LEFT: A Latency and Energy Efficient Flexible TDMA Protocol for Wireless Sensor Networks

Sachin Gajjar
Institute of Technology, Nirma University, Ahmedabad-382481, India
E-mail: sachin.gajjar@nirmauni.ac.in

Nilav Choksi
EInfochips India Ltd., C. G. Road, Ahmedabad-380006, India
E-mail: nilav.choksi@eitra.org

Mohanchur Sarkar
Space Application Centre (SAC), Indian Space Research Organization (ISRO), Ambawadi, Ahmedabad - 380015, India
E-mail: msarkar@sac.isro.gov.in

Kankar Dasgupta
Indian Institute of Space Science and Technology, Valiamala, Thiruvananthapuram - 695547, India
E-mail: ksd@iist.ac.in

Abstract—This paper proposes latency and energy efficient flexible TDMA (LEFT), a medium access control (MAC) combined with routing protocol for data gathering from number of source nodes to a master station (MS) in a wireless sensor network (WSN). TDMA provides fairness, collision-free communication and reduces idle listening, which saves network energy. Data latency is reduced by allocating same transmission slots to nodes falling out of interference range of each other. Unlike a conventional TDMA, LEFT provides flexibility through *slot seizing*, wherein a non-holder of a slot can use slot when holder does not have data to send. This increases channel utilization and adaption to dynamic traffic patterns of WSN applications. Further, a node on a multi-hop path towards MS decides to participate in routing based on (i) its location with respect to MS, to forward data in correct direction, (ii) its current status of residual energy, to uniformly distribute energy across network, (iii) its transit traffic load, to prevent local congestion, (iv) its communication link quality, to guarantee reliable data delivery. This decision requires simple comparisons against thresholds, and thus is very simple to implement on energy, storage and computationally constrained nodes. LEFT also encompasses techniques to cater to link and node breakdowns. Experimental analysis of LEFT; Advertisement-based TDMA; Data gathering MAC; Energy Efficient Fast Forwarding and Cross layer MAC protocols using TI's EZ430-RF2500T nodes shows that LEFT is 65% more energy efficient compared to Cross layer MAC. Data latency of LEFT is 27 % less, delivery ratio is 17 % more and goodput is 11 % more compared to Cross layer MAC.

Index Terms—Wireless Sensor Networks protocol design, medium access control, routing, experimentation, performance evaluation, comparative analysis.

I. INTRODUCTION

Wireless sensor networks (WSN) are used in a wide range of applications like precision agriculture, target monitoring or tracking, military operations, and planetary explorations [1]. Typically in most of these applications, nodes coordinate locally to collect data, process and deliver it to a master station (MS). Severe limitations in energy, storage and computational capabilities of nodes demands resource conscious protocols for WSN design [2]. Researchers have focused on design of energy efficient medium access control (MAC) protocols for radio transmission and reception since the radio transceiver consumes maximum energy compared to other hardware subsystems in the node [3]. MAC protocols can conserve energy by turning off the transceiver when not in use, but at the same time must meet data latency, fairness and channel utilization requirements of the application. They must also reduce transceiver energy consumption by preventing or limiting collisions, overhearing, idle listening, over radiating and control overhead [4]. Apart from these, MAC protocols should be: provide scalability; be fault tolerant and adapt to dynamic traffic patterns of applications. To this end, this paper proposes LEFT which achieves these features as discussed below:

- Energy efficiency by: (i) using TDMA based MAC that combines routing information during time slot allocation for uninterrupted data forwarding (ii) techniques for node-by-node congestion control
- Fairness by using TDMA with slot allocation by MS
- Reliable data delivery by: (i) using TDMA for media

access to minimize transceiver interference and assure a communication path (ii) node deciding to participate in routing only if it is located towards MS, has sufficient energy, less traffic load and good communication link quality

- Robustness by local repair techniques to recover from node and link failures
- Scalability by: (i) multi-hop network architecture (ii) distributing transit traffic along multi-hop path towards MS (iii) spatial reuse (allocating same slot to nodes outside each other's interference range)
- Reduced data latency by: (i) strategic back off and backward scheduling to ensure that nodes far from MS are given an early slot position in data transfer frame (ii) spatial reuse
- LEFT adapts to dynamic traffic pattern by providing flexibility through *slot seizing* in which non-holder of a slot can use the slot when holder does not have data to send. Thus node can transmit data as soon as the channel is available to achieve high channel utilization, lower data latency and adapt to application traffic pattern.

The remainder of paper is organized as follows: Section 2 discusses related work. LEFT system model and operation is introduced in Section 3. Section 4 provides experimentation based comparative analysis of LEFT with four protocols of similar complexity. Finally paper is concluded in Section 5.

II. RELATED WORK

Several MAC protocols have been proposed for WSN in the past few years [5]-[6]. They can be coarsely classified into: contention-centered, scheduled-centered and with hybrid-approach. In contention-centered MAC protocols node that plans to transmit listens to the media first and transmits if no other node is transmitting. The promising characteristics of these protocols are: (i) they allocate media on-demand, and hence can easily adapt to traffic and topology changes; (ii) are simple to implement as they do not require coordinator to decide media access mechanism, strict clock synchronization and knowledge of global topology (iii) do not exchange information with their neighbors prior to sending their messages and thus ensure high channel utilization, throughput and low data latency during low contention (iv) are robust as change in topology does not require any additional operations [7]-[8]. However, lack of communication among nodes results in collision, overhearing, idle listening and above all it is difficult to attain fairness. Scheduled-centered MAC protocols try to coordinate network communication by planning a schedule with details like, when each node should transmit, receive, listen or sleep. In a schedule-centered technique like TDMA nodes own a time slot for transmitting and receiving a message. Nodes sleep for rest of the time to save their energy. TDMA-based protocols: (i) guarantee collision-free communication, (ii) reduce idle listening and save energy (iii) guarantee

fairness. However, these protocols: (i) require determining collision-free slots for nodes (ii) synchronization of nodes' clock (iii) result in lower channel utilization especially during low traffic. An efficient TDMA based protocol should: (i) lessen the effect of increase in data latency and decrease in throughput during high traffic loads, (ii) be flexible to adapt to traffic fluctuations and (iii) update its schedule whenever there is a change in network topology. MAC protocols using a hybrid-approach combine features of contention-centered and scheduled-centered protocols. There has been an increased attention towards cross-layer protocols for WSN that use source of information at a layer to benefit protocol at other layer [9]-[10]. For example, in a multi-hop network, routing information (at network layer) can be combined during time slot allocation (at MAC layer) such that transmission slot of transmitter and receiver slot of relay node overlap with each other. Some of the cross-layer protocols and architectures have been discussed in the authors' previous works [11]-[12]. LEFT is a cross-layer protocol whose MAC layer uses routing information for transmission slot allocation. Some of the well referred MAC protocols for WSNs are discussed next. Traffic adaptive MAC (TRAMA) protocol [13] organizes time into frames and each frame into slots designated for CSMA and TDMA based media access. CSMA slots are used for sending control messages and TDMA for transmitting, receiving or forwarding data. Network topology and traffic information is used by hash function in Adaptive Election Algorithm to determine collision free data transmission slots. This results in high channel utilization and energy savings. However, during neighborhood traffic information exchange all the nodes are in active mode which increases their energy consumption. Flow aware MAC (FLAMA) protocol [14] reduces this idle listening by using data gathering tree information to determine the next-hop relay node. This one-hop traffic information is used for planning transmit, receive, and sleep schedules of nodes. In contrast to TRAMA which requires nodes to broadcast traffic information regularly, FLAMA requires it only during random access period of a frame. Sensor MAC (S-MAC) [15] divides network into several virtual clusters consisting of group of nodes following same schedule. Clusters use common periodic listen and sleep schedules and thus conserve energy and prolong network lifetime. However, several clusters and their schedule transmissions result into transmission errors, collisions, large end-to-end delays, simultaneous self schedule selection. Further, nodes belonging to more than one cluster follow multiple schedules which increases their listen periods and hence their energy consumption. Finally, fix duty cycle of nodes cannot adopt to traffic fluctuation. Patten-MAC (P-MAC) allows nodes to dynamically adjust their sleep and wakeup cycles based on their current traffic loads [16]. For this, nodes exchange sleep/wake up schedules within their neighborhood. In the pattern repeat period of a frame, nodes follow the announced sleep/wake schedule. Nodes wake up during their own and neighbor's transmission

