2009 IEEE International Advance Computing Conference (IACC 2009)
Patiala, India, 6–7 March 2009

An Energy Efficient Congestion Control Protocol for Wireless Sensor Networks

V.S.Felix Enigo
Assistant Professor, CSE Department
SSN College of Engineering, Anna University
Chennai, Tamilnadu, India
felixphd@yahoo.com

Dr. V. Ramachandran
Vice Chancellor
Anna University
Trichy, India

***Abstract*— An efficient reliable sensor-to-sink data transport protocol will ensure that the sink can collect enough information and minimize energy consumption of data transport. It should be designed to adjust the reporting rates of sources and adapting to wireless communication conditions.**

We design a congestion control mechanism at the source which reacts based on the sum of the node weights at each node. In this scheme, each node passes its calculated weight upstream. Each node adds its current weight to that it received from a downstream node, and passes this information toward the upstream node. At the end, the source will receive the sum of all weight information from the corresponding downstream nodes and use the it for controlling rates. Each sensor node transmits the data with the adjusted rate. The sink node receives the time series for each sensor node. After collecting enough data, the sink node uses a clustering algorithm to partition sensor nodes according to the sending rates and similarity of data obtained. Then it sends out the cluster information to all sensor nodes and requires the sensor nodes within the same cluster to work alternatively to save energy. The nodes within a cluster adaptively enters into energy saving mode according to a random schedule. By simulation results, we show that our protocol achieves congestion control along with energy saving.

Keywords- Sensor Networks; Congestion Control;

I. INTRODUCTION

The Technical development of minute, economical, low-power, distributed devices has lead to the veracity for local processing and wireless communication [1].. These types of devices are called as sensor nodes. Only a restricted amount of processing is allowed in every sensor node. By synchronizing these nodes with the information from a massive amount of other nodes, it can illustrate the potential to estimate a physical environment in great detail.. A sensor node coordinating to carry out a few definite actions is characterized by employing a collection of consequent sensor nodes. In contrast to the traditional networks, so as to accomplish its responsibilities, the sensor networks depend on dense deployment and co-ordination.

In the early days, the sensor networks comprised of a few number of sensor nodes wired to a central processing station.

Nonetheless, the focus on these days has shifted to wireless, distributed sensing nodes. Yet, it is important to know what is the necessity of distributed, wireless sensing? [1]. Provided that the accurate location of a particular phenomenon is unfamiliar, the distributed sensing permits closer placement to the phenomenon when compared with that a particular sensor. Moreover, in order to overcome ecological impediments like obstructions, line of sight constraints etc. in several cases multiple sensor nodes are necessary. The observed environment does not possess an offered infrastructure in the majority cases of energy or communication. Thus necessitating the sensor nodes to stay alive on minute, finite sources of energy and communicating through a wireless communication channel [1].

The distributed processing capability is possibly an added requisite in case of sensor networks, which is vital as communication is a primary consumer of energy. In case of the centralized system, additional energy depletion is caused owing to the necessity for a number of sensors to communicate over long distances. Processing as much information as achievable in the neighborhood would be an excellent thought as this reduces the total quantity of bits broadcasted.

A. Applications of sensor networks

Sensor networks are employed by various applications, for instance: Environmental monitoring is an application, in which monitoring of air, soil and water, condition based maintenance is dealt with. Habitat monitoring is another application, in which the plant and animal species population and behavior are determined. It is worthy to mention seismic detection, military surveillance, inventory tracking, smart spaces etc. Indeed the sensor networks have the potential to incredibly transform the complex physical system is understood and built owing to the enveloping character of micro-sensors [2].

In this paper, we design an energy efficient congestion control mechanism at the source which reacts based on the sum of the node weights at each node

The paper is organized as follows: Section 2 gives a detailed discussion about the design of transport protocol for wireless sensor networks. Section 3 presents our congestion control algorithm at the transport layer. Section 4 presents the energy minimization algorithm at the sink. Section 5 gives the related work done. Experimental results are given in Section 6 and Section 7 concludes the paper.

II. Design of Transport Protocol for WSN

A. Problem Statement

In traditional networks (e.g., IP networks), congestion control and transport reliability are often coupled into a single protocol solution (e.g., TCP). This approach, however, is not necessarily correct in the context of sensor networks. In sensor networks, the energy expenditure is more important than occasional data loss because of the natural redundancy inherent in disseminated sensor data. Depending on the application and the direction of the information flow, not all data packets require strict reliability. For example, applications that monitor the temperature of a certain geographic region can tolerate occasional packet loss. Therefore, the complex protocol machinery that would ensure the reliable delivery of data is not always needed. Due to this application-specific nature of sensor networks, we argue that there is a need for the separation of transport reliability and congestion control in sensor networks.

