DRMACSN: New MAC protocol for Wireless Sensor Networks

T. Muddenahalli N and F Takawira, member, IEEE
School of Electrical, Electronics and Computer engineering, University of KwaZulu-Natal,
DURBAN, South Africa,
Email:{muddenahalli, ftakaw }@ukzn.ac.za

Abstract — In this paper, we propose a new distributed, load balanced, and route aware medium access control protocol for Code Division Multiple Access (CDMA) based Wireless sensor networks (DRMACSN). *Wireless Sensor networks (WSN)*, is new class of devices, have the potential to revolutionize the capturing, processing, and communication of critical data by first responders. Sensor networks consist of small, low-power, and low-cost devices with limited computational and wireless communication capabilities. They represent the next step in wireless communication's miniaturization, and their power and size make it feasible to embed them into wearable vital sign monitors, location-tracking tags in buildings, and first responder uniform gear. The sensor nodes can only transmit a finite number of bits before they run out of energy. Thus, reducing the energy consumption per bit for end-to-end data transmission is an important design consideration for such networks. We show how our design would improve the MAC protocol with sleep and wakeup schemes that could achieve greater energy savings, reliability and increased system capacity. In this paper, we present findings for the delay, and the throughput of the proposed MAC protocol.

Index Terms— Medium Access Control (MAC), the Time Division Multiple Access (TDMA), Code Division Multiple Access (CDMA), Channel Quality Indicator (CQI), Ready to Receive (RTR), Multiple Access Interference (MAI), Clear Channel Assessment (CCA).

1. INTRODUCTION

Wireless sensor networks have the potential of revolutionizing the way wireless technology has been used over the last decade. Wireless technology has been traditionally used to connect people to each other or to other devices. Cellular systems, wireless local area networks and broadband wireless access aim to provide voice and data communication. Advances in hardware and wireless network technologies have placed us at the doorstep of a new era, where small wireless devices will provide access to information anywhere as well as actively participate in creating smart environments [1]. Recent developments in sensor technology and low power radios have enabled the widespread deployment of sensor networks consisting of small sensor nodes with sensing, computation, and communication and actuation capabilities. Each node has one or more sensors, embedded processors and low-power radios, and is normally battery operated. Typically, these nodes coordinate to perform a common task [2].

The MAC is an important technique that enables the successful operation of a communication network [3]. The important task of the MAC protocol is to avoid collisions by coordinating two interfering nodes that they do not transmit at the same time. There are many MAC protocols that have been developed for wireless voice and data communication networks. Typical examples are the contention based, random based, the time division multiple access (TDMA), CDMA and hybrid based protocols.

We have considered the following attributes in order to design an improved MAC protocol for the wireless sensor networks. The first is the energy efficiency**:** the Sensor nodes are expected to operate on battery power for several years. Therefore, the energy conservation scheme plays a key role in determining the lifetime of the sensor network. Prolonging network lifetime for these nodes is a critical issue [3]. Another important attribute is the scalability to the change in network size, node density and topology. Some sensor nodes may die over time; some new sensor nodes may join later; some sensor nodes may move to different locations. These reasons lead to change in the network topology over time. A good MAC protocol should be able to accommodate such network changes. Other important attributes include fairness, delay, and throughput.

