending data to send. It is otherwise idle for transmission. A node switches itself into the active state at the beginning of each of its receive slots. If it senses a preamble intended for it, it remains in active mode to listen for the rest of the slot duration to receive data from previous tier. Otherwise, it goes back to idle mode. This careful design of active and idle states helps minimizing energy consumption. Slot assignment in DGRAM happens in three levels of hierarchy. At the highest level, the time frame is called the *superframe*. The superframe is divided into subframes, which are assigned to different tiers. The subframes are reused across the tiers such that the transmissions do not interfere. Nodes in two different tiers can transmit at the same time when they are at least $2I$ apart. Each tier is αR wide. If N is the number of subframes, N tier widths should encompass the $2I$ length. This means that $N = 2I/\alpha R$. But we need to take care of border cases of tiers, i.e., a node on the border between N^{th} and $(N+1)^{th}$

tier and belonging to $(N+1)^{th}$ tier could interfere with the first tier. Thus, taking an extra tier width into the calculation and the fact that $I = \beta R$, we have

$$N = \lceil \frac{2\beta}{\alpha} + 1 \rceil \quad (4)$$

Thus, the tiers can be categorized into groups, where nodes belonging to each group can transmit during the same subframe. The tiers belonging to the same group as C_i are:

$$C_{\lceil i+j(1+2\beta/\alpha) \rceil}, j = 0, 1, 2, \dots \quad (5)$$

Each tier thus has a subframe allotted to it. Further, each subframe is split into two halves called *subsubframe*. One half is used by all odd numbered blocks whereas the other is used by all even numbered blocks. This is possible because the even (odd) numbered blocks are not within interference range of each other. Thus, slots are reused at the block level also. Note that reuse of slots helps in reducing the size of the superframe which leads to reduced latency of the network. The innermost two tiers, i.e., tiers C_1 and C_2 are not divided into blocks.

4.3. Slot Assignment at the Node level

Once the subframe allocation is done for tiers and blocks, the node level slot assignment is done. Nodes only transmit data during the slots assigned to them in the subsubframe of the block to which they belong. To achieve this, each node broadcasts a short control message which carries its location, tier ID, block ID and TTL. This exchange of control messages happens at the time of deployment of nodes and is a one time configuration for the network. The location of a node k in tier C_i and block B_j^i is essentially an ordered pair (D_{ij}^k, ϕ_{ij}^k) , where D_{ij}^k is the radial distance of the node from the sink and ϕ_{ij}^k is the angular distance of the node with respect to the geographical north (in clockwise direction) passing through the sink. The TTL is a time to live field which is decremented by one by the receiving node and the message is retransmitted, as long as the TTL field is greater than one. The initial TTL value is chosen such that the control message reaches every node in the block. Since the diametric width of a block is $2I$, TTL value should be $\lceil \frac{2I}{R} \rceil = \lceil 2\beta \rceil$. A simple distributed algorithm (Algorithm 1) is run by every node to assign time slots. A node waits for a sufficiently long time (Line 4) to collect location information from all other nodes in its block. It maintains a sorted array which indexes nodes according to their location. When all the messages are received, the index of a node's location in the sorted array is the node's index in the block, which is then used to assign a slot to the node for transmission. The node's reception cycle or reception slots are then decided as per the discussion in Section 4.7.2. The control messages can be sent using a reliable CSMA/CA MAC protocol like IEEE 802.11. Note that a node might receive control messages from nodes belonging to other tier or other blocks. A node has to discard such messages which do not belong to the same tier and block as its own. A node waits for a sufficiently long amount of time to ensure that it has received control messages from all the nodes belonging to its

tier and block. Figure 3 shows the time slot assignment at various levels when there are six tiers.