An efficient reliable sensor-to-sink data transport protocol will ensure that the sink can collect enough information and minimize energy consumption of data transport. It should be designed to adjust the reporting rates of sources and adapting to wireless communication conditions.

B. Requirements of a Transport Protocol for WSN

The Transport layer protocol should posses the following requirements in a wireless sensor network:

1) *Generic:* The transport layer protocol should be independent of the application, Network and MAC layer protocols to be applicable for several deployment scenarios.

2) *Heterogeneous data flow support:* The transport layer protocol should encompass a network that is supposed to deal with *Continuous* and event driven flows.

3) *Controlled variable reliability:* The transport layer protocol must accumulate energy at each node, so as to endure the loss of few packets in number of applications.

4) *Congestion detection and avoidance:* The protocol should sustain congestion detection and avoidance mechanism, so as to conserve energy by reducing packet retransmissions.

5) *Base station controlled network:* Base station acts as a focal point to perform Mainstream functionalities and computation intensive tasks, for the reason that sensor nodes are energy constrained and limited in computational capabilities..

6) *Scalability:* The protocol should be scalable, with the motive that Sensor networks will include large number of nodes for processing.

7) *Future enhancements and optimizations:* The protocol should be flexible for future optimizations, with the purpose to improve network performance and to provide support for novel applications.

C. Congestion Control at the Source

Congestion Control: Congestion is mostly originated at the base station of the sensor network, since all the other nodes in a network which transmit more number of packets get congregated at this point. Soaring data rates, abrupt burst of data and collisions are other reasons for congestion in sensor networks [13].Energy depletion occurs because of promoting the packets which are dropped by some nodes due to memory limitation. Congestion also increases latency. To reflect on these points, transport layer should be capable to sustain congestion detection and avoidance.

Transport layer protocol is to be designed, which endow with advantageous to support multiple applications in the same network, provide controlled variable reliability, deal with congestion issues, reduce latency and maximize throughput.

1) Causes of Loss:

1) Interference is the main cause of loss in a sensor network. This occurs because of the simultaneous transmission of packets, by number of nodes that are within a range of the transmitting nodes.

2) The overflowed packets in a queue have to be discarded, even if it is successfully received by a node. This also ends results with loss of packets in a sensor network.

2) Types of Congestion: Congestion in wireless networks differs slightly from that of the wired networks. This can be grouped into two types:

Type 1: In a particular area, the nodes within a specified range begin to transmit simultaneously, ensuing type (1) losses that leads to diminish in throughput of all nodes in the area.

Type 2: Within a particular node, the queue or buffer is used to hold packets to be transmitted. The excess of packets leads to overflow which is also the cause of type (2) losses.

III. Proposed Congestion Control Algorithm at Transport Layer

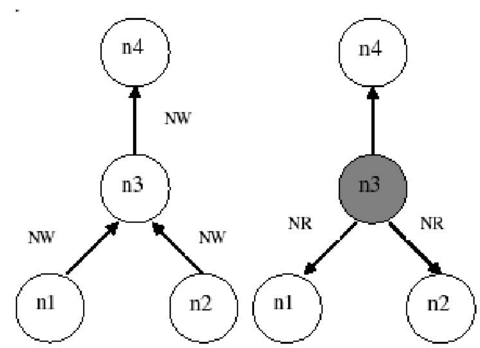

Figure 1. a) Node Weight Propagation b) New Rate propagation

In our protocol, every node in the path of the session executes a congestion control algorithm. The node weight at n1 and n2 is passed to the upstream node n3. When there is no congestion, n3 computes its local weight and adds it to the weight obtained from nodes n1 and n2 (Refer Figure1a). If there is a congestion at n3, a new transmission rate is calculated by n3 and its broadcast this new rate to its downstream nodes n1 and n2. On receiving this, n1 and n2 adjust their transmission rate based on it. (Refer Figure1b). The detailed algorithm is given in the next section.

A. Algorithm

1) Each node calculates its node weight Nw which is stored in a separate Node Weight Packet (NWP) of the following format.