2. RELATED WORK

The S-MAC [3], used the IEEE 802.11 CSMA/CA approach to avoid collision and puts a node to sleep when a neighbor node is transmitting to avoid overhearing. S-MAC uses three novel techniques to reduce energy consumption and support self-configuration. To reduce energy consumption while listening to an idle channel, nodes periodically sleep. Neighboring nodes form virtual clusters to auto-synchronize on sleep schedules. S-MAC also sets the radio to sleep during transmissions of other nodes. The S-MAC protocol essentially trades energy for throughput and latency by utilizing the sleep mode of the radio. A scheduled periodic sleep and listening pattern is used to decrease the idle energy consumption. S-MAC uses the transmission time embedded in RTS/CTS to turn off unintended receivers to avoid the energy consumption caused by overhearing. In [4], the authors proposed a MAC protocol called TA-MAC to modify the contention window mechanism of S-MAC. It employs a fast backoff scheme to reduce the time of idle listening during back off procedure. In B-MAC[5], a carrier sense media access protocol for wireless sensor networks is proposed that provides a flexible interface to obtain ultra low power operation, effective collision avoidance, and high channel utilization. To achieve low power operation, B-MAC employs an adaptive preamble sampling scheme to reduce duty cycle and minimize idle listening. The B-MAC protocol supports on-the-fly reconfiguration and provides bi-directional interfaces for system services to optimize performance, whether it is for throughput, latency, or power conservation. The B-MAC protocol outperforms S-MAC protocol for traffic broadcasting due to more sophisticated CCA and shorter preamble overhead and if the application's required latency is relaxed, S-MAC could achieve lower energy consumption than B-MAC [5]. In X-MAC [6], a short preamble approach is used to reduce energy usage at both the transmitter and receiver and it also reduces per-hop latency, and offers additional advantages such as flexible adaptation to both bursts and periodic sensor data sources. In [7], the authors designed a self-organizing, location ware MAC protocol for DS-CDMA based sensor networks. This protocol tries to conserve energy by deleting the redundant nodes in the neighborhood. In [8], the authors proposed a CDMA based power controlled MAC protocol for mobile ad hoc networks. This protocol improves the network throughput and to minimize the energy consumption.

In this paper, we present a new MAC protocol with the following goals: distributed, route aware to reduce energy consumption, a protocol that has good collision avoidance ability and reduced idle listening time.

3. PROPOSED MAC PROTOCOL DESCRIPTION

We use the spread slotted aloha discussed in [9, 10] as a channel access scheme. Each sensor node is assigned with the receiver based code [11, 12] for the Preamble, the Ready to Receive **(**RTR), and the *DATA* transfer purposes. The sensor node also use a common code [11,12] for broadcasting the RTR message, the *data_send* (ds) and the sleep sensing signal to neighbor nodes.

In the proposed MAC protocol, the time slots are divided into frames that are made up of a preamble slot and a DATA slot. The nodes compete for empty mini slots in a frame according to the spread slotted aloha principle.

Data massages are fixed size in length and are packetized with the fixed length packet size, *M* bits. Messages with multiple packets are transmitted over several continuous slots. When a sensor node needs to transmit a packet, each bit is spread using a receiver based code [11, 12]. We assume that all transmissions (Preamble, RTR, Busy tone/Datasend and DATA) begin at the beginning of the new frame only.

3.1Packet Formats

The following are the packet formats used in the protocol:

a. Preamble:

This packet has four entries: source node id, a total data to be transmitted, and the battery power level at which preamble is being transmitted at the transmitting node.

b. Ready to receive (RTR):

This packet has three entries: node id, Channel quality indicator (CQI), and Receiver code.

c. **Data**:

This packet contains data block, the receiver code, sequence number, more bit flag.

d. Acknowledgement(ACK):

This packet contains the Node Identifier (ID), CQI, battery power level that is remaining.

e. Neighbor node table:

This table is used to maintain the number of neighbors of the node. This table has three fields: ID's of nodes, Simultaneous transmissions (S_T) value, Code.

3.2 Protocol phases:

We assumed that each node can exist in one of three states: Sleep, Wakeup or Active states as depicted in Figure 1.

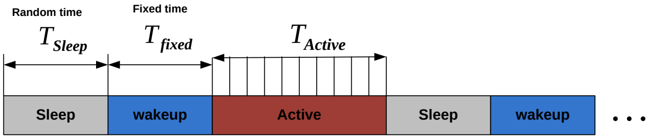


Figure 1: Protocol Phases for a node

We assume that a route from the source node to the destination node has already been established using the routing protocol [13]. For example, in the Figure 2, Nodes A, B, and C are in the routing path and Nodes E-M are outside the routing path and do not participate in the communication. Nodes exchange their neighborhood information by broadcasting table information to all its immediate neighbors. This ensures that all neighboring nodes have the updated value about the current simultaneous transmissions.