Algorithm 1 Node_level_slot_assignment($C_i, B_i^j, H, S_i, rad_distance, angle_distance$)

```

1: /*rad.distance is radial distance of the node from the sink*/
2: /*angle.distance is angular distance of the node from the sink*/
3: sorted_location_array[0] = {rad.distance, angle.distance}
   /* nodes are sorted in an increasing order of location */ /*
   control_msg_timer is sufficiently large so that control messages from
   all other nodes in the block  $B_i^j$  are received before it expires */
4: while control_msg_timer not expired do
5:   Get the next control message
6:   if (control_message.tier !=  $C_i$ ) OR
   (control_message.block !=  $B_i^j$ ) then
7:     drop the control_message
8:     continue
9:   end if
10:  if control_message.TTL > 1 then
11:    control_message.TTL--
12:    broadcast control_message
13:    let rad.distance', angle.distance' be the location information
   in the control_message
14:    insert the above location information into
   sorted_location_array[] such that the array remains sorted
   /* The array is sorted in increasing order of  $\phi$  (angular distance). In
   case of tie, it is then sorted in increasing order of  $D$  (radial distance)
   */
15:  end if
16: end while
17:  $k$  = index of this node's location information in
   sorted_location_array[]
   /* this is the  $k^{th}$  node in the block  $B_i^j$  */
18: Calculate  $P_i, Q_i^j, U_{ij}^k$  using (15), (16), (17) respectively
19: Mark slot  $(P_i + Q_i^j + U_{ij}^k + 1)$  for transmission

```

4.4. Calculation of Size of Superframe

In DGRAM we make sure that there is no queuing in the nodes. This makes it easier to provide deterministic delay guarantee. Since the number of nodes in the outer tier is more than that in the inner tier and the nodes in any tier are given one timeslot each, nodes in the inner tiers aggregate data so that it can be relayed together with the node's own (locally sensed) data in a single slot. This is a reasonable assumption made by many TDMA protocols such as [5]. The outermost tier C_H will allot one slot per node. Hence, we have

$$A_H = \pi\alpha^2 R^2 (2H - 1) \quad (6)$$

$$S'_H = \frac{\lambda A_H}{Z_H/2} \quad (7)$$

In (7), the numerator is the total number of slots assigned in tier C_H and the denominator is the number of block pairs in C_H . Recall that a subframe is divided into two halves, one of which is used by odd numbered blocks and the other half is used by even numbered blocks.

The values for the next tier C_{H-1} can be calculated as follows.

$$A_{H-1} = \pi\alpha^2 R^2 (2H - 3), \quad S'_{H-1} = \frac{2\lambda A_{H-1}}{Z_{H-1}} \quad (8)$$

Generalizing the above, we have

$$A_i = \pi\alpha^2 R^2 (2H - 1 - 2(H - i)), \quad S'_i = \frac{2\lambda A_i}{Z_i} \quad (9)$$

Since tier C_1 and C_2 are not divided into blocks, the corresponding parameters for this tier are given by

$$\begin{aligned} A_1 &= \pi\alpha^2 R^2, & S'_1 &= \lambda A_1 \\ A_2 &= 3\pi\alpha^2 R^2, & S'_2 &= \lambda A_2 \end{aligned} \quad (10)$$

It is obvious from the above discussion that the number of slots allotted for concurrent tiers from different groups are not equal. Hence, we consider the largest number of slots required by a tier in a concurrent group of tiers to be the subframe size for that set of tiers. For example, for a network with six tiers and two groups, if C_1, C_2 and C_3 belong to the first group, C_4, C_5 and C_6 form the next group, C_1 and C_4 transmit concurrently and form part of a set of tiers that transmit concurrently. Similarly, C_2 and C_5 are transmit concurrently and C_3 and C_6 transmit concurrently. The size of the subframe in which C_1 and C_4 transmit has the greater value among the number of slots needed by C_1 and that by C_4 . Likewise, if C_2 has a larger number of slots compared to C_5 with which it transmits concurrently, the size of the second subframe in the super frame is taken to be that of C_2 . Let the size of the subframe actually needed by the p^{th} tier be denoted by S'_p . Let the i^{th} subframe of the superframe be denoted by S_i . Thus,