BufSize
ChBusyRatio
HopCount
NodeWeight

Where BufSize is the buffer size, $ChBusyRatio$ (Rb) [16] is the channel busyness ratio which represents the interference level. It is defined as the ratio of time intervals when the channel is busy due to successful transmission or collision to the total time. We observe that the IEEE 802.11 is a CSMA-based MAC protocol, working on the physical and virtual carrier sensing mechanisms. Network allocation vector (NAV) specifies the status of the particular channel. It indicates the status as busy when the measuring node is sending or receiving packets, or else it is said to be in idle state. Now, the Node weight at n1 can be calculated by

$$Nw_{n_1} = (BSize_{n_1} * Rb_{n_1}) \quad (1)$$

Then, the NWP at n1 will be in the following form

$BSize_{n1}$
Rb_{n1}
Hc_{n1}
Nw_{n1}

2) EEDC adopts the method of explicit congestion notification with some modification. Each EEDC data packet has a *congestion notification* bit in its header. Every sensor node maintains two thresholds Bmax and CBmax in its buffer. When BufSize reaches Bmax, and $ChBusyRatio$ reaches CBmax, the node will set the congestion notification bit in every packet it forwards.

3) The NWP at n1 is passed to the upstream node n2, along with the data packet.

4) On receiving this NWP from n1, n2 will first check for the congestion notification bit. If it is not set, it will simply computes its node weight and adds it to the weight obtained from node n1 and passes on to the next upstream node n3, using the same method.

$$Nw_{n_2} = Nw_{n_1} + Nw_{n_2} \quad (2)$$

Then, the NWP at n2 will be in the following form

$BSize_{n1}$	$BSize_{n2}$
Rb_{n1}	Rb_{n2}
Hc_{n1}	Hc_{n2}
Nw_{n1}	Nw_{n2}

5) On the other hand, if the congestion bit is set, the node n2 calculates the new transmission rate based on the node weight as

$$N_{rate} = (A_{rate} / Hc) - \sum_{i=1}^{n} Rbi \quad (3)$$

Where A_{Rate} is the arrival rate of packets at node n2, which is given by.

$$A_{Rate} = NP / T \quad (4)$$

Here NP - is the number of packets received and T is the time of interval for the packet transmission.

6) Then, this Nrate value is propagated downstream towards the source nodes (n1), as an NACK packet.

7) On receiving this NACK packet, the downstream nodes will adjust their rate to Nrate, and retransmit the data again.

IV. Energy Minimization Algorithm at the Sink

A. Algorithm

1) Each sensor node transmits the data with the adjusted rate.

2) The sink node receives the time series for each sensor node.

3) After collecting enough data, the sink node uses a clustering algorithm to partition sensor nodes according to the sending rates and similarity of data obtained.

4) It is required that the sensor nodes within the same cluster, to work alternatively to save energy. So the sink broadcasts the scheduling & cluster information to all sensor nodes .The broadcast information includes:

- Cluster Id,
- Node Id,
- Sleep Wakeup time slots

5) On receiving the broadcast packet, the nodes within a cluster adaptively enters into energy saving mode according to the schedule.

The sensor nodes inside a single cluster include great resemblance in observations. Therefore, we have developed a dynamic clustering algorithm to form a set of disjoint clusters.

B. Clustering Sensor Nodes

The time series values of two sensor nodes n1 and n2 are said to be dissimilar, if (i) their magnitude are different (ii) if the distance between n1 and n2 are greater than a pre-defined threshold $D_{\max}$.

Based on the dissimilarity measure (DM) of the sensor nodes, we have to partition the nodes such that $DM < DM_{thr}$, where DM_{thr} is the dissimilarity threshold. For this, we need a clustering algorithm which minimizes the number of clusters to maximize the energy savings.

The clustering problem could be modeled as a clique-covering problem. We construct a graph G such that each sensor node is a vertex in the graph. An edge (u, v) is drawn if $DM_{(u,v)} <= DM_{thr}$. We can see that, a cluster is a clique in the graph. The clique-covering problem does not permit for constant approximation, since it is already been proved as a NP-complete one. We can use some greedy algorithm to obtain a rough approximation. The algorithm heuristically chooses cliques in such a way that it cover more vertices that have not been clustered previously. The search starts from the vertex with the largest degree until all vertices are covered. The output of this algorithm is a set of cliques that cover all vertices.

