
Reservation-based protocol for monitoring applications using IEEE 802.15.4 sensor networks

Vidya Krishnamurthy*

Department of Electrical and Computer Engineering,
Clarkson University,
Potsdam, NY 13699-5722, USA
E-mail: krishnav@clarkson.edu
*Corresponding author

Edward Sazonov

Department of Electrical and Computer Engineering,
Clarkson University,
Potsdam, NY 13699-5720, USA
E-mail: esazonov@clarkson.edu

Abstract: The IEEE 802.15.4 and Zigbee are protocols aimed at low-duty and low-power wireless sensor networks. Continuously monitoring applications such as applications of structural health monitoring could benefit from low power consumption of the IEEE 802.15.4 radio chips. However, existing implementation of IEEE 802.15.4 has limited applicability to 'proactive', high-traffic monitoring sensor networks with constant data streams. In this paper, a Time Division Multiple Access (TDMA) scheduler was designed and implemented to resolve issues with collisions and interference, bandwidth usage and delivery of low latency data in a healthy network and to significantly minimise power consumption by sensor nodes. These improvements maintained compatibility with non-TDMA aware nodes. This scheduler was simulated using Network Simulator-2 for different network scenarios that show a remarkable improvement in the network throughput. The proposed mechanism was also successfully tested and implemented in Wireless Intelligent Sensor and Actuator Network (WISAN).

Keywords: sensor networks; wireless; IEEE 802.15.4; monitoring applications; TDMA; time division multiple access; structural health monitoring; WISAN; wireless intelligent sensor and actuator network.

Reference to this paper should be made as follows: Krishnamurthy, V. and Sazonov, E. (200X) 'Reservation-based protocol for monitoring applications using IEEE 802.15.4 sensor networks', *Int. J. Sensor Networks*, Vol. X, No. Y, pp.xx-xx.

Biographical notes: Vidya Krishnamurthy is a PhD candidate in Electrical and Computer Engineering at Clarkson University. She completed her MS in Electrical and Computer Engineering from Clarkson University in 2006 and BTech in Electronics and Communication Engineering from Indraprastha University, India in 2003. Her current research interests include wireless sensor networks, network simulations and signal processing.

Edward Sazonov is an Assistant Professor in the Department of Electrical and Computer Engineering at Clarkson University, Potsdam, New York, USA. His research interests are focused on the area of ambient intelligent systems, including sensor network applications in bioengineering and structural health monitoring, self-powered devices and energy harvesting, and ambient and wearable intelligent devices.

1 Introduction

Wireless Sensor Networks (WSN) consist of a group of sensors or nodes that use wireless links to perform distributed sensing tasks. They combine simple wireless communication, minimal computational facilities and sensing of the physical environment for an application-specific sensor network (Karl and Willig, 2003). The field of sensor networks has experienced tremendous growth in

the recent years with a huge variety of applications. This paper focuses on proactive sensor networks (Sazonov et al., 2004a) that need to deliver constant data streams as opposed to event driven reactive networks.

1.1 Monitoring applications

Applications targeted towards maintaining the health of structures such as bridges may require continuous monitoring

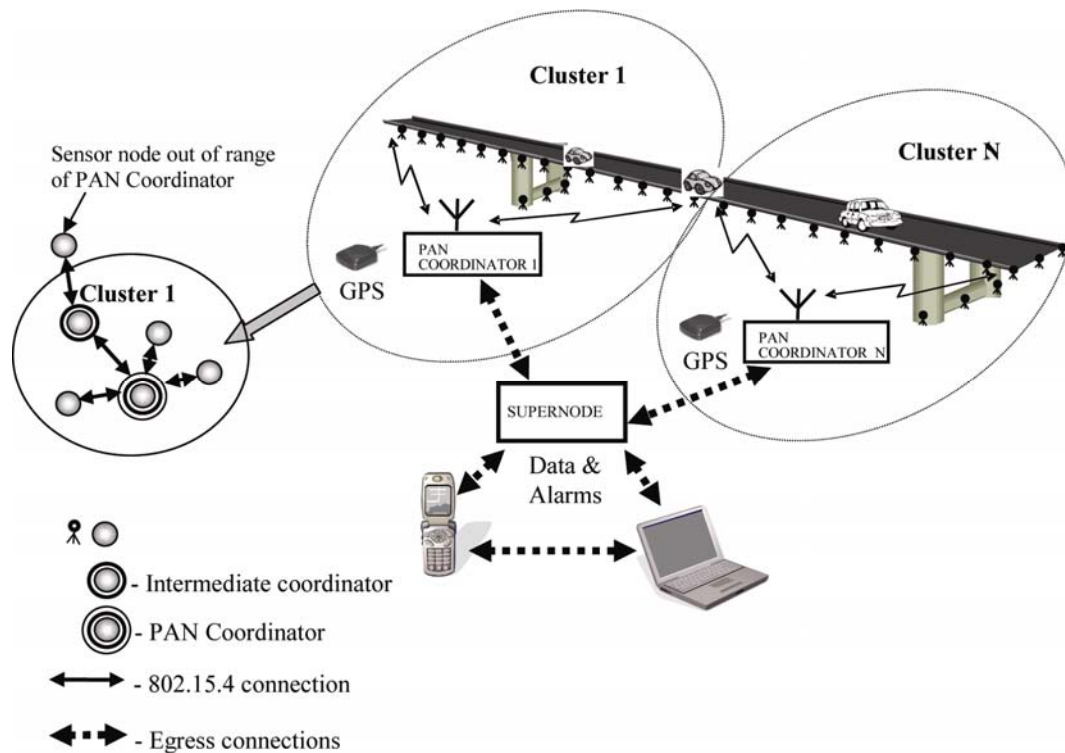
of the status of the system using sensor nodes placed at various locations on the structure. These nodes have sensors that monitor various aspects of the system and send periodic data to a central node. Such a monitoring application requires a network with very low power consumption, real-time, low latency and reliable data. Ideally, the cost of an installed monitoring system should constitute no more than 1–2% of the structure’s cost. Sensor networks can replace extensive cabling of hundreds of sensors that create a reliability problem, and implies high troubleshooting costs (Sazonov et al., 2006). Timely delivery of critical information is extremely important for monitoring applications. Ideally the wireless network should behave in a manner similar to a wired system, where the data samples from different channels are normally synchronised and carry very little latency. Static placement of the nodes enables to utilise an energy efficient hierarchical cluster architecture where most of the communications between sensor nodes and dedicated cluster heads are performed in single hop. A hierarchical cluster architecture inherits the best of star and mesh networks. The cluster heads are an egress to higher bandwidth wired or wireless networks.

An example of such a network monitoring the health of bridges (Sazonov et al., 2004b) is shown Figure 1.

1.2 Why IEEE 802.15.4 protocol?

Traditionally, the most popular medium-access protocol for any wireless network (De Simone and Nanda, 1996) had been the IEEE 802.11 (P802.11, 1995). But its four-way handshaking overhead (Bianchi, 2000), used to increase the reliability of data transfer, proved to be too expensive for low data rate energy scavenging applications. This mechanism also affects the network latency (Zheng and Lee, 2004a) thus making it unfit for monitoring applications with large number of sensor nodes. Thus, IEEE 802.15.4 protocol (P802.15.4/D18, 2003) emerged as a new standard for Low Rate wireless Personal Area Networks (LR-PANs) (Lu et al., 2004). The popularity of the low-power, low-rate IEEE 802.15.4 protocol and availability of low cost, low-power RF chips has led us to use it for developing the low cost Wireless Sensor Networks targeted towards monitoring applications.

Figure 1 A hierarchical cluster-based wireless sensor network for monitoring the structural health of bridges



1.3 Drawbacks of standard CSMA/CA-based protocols

Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) works well for low duty cycle applications but suffers from the following drawbacks for proactive monitoring applications:

- 1 Low reliability of data transfer, that is, there is no guarantee that a node would be able to access the medium for transfer of a single packet before the

allowed maximum number of tries for a packet get over.

- 2 High probability of two or more nodes sensing the channel to be idle in large and dense networks. Due to limited randomness provided by the Back-off Exponent (BE being an integer between 0 and 3), there is high probability of two nodes transmitting their data packets at the same time, hence leading to an increased probability of collisions. These problems lead to an ineffective usage of the available

bandwidth, decreased throughput and increased latency of the network.

- 3 Inefficient usage of the available battery power due to:
 - a *Idle listening*: When a node does not send or receive a packet but keeps listening to the transmissions in the channel.
 - b *Retransmissions due to collision*: If a packet is lost due to collision, it needs to be retransmitted. This causes wastage of bandwidth and power and a decrease in reliability of data transfer (Mistic et al., 2006a). As the further simulations show, the energy loss due to retransmissions can lead up to 600% increase in power consumption.

Simulations have shown a significant decrease in bandwidth utilisation and throughput under heavy load (Bianchi et al., 1996; Bianchi, 1998; Bianchi, 2000; Tantra et al., 2004). At the same time, energy consumption and latency grow significantly. The poor performance of the slotted CSMA/CA in IEEE 802.15.4 protocol has also been analysed using discrete time Markov chain models (Mistic et al., 2006b). This paper attempts to improve the reliability of data transfer for proactive monitoring networks along with increase in the bandwidth usage and reduction in power consumption and latency. Such improvement is achieved through use of a reservation-based protocol.

1.4 Review of wireless reservation-based protocols

A reservation-based scheme applied to the described application scenario could alleviate the problems caused by CSMA/CA in high duty cycle applications. There are no known reservation-based schemes for IEEE 802.15.4 but a review of existing publications has indicated a number of reservation-based techniques developed for IEEE 802.11 contention-based protocol that have been proposed and implemented on ad hoc WSNs. These mainly reduce loss of energy occurring due to idle listening, control packet overhead, collisions and overhearing (Miller and Vaidya, 2004).

The IEEE 802.11 standard specifies a Power Saving Mode (PSM) (P802.11, 1995) that attempts to reduce energy loss. In PSM, a certain period at the start of every Beacon Interval (BI), called *Announcement Traffic Indication Map (ATIM) window*, is used by all nodes to advertise any packets that they need to send using an ATIM packet. If the ATIM packet is acknowledged, both the source and the destination remain 'on' after the ATIM window to send and receive the advertised packet. Although a simple protocol to implement, the 802.11 PSM suffers from several disadvantages. Even if a node has only one packet to send, it needs to remain powered on for the entire BI, potentially leading to large energy wastage. Also, since the protocol uses 802.11's CSMA/CA mechanism to avoid collisions, it might not work well when a large number of competing nodes try to send their packets immediately after the ATIM window rather than spreading the packet transmissions out uniformly over BI.