$$S_i = \text{Max}(S'_i, S'_{i+N}, S'_{i+2N}, S'_{i+3N} \dots) \quad (11)$$

Hence, the size of the superframe T is decided by the tiers that have the maximum number of slots among a set of tiers that transmit concurrently. N is the number of tiers in each group and is given by (4). Thus, the number of slots in the superframe is given by

$$T = \sum_{i=1}^N S_i \quad (12)$$

As an example, the division of slots in T for an area with 6

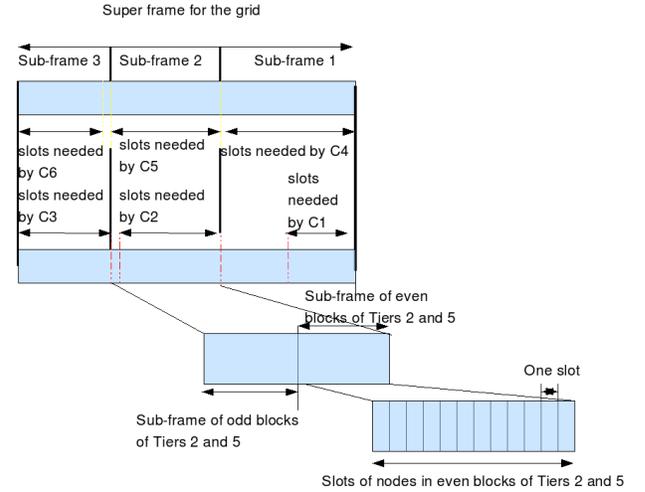


Figure 3. Time Slot Assignment at Different Hierarchy

4.5. Dimensioning of DGRAM Parameters

For a network with given β and H , the value of T depends on the node density λ . The minimum node density that can keep the network connected, in turn, depends on the value of α , as described by [9]. [9] studies the effect of α for a given

λ on the connectivity, which is the number of nodes that can potentially relay data from a given node. In Figure 4, consider node A transmitting data. As per DGRAM, the only nodes that can relay data from this node are those in the hatched region. The hatched area is the region of overlap of two circles: the first is a circle centered at the node with radius R , and the second is a tier circle centered at the sink with radius $\alpha(n-1)R$. α has to be less than unity for a node in the n^{th} tier to be able to relay data to the $(n-1)^{\text{th}}$ tier. The number of nodes in this region of overlap is a measure of connectivity of the network, since only nodes in this region will ultimately relay data.

From Figure 4, the depicted region of overlap is the minimum area possible for all nodes in tier n , since the sender node is at the edge of the tier. This area will be minimized with respect to n when n is as small as possible, i.e., at $n_{\min} \lfloor \frac{1}{\alpha} + 1 \rfloor$.

This is because all nodes in the $\lfloor \frac{1}{\alpha} \rfloor^{\text{th}}$ tier or lower are within a distance R from the sink and hence can communicate directly with the sink. By the cosine rule of triangles we have

$$\begin{aligned} \cos\psi &= ((2n_{\min} - 1)\alpha^2 + 1)/2n_{\min}\alpha \\ \cos\Omega &= ((n_{\min}^2 + (n_{\min} - 1)^2\alpha^2 - 1)/2n_{\min}(n_{\min} - 1)\alpha^2 \end{aligned}$$

The area of the shaded region is then given by

$$A_{\text{conn}} = R^2(\psi + (n_{\min} - 1)^2\alpha^2\Omega - n_{\min}\alpha\sin\psi) \quad (13)$$

Thus, the average number of nodes in this area is given by

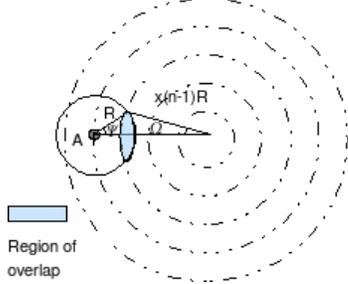


Figure 4. Minimum Density for Connectivity of the Network

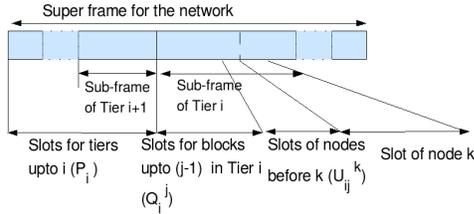


Figure 5. Transmit Cycle of the k^{th} Node in B_j^i

$A_{\text{conn}}\lambda$. For a connectivity of one, the minimum value of the node density for a given α is $1/A_{\text{conn}}$. We have chosen a value of 0.51 for α , as this ensures that a node in a tier can hear even from the farthest node in the outer tier, resulting in zero packet loss.