C. Random Scheduling Method for Each Cluster

To evaluate DM_{thr}, we need atleast M sample values. The time T required to collect M samples is divided into a sequence of N time slots. For each time slot, the sensor node is set out with the probability P in its exertion state. In order to obtain the equivalent detection delay for each cluster, P can be fixed as follows

$P = P_{\min}$ if Size (cluster) $> T_{size}$ and

$P = P_{\max}$ if Size (cluster) $< T_{size}$

Where T_{size} is the average size of the cluster.

Without commencing any communication overhead intended for synchronization, the workload allocation between the entire sensor nodes has been evenhanded by randomization. In addition, the exchange between energy saving with detection delay for cluster split could be simply controlled by modifying the value of M.

Each active sensor node collects latest M samples and transmit to the sink. Then for each cluster, the sink calculates DM_{thr} at the end of each time slot. If even a single active sensor node is located reporting an appreciably dissimilar information by the sink, the existing cluster must be split.

V. Related Work

Cheng Tien Ee and Ruzena Bajcsy [3] proposed a distributed and scalable algorithm that eliminates congestion within a sensor network, and that ensures the fair delivery of packets to a central node, or base station.

Joanna Kulik et al. [4] proposed a family of adaptive protocols, called SPIN (Sensor Protocols for Information via Negotiation), that efficiently disseminates information among sensors in an energy-constrained wireless sensor network. They have used meta-data negotiations to eliminate the transmission of redundant data throughout the network.

Chonggang Wang [5] proposed an overview for transport protocols for Wireless Sensor Networks (WSNs). They have highlighted the unique aspects in WSNs, and described the basic design criteria and challenges of transport protocols including energy-efficiency, quality of service, reliability, and congestion control.

Wei Ye et al. [6] proposed S-MAC a medium-access control (MAC) protocol designed for wireless sensor networks. They have applied message passing to reduce contention latency for sensor-network applications that require store-and forward processing as data move through the network.

Joseph Polastre [7] proposed B-MAC; a carrier sense media access protocol for wireless sensor networks that provides a flexible interface to obtain ultra low power operation, effective collision avoidance, and high channel utilization.

Tijs van Dam [8] described T-MAC, a contention-based Medium Access Control protocol for wireless sensor networks.

Yung Yi and Sanjay Shakkottai [9] proposed a congestion control over multihop wireless networks. They have developed a fair hop-by-hop congestion control algorithm with the MAC constraint being imposed in the form of a channel access time constraint, using an optimization-based framework.

Jeremy Elson and Deborah Estrin [10] described the synchronization requirements of future sensor networks and presented an implementation for low power synchronization scheme, post-facto synchronization.

Yogesh G. Iyer [12] proposed a transport layer protocol that can support multiple applications in the same network, provide controlled variable reliability, address congestion issues, reduce latency and maximize throughput.

Yogesh Sankarasubramaniam et al. [13] proposed a new reliable transport scheme for WSN and the event-to-sink reliable transport (ESRT) protocol. They have developed a ESRT solution o achieve reliable event detection in WSN with minimum energy expenditure.

Chieh-Yih Wan [14] proposed a new reliable delivery transport paradigm for sensor networks called Pump Slowly Fetch Quickly (PSFQ). But it didn't handle energy efficiency.

The design of an energy-efficient congestion control scheme for sensor networks called CODA was proposed in [15]. It is an end-to-end rate adjustment scheme based on queue length and channel status.

Since ESRT and CODA are end-to-end based, the congestion status of individual hops could not be estimated.

Yangfan Zhou et. al [17] proposed a Price-Oriented Reliable Transport protocol (PORT) . In this scheme, a node price is calculated for each node. A node's node price is the energy consumed by all the in-network nodes for each packet successfully delivered from the node to the sink. A weighted average is calculated which is the sum of the node price and link communication cost. This node price information is propagated towards the downstream nodes. Whenever congestion is detected, a portion of the traffic is shifted to the nodes with low node price.

Since the sink feedbacks the optimal sending rate to the source, it is a end-to-end reporting scheme. Also the node price calculation and link loss estimation involves high complexity and introduces additional delay.

VI. Experimental Results

A. Simulation Model and Parameters

We use Network Simulator (NS2) to simulate our proposed protocol. In our simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. We use the distributed coordination function (DCF) of IEEE 802.11 for wireless LANs as the MAC layer protocol. It is possessed with the functionality to notify the network layer about link breakage.