Power Aware Multi-Access protocol with Signalling for Ad Hoc Networks (PAMAS) is a multi-access protocol developed by Singh and Raghavendra for ad hoc radio networks (Raghavendra and Singh, 1998). This protocol provides a separate signalling channel that enables to power off the nodes that are not transmitting or receiving. All control signals, such as the Ready to Send-Clear to Send (RTS-CTS) handshake signals, are exchanged over this signalling channel. One disadvantage of this protocol is that it ignores the idle listening problem. Also, as it requires two independent radio channels, this in most cases indicates two independent radio systems on each node which increases the hardware requirements per node.

The *S-MAC* mechanism was proposed by Ye et al. (2002) for multi-hop ad hoc wireless sensor networks. Here, each node periodically sleeps during transmissions of other nodes to reduce energy consumption due to idle listening. Neighbouring nodes form *virtual clusters* to auto-synchronise sleep schedules. If multiple neighbours want to talk to a node, they contend for the medium when the node is listening using the IEEE 802.11 contention mechanism (P802.11, 1995). Each node maintains a *Schedule Table* that stores the schedule of all its neighbours. Schedule tables are chosen by random selection using a SYNC packet. Since multiple senders may want to send to a receiver at the same time, they need to contend for the medium to avoid collisions. This protocol would not work well when every node may need to perform periodic transfer of data since it does not avoid the problem of contention. A schedule needs to be set up for every transmission, and a large amount of memory may be required to store and maintain neighbour's schedules. T-MAC is another contention-based medium access protocol proposed by van Dam and Langendoen (2003) that provides an extension to S-MAC by adjusting the length of time the sensors are awake between sleep intervals based on communication of nearby neighbours.

Miller and Vaidya (2004) presented *On-Demand TDMA Scheduling for Energy Conservation* where two variants of an on-demand Time Division Multiple Access (TDMA) MAC protocol have been proposed. A flow of data packets is sent every T_{cycle} time. An ACK is sent if the data are received correctly. If otherwise, a data packet is not received in a scheduled slot, a NACK packet is sent. The NACK packet has a bit which indicates whether a collision was occurred or the packet was just received with an error or lost. Each sensor maintains a table of K_{cycle} entries ($T_{\text{cycle}} = K_{\text{cycle}} \times T_{\text{slot}}$) indicating in which slots the sensor should be awake to transmit and in which it should be awake to receive data from a neighbour. At the beginning of every cycle all nodes wake up for a few slots, called the *FLOW-ADV* window, which takes a total time of T_{adv} . In this time, the packet transmission in slots is scheduled by transmitting a *FLOW-ADV* packet using contention mechanism. This is similar to 802.11's ATIM window except flows, instead of packets, are being advertised. A node could transmit a *FLOW-DEL* message during the flow's scheduled data slot to destroy a scheduled slot. Again, this method does not guarantee channel access to all the nodes that have

messages. There is contention in the *FLOW-ADV* window that restricts the access to only a few nodes at a time, again not guaranteeing reliability of data transfer to all the nodes that have data at the same time.

Another approach was proposed by Sichitiu (2004) in *Cross-layer Scheduling for Power Efficiency in Wireless Sensor Networks*. Here, time-synchronised sensors form on-off schedules that enable the sensors to be awake only when sending or receiving data and sleep the rest of the time. The network is divided into two phases: *the setup and reconfiguration phase* and *the steady state phase*. In the former, schedules of data flow are set up which are used in the latter. Each node maintains a schedule table that specifies when the node has to sample (*S*) the data, transmit (*T*) a packet to the next hop on the path or receive (*R*) a packet that needs to be transmitted further. A node selects a route to the base station by using any of the available routing protocols and then tries to set up this selected route by sending a special route setup (*RSETUP*) packet on the selected route. At each intermediate node, the *RSETUP* packet finds a time when the data packet could be scheduled without colliding with other nearby transmissions. The appropriate entries are then appended in the temporary schedules of the two nodes. The actions in the temporary schedule are copied into the permanent schedule if the route acknowledgment (*RACK*) is sent from the base station on the reverse path, or are purged upon timeout. A disadvantage of this method would be that the transmission times are first decided by the nodes and then require the approval of the base. Hence, each node is initially unaware of the schedule of the other nodes. When the base slot has to perform this resolution and for a large network, this would be complicated. This protocol may cause significant data packet loss while trying to set up a flow by causing multiple collisions with the *RSETUP* packet. Also, this protocol may destroy an existing flow while setting up a new flow. The *Traffic-Adaptive Medium Access Protocol* (TRAMA) (Rajendran et al., 2003) is similar to the *On-Demand TDMA Scheduling for Energy Conservation* that uses a distributed election scheme based on information about traffic at each node to determine which node can transmit at a particular time slot. It uses a node's two-hop information to derive a collision-free transmission schedule but has the same limitations as the On-Demand CSMA. Other similar methods have been proposed by van Hoesel et al. (2003) where nodes are provided with the topology of the network and each node autonomously chooses a time slot. In this protocol, time is divided into so-called frames and each frame is divided into *time slots*. Each slot is assigned to a particular node that decides which nodes will communicate during the slot. Each slot has three sections: (a) Communication Request (CR) section where nodes can request for the slot, (b) Traffic Control (TC) section where the owner of the slot decides which nodes will communicate in the slot and (c) data section where the data transfer takes place. New nodes autonomously pick a time slot when they join the network by gathering the list of all the occupied time slots and

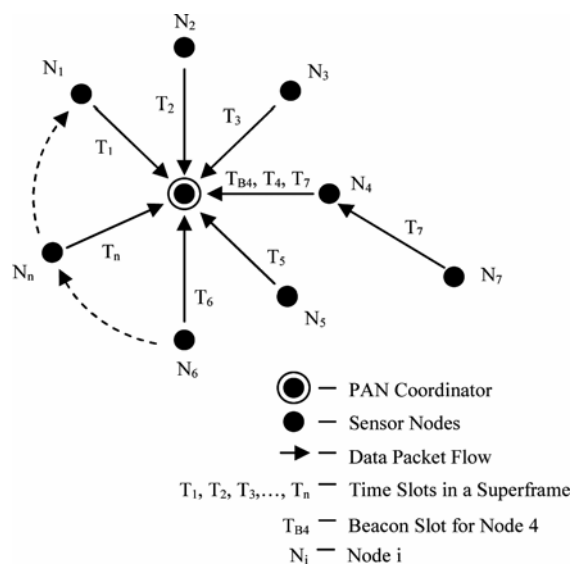
choosing the one that is free. This protocol does not avoid the potential problem of collisions in the CR section; hence, a collision handling mechanism needs to be incorporated. This protocol does not provide reliability of data transfer by all nodes in the network due to unequal assignment of the slots in the CR section.

Though all the above methods provide effective conservation of energy, they do not help much in conserving the bandwidth or increasing reliability of data transfer and latency of the networks. Thus, in order to maximise the bandwidth usage, minimise power consumption while retaining the flexibility of MAC protocol (CSMA/CA) and complying with the desired IEEE 802.15.4 protocol, a TDMA-based method is proposed and developed. The next sections describe a low-power extension for IEEE802.15.4 protocol along with a detailed description, simulation and implementation of the scheme.

2 Methods

TDMA is digital transmission technology that allows a number of nodes to access a single Radio Frequency (RF) channel without interference by allocating unique time slots to each user within each channel (<http://en.wikipedia.org/wiki/TDMA>). Figure 2 shows the basic transmission scheme of a general TDMA network consisting of a single PAN Coordinator and a number of sensor nodes, say, N_1, N_2, \dots, N_n . The nodes exchange data with the PAN Coordinator only during an allotted time slot (T_1, T_2, \dots, T_n , as shown in Figure 2), either directly or through an intermediate node within the transmission range of the PAN Coordinator. Thus, we suggest implementing this time-based reservation scheme at the Service Specific Convergence Layer (SSCS) of the devices in the network while allowing the lower layers to use CSMA/CA channel access mechanism in each of the virtually assigned time slots.

Figure 2 TDMA-based transmission scheme



2.1 Proposed method

For the considered application scenarios with static placement of nodes on a structure, a beacon-enabled star topology that provides single-hop communication and time synchronisation for nodes is preferred inside a cluster. However, for large-sized networks with nodes lying outside the transmission range of the coordinator, a provision for a multi-hop cluster should be provided. Figure 1 shows the two-level hierarchical cluster architecture of a sensor network used in structural monitoring application.

According to the IEEE 802.15.4 protocol, each node associates with the network through a well-defined association procedure during which it requests bandwidth for transmission of data. The PAN Coordinator assigns time slots to each node based on a scheduling algorithm which ensures maximum bandwidth utilisation. Thus, every node N_n would virtually find the assigned time slot T_n to be free and could exchange data using the CSMA/CA access mechanism. Due to static placement of sensor nodes on structures, these slot assignments are done at the time the node joins the network and may be assumed to be static throughout the lifetime of the node, unless a change is requested by the node or the coordinator. The proposed scheme is also capable of working with non-TDMA aware nodes, mobile nodes and multi-hop routing. In the proposed scheme, every sensor node makes requests for bandwidth when it associates with the network and transmits or receives data in these fixed time slots.

According to the CSMA/CA mechanism used in IEEE 802.15.4, every slot transmission is aligned with a back-off period boundary. This smallest time slot or the back-off period is defined by the MAC layer as a *Unit Backoff Period* and has a fixed value of 20 symbols or 80 bits. The Beacon Order (BO) may be initialised to any value between 0 and 14. BI is dependent on BO as

$$BI = 15.36 \times 2^{BO} \text{ ms.} \quad (1)$$

A fixed number of BIs n_B define the scheduling period T of the network. The traffic flow in the network is considered to be periodic with this scheduling period T . This scheduling period expressed in back-off periods is called as B . Thus, for any BO and n_B , the scheduling period of the network is

$$T = n_B \times 15.36 \times 2^{BO} \text{ ms} \quad (2)$$

$$B = n_B \times 48 \times 2^{BO} \text{ backoff periods.}$$

A TDMA-based network divides the available bandwidth, T , into time slots and allots fixed slots to every node that wishes to transfer periodic data on the network. A potential problem the network could face is when any node transmits out-of-turn, that is, other than its allotted slot. An example of such a transmission could be when a node is attempting to associate with the network and causes collision with the node that has been scheduled to transmit in the slot. In such situation, the scheduled data must be retransmitted to maintain reliability. Thus, a time slot in every BI is reserved for any out-of-turn

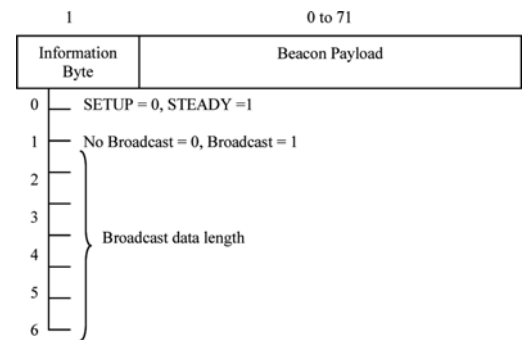
transmissions. This period is known as the *Reserved Time Slot*, T_R . Each node in the network knows the availability of the reserved time slot. Hence, any packet that was not able to be transmitted in the allotted time slot is transmitted in the reserved time slots using CSMA/CA medium access. A time slot immediately after the beacon is marked as T_R . The size of this slot is an important parameter for analysis (performed in a later section of this paper).