4.6. Calculation of Worst Case Delay

The worst case delay occurs when an event occurs in the outermost tier C_H and when a node in that tier which senses

the event just misses its assigned slot. So it has to wait for time T before it can transmit the packet. Once the data is transmitted out of the source node, it is transported in $\lceil i/N \rceil$ cycles by the nodes in the inner tiers. $\lceil i/N \rceil$ is the group number to which the source node belongs. Once the data is on its way, it is transmitted in subsequent subframes in a continuous manner by the node in the respective tier until it reaches the sink. This translates to total number of subframes assigned to all the tiers in the network for an event occurring in the tier C_H . But the number of tiers H may not be an exact multiple of the number of subframes N in a superframe. Hence the worst case delay is given by

$$d_{\text{worst}} = T + \lceil H/N \rceil T \quad (14)$$

For a network with a parameter of α , and whose maximum delay is to be $T + \lceil H/N \rceil T$, the parameter λ has to be chosen appropriately.

4.7. Sleep and Wakeup Cycle of Nodes

4.7.1 Transmission Cycle of a Node

Since DGRAM follows a TDMA schedule, each node transmits only during its assigned slot in a superframe. Each node in a tier (i.e., C_i) has one slot per superframe to transmit aggregation of the data it has sensed and any data it has to relay from the outer tiers. Consider a node belonging to the tier C_i and block B_j^i . If the node is the k^{th} node in its block and tier (as per the ordered pair of its location (D_{ij}^k, ϕ_{ij}^k)), the slot it should transmit in is calculated as follows. Let the number of slots assigned to the outer tiers before the tier C_i in the superframe be P_i . This is given by

$$P_i = \begin{cases} \sum_{l=(i+1) \bmod N}^N S_l & \text{if } i \bmod N \neq 0 \\ 0 & \text{if } i \bmod N = 0 \end{cases} \quad (15)$$

Let the number of slots assigned to the blocks before block B_j^i in the subframe assigned to C_i be Q_i^j . Q_i^j is given by

$$Q_i^j = ((B_j^i - 1) S_i / 2) \bmod S_i \quad 1 \leq i \leq H, 1 \leq j \leq Z_i \quad (16)$$

Let the number of slots before this node in the subsubframe of its block be U_{ij}^k . Since this is the k^{th} node in its block B_j^i ,

$$U_{ij}^k = (k - 1) \quad (17)$$

Hence, this node transmits in the slot $(P_i + Q_i^j + U_{ij}^k + 1) \bmod T$. This is illustrated in Figure 5.

4.7.2 Receive Cycle of a Node

A node in block B_j^i receives data from its previous tier $C_{(i+1)}$ ($1 \leq i < H$). The number of slots in the tier C_i is given by

$$n_i = \begin{cases} S_i Z_i / 2 & \text{if } i > 2 \\ S_i & \text{if } i \leq 2 \end{cases} \quad (18)$$

The number of slots per block in the tier C_i is given by

$$m_i = \begin{cases} S_i / 2 & \text{if } i > 2 \\ S_i & \text{if } i \leq 2 \end{cases} \quad (19)$$