slots. Request To Send (RTS)/ Clear To Send (CTS) /DATA/ Acknowledgement (ACK) is used for reliable data transmission. P-MAC is simple to implement and provides good adaptability to changing traffic loads. However, if there are frequent traffic changes, pattern exchange happens frequently resulting in significant increase in energy consumption due to computing and communication power required for updating and transmitting pattern information. Funneling-MAC (F-MAC) is a hybrid TDMA and CSMA/CA MAC protocol for addressing the funneling effect in multi-hop network with many to one traffic pattern [17]. Funneling effect results in a significant increase in transit traffic near MS (funneling region) and thus packet congestion, collision, loss, delay and energy consumption occur as events move closer towards the MS. Pure CSMA/CA, is implemented in the funneling region and also network wide. In addition, local TDMA scheduling is also used in the funneling region to provide additional scheduling opportunities. TDMA schedule preparation and determining the depth of funneling region is done by MS to offload the work of resources-limited nodes. Localized TDMA solves the scalability problem of TDMA. However, the protocol does not concentrate on high channel utilization or adapting to dynamic traffic pattern. Data gathering MAC (D-MAC) protocol [18] is for data gathering from several nodes to a MS along a multi-hop path. An interval in node's schedule is divided into: receiving (to receive packet and send ACK), sending (send packet to relay node and receive ACK), and sleeping periods (turn off radio to save energy). Based on depth d in the data gathering tree, a node sets its wake-up schedule $d\mu$ ahead from MS schedule where μ is length of receiving and sending periods. It then periodically goes into receiving, sending, and sleeping states like a multi-hop chain. Thus when there is no collision, data is forwarded sequentially along a multi-hop path to the MS without sleep latency. D-MAC combines media access with data forwarding which decreases data latency. Further, since nodes are awake only when it has to forward data, energy consumption of node decreases. However, D-MAC does not consider node fairness and interference between nodes in same depth is to be handled carefully. Finally, nodes outside routing paths waste energy in unnecessary idle listening. In Crankshaft protocol [19] each frame is divided into slots for unicast communication followed by slots for broadcast communication. Each node is allocated one unicast slot in a frame corresponding to its MAC address. Nodes scheduled to receive data wake up during their unicast slot, while rest of the nodes remain sleeping. Senders that want to send data to scheduled node contend for transmission media and winner starts transmission. Slot of intended receivers is known from their MAC address. Nodes wake up and listen for incoming messages during the broadcast slots. Crankshaft increases network lifetime by reducing idle-listening and overhearing as senders are only awake during contention period of a unicast slot. However, slot structures of Crankshaft are inflexible limiting its usage to applications requiring periodic

measurements and increased lifetime. Demand-wakeup MAC (DW-MAC) [20] proposes media access scheduling that ensures data transmissions do not collide at their intended receivers. For this a mapping function of data period (T_{data}) and following sleep period (T_{sleep}) is defined as,

$$T_i^S = T_i^D \times \frac{T_{sleep}}{T_{data}} \qquad (1)$$

where T_i^D is time difference between time instance t_i in a data period and beginning of that data period, T_i^S is time difference between start of sleep period and corresponding mapped time instance during sleep period. With this mapping when the scheduling frame is transmitted it determines offset of DATA packet exchange. A reservation frame SCH containing routing information like source address and packet sequence number is transmitted to avoid multiple packet broadcasts. The mapping function reduces collision but because of it DATA packet exchange during sleep periods are interlaced by gaps with a size proportional to T_{sleep}/T_{data} ratio which increases the latency. Further, data and sleep period durations are fixed and collisions between data and ACKs is not taken care of which increases energy consumption of the network. Adaptive Scheduling MAC (AS-MAC) [21] improves DW-MAC by introducing a flexible Reserved Active Time (RAT) whose length can be changed in each operational cycle to adapt to variable traffic load. Further, an enhanced mapping function is proposed to prevent the collisions between data and ACKs in most of the scenarios. However, both DW-MAC and AS-MAC require a specific sequence of transmissions of SCH frames i.e, a node is allowed to start its negotiation after its predecessors and before its successor, to ensure fast packet delivery. Optimized MAC (O-MAC) proposed in [22] decreases energy consumption due to idle listening and overhearing by adjusting node's duty cycle on the basis of network traffic. The network traffic is calculated based on node's message queue. However, O-MAC is not proposed as an independent protocol and is shown to improve energy and latency performance of S-MAC for varying network traffic. Zebra MAC (Z-MAC) is a hybrid protocol that behaves like CSMA under low contention to achieve high channel utilization and low latency [23]. Under high contention, it behaves like TDMA to achieve high channel utilization and also reduce collisions among two-hop neighbors. The time slot assignment is performed by Distributed Randomized TDMA Scheduling (DRAND), a distributed and scalable scheduling algorithm [24]. Each node reuses its assigned slot periodically in every predetermined period, called frame. A node may transmit during any timeslot after performing carrier sensing. Size of initial contention window is adjusted such that slot owners are always given earlier chances to transmit than non-owners. When owners do not have data to transmit, non-owners can use the slot. Thus, Z-MAC saves energy

by dynamically adjusting between CSMA and TDMA depending on the network contention. It also becomes more robust to time synchronization faults, slot assignment failures, varying topology, channel conditions and traffic compared to pure TDMA. However, slot assignment algorithm run at each node incurs high energy cost and in case of large changes in topology, it is required to re-run the costly initial network setup. Finally, inclusion of a small contention window in every slot leads to increase in latency and energy consumption during high contention. Energy Efficient and Fast Forwarding (EEFF) [25] is an energy efficient MAC protocol coupled with dynamic minimum hop routing. Nodes are not synchronized and perform their active-sleep schedule individually. The data is send without any preambles to save energy. A node that wants to send data, does carrier sensing and if it finds media to be idle, it broadcasts RTS packet containing receiver's address. Nodes receiving RTS decide whether to accept the request and become relay nodes based on their active or sleep condition and hop latency to MS. Nodes with less number of hops from MS wake up first and become candidate receivers which compete to send CTS packet. The winner sends CTS and keeps awake to receive data. Nodes broadcast their local routing information periodically to build minimum hop routing table information for data transmission to MS. EEFF achieves low latency and high energy savings as its dynamic routing approach selects relay nodes during but not before the transmission and it sends data without any preambles. On the downside, latency estimation by relay nodes and routing protocol does not consider varying traffic which might result into improper relay nodes. Work in [26] proposes Low energy and Latency MAC and Routing (LEMR) cross layer protocol that combines scheduling with channel polling. The channel polling intervals are shifted according to distance of node from MS. Thus, intermediate nodes do not have to wait for a complete channel polling cycle to forward data towards MS decreasing data latency. Each node has a table which has: (i) its own hop distance to MS and all its neighbors (ii) power level of signals received from each neighbor. This information is used for selecting the relay node during multi-hop routing. Speedy and Energy Efficient data delivery MAC (SPEED-MAC) [27] uses a SIGNAL packet during signaling period to reserve a transmission slot for upcoming DATA period, and to detect collision from multiple event-senders. The proposed signaling wake up works well with single-source event. However, with multisource event it will cause several contentions and collisions near the sink increasing the latency and energy consumption. Advertisement based time-division multiple access (ATMA) [28] is a distributed TDMA-based MAC protocol in which time is divided into frames consisting of SYNC (for node's clock synchronization), advertisement (ADV, for slot reservation through contention) and data (DAT for contention-free data exchange) periods. ATMA prevents energy waste through advertisements and reservations for slots using the traffic information. To reduce delay and energy consumption

incurred by ADV period optimal value of ADV period is to be decided using traffic and node density. This gets complicated when topology changes. Medium Reservation Preamble based MAC (MRPM) Protocol [29] addresses long idle listening problem of S-MAC by combining contention, SYNC and DATA periods into a contention period followed by a short listen period. Transmitters contend at the beginning of each contention period. During listening period, the winner sends a $SYNC_{RTS}$ packet (combination of SYNC and RTS), to which the recipient node responses with CTS. Neighboring nodes along the path towards MS overhear CTS and schedule themselves for further data reception after Advanced Adaptive Listening (AAL) duration given by,

$$ALL_{duration}=2(t_{ack}+t_{data})+t_{contention-period}+t_{listen-period}, \quad (2)$$

where t_{ack} and t_{data} are time taken to transmit corresponding packets and $t_{contention-period}$ and $t_{listen-period}$ are contention and listen periods. Thus, idle listening is significantly reduced compared to S-MAC. However overhearing nodes must know beforehand whether they are intended forwarders or not. Cross-Layer MAC protocol (CL-MAC) for WSNs [30] is designed to handle multi-packet, multi-hop and multi-flow traffic patterns while adapting to varying traffic loads. For this, flow setup packets (FSP) are used that serve as RTS to destination node and CTS to source node. FSP utilizes routing layer buffer and all flow setup requests from neighbors to transmit multiple data packets over multiple multi-hop flows. Thus, CL-MAC's scheduling decisions considers recent network status and optimizes its scheduling mechanism accordingly. However, the protocol is best suited for heterogeneous WSNs where a single node incorporates several sensors or different types of nodes monitoring different phenomenon. Several energy efficient routing protocols for WSN are discussed in the literature [31]-[36].