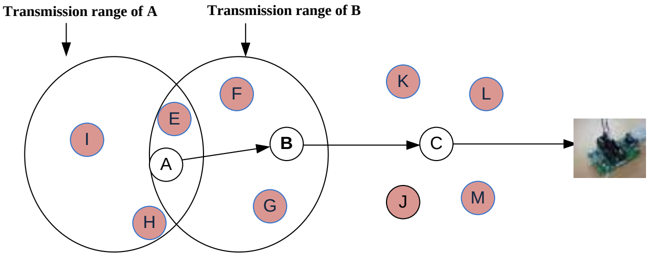


Figure 2: Path from Node A to a Sink.

A node goes into sleep state when it does not have data to transmit in its buffer when there are no transmissions from other nodes or when there is no reply from any one of its neighboring nodes. An individual node will sleep for a random time interval, $T_{Sleep} = [0, \cdots, T_{max}^{Sleep}]$, the duration of which is geometrically distributed with mean $\frac{1}{T_{Sleep}}$.

In the wakeup state, a node enters *send preamble mode* when a node generates data or has data in the buffer. Each node manages two types of packets in the buffer. One of these, called the source queue, stores packets coming from its own packet generator. The other one, called the relay queue, stores the incoming packets that have to be relayed.

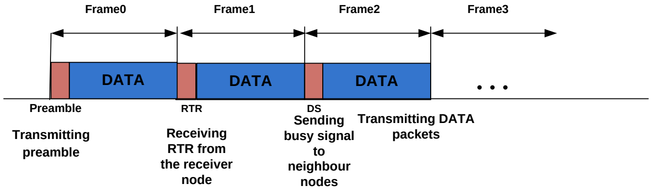


Figure 3: A basic MAC protocol scheme

Node, A, sends a preamble to the neighboring Node B, on B receiver code at the beginning of the new frame (i.e., Frame 0 in Figure 2). The sender node waits for the RTR packet in the wakeup state. Upon receiving the preamble, the neighbour Node, B, prepares *RTR* packet, which includes a CQI value (Initially a initial CQI value will be transmitted. For subsequent transmissions, the receiver nodes would estimate using the method described in section 3.3), code on A's receiver code at the beginning of the next new frame (i.e., Frame 1 in Figure 3).

Before sending this RTR packet, the received neighbor node first checks the state of the network by checking the following two threshold conditions:

Threshold T1: It checks whether the number of simultaneous transmissions (ongoing transmissions) exceed the blocking threshold α, $S_T < \alpha$.

Threshold T_2: It also checks whether the number of simultaneous transmissions β, S_T exceed the channel load, $S_T < \beta$.

If both conditions are satisfied, the received neighbour node sends back the RTR packet which also contains the receiver node ID, and the CQI value to the sender node on A's receiver code as shown in Figure 4a and updates its $\boldsymbol{S_T}$ value in its neighbour node table. This node also broadcasts a RTR to all of its neighbour nodes using a common code.

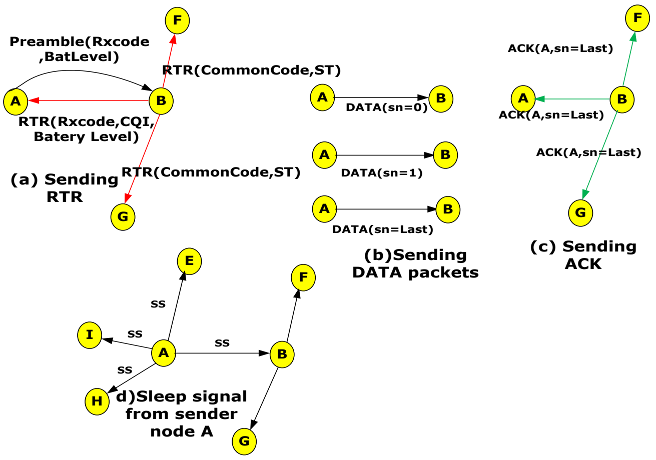


Figure 4: a) Sending RTR b) Data Transmission c) Sending ACK d) Sending sleep signal to neighbor nodes

Upon receiving the RTR, every other neighbour node (E, F and G) updates the number of simultaneous transmissions, S_T in their neighbour node table.
If there no reply from a neighbor node within its slot time interval,T_i, or there are no preamble transmissions from any of its neighbor nodes, the sender node sets its random back off timer and goes into back off state.