Let us consider the k^{th} ($k \geq 1$) node of block B_j^i . The index of this node in the tier C_i would be at least $k' = (B_j^i - 1)m_i + k$. This node should receive from

nodes $\lfloor \frac{n_{i+1}}{n_i}(k' - 1) + 1 \rfloor$ (denote this as $startnode_i^k$) to $\lfloor \frac{n_{i+1}}{n_i}k' \rfloor$ (denote this as $endnode_i^k$) of the tier C_{i+1} . The $startnode_i^k$ belongs to the block $\lceil startnode_i^k/m_{i+1} \rceil$. The $endnode_i^k$ belongs to the block $\lceil endnode_i^k/m_{i+1} \rceil$. The index of the $startnode_i^k$ in its block is $(startnode_i^k \bmod m_{i+1})$ and the index of the $endnode_i^k$ is $(endnode_i^k \bmod m_{i+1})$. The transmission slots of the nodes from the $startnode_i^k$ to the $endnode_i^k$ are calculated as per the formulae discussed in the Section 4.7.1, so that this node can go into listen mode only for those slots, and then go into active mode only if there is a transmission for it. Nodes in the tier C_H need not receive data from any other node and hence do not have a receive cycle.

Thus, for a node in the tier C_i and block B_i^j , to know when it has to wake up for transmission or reception, it needs to know the required parameters of the network. Then it can calculate the values of P_i , Q_i^j , U_{ij}^k to get its transmission slots. Then knowing the transmission slots of nodes in tier C_{i+1} from which it should receive data, it can calculate the slots in which it should receive data. For all other slots in the superframe the node goes to sleep. This fully distributed slot calculation is a major advantage of the DGRAM, which distinguishes it from other TDMA protocols for WSNs, which typically have a centralized slot assignment. Also note that the coordination of sleep and wake up cycle helps in transporting data from source to the sink across the tiers, thereby eliminating the need for a separate routing protocol. Thus, DGRAM does not require any routing protocol to transport data from source to the sink. This also means, that energy which otherwise would have been spent in finding routes is conserved in a DGRAM network.

5. Energy Consumption in DGRAM

In this section, we calculate the maximum power consumption of a node in the network running DGRAM. The notation used in this calculation is given in Table 2. In a WSN, power consumption can be because of event sensing, transmitting event reports to the sink and exchanging control information. We examine the power consumed due to event reporting only, since the first component is common to all protocols and the last component is negligible, as control messages are only exchanged one time in DGRAM at the time of deployment of nodes. The duration of each TDMA slot is ρ . The idle time between slots is T_{idle} . If T is the total superframe time,

$$T_{idle} = T - \rho \quad (20)$$

The life time of a node is dependent on the TDMA frame time and the number of TDMA slots for which the node remains active for transmission or reception. The minimum energy spent by a node during a single TDMA frame is the energy spent by it when it has no packets to receive or send. If the node is supposed to receive from d nodes, it has to check the channel in the beginning of d slots for a possible transmission. Hence:

$$E_{min} = d * E_{grx} + E_{idle} \quad (21)$$

The maximum energy spent by a node in a TDMA frame is when it receives in all its d slots and transmits in its slot.

$$E_{max} = E_{min} + d * E_{rx} + E_{tx} \quad (22)$$

Table 2. Notations used in Calculation of Power consumption

| Notation | Definition |
|---------------------------------------|--|
| ρ | duration of a single time slot |
| T_{grx} | duration of channel check at the beginning of a reception slot |
| T_{idle} | Time for which the node is idle in a superframe |
| $T_{minlife}$ | minimum life time of a node |
| P_{idle} | Power consumption when the radio is idle |
| P_{rx} | Power consumption with the radio receiving |
| P_{tx} | Power consumption with the radio transmitting |
| E_{rx} $= P_{rx} * \rho$ | Energy spent by a node to receive in a single slot |
| E_{tx} $= P_{tx} * \rho$ | Energy spent by a node to transmit in a single slot |
| E_{max} | Maximum energy spent by a node per superframe |
| E_{min} | Minimum energy spent by a node per superframe |
| E_{grx} $= P_{rx} * T_{grx}$ | Energy spent checking the channel at the beginning of a reception slot |
| E_{idle} $= P_{idle} * T_{idle}$ | Energy spent by a node while idle in a superframe |