LEFT differs from the above mentioned protocols as it provides various needs of WSN like fairness, reliable data delivery, local repair, scalability, reduced latency and adaptation to dynamic traffic patterns along with energy efficient routing.

III. LEFT DETAILS

A. System Model

• **Nodes:** Nodes in the network are: (i) equipped with transceiver to transmit/receive data within one-hop neighborhood (10 m to 80 m) (ii) can vary transmit power depending on distance from receiver (iii) can run simple protocols (iv) have unique hard-wired identification (ID) (v) have priorities associated with them. MS is data gathering point and has good energy, storage and processing capabilities. With high gain antenna it can broadcast to entire network. Nodes that provide all the above features are available in the market [37]. Each

node periodically reports its own measurement to MS through multi-hop transmission.

- **Network:** Number of homogenous stationary nodes are randomly distributed within an M x N sensing field with MS at corner of the field. Nodes are organized into layers as described next. The first layer is a circle with center as MS and radius R_{max} m (maximum transmission range of node). In general, N^{th} layer is a circular ring with center as location of MS, outer radius of $R_{max}xN$ m and inner radius of $R_{max}x(N-1)$ m. The layering is stopped when $R_{max}xN \geq M$ m which ensures that entire sensing field is divided into layers. Fig. 1 shows network layering, route setup and data transfer in LEFT. Data aggregation and compression is assumed to be done by the upper layers and not in the scope of LEFT protocol design.
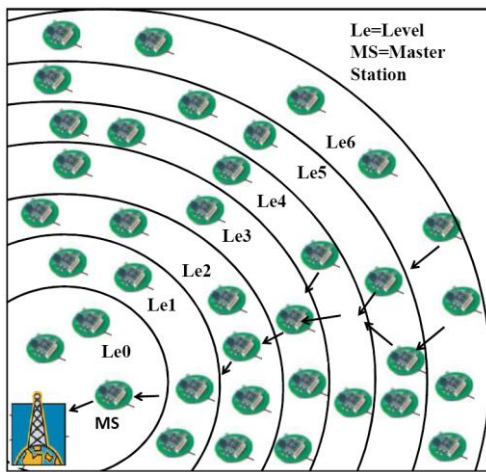
Fig. 1. Network layering, route setup and data transfer in LEFT
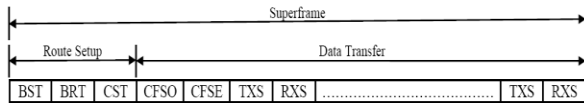
Fig. 2. Time line showing steady-state phase of LEFT

B. LEFT Operation

The operation of LEFT consists of network layering and steady state phase.

- **Network layering:** Nodes learn their layers through a sequence of steps described next.

Step 1. MS sends a SETUP_MESSG (containing its ID, location information, its transmit power details, timing information for nodes to synchronize their clocks, data (if any)) to reach R_{max} m radius from itself.

Step 2. Each node uses received signal power to find its distance from MS. For this, it uses two-ray ground radio propagation model [8]. Thus, distance between MS and node i is given by,

$$distance(i) = \sqrt[4]{\frac{P_{MSt}G_{MSt}G_{ir}h_{MSt}^2h_{ir}^2}{P_{ir}L}} \qquad (3)$$

where P_{MSt} and P_{ir} is power, G_{MSt} and G_{ir} is gain, h_{MSt} and h_{ir} is height above ground for MS transmitting antenna and node i receiving antenna respectively. L is path loss. The nodes then find their layer based on their distances from MS. At the end of first round of SETUP_MESSG, nodes in layer 1 know their layer. Steps 1 and 2 are then repeated with MS increasing its signal strength to reach consecutive layers in each iteration. At the end of network layering phase, each node knows its layer.

- **Steady state phase:** Time in the steady-state phase is divided into super-frames as shown in Fig. 2. Each super-frame is divided into two sub-frames: route setup and data transfer. Route setup sub-frame consist of following slots: beacon slot (BST), broadcasting slot (BRT) and carrier sensing slot (CST). The data transfer frame consist of: compound function slot for odd numbered layers (CFSO), compound function slots for even numbered layers (CFSE), transmitting slot (TXS) and receiving slot (RXS). Minimum number of slots in a data transfer frame is more than number of nodes in network so that all the nodes including MS can have at least one slot per frame for transmission.

- **Route setup:** The multi-hop route from nodes to MS is set up through a sequence of steps described below.

Step 1. MS boots up in scheduled mode. Rest of the nodes boot up in scanning mode.

Step 2. In BST, MS broadcasts BEACON_MESSG (containing beacon interval, super-frame duration, sub-frame duration and slot duration) for nodes to synchronize their clocks.

Step 3. In BRT, MS periodically broadcasts SLOT_DISTRI_MESSG (containing node's ID, number of slot allocated to node (based on its priority as indicated in the SCHEDULE_MESSG received in previous super frame), numbers of unallocated slots). During CFSO, nodes in odd layers are in transmitting and even in receiving state. During CFSE, nodes in even layers are in transmitting and odd in receiving state. These compound function slots are used for miscellaneous functions as described in the subsequent discussions.

Step 4. Node in scanning mode (say node i) is in listen mode and when it receives SLOT_DISTRI_MESSG from MS it moves to harmonized mode. It then selects an unoccupied slot as its TXS, selects one of its neighbor nodes in scheduled mode as its parent node, and sends out a SCHEDULE_MESSG (containing node details like its ID, layer, priority, aging factor) both in CST and its TXS.

Step 5. Nodes in scheduled mode transmit data in their allocated TXS and listen for transmissions from their neighbor nodes (in scanning mode) in CST and RXS. They turn off their radio during idle slots (IDT) to save energy.

Step 6. When a scheduled node f hears transmission in CST, meaning one or more of its neighbor nodes may set it as parent node, it listens during all its idle sots (IDT). Upon receiving SCHEDULE_MESSG of node i, node f decides whether to become parent of i based on *parenthood willingness, PW(f)*. A node f decides to

become a parent of node i if its $PW(f)$ is 1. The $PW(f)$ is determined as follows,

$$PW(f) = \begin{cases} 1, & if \begin{cases} L(i) > L(f) \\ E_{res}(f) > E_{Thresh} \\ TPR(f) \leq TPR_{Thresh} \\ LQI_{SCHM}(i) > LQI_{Thresh} \end{cases} \\ 0, & otherwise \end{cases} \qquad (4)$$

The *parenthood willingness* is set to one if all three conditions in (4) are satisfied, where each condition represents definite communication functionality. The first condition $L(i)>L(f)$, checks if node i is at a higher layer than f which ensures that packet is forwarded in right direction towards MS. Second condition, i.e., $E_{res}(f)>E_{Thresh}$, ensures that residual energy of node f is above a minimum value, E_{Thresh} to maintain a uniform distribution of energy consumption throughout the network. Third condition, $TPR(f)<TPR_{Thresh}$, limits the traffic a node will transit. More specifically, a node accepts to become a parent if its transit packet rate (*TPR*) is below some threshold value TPR_{Thresh}. *TPR* quantifies node's relaying efficiency and does a passive local congestion control and traffic load distribution. The fourth condition, i.e., $LQI_{SCHM}(i)>LQI_{Thresh}$, checks if Link Quality Indicator (LQI) of the SCHEDULE_MESSG (LQI_{SCHM}) from node i is above some threshold value i.e LQI_{Thresh}. Link Quality indicator (LQI) characterizes strength and/or quality reception of a packet at a node [38]. Thus, for a node to become a relay it should have a sufficient value of LQI to ensure reliable parent-child communication link quality. If $PW(s)$ is 1 node decides to become a parent, it regards sender as its child and forwards SCHEDULE_MESSG to its own parent. If more than two nodes satisfy *parenthood willingness* node with least ID will be parent. This process is repeated until SCHEDULE_MESSG reaches MS.

Step 7. On receiving SCHEDULE_MESSG, MS allocates an unoccupied slot on first come first serve basis for the request and adds slot allocation information in next SLOT_DISTRI_MESSG. The overall slot allocation scheme at MS considers node priorities (prioritized nodes are given multiple slots) and also allows a slot to be shared among many nodes as long as nodes' transmission ranges are not within interference range of each other. This maximizes spatial slot re-use increasing concurrency in channel usage and reducing data latency.

Step 8. When a node in harmonized mode hears SLOT_DISTRI_MESSG from MS, it goes into scheduled state and changes its TXS to that allocated by MS. Its parent uses SLOT_DISTRI_MESSG to set its RXS as TXS of its child (combing routing information with slot allocation). Nodes is scheduled mode switch to receive state for their scheduled RXS, transmit state for scheduled TXS, or else turn off the radio module, if they do not hear transmission in CST.