In our simulation, sensor nodes of sizes 25, 50, 75 and 100 deployed in a 1000 mr x 1000 m rectangular region for 50 seconds simulation time. All nodes have the same transmission range of 250 meters. The simulated traffic is Constant Bit Rate (CBR). For each scenario, ten runs with different random seeds were conducted and the results were averaged.

No. of Nodes	25,50,75 and 100
Area Size	1000 X 1000
Mac	802.11
Simulation Time	50 sec
Traffic Source	CBR
Packet Size	512
Transmit Power	0.360 w
Receiving Power	0.395 w
Idle Power	0.335 w

B. Performance Metrics

We compare the performance of our proposed EECC protocol with CODA and ESRT protocols. We evaluate mainly the performance according to the following metrics:

Aggregated Throughput: We measure aggregated throughput in terms of no. packets received.

Average Energy Consumption: The average energy consumed by the nodes in receiving and sending the packets are measured

Packet Delivery Ratio: It is the ratio of the fraction of packets received successfully and the total no. of packets sent.

The performance results are presented graphically in the next section.

C. Results

A. Varying No. Of Nodes

In the first experiment, we measure the performance of the protocols by varying the no. of nodes as 25, 50, 75 and 100.

Figure 2 shows that the average energy consumed by the nodes in receiving and sending the data. Since EECC make use of energy efficient scheduling, the values are considerably less in EECC when compared with CODA and ESRT.

From Figure 3, we can see that the packet delivery Ratio (PDR) for EECC increases, when compared to ESRT and CODA.

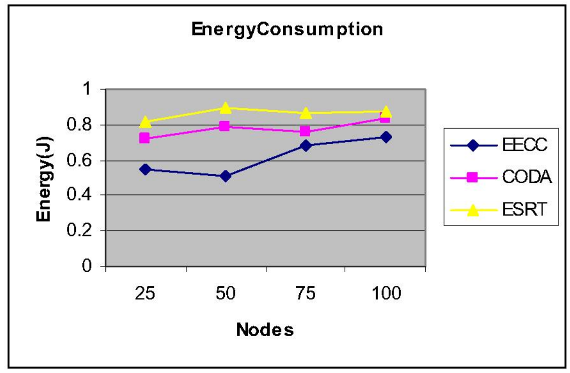

Figure2. Nodes Vs Energy

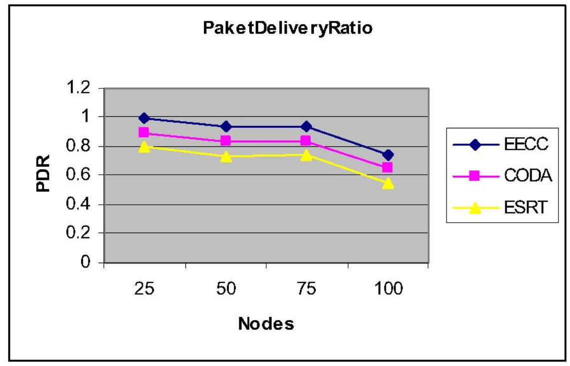

Figure 3. Nodes Vs Delivery Ratio

B. Varying the Transmission Rate

In the second experiment, we measure the performance of the protocols by varying the transmission rate as 250,500,750 and 1000 Kb.

Figure 4 shows that the average energy consumed by the nodes in receiving and sending the data. Since EECC make use of energy efficient scheduling, the values are considerably less in EECC when compared with CODA and ESRT.

From Figure 5, we can see that the packet delivery ratio (PDF) for EECC increases, when compared to CODA and ESRT.

When we measure the aggregated throughput for various rates, EECC throughput increases when compared to ESRT and CODA. Figure 6 shows this result.