The maximum power consumed by a node over a TDMA frame can be computed as follows:

$$P_{max} = E_{max}/T_{frame} \quad (23)$$

The minimum lifetime of a node can be computed as follows:

$$T_{minlife} = (E_{capacity}/E_{max}) * T_{frame} \quad (24)$$

A typical Low Power Listen CSMA (LPL-CSMA) approach balances long preamble transmit times with the frequency of channel activity checks. For a given event rate, checking the channel more or less frequently can be inefficient. More importantly, the end-to-end latency is directly proportional to the channel check interval, which must be fixed for optimal lifetime. DGRAM alleviates this problem as the channel checks have to be performed for receiving only during the beginning of the reception slots. This minimizes the energy expended in channel sensing and at the same time reduces the latency.

6. Simulation results

Table 3. Physical parameters used in simulation

| Parameter | Value | Parameter | Value |
|------------|-------------|-----------|----------|
| λ | 1/400 m^2 | ρ | 2.048 ms |
| R | 100m | β | 1.0 |
| P_{rx} | 150mW | P_{tx} | 150mW |
| P_{idle} | 30mW | Data rate | 50kbps |

We simulated DGRAM using ns2 for different radii of the sensing area and for different event rates. For all our simulation experiments we used the parameter values given in Table 3. The values of the physical parameters of the nodes are taken from [6], which contains representative values for μ amps sensor nodes. The μ amps nodes are designed for transmitting data upto 1Mbps within a range of 100 meters. We have assumed a data rate of 500kbps for our experiments. Duration of each time slot (ρ) is taken to be the duration for which packet of 128bytes can be transmitted over a link of 500kbps

bandwidth, i.e., $\rho = \frac{128 \times 8}{500} ms = 2.048 ms$. The simulation is done for a circular region with the sink at the center. The sensor nodes are randomly and uniformly distributed with a density $\lambda = 1$ node per 400 square meters. Each node knows its position and that of the base station, the node density λ , the transmission radius R and the interference radius I .

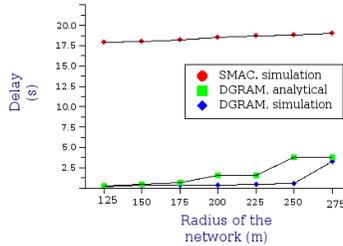


Figure 6. Effect of the Network Radius on the Delay

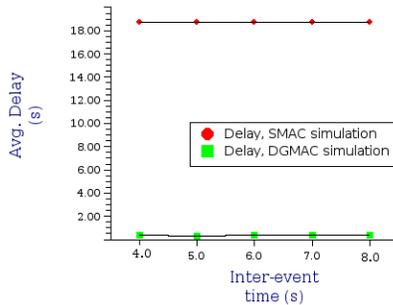


Figure 7. Effect of Event Rate on the Average Delay

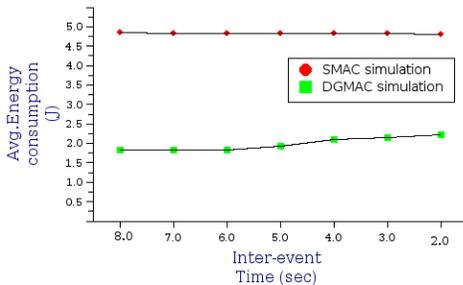


Figure 8. Average Energy Consumption versus Event Rate

We are interested in finding out the performance of the protocol in the worst case scenario. Hence, events were generated such that the inter-event time is uniformly distributed with the desired average rate. Further, when an event generation time is up, then all the nodes in the network are handed down an event with a timestamp which is uniformly distributed with the average being the time of generation of the event. This scenario captures the worst case scenario when a phenomenon occurs which affects the entire sensing area (e.g., a biohazard which encompasses the entire sensing area) and hence all the nodes in the network would sense an event almost at the same time.