After each node has been scheduled, spanning tree of network rooted at MS is constructed.

• **Data transfer:** During data transfer frame each child node transmits DATA_MESSG (containing parent's ID, its own ID, data, status indicator (SI) (marks energy level of node, set to one when node is about to die), link indicator (LI) (marks parent-child communication link quality, set to one when link quality is bad (known from LQI value)) through multi-hop transmission. For this, TXS of a node overlaps with RXS of its parent. Each parent node forwards DATA_MESSG to its own parent until it reaches MS. It also sends DATA_RESP_MESSG (containing parent's ID, child's ID, choke bit (CB) and retransmit bit (RB)) during the compound function slots under special circumstances as explained in the subsequent discussions. The cross-layer interaction between MAC and routing protocol requires only nodes involved in a communication to be awake during their associated slots. This results in efficient usage of energy, storage and bandwidth resources. The data transmission slot allocation is done centrally by MS and hence data reaches MS through collision-free multi-hop transmissions.

• **Orphan Nodes:** There may be conditions when a node may not find any feasible neighborhood node that satisfies required parenthood conditions. To overcome this, node tries to get a parent by sending SCHEDULE_MESSG for K_{max} times and increases its aging factor each time it sends the message. When aging factor goes beyond a threshold parent node in its neighborhood accepts the orphan node as its child irrespective of its *parenthood willingness* condition.

• **Strategic back off and backward scheduling:** As the slot allocation is centrally done by MS, collisions can only happen when two or more non-scheduled nodes select same TXS to transmit SCHEDULE_MESSG. To avoid collisions, node f strategically backs off for T_{bk} time interval given by,

$$T_{bk}(f) = \frac{residual\ energy(f)}{layer(f) \times aging\ factor(f)} \qquad (5)$$

This also makes SCHEDULE_MESSGs of nodes at layers far from MS back off for less amount of time, making their SCHEDULE_MESSG reach earlier to MS (backward scheduling) and thereby get an earlier slot position in the frame. This ultimately decreases the overall data latency.

• **Node-by-node local congestion control:** Traffic sources for non-leaf nodes are: data generated by its sensing unit and transit data from its children. The local cross-layer congestion control mechanism of LEFT is incorporated by: (i) passive congestion control: the third condition in equation (4) limits node's transit data. (ii) active congestion control: each node explicitly regulates rate at which data packets are generated by sensing unit. For this if a child node f receives DATA_RESP_MESSG

with CB bit set to one from its parent in CFS it comes to know that subnet is congested and multiplicatively decreases its data traffic $\sigma(f)$ as,

$$\sigma(f) = \frac{1}{d} \times \sigma(f) \qquad (6)$$

where d is decreasing factor.

- *Slot seizing*: If a node is a holder of a TXS slot and does not have data to send, its child can take away the slot if it has data to send. For this, parent switches to receive mode when it does not have data to send. Its child takes a back off of $T_{bkchild}$ ($TXS > T_{bkchild} > T_{bkparent}$). On expiration of back off timer it does clear channel assessment and transmits DATA_MESSG if channel is free. Thus, node can transmit as soon as channel is available increasing channel utilization and decreasing latency.

- **Adaptation to link breakdown:** The communication link between parent and child nodes may fail due to external environmental factors like fog, rain, humidity or buildings and other obstacles [8]. These breakdowns can be temporarily or permanent depending on environment and obstacles.

Temporary link breakdowns: When a parent does not receive a DATA_MESSG when it expects one from its child during its RXS slot, it requests a data retransmission by sending a DATA_RESP_MESSG with RB bit set to one in CFS. The child sends requested data in the next data transfer frame. Furthermore, if a parent is required to send data retransmission request repeatedly, it sends DATA_MESSG with LI bit set to one to MS. MS then allocates more TXS/RXS to nodes connected with that link. On link re-establishment parent sets LI bit to zero and MS resumes original slot allocation.

Permanent link breakdowns: If node receives DATA_RESP_MESSG with RB set to one for a number of data frames it accepts the link to be too noisy, assumes itself to be orphan and decides to find a new parent.

- **Adaptation to node deaths:** During course of simulation nodes use up their energy and are said to be dead when they can no longer transmit or receive data. When a node is about to die it sets SI bit to one. Parent removes dying child from its child list and does not go in listen mode during RXS of dying child. On receiving DATA_MESSG with SI bit set to one, MS marks node as a dead and sets its time slot as unoccupied in SLOT_DISTRI_MESSG. When a node finds its parent as dead, it repeats scheduling process to select a new parent retaining its child information.

IV. PERFORMANCE EVALUATION

An experimental approach is taken to demonstrate LEFT's real world feasibility and also compare its performance with ATMA, a distributed-TDMA protocol and D-MAC, EEFF and CL-MAC which are MAC and routing based cross layer protocols.

- **Experimental setup:** All the protocols are implemented on EZ430-RF2500T [37] nodes from TI. The nodes are programmed using IAR embedded workbench [39]. The node's controller is highly integrated ultra-low power MSP430F2274 from TI working at 16 MHz with 16-bit data bus, 32 KB flash and 1 KB RAM sufficient to code the protocols for experimentation. The USB debugging interface enables node to remotely send and receive data from a PC using the MSP430 application UART and thus work as MS. The interface can also be used for node programming and debugging. Node's radio is TI's CC2500 operating in 2400.0-2483.5 MHz frequency band (non-licensed ISM band). It uses GFSK modulation and its bit rate is 250 Kbps. Each node uses two alkaline AAA batteries with total 2000 mAh (2*1000 mAh) capacity, supply voltage of 3.0 V (2*1.5 V) and total initial energy of 10,800 J (2*5400J). Radio can be operated in transmitting, receiving, idle, sleeping modes and it consumes 29.88 mW, 38.16 mW, 4.5 mW and 1.2 μW of power in each of the modes respectively. These features and some more as depicted in the authors' previous work [2], led to selection of EZ430-RF2500T nodes for protocol experimentation. The experimental testbed comprises of 20 nodes deployed in a 37 m x 92 m area with MS at upper end center (18, 92) coordinates in a laboratory room. Network is organized into four layers with inter-layer distance of 23 m as shown in Fig. 3. Thus, a fairly strict, small and dense
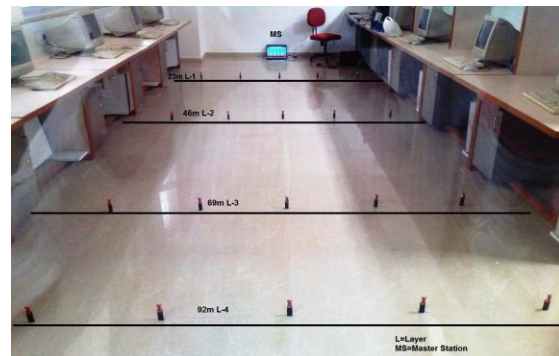

Fig. 3. Testbed with twenty nodes and one MS

multi-hop radio environment is constructed for experimentation. The parameter settings of LEFT are discussed next. The control packet length is up to 8 bytes, data packet length is up to 176 bytes (parent nodes at each layer append their data with the data received from their child), TDMA slot size is 6 ms (large enough to send an entire data packet), data transfer frame length is of 156 ms (to accommodate data slots for 20 motes, 1 slot for MS and 5 control slots super-frame length is 174 ms and data (readings of internal temperature sensor of node) reporting time is every 3 secs. For ATMA, D-MAC, EEFF and CL-MAC the default settings described in [18], [25], [28] and [30] respectively are used. The data packet length, reporting period and network deployment is same for all the protocols for fair comparison. For ATMA and D-MAC, MTE protocol [40] is used for routing.

A. LEFT Parameters

Parameters that affect LEFT operation are E_{Thresh}, TPR_{Thresh} and LQI_{Thresh}. The most efficient values of these thresholds are determined in this sub-section.

- **Energy threshold E_{Thresh}**: Each intermediate relay node in LEFT is required to forward its own data and the transit data from its child to the next hop. Maximum distance to next hop will be when relay node is at lower end of its layer and its next hop node is at upper end of its layer. Under this condition relay node will require energy large enough to transmit to node at a two layer distance. Thus, E_{Thresh} has to be large enough for transmitting data at a distance of two layers. It is set to 14 mJ for experimentations done in the paper.