On the other hand, if a receiver sends the RTR to a sender and doesn't receive any packet from the sender within $\boldsymbol{T_i}$ the timer expires. When the timer expires, the communication is released, and the receiver will enter the wakeup state to receive preamble requests from the other neighbor nodes within slot time interval,T_i.
Upon receiving the ready to receive signal message, the sender node, A, on the other hand, prepares its data packets for transmission using node B's receiver code and correspondingly updates its S_T value in the neighbour node table.

In the proposed MAC, long data messages are fragmented into many small fragments, and our protocol would transmit them in a burst as in SMAC [3] except that our protocol continues to send the packets without waiting for the ACK signal after every fragment transmissions(refer Figure 5).

The sender and the receiver would reserve the medium for the entire transmission of all the fragments. The sender node sends all the data packets and waits for the ACK back from the receiver node as shown in Figure 4 c).

When the receiving node receives data in full form (received without errors), it sends acknowledgement back to the sender node and a CQI. It then continues to be in the listen/wakeup state for further transmissions from the same sensor node or from any of the other neighbor nodes otherwise it sets random timer, goes to sleep mode and this node also updates (decrement by one) the values of S_T in its neighbour node table. The sleep signal (SS) is also sent to all neighboring nodes (E, F and G) so that these nodes would update (decrement by one) the values of S_T in its neighbour node table.

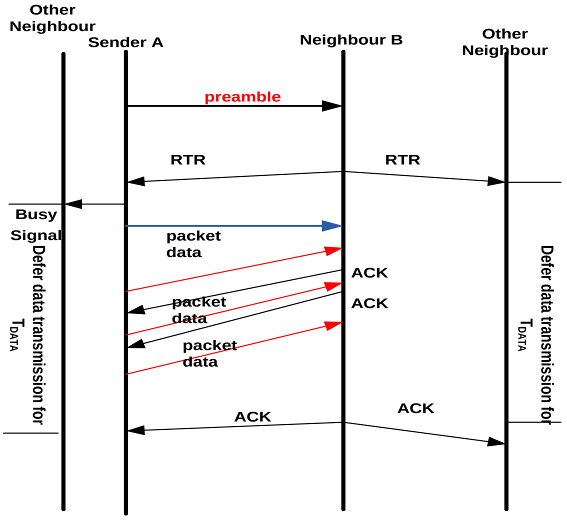


Figure 5: Successful transmission between Nodes A and B

After finishing the current data packet transmission, the sender node still has data packets in its buffer. The sender node however checks its battery power level in order to continue with further transmissions. If the battery power level is sufficient enough, it continues to be in the active mode to continue its data transmission using the same receiver code.

The data transmission will follow the above discussed procedure. Otherwise it checks for the transmission request from any of its neighbor nodes. Only if there no transmission request during this time interval from any of its neighbor nodes; the sleep signal (SS) is sent to all of its neighbor nodes (B, E, H and I), so that these nodes would update (decrement by one) the value of number of simultaneous transmissions, S_T in their neighbour node table. It goes to sleep mode by setting its timer as shown in Figure 4 d).

The nodes that are blocked or backed-off compete for the available slot using slotted spread aloha after a random back off timer expires. If a slot is available, it starts sending the preamble to its neighbour nodes. If slots are not available, it checks for the preamble transmissions from any of its neighbor nodes. Otherwise it enters again into back off state and sets its random back off timer. After the random back off timer expires, if there is still no request or availability of slots after two retries, it gives up and go into sleep mode.

For those messages that are corrupted due to CDMA multiple access interference (MAI), or if errors are detected in the received packet, the received node generates a negative acknowledgement (NACK) as shown in Figure 6 and sends it to the sender, over the reverse channel, as a retransmission request. In our MAC, we retransmit only that fragment using selective repeat automatic repeat request (SR-ARQ). Upon reception of the NACK, the transmitter sends again the packet to the receiver. Retransmissions can continue until the packet is correctly received or that the maximum number of allowed retransmissions is attained; in the latter case , the packet is considered lost.