In Figure 6, as expected, the average delay increases as the radius of the network increases. This graph is very useful

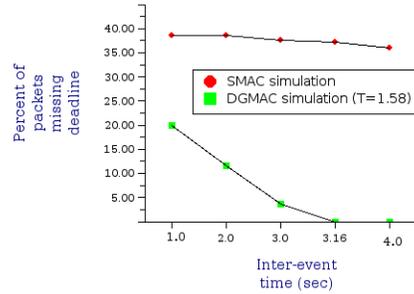


Figure 9. Number of Packets Missing Deadline as Event Rate Changes

in designing the radius of the sensing area so that the maximum delay is less than a certain desired deadline. In case the sensing area is more than that determined by the deadline constraint, the area should be split into multiple circular clusters of smaller radius to satisfy the delay constraint. Here again, the analytical maximum delay value is always greater than the average delay obtained through simulation. A similar simulation with SMAC shows that the average delay is much higher compared to that of DGRAM for any network radius. Since S-MAC is contention based, for high event rates, it encounters a lot of collisions resulting in packet retransmission. This leads to higher average delay of the packets.

Figure 7 shows the variation of average delay as the inter-event time is varied. We have plotted the graph only for inter-event time greater than the maximum possible delay (super-frame size for these experiments is $1.393 sec$, hence, the maximum delay as per 4.6 is $2.786 sec$). When inter-event time is less than the maximum possible delay, the event rate becomes too high, which leads to buffering of events and eventually loss of events due to buffer overflow. The average delay normally is more or less constant as the event rate decreases, which is to be expected since the nodes always transmit in fixed slots irrespective of the event rate. It is observed from this figure that SMAC always has a much higher delay than DGRAM.

Figure 8 presents the average energy consumed by a node as the inter-event time (with events occurring at all nodes) decreases. The energy consumed remains nearly constant even though the inter-event time changes as long as the inter-event time is less than the maximum possible delay ($2.786 sec$). This is due to the fact that the event rates taken in the experiment are low and when event rate is low, energy required to power on or off the radio dominates the energy consumption compared to the energy spent in transmission or reception. The energy needed to power up or down the radio of a node in a time interval is fixed because a node switches on and off for a fixed number of times in a TDMA cycle. This is a good characteristic of DGRAM, since as long as the event rate of each node is below the threshold, the network can handle different rates without affecting energy consumption. The average energy spent using SMAC is much higher, because of possible collisions when all nodes generate packets. This can be seen from Figure 8.

Finally, Figure 9 shows how DGRAM guarantees delivery of events (packets) to the sink. As inter-event time (at each node) increases, the percentage of packets with delay exceeding the maximum allowable delay, i.e., packets missing deadline, decreases. But when inter-event time is less than maximum possible delay, the network has packets missing the deadline. As long as inter-event time is less than the maximum possible delay, DGRAM is designed in such a way that the packets flow towards the sink without exceeding the maximum possible delay as in 4.6. If two or more events are generated in a duration less than equal to this value (2.786sec for the simulation scenerio), then there will be queueing at the source node which will have a cumulative queueing effect at nodes in subsequent tiers, eventually leading to packets missing deadline. As the inter-event time approaches maximum possible delay, packet loss fast approaches zero and when it is more than this maximum possible delay, the network does not have any packet delay exceeding the deadline. Thus, DGRAM guarantees delivery of packets within the maximum delay as long as inter-event time is more than the maximum possible delay. SMAC on the other hand, has a large percentage of packets missing deadline in a similar scenario.

7. Conclusion

We have presented a new contention-free TDMA based MAC and routing protocol, DGRAM, which can provide deterministic delay guarantee. We presented detailed design of time slot assignment, transmission and reception cycle of nodes. We also provided the worst case delay analysis of DGRAM. Our simulation results validated that the actual delay is always less than the analytical delay bound for which DGRAM is designed. Thus, DGRAM can be used for hard realtime applications such as bio-hazard detection, radioactive emission control etc. DGRAM is designed to handle inter-event time less than the worst case delay. That is, DGRAM can guarantee delay bound and zero packet loss as long as inter-event time is greater than the worst case delay. We also learnt that as long as the inter-event time at each node is less than the maximum allowable delay, the energy consumption of the network remains almost constant. This characteristic of DGRAM can be exploited while choosing various operating parameters of the protocol.