The basic assumptions of the network model are:

- 1 Network operates using two types of channel access mechanisms:
 - a TDMA_CSMA (For data transmissions in allocated time slots): The CSMA BE is set to 0, in order to avoid delay in channel access. Since the data packets are transmitted in pre-defined time slots, they do not require back-offs. This gives higher priority to information from a node that has been assigned the slot over any other node.
 - b IEEE_CSMA (For any other packet transmission): The CSMA BE is chosen at random.
- 2 Traffic is assumed to be of Constant Bit Rate (CBR).
- 3 Depending on the data rate, a node may have more than one slots scheduled in a BI. This scheduling is static for the node unless the node requests a change or decides to leave the network.

The network operates in two distinct phases, the Setup Phase and the Steady State Phase.

Figure 3 Beacon payload



The *setup phase* occurs when the network is started for the first time. This phase is indicated by setting a bit in the beacon payload as shown in the beacon payload format in Figure 3. No data transmission takes place in this phase. Time slot allocation is mostly done during this phase. Nodes are allocated time slots during the association procedure. Each node scans the set of frequency channels to obtain a list of available PAN Coordinators in its range. It selects the PAN Coordinator that is in nearest proximity, matches the network ID and sends a request for association along with the required *scheduling request parameters*. This request is sent by using the normal IEEE_CSMA

routine. The PAN Coordinator on receiving the association request sends an acknowledgement and tries to schedule time slots for the node to accommodate the entire bandwidth requested.

The scheduler works as follows. The node sends a scheduling request consisting of the number of bytes of data (χ bytes), time period for transmission of data (t_{node} ms), direction of data transfer ($txdir$; from PAN Coordinator to sensor node or from node to PAN coordinator) and the type of scheduling required ($sctype$). There could be two types of scheduling – hard scheduling or soft scheduling. A hard scheduler assigns data time slots to the node for χ bytes exactly at an interval of t_{node} ms as requested by the node. Thus, a hard time scheduler maintains the periodicity of the node which may be important for control applications. A soft scheduler, on the other hand, assigns the node any data time slot (of size χ bytes) anywhere within the t_{node} ms. The selection of the type of scheduling depends on the frequency of availability of data and the buffer size of the node. For example, if a node has a buffer size of just one packet and samples data at fixed intervals, then it needs to transmit the data as soon as it is sampled. Thus, hard scheduling would be preferred. On the other hand, if a node has a larger buffer size, then using a soft scheduler provides more flexibility for scheduling slots. A time slot may be inserted anywhere every t_{node} ms. Scheduling slots in succession helps reduce energy wastage of each node by eliminating multiple sleep-wake cycles. For calculation of any time slot, the following two criteria must be satisfied:

- 1 The maximum time of transfer of the data is used; that is, all the inter-frame spacings are considered maximum (P802.15.4/D18, 2003).
- 2 Every time slot has to be rounded off to the next back-off period.

The scheduler finds the total number of slots that the node needs to be allotted in the network time period. The type of scheduling is determined based on the requested type of scheduling, $sctype$. The requested direction of data transfer, $txdir$ determines the overhead for each packet. A node may be ‘un-schedulable’ if the total requested bandwidth is less than that available. The slots assigned to the nodes are stored by the PAN Coordinator as a linked table structure.

After the scheduling is completed, the *scheduling response parameters* are returned to the node in the payload in the association response packet that is sent as a part of the standard association request. These parameters contain information about the allotted slots in terms of beacon ID, slot size (maximum amount of data that may be transferred in the slot) and location of the slot (how many back-off intervals to wait for the slot). These parameters are also stored by the sensor node in a linked table structure. If slots could not be allocated for the node, the PAN Coordinator sends an association response indicating a failure. On successful association, the node waits for the steady state and transmits data in the allocated slots. The node association algorithm can be summarised as shown in Figure 4.

After a minimal number of nodes have been associated in the network or the association requests fail repeatedly due to lack of any more slots, the network enters the *steady state phase*. The bit in the beacon payload indicates if the network is in steady state. Each node transmits data in its assigned time slot. The different traffic types in the network are handled as follows:

- 1 *Data packets from Sensor Node to PAN Coordinator:* These flows occur in the scheduled time slots.
- 2 *Data packets or messages from PAN Coordinator to Sensor Node:* The PAN Coordinator indicates the device address in its pending addresses field in the beacon. The node sends a data request in its allotted slot to the PAN Coordinator which is followed by an acknowledgement by the PAN Coordinator along with the data or message.
- 3 *Broadcasts:* The PAN Coordinator sends broadcast messages as the beacon payload.

The reserved time slot takes care of further associations, retransmissions, request for BW increase or decrease and messages from PAN coordinator to devices (maximum seven nodes according to the standard (P802.15.4/D18, 2003)).

Figure 4 Node association algorithm during setup phase

```

1: channel scan
2: if coordinators not found
3:   association fail
4:   goto 1
5: elseif no coordinators permit association
6:   association fail
7:   goto 1
8: else
9:   select a proper coordinator
10:  send association request to coordinator with
    scheduling request parameters
11:  wait for ACK
12:  if ACK not received
13:    association fail
14:    goto 10 for 3 times then goto 1
15:  else
16:    send data request to the coordinator
17:    wait for ACK
18:    if ACK not received
19:      association fail
20:      goto 16 for 3 times then goto 14
21:    else
22:      wait for association response
23:      if association response not received
24:        association fail
25:        goto 16 for 3 times then goto 14
26:      elseif association not granted
27:        association fail
28:        select different coordinator or goto 1
29:      else
30:        association success
31:        store scheduling response parameters
32:        synchronize with next beacon

```

In the steady state, the PAN Coordinator periodically transmits beacons with all the information in its payload as shown in Figure 3. Each node wakes up to receive these beacons periodically and goes back to sleep if no action is required. In the reserved slot immediately following the beacon, the PAN Coordinator waits for any retransmission or association request. If an association request is received, it is processed and slots are assigned to the node. Any retransmission received is acknowledged and stored with proper time stamp. After the reserved time slot, the PAN Coordinator waits for data transmissions from the nodes that have been allocated slots. Any unscheduled transmission in the allocated slots could also be processed by the PAN Coordinator, for example, association requests from other nodes. Any information that needs to be sent by the PAN Coordinator to all nodes (broadcast) is sent in the payload of the beacon (Figure 3). The PAN Coordinator is responsible for maintaining the network. It is active all the time and continuously receives data from associated nodes. This operation of the PAN Coordinator in steady state can be summarised as in Figure 5.

Figure 5 Operation of PAN Coordinator in steady state

```

1: send beacons periodically
2: if setup phase
3:   send beacons indicating setup state
4:   wait for association requests
5:   if association request
6:     if number of nodes < max. required or nodes still
        schedulable
6:       assign slots using scheduling algorithm
7:       send association response
8: elseif steady state phase or no. of nodes = max. nodes
9:   send beacons indicating steady state
10:  if reserved slot
11:    wait for association requests and
        retransmission
12:    if association request
13:      assign slots using scheduling algorithm
14:      send association response
15:    if retransmission
16:      store data of node with time stamp
17:  if assigned slot
18:    if data from node assigned to the slot
19:      store data of node with time stamp
20:    else if packet from other node
21:      process data accordingly
22:  if broadcast message
23:    send as beacon payload
24:  if message or data to particular node
25:    add node address in list of pending addresses
        in beacon
26:    wait for data request from node
27:    if data request
28:      send the data or message to node

```

After a sensor node gets its scheduling response parameters, it goes into sleep mode and wakes up only after the network is in steady state and the beacon

indicates start of data transmission. In this state, the node that has a data packet to be transmitted checks for the next available scheduled slot and sends the data using TDMA_CSMA mechanism. If any transmission is unsuccessful, the node transmits the data in the next available slot within the beacon or in the reserved time slot in the next beacon. The node also checks for any broadcast message or any messages destined for the node in the list of pending addresses in the node. The node also synchronises with every beacon. The working of an associated node in steady state is summarised in Figure 6 and also depicted in Figure 7.

2.2 Theoretical analysis of the proposed method

A specific application may require a different number of sensor nodes in a cluster and different bit rates from each of the nodes. In order to establish limits on such parameters of the network as the maximum number of nodes and optimal duration of the reserved time slot, a theoretical analysis of the proposed method is performed. The maximum size of the network is restricted by the maximum possible bandwidth that may be allocated to the nodes. The theoretically possible maximum bandwidth is 250 kbps which does not account for overhead in the network due to packet header, interframe spacing, acknowledgement frames and beacon frames. Consider a network of beacon order BO and scheduling period T (equation 2) with T_B and T_R being the beacon transmission time and reserved slot time, respectively. Let every node transmit χ bytes of data t_{node} seconds, that is, γ kbps. The total throughput of the network with each node allotted S_{node} time slots is

$$\pi_{\text{th}} = \frac{\chi N S_{\text{node}} 8}{T} \text{ kbps.} \quad (3)$$

Let α bytes be the total overhead including the maximum interframe spacing (LIFS), acknowledgement size and maximum wait time for an acknowledgement (t_{ACK}). From Equations (1), (2) and (3), the throughput of the network is

$$\pi_{\text{th}} = \frac{\chi^* (48 \times 2^{\text{BO}} - (T_B + T_R)) 8}{15.36 \times 2^{\text{BO}} \left\lceil \frac{\chi + \alpha}{10} \right\rceil} \text{ kbps.} \quad (4)$$

For maximum value of α and maximum data packet size (127 bytes including headers), the time slot required is given as $T_s = 20$ back-off slots.

$$\Rightarrow \pi_{\text{th}} \approx \frac{[48 \times 2^{\text{BO}} - (T_B + T_R)] 8 \chi}{15.36 \times 2^{\text{BO}} \times 20} \text{ kbps.}$$