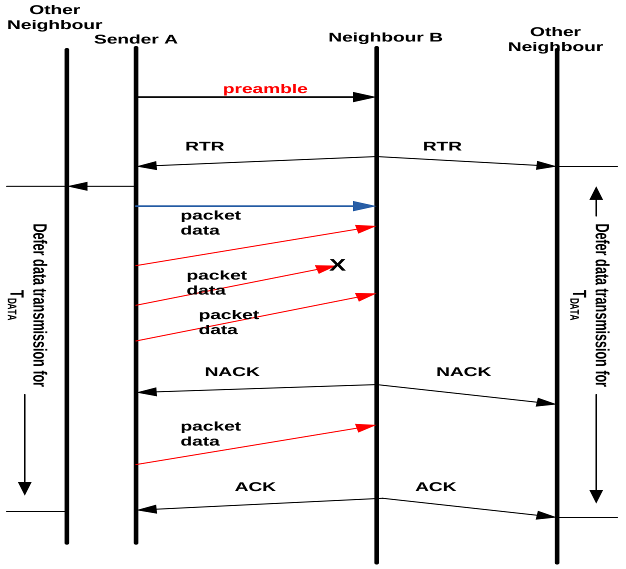


Figure 6: Unsuccessful transmission between Nodes A and B

3.3 Estimation of Channel Quality Indicator values

A CQI value is basically a 5-bit feedback value from the receiving sensor node to the transmitting sensor node. Each CQI value represents a specific combination of the number of codes; modulation type and transport block size. The range of values of the CQI index is from 0 to 31, where each value indicates the maximum transport block size that can be correctly received with 90% probability [14]. We can compute the Signal to Noise Ratio (SNR) from the received signal [15],

$$SNR = \frac{P_{rx}}{I_{Inter} + I_{Extra} + N} * \text{PG} \qquad (1)$$

where P_{rx} received power is level of the sender node, I_{Inter} is the intra-cell interference, I_{Extra} is the extra-cell interference, N is the noise power , $PG = W/R_b$ where W is the spreading bandwidth and R_b is the bit-rate.
Furthermore, from the measured value of SNR the CQI for the channel is computed using the following formula [14]:

$$CQI = \begin{cases} 0 & if\ SNR \le -16 \\ \left\lfloor \frac{SNR}{1.02} + 16.61 \right\rfloor & if -16 \le SNR \le 14 \\ 30 & if\ 14 \le SNR \end{cases} \qquad (2)$$

3.3 Parallel transmission on a slot

Proposed MAC protocol allows multiple reservations for preamble minis lot (based on availability of minis lots) on the same slot using spread slotted aloha. The receive node indeed maintains a list of senders that managed a successful reservation and will poll them in the successive frames. This feature is illustrated in Figure 7 that shows two successive reservations on the same slot *i*.

a. In frame j, the PREAMBLE dialogue starts the connection between nodes 0 and 3:
b. Node 0 sends a PREAMBLE request with its destination node address and battery level. Node 3 sends back the RTR. The RTR is also received by Nodes 1 and 4, that is now they are aware of a communication on slot i with Node 3 as receiver.
c. During the data phase, Node 0 is allowed to transmit the first packet and sequence number (sn) 0.
d. In frame j+1, Node 1 establishes a connection with Node 3 or its neighbor Node 2. Node 1 sends preamble request. Node 3 or the immediate neighbor of Node 2 will send a RTR; Node 3 acknowledges the packet transmitted by Node 0 in frame j, and either Node 1 or Node 3 continues to receive data packets from Node 1.
e. In frame j+2, Node 3 now polls Node 0. With the RTR, it also acknowledges the data packet of Node 1 with sequence number 0.
f. In frame j+3, Node 3 or Node 2 polls Node 1 and acknowledges the data packet of Node 0 with sequence number 1.