References

- [1] Cohen, R. Kapchits, B. . An Optimal Algorithm for Minimizing Energy Consumption while Limiting Maximum Delay in a Mesh Sensor Network. In *IEEE International Conference on Computer Communications, (INFOCOM)*, pages 258–266, 2007.
- [2] Sohrabi, K.; Pottie, G.J. . Performance of a novel self-organization protocol for wireless ad-hoc sensor networks. In *IEEE Vehicular Technology Conference*, pages 1222–1226, 1999.
- [3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks (Amsterdam, Netherlands: 1999)*, 38(4):393–422, 2002.
- [4] Amre El-Hoiydi and Jean-Dominique Decotignie. WiseMAC: An Ultra Low Power MAC Protocol for Multi-hop Wireless Sensor Networks. In *ALGOSENSORS*, pages 18–31, 2004.
- [5] R. M. Anthony Rowe and R. Rajkumar. Rt-link: A time-synchronized link protocol for energy-constrained multi-hop wireless networks. In *IEEE Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, volume 2, pages 402–411, 2006.
- [6] E. Shih, S.-H. Cho, N. Ickes, R. Min, A. Sinha, A. Wang and A. Chandrasekharan. Physical layer driven algorithm and protocol design for energy-efficient wireless sensor networks. In *IEEE International conference on Mobile computing and networking (Mobicom)*, pages 272–287, 2001.
- [7] S. Ganeriwal, R. Kumar, and M. B. Srivastava. Timing-sync protocol for sensor networks. In *ACM International conference on Embedded networked sensor systems (SenSys)*, pages 138–149, 2003.
- [8] H. M. Khan, S. Olariu, and M. Eltoweissy. Efficient Single-Anchor Localization in Sensor Networks. *DSSNS*, 0:35–43, 2006.
- [9] S. Kulkarni, A. Iyer, and C. Rosenberg. An address-light, integrated MAC and routing protocol for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 14, 2006.
- [10] Lu, G.; Krishnamachari, B.; Raghavendra, C.S. An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks. In *Parallel and Distributed Processing Symposium*, pages 224–233, 2004.
- [11] D. Lucarelli and I.-J. Wang. Decentralized synchronization protocols with nearest neighbor communication. In *ACM International conference on Embedded networked sensor systems (SenSys)*, pages 62–68, 2004.
- [12] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi. The flooding time synchronization protocol. In *ACM International conference on Embedded networked sensor systems (SenSys)*, pages 39–49, 2004.
- [13] N. B. Priyantha, H. Balakrishnan, E. Demaine, and S. Teller. Anchor-Free Distributed Localization in Sensor Networks. Technical Report TR-892, MIT LCS, April 2003.
- [14] A. Savvides, C.-C. Han, and M. B. Srivastava. Dynamic fine-grained localization in Ad-Hoc networks of sensors. In *IEEE International Conference on Mobile computing and networking (Mobicom)*, pages 166–179, 2001.
- [15] Sungrae Cho, Kalyani Kanuri, Jin-Woong Cho, Jang-Yeon Lee and Sun-Do June. Dynamic Energy Efficient TDMA-based MAC Protocol for Wireless Sensor Networks . In *IEEE Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services (ICAS-ICNS)*, page 48, 2005.
- [16] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. In *IEEE Transactions on Wireless Communications*, pages 1(4):660–670, 2002.
- [17] Wei Ye and John Heidemann and Deborah Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12(23):493–506, 2004.
- [18] A. Woo and D. E. Culler. A transmission control scheme for media access in sensor networks. In *IEEE International conference on Mobile computing and networking (Mobicom)*, pages 221–235, 2001.
- [19] T. Wu and B. S. Reducing inter-cluster TDMA interference by adaptive MAC allocation in sensor networks. In *IEEE International Symposium (WoWMoM)*, pages 507–511, 2005.