- **Transit packet rate threshold TPR_{Thresh}**: First upper bound for total transit packet rate that will prevent congestion is derived for each node. The overall input packet rate at node f, α_f can be denoted as,

$$\alpha_f = \alpha_{fgen} + \alpha_{frel} \qquad (7)$$

where α_{fgen} is generated packet rate, α_{frel} is overall transit packet rate of node f. If α_{if} is transit packet rate from node i to f and N_f^{inp} is set of nodes from which node f receives transit packets then, $\alpha_f = \alpha_{fgen} + \sum_{i \in N_f^{inp}} \alpha_{if}$. Thus,

$$\alpha_f = \alpha_{fgen} + \sum_{i \in N_f^{inp}} \alpha_{if} \qquad (8)$$

Since node f wishes to transmit all the packets in its buffer, its overall output rate is given by,

$$\beta_f = \alpha_{fgen} + \alpha_{frel} + \alpha_{fret} \qquad (9)$$

where α_{fret} is packet retransmission rate which will be very less since relay node is selected by considering LQI of the link. The output rate is higher than input rate since node retransmits packets that are not successfully sent. According to (8) and (9), in a data transfer sub-frame T_{sub}, average time node f spends in transmitting (T_{tx}) and receiving (R_{rx}) is,

$$T_{tx} = (\alpha_{fgen} + \alpha_{frel} + \alpha_{fret}) \times T_{sub} \times T_{pkt} \qquad (10)$$

$$R_{rx} = \alpha_{frel} \times T_{sub} \times T_{pkt} \qquad (11)$$

where T_{pkt} is average duration to transmit a packet from a node to its next hop. If a node transmits and receives packet within time slot given to it, congestion in the node can be prevented. Because of time slot allocation technique, a node is minimally active for T_{tx}/N sec, where N is number of nodes in the network. Therefore,

$$\frac{T_{tx}}{N} \geq (\alpha_{fgen} + \alpha_{frel} + \alpha_{fret}) \times T_{sub} \times T_{pkt} + \alpha_{frel} \times T_{sub} \times T_{pkt} \qquad (12)$$

$$\frac{T_{tx}}{N} \geq (\alpha_{fgen} + 2\alpha_{frel} + \alpha_{fret}) \times T_{sub} \times T_{pkt} \qquad (13)$$

Thus the input relay packet rate α_{frel} is bounded by,

$$\alpha_{frel} \leq \alpha_{frel}^{Th} \qquad (14)$$

where the threshold of relay rate α_{frel}^{Th} is given by,

$$\alpha_{frel}^{Th} = \frac{1}{2NT_{pkt}} - \frac{\alpha_{fgen}}{2} - \frac{\alpha_{fret}}{2} \qquad (15)$$

From the above analysis it can be concluded that, congestion at a node can be prevented by throttling input transit rate. This is incorporated in LEFT through *parenthood willingness* where nodes participate in routing only if equation (4) is satisfied. The implementation of equation (15) requires node to calculate parameters T_{pkt}, α_{fgen} and α_{frel}. The generated packet rate, α_{fgen}, can be calculate from rate of packets injected from sensors. The packet error rate, α_{frel}, is stored as average of packet loss rate encountered by node. Finally, T_{pkt} is determined by using delay encountered in sending previous packet by node. All these values are updated by nodes at regular time intervals.

- **Link Quality Indicator LQI_{Thresh}**: For EZ430-RF2500T nodes with CC2500 radio LQI gives an estimate of how easily a received signal can be demodulated by accumulating magnitude of error between ideal constellations and received signal over 64 symbols immediately following the sync word [41]. LQI measurement is performed for each received packet, and result is reported to MAC sub-layer as an integer ranging from 0 to 255. The minimum and maximum LQI values (0 and 255) are associated with lowest and highest quality reception detectable by radio and values in between are distributed between these two limits. In Fig. 4, effect of LQI_{Thresh} on total throughput at MS and in Fig. 5 its effect on latency is shown. The No LQI is when the fourth condition in equation (4) is not implemented i.e nodes check for participating as a relay node irrespective of LQI_{Thresh}. It can be seen that increasing LQI threshold, improves throughput and decreases latency up to a certain LQI_{Thresh}. Above this value, throughput and latency performance degrades. This shows a very conservative operation of LEFT leads to a degraded performance. Since, for both the metrics, LQI_{Thresh} of 60 results in most efficient performance, this value is used for experimentations done in the paper.

B. Comparative Evaluation

All the experiments were run five times with same setup and average values were adopted as results.

• **Energy consumption of network:** All 20 nodes begin with 10,800 J of energy and an unlimited amount of data to be sent to MS. Sum of energy consumed by all nodes in the network is traced at an interval of 3000 secs. Fig. 6 shows sum of energy consumed in LEFT is up to 65% less compared to CL-MAC. The energy savings in LEFT is due to following: centralized generation of TDMA schedule which guarantees collision-free communication, reduced idle listening since nodes sleep until their next communication activity and strategic back off which avoids collisions during slot requests to MS. Use of RTS/ACTS/ACK for each message transfer in EEFF and interference between nodes on nearby branches of data gathering tree in D-MAC results in significant energy consumption. ATMA does not exploit inter layer interactions and hence does not achieve energy performance gains.
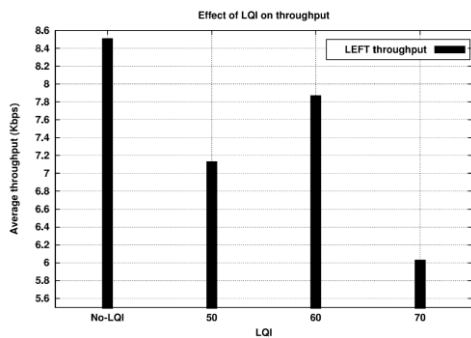
Fig. 4. Average throughput for different values of LQI_{Thresh}
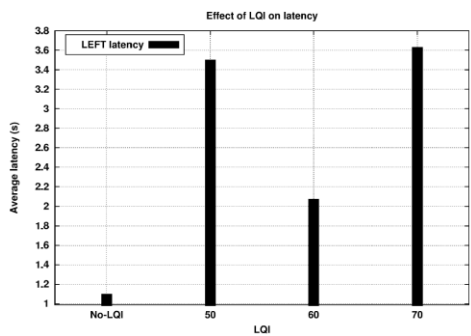
Fig. 5. Average latency for different values of LQI_{Thresh}
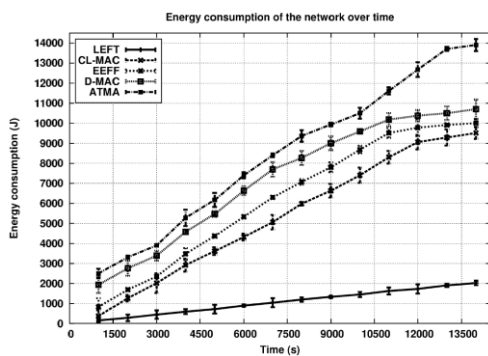
Fig. 6. Total energy consumption of the network over time

• **Data delivery to MS:** In most of the cases merit of a WSN protocol is an application-specific measure. But one application-independent method of determining merit of the protocol is to determine total data packets received at MS. The analysis of sensed environment will be more precise if more data is received at MS. Fig. 7 shows total number of data packets received at MS over time. LEFT sends maximum packets to MS compared to any other protocol. This is because node-by-node congestion control (active and passive) and selection of reliable parent-child pair link minimizes the packet drops in LEFT.

• **Network energy consumption with distance:** Fig. 8 shows average energy consumption of nodes in percentage from layer 1 (at 23 m from MS) to layer 4 (at 92 m from MS). Variation in energy with distance is 1.11 % in LEFT, 1.14 % in D-MAC, 1.30 % in CL-MAC, 1.34 % in EEFF and 1.91% in ATMA. This is because nodes in LEFT decide to participate in routing only if it has sufficient residual energy and less traffic load. These checks uniformly distributes energy load amongst all nodes in the network.

• **Packets received with distance**: Fig. 9 shows total number of packets received from nodes at different layers (distances from MS). As the distance increases, variation in received packets in LEFT is 9%, in CL-MAC it is 13%, in EEFF it is 14 %, in D-MAC it is 14% and in ATMA it is 12 %. Due to centralized TDMA approach in LEFT, nodes at each layer gets a chance to transmit data packet. Further due to parenthood *willingness* condition only the nodes which have sufficient residual energy, good communication link quality and in the direction towards the MS take part in communication to make sure they deliver packets to MS correctly. Hence, variation in packet received with distance is least in LEFT.

• **Throughput with varying media contenders:** Fig. 10 compares throughput of all the protocols with varying number of contenders (sources). LEFT consistently achieves higher throughput because of appropriate selection of relay node and spatial reuse of slots. Further, during less number of contenders slot seizing and with more contenders congestion control technique increases throughput in LEFT. In case of ATMA, as the number of sources increases energy consumption incurred by ADV period increases and effects of clock drifts gets worst which in turn degrades its throughput.