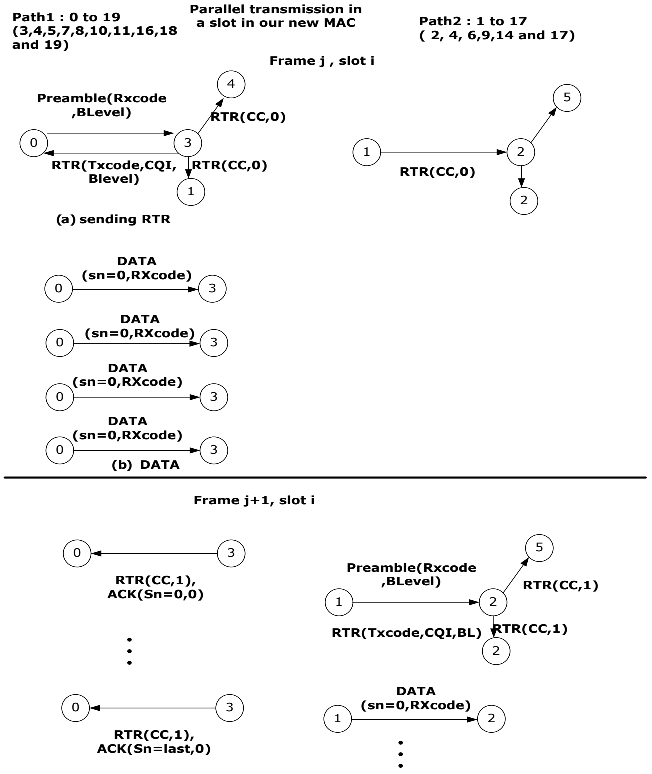


Figure 7: Parallel Transmissions on a Slot.

4. SIMULATION RESULTS

We created an event based simulator in VC++ 2008 to simulate the MAC protocol. The parameters used in the simulation are shown in Table 1. We assumed 100 sensor nodes are deployed randomly in a 1000mx1000m area as shown in the Figure 8 and also assumed that each node can reach all its neighbors within its radio range. The parameters to compute SNR are taken from [15]. We also assumed that a route from the source node to the destination node is known. Energy consumption is based on the amount of energy the sensor nodes use: 79 mA while transmitting, 49 mA while receiving, 50 μA while sleeping and 3 mA while idle.

Table 1: Simulation parameters

Parameters	Values
Number of sensor nodes	100
Spreading factor	16
Link rate	Variable depending on the CQI
Message Length	1024 bits
Packet size	128 bits
Average Message Length	4 packets
Preamble Length	10 bytes
RTR	14 bytes
DATA	128 bytes
ACK	10 byte
Number of neighbours	Varying

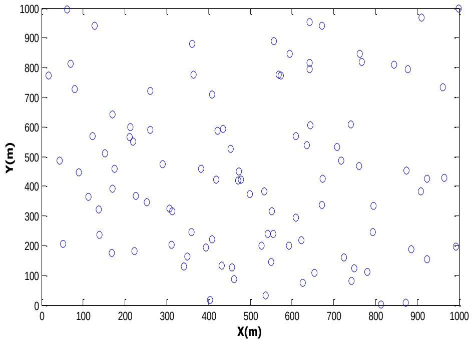


Figure 8: Sensor nodes randomly deployed in 1000x1000 m^2 area.

We calculated neighbors for each node. Figure 9 displays list of the neighbor nodes generated for each node. We have generated non-redundant nodes in the neighbor list to conserve energy.

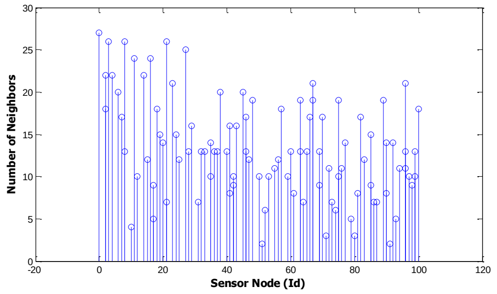


Figure 9: Histogram of the generated neighbor list for each node.

We have considered the battery level for each node for the simulation purpose. Figure 10 shows the initial battery level each node in the system.

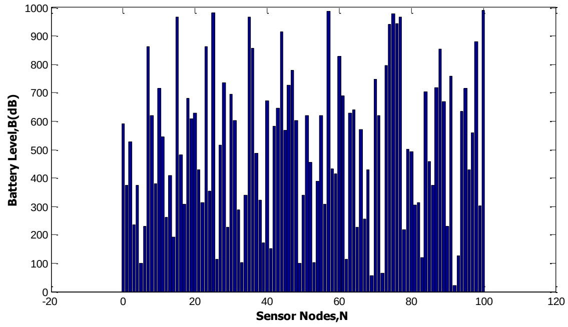


Figure 9: Histogram of the battery level of nodes in the system

We have generated some data packets for the simulation and Figure 10 shows the packet arrival per slot.