The maximum number of nodes that can successfully join to form a network is limited by this maximum throughput as

$$N^{\text{max}} = \frac{\pi_{\text{th}}}{\gamma}. \quad (5)$$

Figure 6 Operation of the sensor node in steady state

- 1: synchronize time with the next beacon after association response
- 2: wake-up every beacon and check bit for steady state
- 3: **if** setup state
- 4: wake-up every beacon and check bit for steady state
- 5: **elseif** steady state and start of data transmission
- 6: look scheduled parameters for next beacon id to listen
- 7: sleep till that time
- 8: wakeup and check beacon id
- 9: **if** beacon id not equal to scheduled beacon id
- 10: check beacon order, Pan Id
- 11: calculate time error
- 12: synchronize time with current beacon
- 13: **if** early wake-up
- 14: set parameters to wake-up for current beacon
- 15: **else**
- 16: discard transmissions for the beacon
- 17: set parameters to wake-up for next scheduled beacon
- 18: **else**
- 19: synchronize time with current beacon
- 20: look for broadcast message in beacon
- 21: look for pending address list to see if node address present
- 22: **if** node address in pending address
- 23: send data request in the reserved slot
- 24: get data from PAN Coordinator
- 25: sleep till scheduled data transmission slot
- 26: wake-up and transmit packet
- 27: **if** success
- 28: sleep till next slot or beacon (whichever is earlier)
- 29: **else**
- 30: retransmit in reserved slot in next beacon

In an ideal network, there would be no collision or interference with the packets transmitted in an allotted

time slot. But, in general, in order to compensate for packet collisions, a time slot in every BI, called the reserved time slot, is provided. The size of this slot is critical for maintaining the reliability of the network. The size of the reserved time slot which provides for all the retransmissions required is the optimal value of the reserved time slot for a network with a certain noise level. In order to establish this optimal value of the reserved time slot, let the size of the reserved time slot be kept at the maximum bandwidth left after all the possible nodes are scheduled. Hence, the size of the slot may be obtained as a function of the number of nodes as

$$T_R = \frac{(n_B[48 \times 2^{BO} - T_B]) - \sum_{i=1}^{num_nodes} (S_{node} * T_s)_i}{n_B} \text{back-off slots.} \quad (6)$$

For a network with all nodes transmitting χ bytes of data at a constant rate of γ kbps, T_R can be represented as

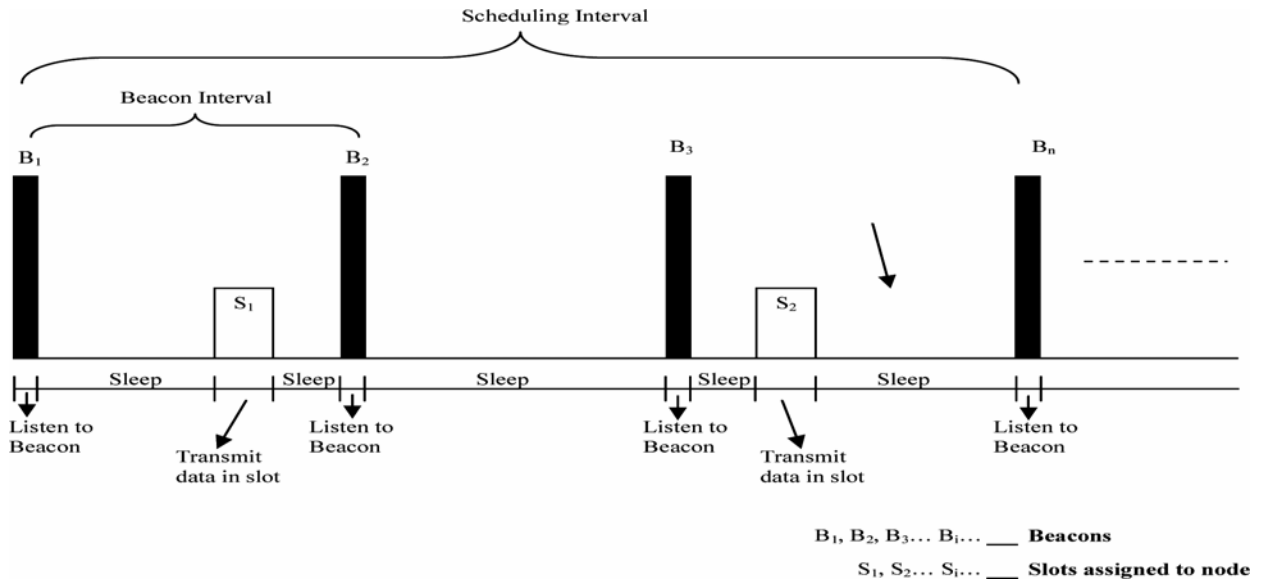
$$T_R = [48 \times 2^{BO} - T_B] - N \frac{15.36 \times 2^{BO} \times \gamma}{8\chi} T_s \text{ back-off slots.}$$

For $\chi = 100$ bytes, $BO = 4$ and $\gamma = 3.2$ kbps, $T_B = [T_B^{max}] = 15$ and $T_s = 20$ back-off slots ,

$$T_R = 753 - 19.6608 \times N \text{ back-off slots.} \quad (7)$$

Thus, for no reserved time slot, a network with 38 nodes may be realised with 3.2 kbps data rate per node, 76 nodes with 1.6 kbps and 153 nodes with 0.8 kbps. Due to noise in the simulation (minimum noise due to routing packets, as described in the next section), the optimal size of the reserved slot, when determined by simulations, reduces this number of nodes to 34, 68 and 136, respectively, for the three data rates. The maximum theoretical throughput of the network, when every node transmits 100 bytes of data at the rate of 3.2 kbps and the BO is taken as 4 and optimal values for T_B and T_R are taken, is calculated as about 108.8 kbps.

Figure 7 Node operation in steady state phase



Let P_n be the percentage of noise in the network that includes noise in the channel and the noise induced due to collisions with routing packets. The minimum size of the reserved time slot (T_R) should be such that the number of nodes accommodated in the remaining time should be able to transmit any missed packet (due to collisions or channel error) in the period T_R . Hence, the total number of packets (ψ_{total}) transferred in the scheduling period of the network is given as

$$\begin{aligned} \psi_{\text{total}} &= N\psi_{\text{node}}T_{\text{avail}} \\ \psi_{\text{node}} &= \frac{\gamma \times 1000}{8 \times \chi} = \frac{\gamma \times 1000}{8 \times 100} = \frac{5 \times \gamma}{4} \text{ packets/sec,} \\ T_{\text{avail}} &= n_B [15.36 \times 2^{\text{BO}} - ((T_B + T_R)80/250)], \\ \Rightarrow \psi_{\text{total}} &= N \frac{5r}{4} n_B [15.36 \times 2^{\text{BO}} - ((T_B + T_R)80/250)]. \end{aligned} \quad (8)$$

For a successful scheduling, considering that a packet should be retransmitted in a single attempt, the total available reserved time slot size should account for retransmissions for all the dropped packets.

$$\begin{aligned} \psi_{\text{total}} P_{\text{noise}} &= \frac{n_B T_R}{T_s^{\text{max}}} = \frac{n_B T_R}{20}, \\ \Rightarrow N &= \frac{T_R}{25r [15.36 \times 2^{\text{BO}} - ((T_B + T_R)80/250)] P_{\text{noise}}}. \end{aligned} \quad (9)$$

A mixed network is a network consisting of both TDMA based and pure CSMA/CA based nodes. Consider for example a network with fixed as well as mobile nodes where the mobile nodes enter and leave quickly. In such a scenario, it would not be feasible for time slot allocation through the regular association procedure due to time overhead of the association procedure. However, these nodes could communicate through the regular CSMA/CA mechanism and the impact of this unmanaged CSMA/CA traffic is thus studied. The TDMA-based nodes (N_T) are allocated slots during the association procedure. The CSMA/CA based nodes (N_C) transmit data whenever they sense the channel to be free. Let the total number of nodes in the network be $N_T + N_C$. Consider a network with N nodes and ζ percentage of TDMA nodes. Let num_slots be the total number of time slots required per node for independent transmission of data (without collision) in n_B BIs of the network. The TDMA nodes are already provided with this bandwidth. Every slot is allotted for transmission of a single maximum-sized data packet (for maximising the throughput). Thus, let the number of slots per BI per node be $S_{\text{node}}^{\text{BI}}$.

$$S_{\text{node}}^{\text{BI}} = \frac{S_{\text{node}}}{n_B}.$$

Hence, the total bandwidth required per BI is given by

$$\begin{aligned} T_{\text{avail}} &= [S_{\text{node}}^{\text{BI}} * N] T_S \text{ back-off periods} \\ &= [S_{\text{node}}^{\text{BI}} (N_T + N_C)] 20 \text{ back-off periods.} \end{aligned}$$

Let ρ be the probability of collision using CSMA/CA and P_{noise} be the percentage of channel noise in the network. The probability of collision for a node in a TDMA only

network is ideally equal to 0. Let P_{tdma} be the probability of collision of a CSMA/CA node packet with a TDMA node packet. Let P_{csma} be the probability of collision of a CSMA/CA node packet with another CSMA/CA node packet. Hence, the number of dropped packets is given by,

$$\begin{aligned} \psi_{\text{dropped}} &= [S_{\text{node}}^{\text{BI}} (N_T + N_C)] P_{\text{noise}} + 2N_C (P_{\text{tdma}} + P_{\text{csma}}) \\ &\text{where } (P = (P_{\text{tdma}} + P_{\text{csma}})). \end{aligned}$$

The time for retransmission for the packets dropped by TDMA nodes is $T_R^{\text{tdma}} = T_R$ and for CSMA/CA nodes, $T_R^{\text{csma}} = [BI - T_B]$ back-off periods. Let the number of packets to be retransmitted by TDMA nodes be given by R_{tdma} and that by CSMA/CA nodes be given by R_{csma} ,

$$\begin{aligned} R_{\text{tdma}} &= n_{\text{slots}} N_T P_{\text{noise}} + N_C P_{\text{tdma}} \\ R_{\text{csma}} &= n_{\text{slots}} N_C P_{\text{noise}} + N_C (P_{\text{tdma}} + 2P_{\text{csma}}). \end{aligned}$$

Hence, R_{tdma} and R_{csma} give the minimum number of packet retransmissions required for packets dropped in a single BI. Therefore, for a stable mixed network, the minimum criteria to be satisfied will be

$$\begin{aligned} R_{\text{tdma}} T_S &\leq T_{\text{tdma}} \\ R_{\text{csma}} T_S &\leq (T_{\text{csma}} - R_{\text{tdma}} T_S). \end{aligned}$$

For the next BI, in addition to the original number of packets dropped, we have additional dropped packets due to the probability of collision of the retransmitted packets. Hence,

$$\begin{aligned} \psi_{\text{dropped}} &= [R_{\text{tdma}} + R_{\text{csma}}] + [R_{\text{tdma}} + R_{\text{csma}}] P_{\text{noise}} + 2R_{\text{csma}} (P_{\text{tdma}} + P_{\text{csma}}) \\ &= [R_{\text{tdma}} + R_{\text{csma}}] (P_{\text{noise}} + 1) + 2R_{\text{csma}} (P_{\text{tdma}} + P_{\text{csma}}). \end{aligned}$$

Hence, the number of retransmissions required in the i th beacon is given by,

$$\text{num_retx}_i = [R_{\text{tdma}} + R_{\text{csma}}] (P_{\text{noise}} + i) + 2R_{\text{csma}} (P_{\text{tdma}} + P_{\text{csma}}). \quad (10)$$

The maximum time for TDMA retransmissions in the i th beacon is $i * T_R$ while for CSMA/CA nodes is $i[BI - T_B]$ back-off periods. The maximum number of retries for transmission of every packet is 3; hence for stability of the network, within three tries, a packet generated must be transmitted.