• **Scalability:** For analyzing scalability of protocols number of source nodes is varied from 2 to 20 and total energy consumed in the network, average data latency and delivery ratio is measured. Fig. 11, 12 and 13 show that as number of source nodes increases energy consumed in the network and data latency increases and delivery ratio decreases. This effect is least in LEFT since it is the most scalable protocol. This is because, LEFT does uniform distribution of energy by sharing transit traffic by placing an upper limit on traffic a node can relay. Further, concurrent transmissions, combined MAC and routing approach, and exchange of transmission slots decreases LEFT's latency. Delivery ratio of LEFT is impressive because of its techniques to adapt to link/node failures, techniques for local congestion control, selecting

parent which is energy efficient and on the right path towards MS.

• **Fairness Index:** In many applications, particularly when bandwidth is scarce, it is important to ensure that MS receives information from all sources in a fair manner. Fairness index of throughput of data packets from each source node is measured with varying number of packets per minute. The results plotted in Fig. 14 show that LEFT is the fairest protocol. This is because of its centralized TDMA approach. CL-MAC does schedule computation based on buffered packets and flow setup requests without considering topological routing information. If flow setup requests are not correctly received it degrades fairness index. ATMA is comparatively fair because it uses TDMA for scheduling. D-MAC and EEFF do not consider fairness in their design.
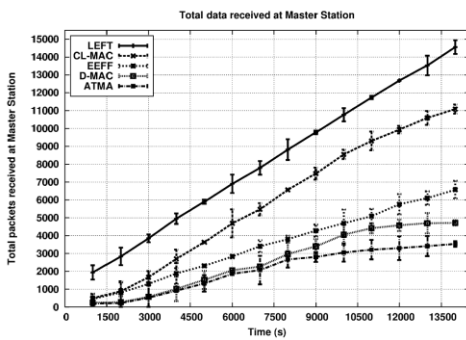
Fig. 7. Total data received at MS over time
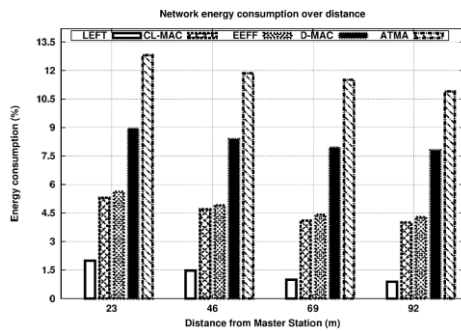
Fig. 8. Average energy consumption over distance
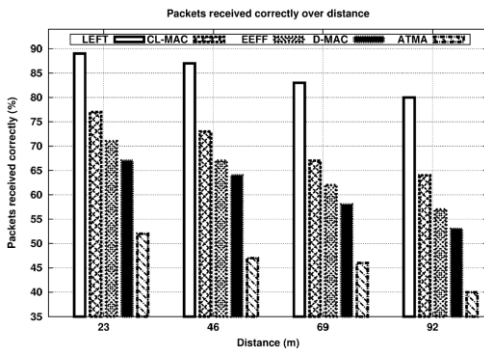
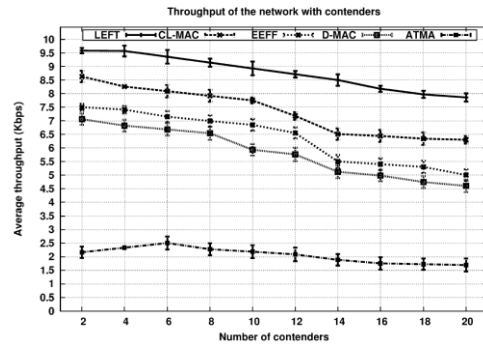Fig. 9. Packet received correctly over distance

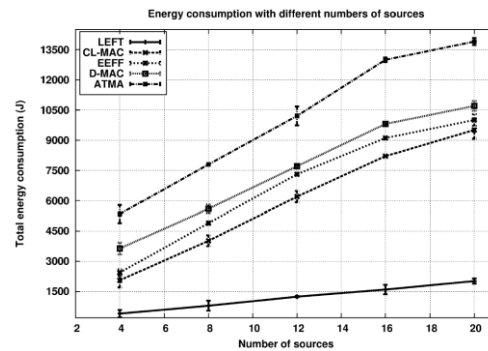Fig. 10. Average throughput with varying contenders

Fig. 11. Total energy consumption with varying sources

• **More results:** Table 1 shows LEFT has maximum goodput because it requires minimum number of control packets compared to other protocols. Initial network setup time of LEFT is maximum because each node is required to get scheduled and wait for slot distribution message transmission from MS. With techniques to adapt to link and node breakdown network recovery time of LEFT is least amongst all the protocols.

Table 1. Experimental results

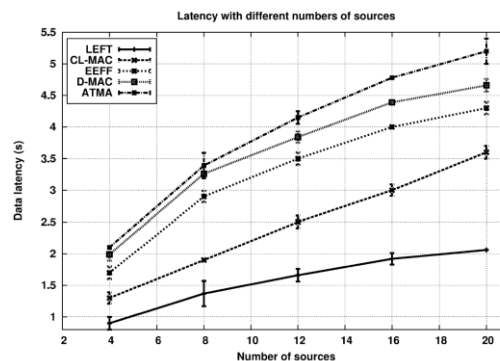| Parameter	LEFT	EEFF	D-MAC	CL-MAC	ATMA
Goodput (%)	88	77	86	84	85
Initial Network setup time (s)	5.13	3.83	2.56	3.43	3.93
Network recovery time (s)	5.12	7.10	9.10	10.15	9.24

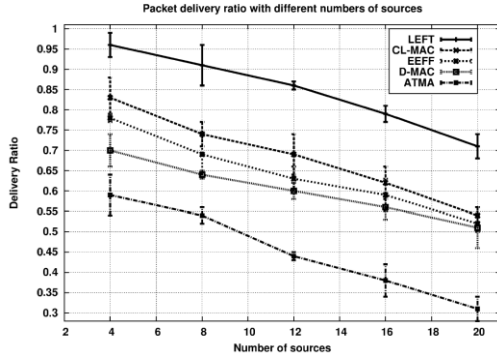Fig. 12. Average data latency with varying sources

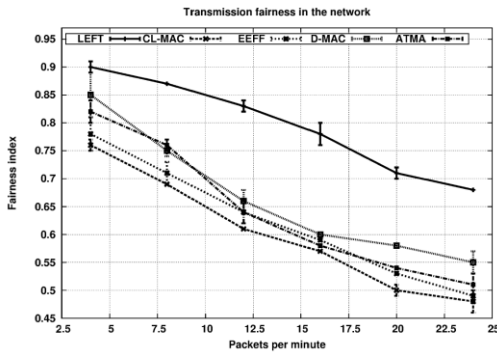Fig. 13. Delivery ratio with varying sources

Fig. 14. Fairness index with varying source traffic

• **Implementation complexity:** The use of local information for routing decisions and absence of a global routing eliminates requirement for storage of network-wide routing data at each node in LEFT. MAC and routing based cross layer approach decreases memory requirement of LEFT. Thus, LEFT implementation requires 52 kB size of controller's flash memory.

• **Network lifetime in LEFT:** Lifetime of LEFT can be estimated in hours as,

$$L = \frac{Q}{E_{hour}} \qquad (16)$$

where Q=2000 mAh, energy from 2 AAA alkaline batteries of 1.5 V and E_{hour} is total energy consumption of a node in an hour. The energy consumption of a single node is estimated as sum of energy consumed during active (sending, receiving) and sleep states. The transceiver activity starts with beacons send by MS and after that nodes send or receive data. Thus, it is appropriate to determine energy consumption in relation to beacons as follows,

$$E_{hour} = E_{tx} + E_{rx} + E_{sleep} \qquad (17)$$

$$= (B \times (t_{startup}I_{startup} + t_{tx}I_{tx} + t_{rx}I_{rx} + t_{shutdown}I_{shutdown}))$$

$$+ T_{sleep}I_{sleep} \qquad (18)$$

where, E_{tx}, E_{rx}, E_{sleep} is node's hourly energy consumption when transmitting, receiving and sleeping respectively; I_{tx}, I_{rx}, I_{sleep}, $I_{startup}$, $I_{shutdown}$ is (average) current consumption when transmitting, receiving, sleeping, starting up and shutting down transceiver respectively; t_{tx}, t_{rx} are times of transmitting and listening for data; $t_{startup}$, $t_{shutdown}$ are times to startup and shutdown the transceiver; T_{sleep} is total sleep time of node in an hour; B is number of beacons in an hour. For all calculations total energy consumed by transceiver when sending, receiving and sleeping also contains also energy consumed by microcontroller in these states. B, the number of beacons in an hour can be estimated as,

$$B = \frac{T_{hour}}{T_{beacon}} \qquad (19)$$

where T_{beacon} is beacon period (within the hour), and expressed in secs as T_{hour} is in secs. Substituting the values of T_{beacon}=3400(beacons send) x 6 ms (slot) = 20.4 secs and T_{hour} = 3600 secs, B ≈ 176. Node sends and receives data periodically, after the beacon. Knowing transceiver data rate and amount of data transmitted, the single transmit (t_{tx}) and receive (t_{rx}) times are estimated. Apart from when transmitting and receiving node sleeps rest of the time. Thus, sleep time in an hour can be determined by subtracting transmit and receive times as,

$$T_{sleep} = T_{hour} - B \times (t_{startup} + t_{tx} + t_{rx} + t_{shutdown}) \qquad (20)$$

where T_{hour}, t_{tx}, t_{rx}, $t_{startup}$, $t_{shutdown}$ are in secs. Substituting their values,

$$T_{sleep} = 3600 - [176 \text{ x } \{(1+320+400+1) \text{ x } 10^{-3}\}]$$
$$= 3472.928 \text{ secs}$$