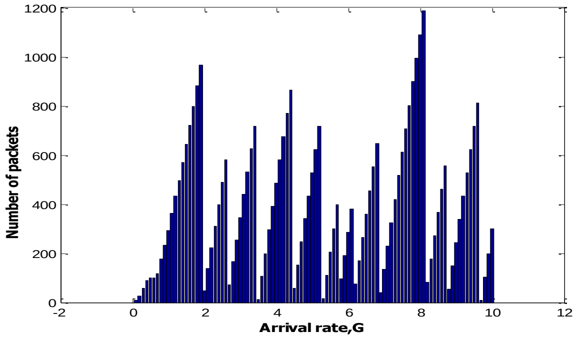
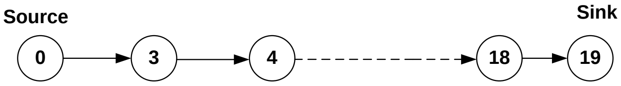


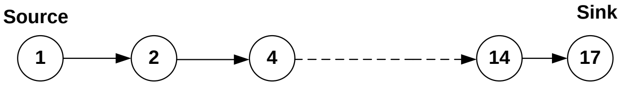
Figure 10: Histogram of the packet arrivals per slot.

We assume that our routing information is available to our MAC protocol. Based on this assumption, we generated a routing table which contains path information after calculating the neighbor nodes for each node in the network. For example: the neighbor nodes for Node 0 are: 3,4,5,7,8,10,11,16,18 and 19.

We assumed Node 0 as the source node and Node 19 as its destination in the Path1.



Similarly, neighbor nodes for Node 1 are: 2, 4, 6,9,14 and 17, these nodes will be there in second path 2.



Throughput: The throughput is defined as the expected value of the number of successful packets transmitted in a slot. The input of the channel contains newly generated packets at rate λ packets per slot. We assume that the new and retransmitted packets are Poisson distributed, the offered load is thus Poisson distributed with rate, $G = \lambda * s,$ where s is the slot time.

We can compute Throughput S as,

$$S = \frac{Number\ \ of\ a\ pckets\ transmitted}{Total\ \ Time\ taken\ to\ transmit}\ \text{per slot.}$$

Figure 11 illustrate average throughput versus offered load G, for various values of blocking threshold $\propto$.

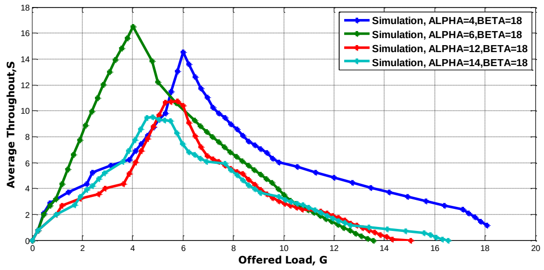


Figure11: Effect of blocking threshold, $\propto = 4,6,12,$ and 14, on the average throughput, $\beta = 18$.

Figure 12 shows the average delay versus offered load. We plot for $\propto$ = 4, and 14 to check the average delay when the load changes.

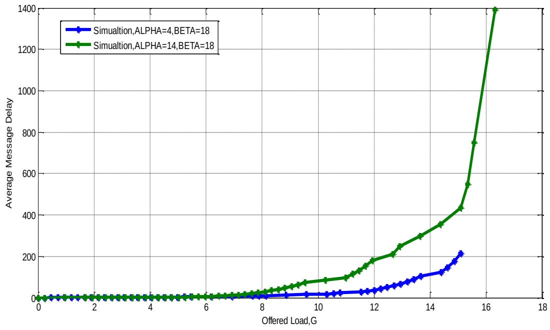


Figure12: Effect of varying blocking threshold, $\propto = 4$ and 14, $\beta = 18$, on the average delay.

5. CONCLUSIONS

In this paper, we have presented a new MAC protocol based on CDMA for wireless sensor networks. An interesting property of our protocol is that the received sensor nodes check the channel overload condition before proceeding to send RTR packets after receiving a preamble from the sender sensor node. This improves in the probability of success of those nodes transmitting before the overload condition.