Provided analysis enables establishing the optimal duration of the reserved time slot for a network of given size (equation 7) or establishing the upper bound on allowed noise levels (equation 9) that would not cause unrecoverable disruption of network traffic. Both of these estimates are extremely important for practical monitoring applications.

2.3 Simulations

In order to test the performance of the proposed network model, a simulation environment of the actual network was developed using the simulator Network Simulator ns-2 (Varadhan, 2000). The IEEE 802.15.4 extension to the simulator was provided by Zheng (Zheng and Lee, 2004b) which was used for analysing the performance of the current model. The TDMA Scheduler was

implemented at the Service Specific Convergence Sublayer (SSCS) that acts as an interface between the MAC and the upper layers as shown in Figure 8. The SSCS layer is an implementation specific module that provides access to the MAC primitives and allows for their modification for any specific application. The topology used for performing simulations consists of an array of 11×11 nodes, with a distance of 3 m between them, arranged as shown in Figure 9. A wireless channel with two-ray ground propagation model is chosen. Each node is chosen to have omni-directional antenna with a maximum reception threshold of 100 m. The node at the centre of the topology (node number 60) is the PAN

Coordinator. Every node joins the network by associating with the PAN Coordinator and transmits data packets using CBR traffic type. The maximum number of packets allowed in the interface queue is fixed to be 4, and the packets are transmitted using single-hop AODV routing protocol.

Figure 8 Embedding TDMA Scheduler in ns-2

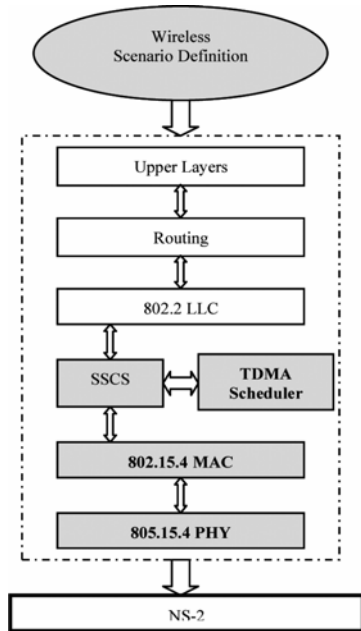
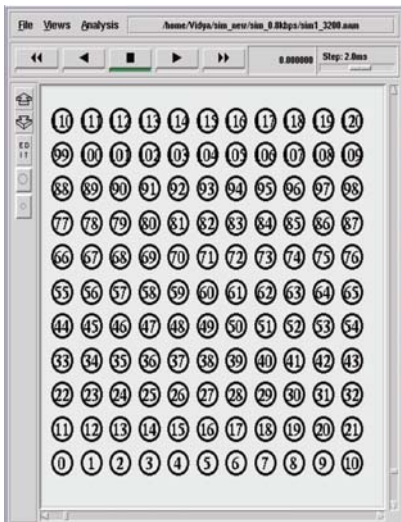


Figure 9 Simulation topology



In order to analyse the drawbacks of CSMA/CA, we performed simulations of traditional IEEE 802.15.4 network. A beacon-enabled slotted CSMA/CA was chosen where beacons are sent by the PAN Coordinator and all nodes in the network synchronise with the beacon. A single-hop Star topology was considered. The network performance was analysed using two transmission scenarios; first where a single node sends data to the cluster head or PAN Coordinator at CBR, and, second, where 20 nodes send data to the PAN Coordinator at CBR. The traffic load was varied from 5 to 200 kbps, and results were averaged over 10 different random seeds. The highest throughput achieved when there is one source is around 110.67 kbps as shown in Figure 10(a). This is below the nominal value of 250 kbps due to the presence of random back-offs, acknowledgement messages and settings for large inter-frame spacing. This value is close to the theoretical maximum throughput obtained at 108.8 kbps in Section 2.2. The decrease of the theoretical throughput from the ideal value of 110.67 kbps is due to rounding off to the next back-off interval for scheduling and accommodating a finite integer number of slots in a BI. For multiple sources, throughput rises to about 22 kbps and then begins to fall due to collisions. The packet delivery ratio also falls drastically with an increase in data rate as shown in Figure 10(b).

In order to analyse a continuously monitoring wireless sensor network with a number of nodes sending data at a CBR, a set of simulations was performed by fixing the traffic flow rate from each node and varying the number of nodes in the network. A network scenario with up to 36 nodes transmitting the maximum-sized packets data at a fixed constant data rate of 3.2 kbps is taken. The obtained throughput (25 kbps maximum) and packet delivery ratio (16%) are very low (Figure 11) when compared to the ideal theoretical networks. This in turn shows that in order to deliver all the packets, the nodes need to retransmit each packet six times, thus increasing the power consumption and latency of the network by 600%.

To analyse the performance of the proposed model, the following sets of simulations were performed.

- 1 A comparison of the standard IEEE 802.15.4-based networks without TDMA slot scheduling and the networks built with the TDMA time slot scheduling.
- 2 A comparison of the throughput and delivery ratio of packets at different data rates of 3.2 kbps, 1.6 kbps and 800 bps per node, respectively, with the theoretical limits.
- 3 The effect of variation of the size of reserved time slot on throughput and packet delivery ratio of the network.
- 4 The effect of the percent of nodes in the network using the TDMA slot scheduling on the throughput and packet delivery ratio of the network.
- 5 The effect of noise on the delivery ratio of packets.

Figure 10 (a) Throughput and (b) packet delivery ratio for existing networks for 1 and 20 source nodes using IEEE 802.15.4 CSMA/CA

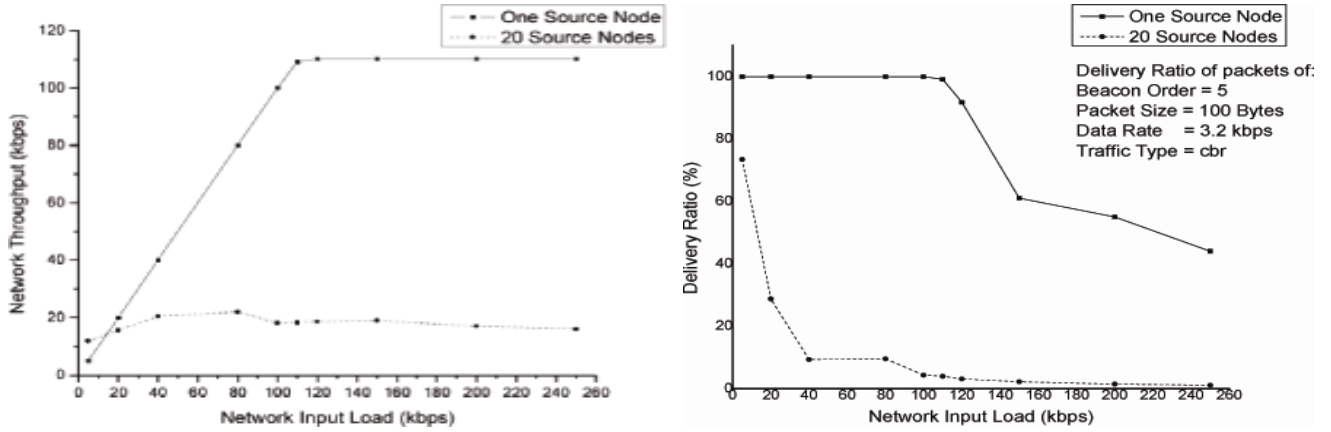
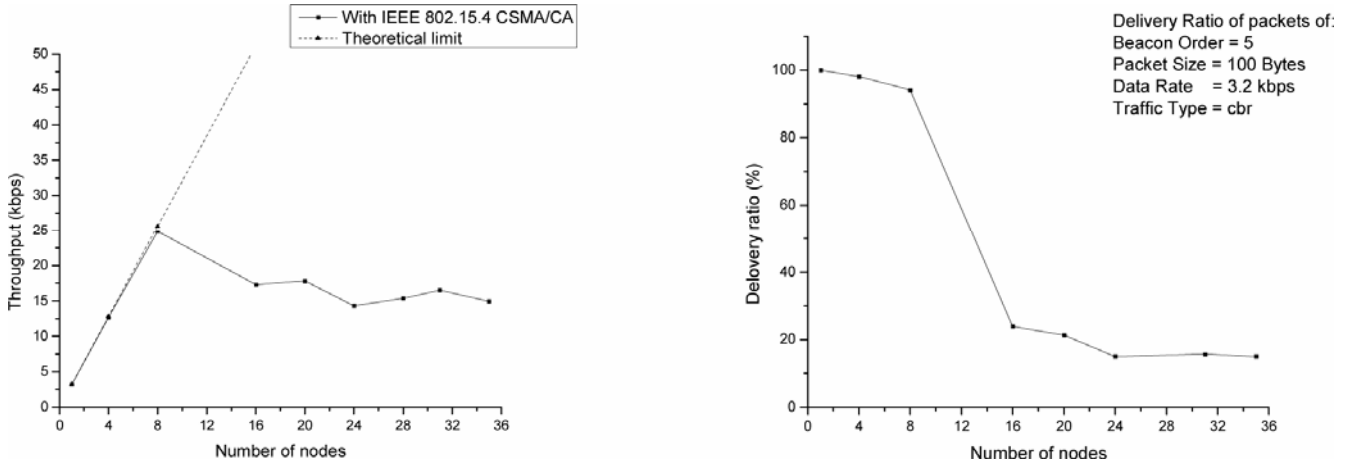


Figure 11 (a) Throughput and (b) packet delivery ratio of typical sensor networks transmitting maximum-sized packets at CBR of 3.2 kbps using IEEE 802.15.4 CSMA/CA



The results were averaged from three random seeds (random number creating an independent simulation scenario when varied) of the same simulation. The upper layers of the simulator generating the CBR traffic add an extra 10 byte header hence limiting the maximum raw data packet size possible from 110 (according to the standard) to 100 bytes. According to the simulator, every node before starting transmission of the CBR packets needs a route to the PAN coordinator to be set up. For our analysis, we set up a single hop route by exchange of six routing packets (a broadcast AODV and ARP, and *RSETUP* AODV and ARP with ACKs). For analysis, we assumed this can affect up to four maximum packet transmission slots in each routing attempt; and this corresponded to the minimal noise in the network. Due to this probability of a node not getting access to the channel due to unsuccessful *RSETUP* (as the network load increases), there is a reduction in this maximum throughput resulting in $N^{\max} = 31$ and $\pi_{th} = 99.2$ kbps.