For measuring current consumption of network, a 10 Ω resistor is placed across battery and board as shown in Fig. 15. Voltage drop across the resistor is measured using digital storage oscilloscope when node is transmitting, receiving, sleeping and shutdown as in the Fig.16. The current consumption is then measured using ohm's law V = IR [42]. Substituting measured values in equation (18), gives E_{hour}=893.684 µA. Substituting E_{hour} in equation (16) gives,

$$L = \frac{Q}{E_{hour}} = \frac{2000mAh}{893.684µA} = 2.2379x10^3 \approx 93 days$$

Thus, lifetime of LEFT is 93 days when the data reporting time is every 3 secs.

V. CONCLUSION

This paper presents a TDMA protocol called LEFT that ensures strengths of TDMA like fairness and energy efficiency and offsets its weakness like non-adaptability
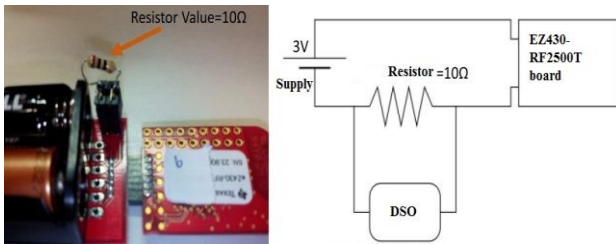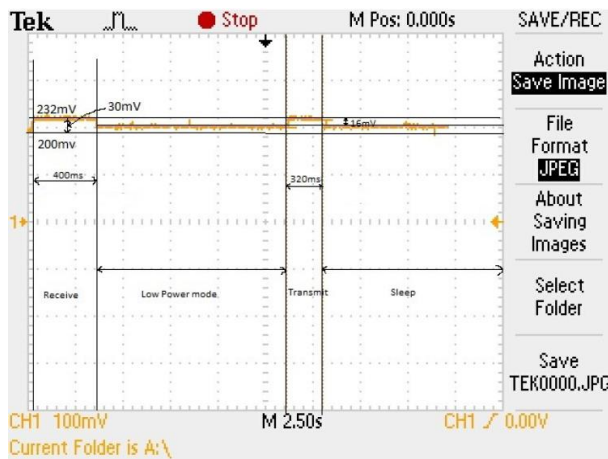
Fig. 15. Setup for current measurement

Fig. 16. Voltage readings on DSO

to traffic pattern, lower channel utilization, lower throughput and higher latency during low load. Common framework for route set up and data transfer in LEFT respects energy, computational and storage constraints of nodes. Experimental results show that LEFT outperforms ATMA (MAC protocol with hybrid-approach) and D-MAC, EEFF and CL-MAC (MAC and routing based cross-layer protocols).

ACKNOWLEDGMENTS

REFERENCES

[1] S. H. Gajjar, S.N. Pradhan, K.S. Dasgupta "Wireless Sensor Networks: Application led research perspective", *Proc. of IEEE Recent Advances in Intelligent Computational Systems*, pp. 025 – 030, 2011.

[2] Sachin Gajjar, Nilav Choksi, Mohanchur Sarkar, Kankar Dasgupta, "Comparative analysis of Wireless Sensor Network motes", *Proc. of International Conference on Signal Processing and Integrated Networks*, pp. 426 – 431, 2014.

[3] Tifenn Rault, Abdelmadjid Bouabdallah, Yacine Challal "Energy efficiency in wireless sensor networks: A top-down Survey", *Elsevier Journal of Computer Networks* vol. 67(4), pp. 104-122, 2014.

[4] A. Bachir, M. Dohler, T. Watteyne, K. Leung, "MAC essentials for wireless sensor networks", *IEEE Communications Surveys and Tutorials*, vol. 12 (2), pp. 222–248, 2010.

[5] Ratnadip Adhikari, "A meticulous study of various medium access control protocols for wireless sensor networks", *Elsevier Journal of Network and Computer Applications*, vol. 41, pp. 488–504, 2014.

[6] S.H. Gajjar, S.N. Pradhan, K.S. Dasgupta, K.V. Singhla, K.P. Zinzuvadia, "Comparative analysis of Medium Access Control Networks protocols for Wireless Sensor Networks," *Proc. of IEEE India Conference*, pp. 1–4, 2011.

[7] Messaoud Doudou, Djamel Djenouri, Nadjib Badache, Abdelmadjid Bouabdallah, "Synchronous contention-based MAC protocols for delay-sensitive wireless sensor networks: A review and taxonomy", *Elsevier Journal of Network and Computer Applications*, vol. 38, pp. 172–184, 2014.

[8] Kaveh Pahlavan, Prashant Krishnamurthy, "Principles of Wireless Networks A Unified Approach", *Prentice Hall Publication*, 2002.

[9] S. Jagadeesan, V. Parthasarathy, "Cross-Layer Design in Wireless Sensor Networks", *Advances in Intelligent and Soft Computing*, vol. 166, pp. 283-295, 2012.

[10] Lucas D. P. Mendes, Joel J.P.C. Rodrigues, "A survey on cross-layer solutions for wireless sensor networks", Elsevier Journal of Network and Computer Applications vol. 34 (2), pp. 523-534, 2011.

[11] S.H. Gajjar, S.N. Pradhan, K.S. Dasgupta, "Cross layer Architectural Approaches for Wireless Sensor Networks", *Proc. of IEEE Recent Advances in Intelligent Computational Systems*, pp. 557 – 562, 2011.

[12] Sachin Gajjar, S. N. Pradhan, K. S. Dasgupta, "Performance Analysis of Cross Layer Protocols for Wireless Sensor Networks", *Proc. of International Conference on Advances in Computing, Communication and Informatics*, pp. 348-354, 2012.

[13] V. Rajendran, K. Obraczka, J. J. Garcia-Luna-Aceves, "Energy-Efficient Collision-Free Medium Access Control for Wireless Sensor Networks", *Proc. of ACM International Conference on Embedded Networked Sensor Systems*, pp. 181-192, 2003.

[14] V. Rajendran, J. Garcia-Luna-Aceves, K. Obraczka, "Energy-Efficient, Application-Aware Medium Access for Sensor Networks", *Proc. of IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, pp. 181-892, 2005.

[15] W. Ye, J. Heidemann, D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks", *IEEE/ACM Transactions on Networks*, vol. 12(3), pp. 493–506, 2004.

[16] T. Zheng, S. Radhakrishnan, V. Sarangan, "PMAC: an Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks", *Proc. of IEEE International Parallel and Distributed Processing Symposium*, vol. 13, pp. 237a, 2005.

[17] G. S. Ahn, E. Miluzzo, A. T. Campbell, S. G. Hong, F. Cuomo, "Funneling - MAC: A localized, sink-oriented MAC for boosting fidelity in sensor networks ", *Proc. of ACM Conference on Embedded Networked Sensor Networks*, pp. 293-306, 2006.

[18] Gang Lu, Bhaskar Krishnamachari, Cauligi S. Raghavendra, "An adaptive energy-efficient and low-latency MAC for tree-based data gathering in sensor networks", *Wiley InterScience Journal of Wireless Communications and Mobile Computing*, vol. 7, pp. 863-875, 2007.

[19] G. Halkes, K. Langendoen, "Crankshaft: An Energy-Efficient MAC-Protocol for Dense Wireless Sensor Networks", *Proc. of European Conference on Wireless Sensor Networks*, pp. 228-244, 2007.

[20] Sun Y, DuS, Gurewitz O, Johnson DB, "DW-MAC: a low latency, energy efficient demand-wakeup MAC protocol for wireless sensor networks", *Proc. of ACM international symposium on Mobile Adhoc Networking and computing*, pp. 53–62, 2008.

[21] Zhao Y Z, Ma M, Miao C Y, Nguyen T N, "An energy-efficient and low-latency MAC Protocol with adaptive scheduling for multi-hop wireless sensor networks", *Computer Communications*, vol. 33, pp. 146–452, 2010.