Future work includes analyzing the protocol performance, and more tests will be done to check the performance of the MAC with different number of nodes and system complexity.

6. REFERENCES

[1] F. Akyildiz, W. Su, Y. Sankara Subramanium and E. Cayirci,"A survey on Sensor Networks", IEEE Communication Magazine, Vol.40(8), pages. 102-114, August 2002.

[2] V. Raghunathan, C. Schurgers, Park.S, and M.B. Srivastava, "Energy-aware wireless micro sensor networks," IEEE Signal Processing Magazine, Vol.19 Issue: 2, Pages. 40 –50, March 2002.

[3] W. Ye, J. Heinemann, and D. Estrin, "An energy efficient MAC protocol for Wireless sensor networks", in Proceedings of the IEEE INFOCOM, New York, June 2002.

[4] H. Gong, M. Liu, Y. Mao, L. Chen, and L. Xie, "Traffic Adaptive MAC Protocol for Wireless Sensor Network", ICCNMC 2005, LNCS 3619, Pages.1134 – 1143, 2005.

[5] J. Polastre, J.Hill and D.Culler, "Versatile Low Power Media Acess for Wireless Sensor Networks", In Proceeding of the 2nd ACM Conference on Embedded Networked Sensor Systems (SenSys), November 3-5,2004.

[6] M. Buettner, G. Yee, E. Anderson, R. Han, "X-MAC: A Short Preamble MAC Protocol for Duty-Cycled Wireless Sensor Networks",4th ACM Conference on Embedded Sensor Systems (SenSys), pages. 307-320,2006.

[7] B. H. Liu, N. Balusu, H. Pham, and S. Jha, "CSMAC: A novel DS-CDMA based MAC protocol for wireless sensor networks**"**, Proceedings of IEEE Global Telecommunications Conference workshop, USA, pages.33-38, 2004.

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

[9] G. Judge and F. Takawira, "Spread-spectrum CDMA packet radio MAC protocol using channel overloads detection and blocking", Wireless Networks, Vol.6, pages. 467–479, 2000.

[10] K. Toshimitsu, K. Yamazato, T. Katayama, M. Ogawa, A. Ogawa, "Novel spread slotted aloha system with channel load sensing protocol", IEEE Journal on Selected Areas in communications, Vol. 12, No. 4, pages. 665-672, May 1994.

[11] L. Hu, "Distributed CDMA code assignment for CDMA Packet Radio Networks', IEEE/ACM Transactions on Networking, Vol 1, No 6, December 1993.

[12] H. X. Tan, W. K Seah and K. M. Chan, "Distributed CDMA code assignment for wireless sensor networks", Proceedings of the IEEE Radio and Wireless Symposium (RWS 2006), San Diego, CA, Jan 17-19, 2006.

[13] D.Jonhnson, Y.Hu and D.Maltz, "The dynamic Source Routing Protocol for Mobile Ad Hoc networks",http:/www.ietf.org/internet-drafs/drafs-ietf-manetdsr09.txt, IETF internet draft, April 2003.

[14] A. Haider and R. Harris," A Novel Proportional Fair Scheduling Algorithm for HSDPA in UMTS Networks", the 2nd International Conference on Wireless Broadband and Ultra Wideband Communications (AusWireless 2007).

[15] F. D. Angelis, I. Habit, G.Giambene and S. Giannetti, "Scheduling for Differentiated Traffic Types in HSDPA Cellular Systems", Global Telecommunications Conference, 2005.

T. Muddenahalli N received a B.Eng (Computer Science and Engineering) degree from Kuvempu University, India in 1995 and an M.Tech (CS&E) degree from VTU, India in 2001. He is currently a LECTURER in the School of E, E and Computer Engineering at UKZN, and is pursuing a Ph.D. in Electronic Engineering at the same university.

Fambirai Takawira received the B.Sc. degree (electrical engineering) with First Class Honours from Manchester University, England, in 1981 and the Ph.D. degree in 1984 from Cambridge University, England. He is currently a Professor in the School of E,E and Computer Engineering at the University of KwaZulu-Natal, Durban, South Africa.