2.3 Network implementation on WISAN

This scheme was successfully implemented and tested on WISAN, a scalable low-power platform for continuously

monitoring sensor networks (Sazonov et al., 2006; <http://www.intelligent-systems.info/wisan.htm>). The WISAN data acquisition modules were built around an ultra-low-power microcontroller MSP430F1611 from Texas Instruments (www.ti.com). The MAC implementation from Chipcon was modified to run on MSP430 processors. The TDMA scheduler was implemented at the upper layers of the PAN Coordinator (shown in Figure 12(a)) in order to schedule time slots for associating nodes (shown in Figure 12(b)) as well as maintain the scheduling table of the network. The upper layers on the node enable the sending of *scheduling request parameters* along with association requests, and they also maintain the scheduling table for the node. Data transmissions occur in the allotted time slots. For comparison with the simulated results, a network scenario with each node sending a constant stream of data packets (maximal size) was taken. The input load on the network was varied and the output throughput and delivery ratios were measured using a packet sniffer. The noise in the test environment was typical of office settings with multiple 2.4 GHz devices (such as cordless phones) and IEEE 802.11g network operating in the vicinity.

Figure 12 (a) WISAN PAN Coordinator and (b) WISAN Node (online version for colours)



3 Results

Analytical results for the proposed method obtained from the simulations performed as described in Section 2.3 are summarised in Figures 13–22. Experimental results of the method obtained from WISAN are shown in Figure 23.

4 Discussion

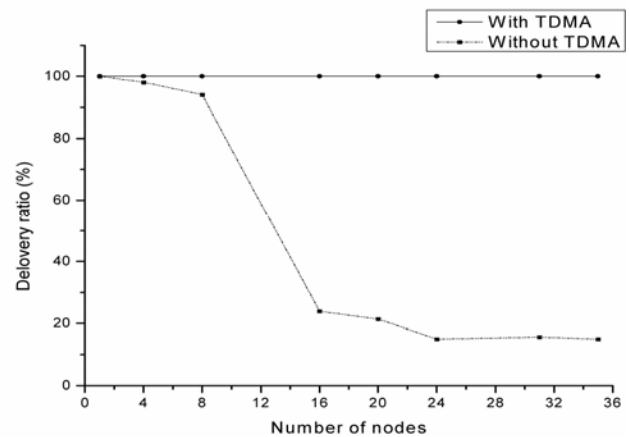
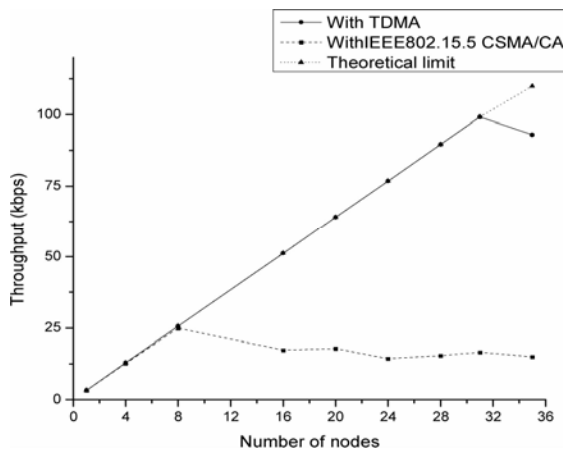
4.1 TDMA vs. standard IEEE 802.15.4 scheduling networks

A comparison of the standard 802.15.4-based networks (Lu et al., 2004; Zheng and Lee, 2004a) without TDMA slot scheduling and networks built with the TDMA time slot scheduling was done using the simulator. Every node is set to transmit data packets of size 100 bytes (maximum raw data packet size) at 3.2 kbps. No random packet error is

induced in the simulation environment. The number of nodes in the network is varied from 1 to 36. Figure 13(a) shows the throughput of the network; that is, the total number of data packets successfully obtained at the receiver for the total time of network simulation. It can be observed that in standard IEEE 802.15.4 networks, the throughput reaches a maximum of about 25 kbps for about 11 nodes and drops down to 22 kbps as the size of the network increases. In contrast to this, the network using the TDMA scheduler can obtain a maximum throughput of up to 99.2 kbps for 31 nodes which is comparable to the maximum theoretical limit established in Section 2.2. Figure 13(b) shows the packet delivery ratio, that is, the ratio of the total number of packets successfully delivered at the receiver to the sum of the number of packets generated and the number of packets retransmitted. It can be observed, as the size of the network increases, the packet delivery ratio remains in the range of 99–100% for TDMA-based networks but drops down to as low as 16% for standard IEEE 802.15.4 networks. Thus, in order to deliver all the packets, the standard IEEE 802.15.4 nodes would need to retransmit each packet up to six times, thus increasing the power consumption and latency of the network by 600%.

The result obtained in Figure 13 shows that standard IEEE 802.15.4 networks suffer greatly since fair channel access is not provided to each node. This reduces the delivery ratio of the packets and also increases the probability of multiple attempts for successful packet transmission which, in turn, increases the energy consumption and latency of the network.

Figure 13 (a) Throughput and (b) packet delivery ratio of the networks with and without using TDMA-based scheduling as a function of increasing number of nodes of the network



4.2 TDMA scheduling networks at different rates

A comparison of the throughput and delivery ratio of packets at different data rates of 3.2 kbps, 1.6 kbps and 800 bps per node, respectively, with their theoretical limits was performed. A network is considered to be saturated when the size of the reserved time slot (optimal) is able to accommodate the retransmissions made and no more additional time slots can be scheduled in the network. Figures 14 and 15 give the throughput and the delivery

ratio for the different sets of data rates. The obtained throughput in Figure 14 remains equivalent to the theoretical limit until the number of nodes in the network gets saturated. As the size of the network becomes large, the probability of transmission of these routing packets in the allocated time slots increases. Considering collisions with the routing packets as described earlier in Section 2.3, more data packets need to contend for the available reserved time slots. Beyond this point, a decrease of throughput from the theoretical limit is seen due to

non-recovery from collisions. Figure 14(a) shows that although the obtained throughput is about 99.2 kbps which is less than the theoretically established limit at 108.8 kbps

(Figure 10), these networks perform much better than the non-TDMA scheduling-based networks that only provided a maximum throughput of about 25 kbps.

Figure 14 Throughput of networks using TDMA scheme for each node transmitting at (a) 3.2 kbps (b) 1.6 kbps and (c) 0.8 kbps for varying number of nodes in the network

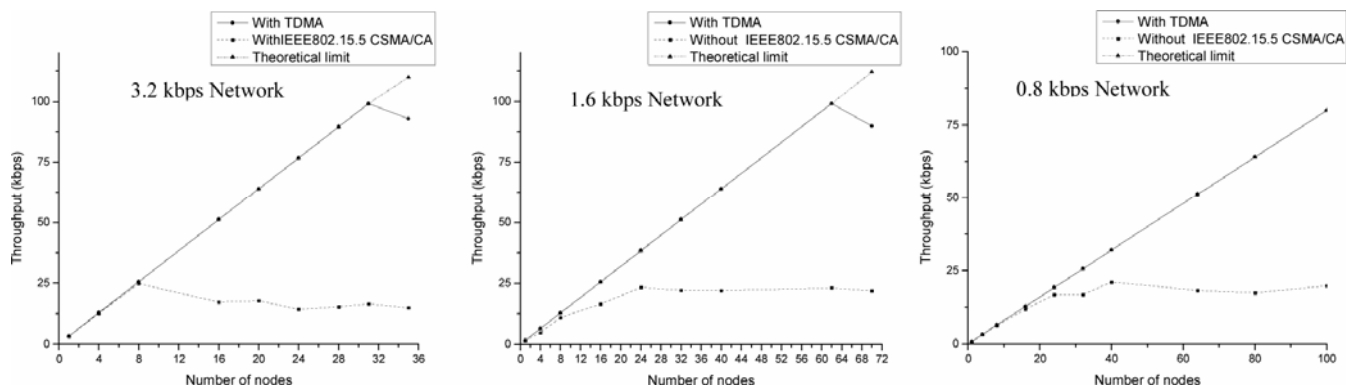
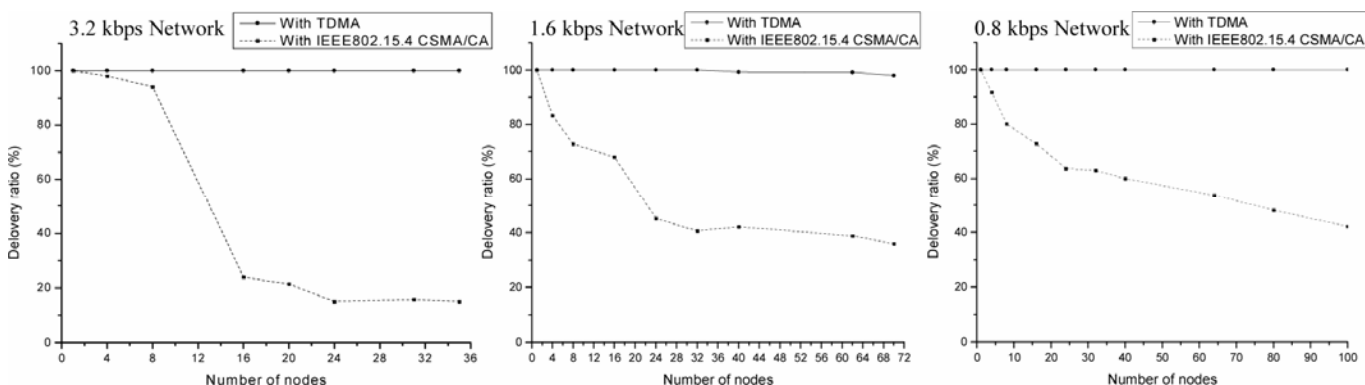


Figure 15 Packet delivery ratio of networks using TDMA scheme for each node transmitting at (a) 3.2 kbps (b) 1.6 kbps and (c) 0.8 kbps for varying number of nodes in the network



4.3 Reserved time slot

As discussed in Section 2.2, the size of the reserved time slot plays an important role in maintaining the reliable delivery of data packets in the network. By performing simulations for the decreasing size of the reserved time slot with the increasing number of nodes using equation (7), the value of the slot for maximum possible throughput for a network with certain parameters is obtained, as shown in Figures 16 and 17. This optimal point gives the maximum number of nodes that may be scheduled for the optimal reserved time slot and provides the most efficient bandwidth utilisation. As Figures 18 and 19 demonstrate, there is an optimal size of the reserved slot for fixed parameters of the network that defines an upper bound on the number of schedulable nodes. Also from equation (9), the probability of collisions with the routing packets is obtained as 0.01% for the optimal network size and the number of nodes that may be successfully scheduled is with probability 31 which is the value obtained using

simulations. Thus, the simulated results match well with the theoretical analysis.