[22] Rajesh Yadav, Shirshu Varma, N. Malaviya, "Performance Analysis of Optimized medium access control for wireless sensor network", *IEEE Sensors Journal*, vol. 10 (12), pp. 1863-1868, 2010.

[23] I. Rhee, A. Warrier, M. Aia, J. Min, "Z-MAC: a hybrid MAC for wireless sensor networks", *IEEE/ACM Transactions on Networking*, vol. 16 (3), pp. 511-524, 2008.

[24] I. Rhee, A. Warrier, Jeongki Min, Lisong Xu, "DRAND: Distributed Randomized TDMA Scheduling for Wireless Ad Hoc Networks", *IEEE Transactions on Mobile Computing*, vol. 8(10), pp. 1384 -1396, 2009.

[25] Tao Zhang, Lijun Chen, Daoxu Chen, Li Xie, "EEFF: A Cross-Layer Designed Energy Efficient Fast Forwarding Protocol for Wireless Sensor Networks", *Proc. of IEEE Wireless Communications and Networking Conference*, pp. 1 – 6, 2009.

[26] Cortes A, Gamboa R, Pena N, Labrador M A., "Low energy and low latency in wireless sensor networks", *Proc. of IEEE International Conference on Communications*, pp. 1-5, 2009.

[27] Choi L, Lee S H, Jun J. A,"SPEED-MAC: speedy and energy efficient data delivery MAC Protocol for real-time sensor network applications", *Proc. of IEEE International Conference on Communications*, pp. 1-6, 2010.

[28] Surjya Ray, Ilker Demirkol, Wendi Heinzelman, "ATMA: Advertisement-based TDMA Protocol for Bursty Traffic in Wireless Sensor Networks", *Proc. of IEEE Global Telecommunications Conference*, pp. 1-5, 2010.

[29] Sthapit P, Pyun J., "Medium reservation based sensor MAC protocol for low latency and high energy efficiency", *Journal of Telecommunication Systems, Springer*, vol. 52 (4), pp. 2387-2395, 2013.

[30] Mohamed S. Hefeida, Turkmen Canli, Ashfaq Khokhar, "CL-MAC: A Cross-Layer MAC protocol for heterogeneous Wireless Sensor Networks", *Elsevier Journal of Ad Hoc Networks*, vol. 11(1), pp. 213–225, 2013.

[31] S. Saqaeeyan, M. Roshanzadeh, "Improved Multi-Path and Multi-Speed Routing Protocol in Wireless Sensor Networks", *International Journal of Computer Network and Information Security (IJCNIS), MECS Publisher*, vol. 4(2), pp. 8-14, 2012.

[32] M.Siddappa, Channakrishna Raju, "Survey on an Efficient Coverage and Connectivity of Wireless Sensor Networks using Intelligent Algorithms", *International Journal of Information Technology and Computer Science (IJITCS), MECS Publisher*, vol. 4(5), pp. 39-45, 2012.

[33] S. Saqaeeyan, M. Roshanzadeh, "IEATH: Improved Energy Aware and Two Hop Multipath Routing Protocol in Wireless Sensor Networks", *International Journal of Computer Network and Information Security (IJCNIS), vol. 4(5), MECS Publisher*, pp. 22-28, 2012.

[34] Robin Guleria, Ankit Kumar Jain, "Geographic Load Balanced Routing in Wireless Sensor Networks", *International Journal of Computer Network and Information Security (IJCNIS), vol. 5(8), MECS Publisher*, pp.62-70, 2013.

[35] Debasmita Sengupta, Alak Roy, "A Literature Survey of Topology Control and Its Related Issues in Wireless Sensor Networks", *International Journal of Information Technology and Computer Science (IJITCS), vol. 6(10), MECS Publisher*, pp. 19-27, 2014.

[36] P. Raghu Vamsi, Krishna Kant, "An Improved Trusted Greedy Perimeter Stateless Routing for Wireless Sensor Networks", *International Journal of Computer Network and Information Security (IJCNIS), vol. 6(11), MECS Publisher*, pp. 13-19, 2014.

[37] eZ430-RF2500 Development Tool User's Guide [Online] available:http://www.ti.com/lit/ug/slau227e/slau227e.pdf

[38] Michele Rondinone, Junaid Ansari, Janne Riihijarvi, Petri Mahonen, "Designing a Reliable and Stable Link Quality Metric for Wireless Sensor Networks", *Proc. of workshop on Real-world wireless sensor networks*, pp. 6-10, 2008.

[39] IAR Embedded workbench [Online] Available: http://www.iar.com/ServiceCenter/Downloads/.

[40] M. Ettus, "System capacity, latency, and power consumption in multi-hop routed SS-CDMA wireless networks", *Proc. of Radio and Wireless conference*, pp. 55–58, 1998.

[41] Thomas Watteyne, "eZWSN Exploring Wireless Sensor Networking", *Technical report, Berkeley Sensor and Actuator Center, UC Berkeley*, pp. 20-21, 2009.

[42] B. L. Theraja, A.K. Theraja, "A Textbook of Electrical Technology: Basic Electrical Engineering in S.I. Units", vol. 1, *S. Chand Publications*, 2007.

Authors' Profiles

Sachin Gajjar received his B.E. degree in Electronics and Communication Engineering from Government Engineering College, Gujarat, India in 1999. He received his M.Tech. degree in Computer Technology from Indian Institute of Technology, Delhi (IIT Delhi) in 2007. He worked at Videocon International Limited, Gandhinagar, Gujarat, India in Product Development Group from June 1999 to December 1999. From January 2000 to December 2000 he worked with Cincinnati Milacron, Ahmedabad, Gujarat, India in Software Development Group. In January 2001, he joined L. D. College of Engineering, Ahmedabad, Gujarat, India as Lecturer in Computer Science and Engineering Department and worked there till August 2002. Since August 2002, he is working as Assistant Professor in Electronics and Communication Engineering Department, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India. Currently he is doing his Ph.D. in the area of wireless sensor networks in Computer Science and Engineering Department, Institute of Technology, Nirma University. His research interests include computer networks and wireless sensor networks.

Nilav Choksi received his B.E. degree in Electronics and Communication Engineering from L.D.R.P. Institute of Technology and Research, Gandhinagar, Gujarat, India in 2011. He received his M.Tech. Degree in Embedded Systems from Institute of Technology, Nirma University, Ahmedabad, Gujarat, India in 2014. He worked at Komoline Electronics Private Limited, Ahmedabad, Gujarat, India in Research and Development Group from June 2011 to July 2012. Since July 2014, he is working as Technical Associate in Embedded Systems group in eInfochips Private Limited, Ahmedabad, Gujarat, India. He is involved in design and development of embedded systems using Texas Instrument's MSP430 series of controllers.

Mohanchur Sarkar received his B.Tech. degree in Computer Science and Technology from Kalyani University, India in 1999. He received his Ph.D. degree in Computer Science and Technology from Indian Institute of Technology (IIT, BHU), Varanasi, India in 2011. He joined Space Applications Centre (SAC), a branch of the Indian Space Research Organization (ISRO) in 2000. Currently he is Scientist/Engineer-SF in SATCOM and Navigation Applications Area at SAC, ISRO. At SAC, he is the focal person for the development of "IRNSS (Indian Regional Navigation Satellite System) Simulator" which mimics the IRNSS constellation. He has worked on Design, Simulation and Realization of Transport Protocols for Mobile Satellite Networks. His research interests include GNSS Simulation, Satellite Interactive Networks (DVB-RCS), Satellite based IP Broadcasting (DVB-S) Network, Satellite based Mobile TV (DVB-SH), Mobile Satellite Network Protocols, Transport Protocols over Satellite based Networks, Delay Tolerant Networks, Deep Space Communication Protocol and wireless sensor networks. He is a registered reviewer of the IEEE Transactions on Wireless Communication.

K. S. Dasgupta received his B.E. and M.E. degrees in Electronics and Telecommunication from Jadavpur University, India in 1972 and 1973 respectively. He received his Ph.D. degree in Computer Science from Indian Institute of Technology, Mumbai in 1990. He joined Space Applications Centre (SAC), a branch of Indian Space Research Organization (ISRO) in 1974. He was involved in the design and development of communication systems and payloads for geosynchronous satellite projects. He led a team that developed various SATCOM applications including, rural telephony and grid computation. He was the chief technical architect of the GSAT-4 regenerative payload. He served as the Director of Development and Educational Communication Unit (DECU), Ahmedabad, India before joining Indian Institute of Space Science and Technology (IIST), Thiruvananthapuram, India. Currently he is Director, IIST since December 2010. He is a recipient of the ISRO performance excellence award for the year 2009. His research interests include Digital Signal Processing, Digital Image Processing, Computer Architecture, Digital Communication and Onboard Processing Satellites.