Figure 16 Maximum network throughput (kbps) obtained by varying the size of the reserved time slot represented as a percentage of BI

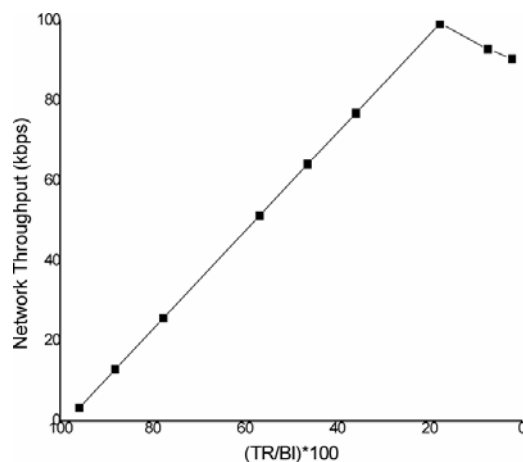


Figure 17 Maximum network throughput (kbps) obtained by varying the number of nodes and keeping the size of the reserved time slot as maximum. Each node transmits at a constant packet of 100 bytes at a fixed rate of 3.2 kbps

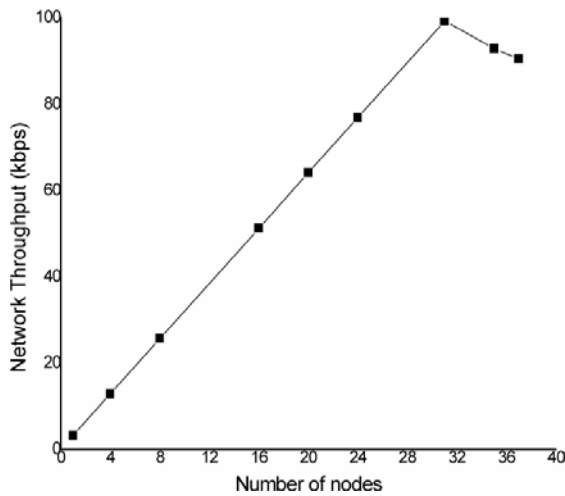


Figure 18 Ratio of the required throughput for lightly loaded, medium loaded and heavily loaded networks with variation in the percent of nodes using the TDMA scheduling

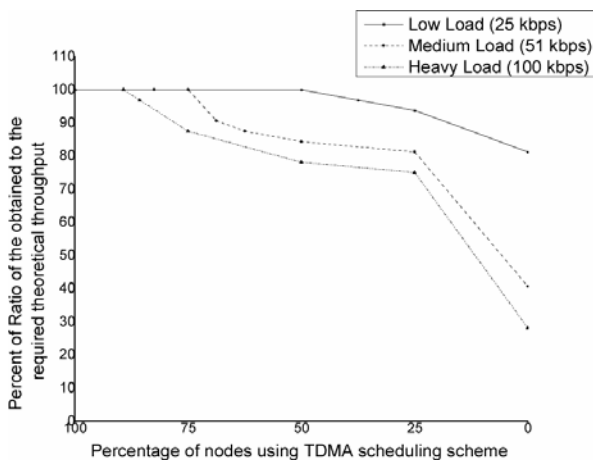
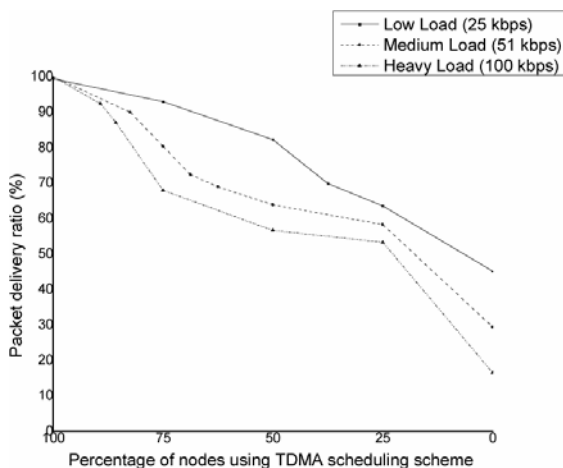


Figure 19 Packet delivery ratio for lightly loaded, medium loaded and heavily loaded networks with variation in the percent of nodes using the TDMA scheduling



4.4 Mixed networks

Any TDMA-based network consisting of a fixed number of nodes has an optimal reserved time slot determined by equation (7). If TR equals the maximum available bandwidth in a BI after all the TDMA-based nodes are scheduled, then this would be the maximum time period available for retransmission of any lost packet. By performing simulation on lightly loaded (25 kbps), medium loaded (51 kbps) and heavily loaded (100 kbps) networks for different percentages of the nodes using TDMA scheduling a comparison is obtained in Figures 18 and 19. It shows that while a lightly loaded network may be used with up to 50% of the nodes using pure CSMA/CA, the medium loaded networks cannot accommodate more than 25% CSMA/CA nodes; and the heavily loaded networks cannot accommodate more than 10% CSMA/CA nodes without loss of performance.

4.5 Effect of noise

The effect of noise on the scheduling scheme was analysed by introducing uniform packet error rate in a network where every node was set to transmit data packets of size 100 bytes (maximum raw data packet size) at a data rate of 3.2 kbps. Channel noise is considered according to equation (9), and varying P_{noise} gives an estimate of the number of nodes that can be successfully accommodated in the network without any loss. By performing simulations for different noise levels, the maximum possible throughput with the reserved slot in each case kept to maximum is obtained as shown in Figures 20 and 21. The optimal point giving the maximum number of nodes that may be accommodated for a given percentage of channel noise decreased from 31 nodes for 1% noise to 12 nodes for 10% noise. The packet delivery ratio also significantly drops, as shown in Figure 22, suggesting that the number of retransmissions increases as noise increases.

Figure 20 Maximum network throughput (kbps) obtained by varying the number of nodes and keeping the size of the reserved time slot as maximum for different noise levels. Each node transmits at a constant packet of 100 bytes at a fixed rate of 3.2 kbps

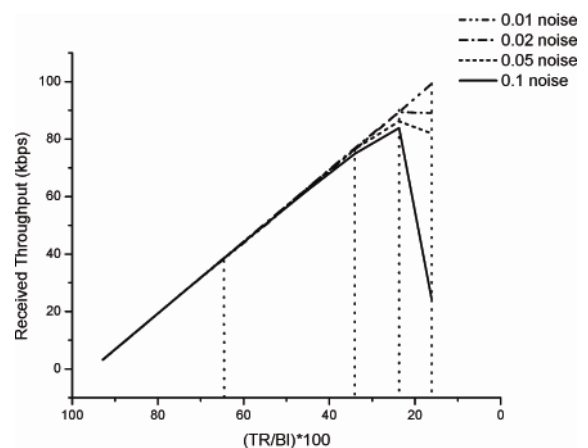


Figure 21 Maximum network throughput (kbps) obtained by varying the size of the reserved time slot represented as a percentage of BI for different noise levels

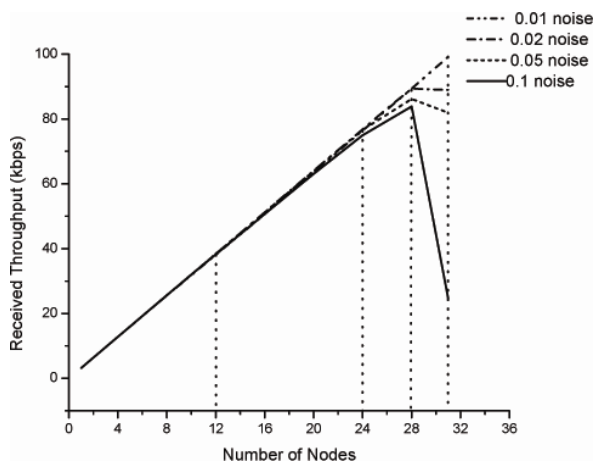
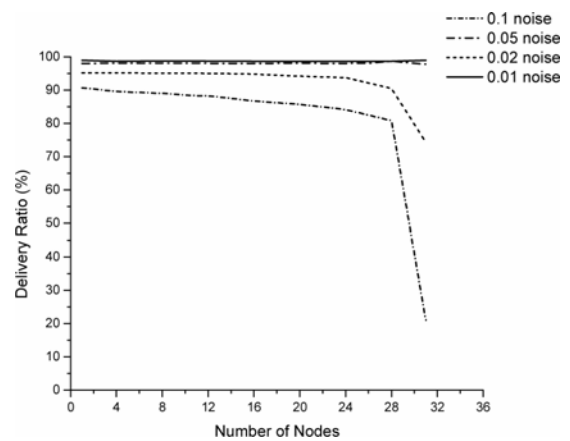


Figure 22 Packet delivery ratio obtained by varying the size of the reserved time slot represented as a percentage of BI for different noise levels

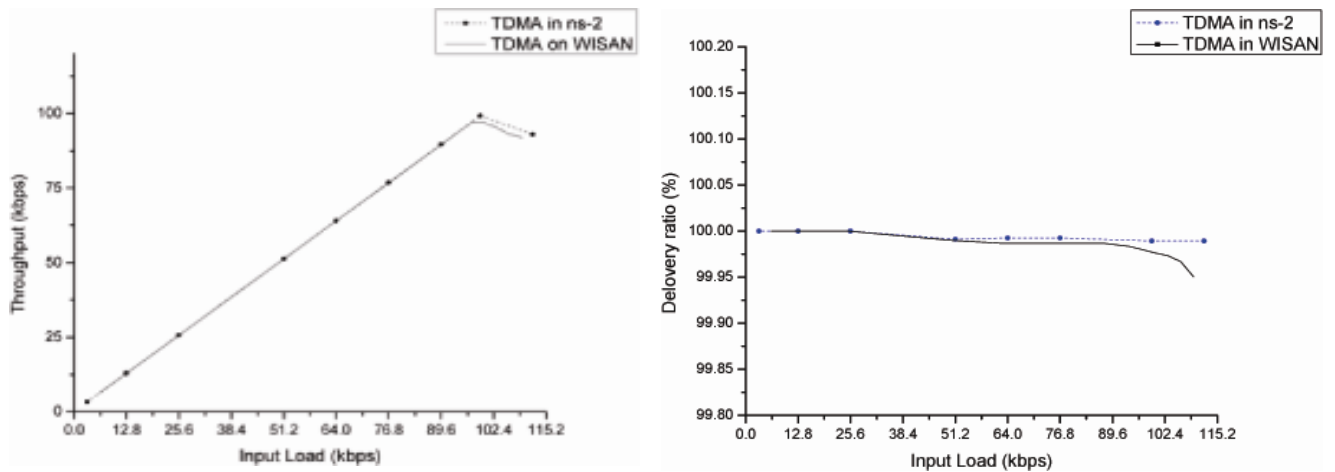


4.6 Scheduler implementation on WISAN

Figure 23 shows that using the TDMA scheduler, the maximum throughput of the network is obtained at 30 nodes (96 kbps) with the packet delivery ratio being nearly 100% along with minimal consumption of the node energy and very low latency in the packet delivery. This network provides 100% reliable data under normal operating conditions, which is highly critical for monitoring applications.

The results obtained from simulations and on WISAN platform show that the protocol enables efficient and reliable data delivery from multiple sensors. The proposed reservation-based protocol could be implemented for an IEEE 802.15.4 monitoring network by choosing the appropriate parameters based on the network requirements. Protocol parameters can be adjusted so that the network can perform reliably in a specific environment. Conducted theoretical analysis is used to establish optimal parameter values. Validity of the analysis has been supported by the simulation results.

Figure 23 (a) Throughput and (b) packet delivery ratio of the network implemented on WISAN compared to that simulated using Network Simulator ns-2



5 Conclusions

In this paper, a reservation-based protocol was developed for IEEE 802.15.4-based low power PANs. A typical continuously monitoring sensor network is considered to have a hierarchical architecture with a number of nodes transmitting a constant amount of information (data) at a constant rate. This scheme was developed to ensure the reliability of such networks, increase the bandwidth usage and reduce power consumption and latency of these networks. Two types of scheduling algorithms were

designed, namely, soft scheduling and hard scheduling. A linked list based data structure is used to store the scheduling table.

This network model was simulated using NS-2 for different scenarios and the results obtained show a remarkable improvement in the networks using the proposed method. The TDMA-based network without noise is seen to handle a load of up to 99.2 kbps with multiple nodes, an improvement of almost 400% as compared to 25 kbps of the networks without the scheduler. This is comparable to the theoretically achievable maximum limit of 110.67 kbps.

Also, the energy consumption and latency due to retransmissions are reduced by a factor of six for large networks. As the amount of channel noise increases, the size of the reserved time slot needs to be varied to get maximum nodes to transmit without any data being dropped. This variation of reserved time slot size with noise has also been analysed. The network model has also been implemented on WISAN hardware platform and tested to produce remarkable results comparable to the simulated results. Thus, the method is seen to work very well for monitoring networks with a large number of nodes requiring reliable delivery of low-latency data streams.

Acknowledgements

The authors acknowledge the New York State Energy Research and Development Authority (NYSERDA) for their financial support.

References

- Bianchi, G. (1998) 'IEEE 802.11 – Saturation Throughput Analysis', *IEEE Communications Letters*, Vol. 2, No. 12, pp.318–320.
- Bianchi, G. (2000) 'Performance analysis of the IEEE 802.11 distributed coordination function' *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 3, pp.535–547.
- Bianchi, G., Fratta, L. and Olivefi, M. (1996) 'Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs', *Proc. PIMRC*, Taipei, Taiwan, pp.392–396.
- De Simone, A. and Nanda, S. (1996) 'Wireless data: systems, standards, services', *Wireless Networks*, Vol. 1, No. 3, pp.241–254.
- Karl, H. and Willig, A. (2003) 'A short survey of wireless sensor networks', *Telecommunication Networks group, Technical Report*, Technische Universitat Berlin, Hasso-Plattner Institute, Potsdam.
- Lu, G., Krishnamachari, B. and Raghavendra, C.S. (2004) 'Performance evaluation of the IEEE 802.15.4 MAC for low-rate low power wireless networks', *Proceedings of IEEE International Performance Computing and Communication Conference (IPCCC'04)*, Phoenix, AZ, pp.701–706.
- Miller, M.J. and Vaidya, N.H. (2004) 'On-demand TDMA scheduling for energy conservation in sensor networks', *Technical Report*, University of Illinois at Urbana Champaign.
- Misic, J., Shafi, S. and Misic, V.B. (2006a) 'Maintaining reliability through activity management in 802.15.4 sensor clusters', *IEEE Transaction on Vehicular Technology*, Vol. 55, No.3, pp.779–788.
- Misic, J., Shafi, S. and Misic, V.B. (2006b) 'Performance of beacon enabled IEEE 802.15.4 cluster with downlink and uplink traffic', *IEEE Transactions on Parallel and Distributed Systems*, Vol. 17, No. 4, pp.361–376.
- P802.11 (1995) IEEE Draft Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, D2.0, July.
- P802.15.4/D18 (2003) IEEE Draft Standard: Low Rate Wireless Personal Area Networks, February.
- Raghavendra, C.S. and Singh, S. (1998) 'PAMAS – Power Aware Multi-Access protocol with Signalling for Ad Hoc Networks' *ACM Computer Communications Review*, Vol. 28, No. 3, pp.5–26.
- Rajendran, V., Obraczka, K. and Garcia-Luna-Aceves, J.J. (2003) 'Energy-efficient, collision-free medium access control for wireless sensor networks', *ACM SenSys 2003*, November, pp.181–192.
- Sazonov, E., Janoyan, K., Jha, R., Nelson, R., Krishnamurthy, V., Gao, Z. and Fuchs, M. (2004a) 'Sensor network application framework for autonomous structural health monitoring of bridges', *Proceedings of Structural Materials Technology (SMT): NDE/NDT for Highways and Bridges 2004*, Buffalo, NY.
- Sazonov, E., Janoyan, K.D. and Jha, R. (2004b) Wireless Intelligent Sensor Network for Autonomous Structural Health Monitoring, Smart Structures/NDE 2004, San Diego, CA.
- Sazonov, E., Jha, R., Janoyan, K., Krishnamurthy, V., Fuchs, M. and Cross, K. (2006) 'Wireless intelligent sensor and actuator network (WISAN): a scalable ultra-low-power platform for structural health monitoring', *SPIE Proc. 6177*, March 2006.
- Sichitiu, M.L. (2004) 'Cross-layer scheduling for power efficiency in wireless sensor networks', *IEEE Infocom 2004*, March.
- Tantra, J.W., Foh, C.H., Bianchi, G. and Tinnirello, I. (2004) 'Performance analysis of the out-of-band signaling scheme for high speed wireless LANs', *IEEE GLOBECOM '04*, Vol. 5, No. 29, pp.3002–3006.
- van Dam, T. and Langendoen, K. (2003) 'An adaptive energy-efficient MAC protocol for wireless sensor networks', *ACM SenSys 2003*, November, pp.171–180.
- van Hoesel, L.F.W., Chatterjea, S. and Havinga, P.J.M. (2003, November) 'An energy efficient medium access protocol for wireless sensor networks', *ProRisc, The Netherlands*.
- Varadhan, K. (2000, August) 'The Ns Manual', *The VINT Project*.
- Ye, W., Heidemann, J. and Estrin, D. (2002) 'An energy-efficient MAC protocol for wireless sensor networks', *IEEE Infocom 2002*, June, Vol. 3, pp.1567–1576.
- Zheng, J. and Lee, M.J. (2004a) 'A Comprehensive Performance Study of IEEE 802.15.4', *IEEE Press Book*.
- Zheng, J. and Lee, M.J. (2004b) 'Will IEEE 802.15.4 make ubiquitous networking a reality?: a discussion on a potential low power, low bit rate standard', *IEEE Communication Magazine*, Vol. 42, No. 6, pp.140–146.

Websites

Texas Instruments, Available online at: www.ti.com.

Time Division Multiple Access definition, Available online at: <http://en.wikipedia.org/wiki/TDMA>.

Wireless Intelligent Sensor and Actuator Network (WISAN), Available online at: <http://www.intelligent-systems.info/wisan.htm>.

List of abbreviations

π_{th}	Theoretical throughput of the network	T_{avail}	Available network time period for data transfer
n_B	Number of beacon intervals forming a network period	T_s^{max}	Slot size required for transmission of a maximum-sized data packet
X	Number of bytes per node (node bytes)	T_S	Slot size per packet
N	Number of nodes in the network	B_{node}	Bandwidth utilised per node in the network
N^{max}	Maximum number of nodes in the network	t_{node}	Rate of data (single packet) transfer for each node (node period)
S_{node}	Number of slots per node	Z	Percentage of nodes in the network using TDMA scheduler
S_{BI}	Number of slots per beacon interval	P_{tdma}	probability of collision of a CSMA/CA node packet with TDMA node packet
S_{node}^{BI}	Number of slots per beacon interval per node	P_{csma}	probability of collision of a CSMA/CA node packet with another CSMA/CA node packet
S_{total}	Total number of slots in the network period	R_{tdma}	Number of packets to be retransmitted by TDMA nodes
T	Total scheduling period (in seconds)	R_{csma}	Number of packets to be retransmitted by TDMA nodes
A	Overhead in each slot (in bytes)	P_{noise}	Percentage of channel noise in the network
Γ	Constant data rate per node (kbps)		
ψ_{total}	Total number of packets transmitted		
ψ_{node}	Packets per node per second		
$\psi_{dropped}$	Number of packets dropped in a single network period		