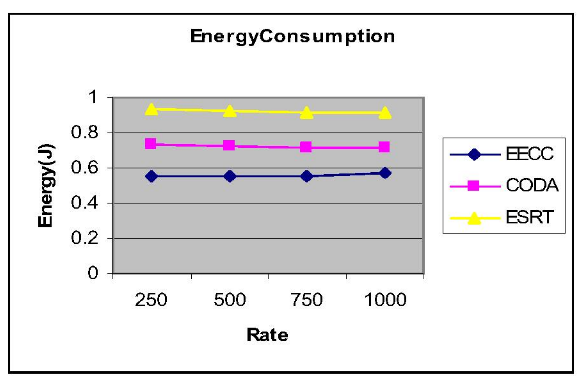

Figure 4. Rate Vs Energy

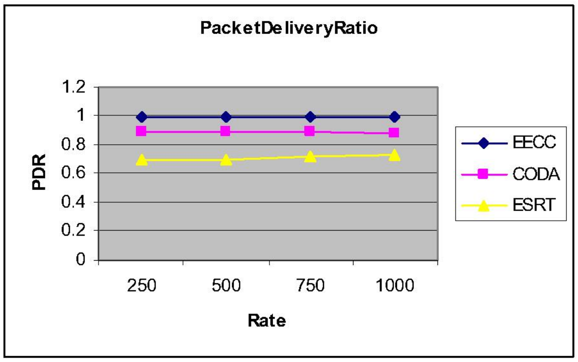

Figure5. Rate Vs Delivery Ratio

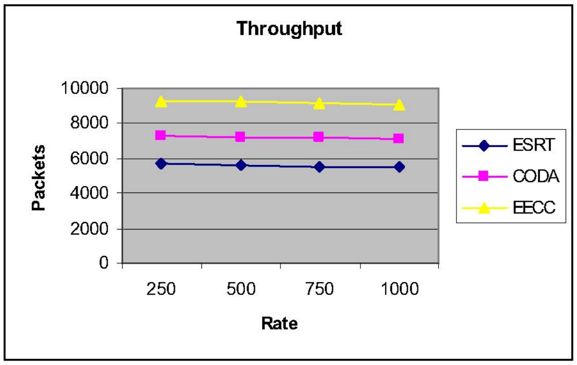

Figure 6. Rate Vs Throughput

VII. Conclusion and Future Work

In this paper, we have designed a congestion control mechanism at the source which reacts based on the sum of the node weights at each node. In this scheme, each node passes its calculated weight upstream. Each node adds its current weight to that it received from a downstream node, and passes this information toward the upstream node. At the end, the source will receive the sum of all weight information from the corresponding downstream nodes and use the it for controlling rates. Each sensor node transmits the data with the adjusted rate. The sink node receives the time series for each sensor node. After collecting enough data, the sink node uses a clustering algorithm to partition sensor nodes according to the sending rates and similarity of data obtained. Then it sends out the cluster information to all sensor nodes and requires the sensor nodes within the same cluster to work alternatively to save energy. The nodes within a cluster adaptively enters into energy saving mode according to a random schedule. By simulation results, we have shown that our protocol achieves congestion control along with energy saving.

Depending on the changes in time series reading of sensor nodes, the sink has to perform re-clustering. Also an efficient MAC protocol is needed to work in power saving mode. In our future work, we aim to solve these issues.

References

[1] Tubaishat, M. et al. "Sensor Networks: An Overview" Potentials, IEEE, April-May 2003.

[2] D. Estrin, R. et al. “Next Century Challenges: Scalable Coordination in Sensor Networks,” In Proceedings of the Fifth Annual International Conference on Mobile Computing and Networks (MobiCOM '99), August 1999, Seattle, Washington.

[3] Cheng Tien Ee, Ruzena Bajcsy, “Congestion Control and Fairness for ManytoOne Routing in Sensor Networks” in *SenSys'04,* November 35, 2004, Baltimore, Maryland, USA.

[4] W.R. Heinzelman et al. “Adaptive protocols for information dissemination in wireless sensor networks”. In Proc. of the 5th Annual International Conference on Mobile Computing and Networking (Mobicom 1999), pages 174{185, 1999.

[5] Chonggang Wang et al. "A survey of transport protocols for wireless sensor networks" Communications, Circuits and Systems, 2005. Proceedings. 2005 International Conference on May 2005.

[6] W.Ye, “An energy efficient mac protocol for wireless sensor networks”. In Proc. of the 21st International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002), pages 1567{1576. New York, June 2002.

[7] Joe Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In Proc. of Second ACM Conference on Embedded Networked Sensor Systems (SenSys 2004), pages 95{107. Baltimore, November 3-5 2004.

[8] T.V.Dam et al. “An adaptive energy-efficient mac protocol for wireless sensor networks”. In Proc. of First ACM Conference on Embedded Networked Sensor Systems (SenSys'03), pages 171{180. Los Angeles, November 5-7 2003.

[9] Yung Yi, and Sanjay Shakkottai , “Hop-by-Hop Congestion Control Over a Wireless Multi-Hop Network” in IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 15, NO. 1, FEBRUARY 2007

[10] Jeremy Elson. “Time synchronization in wireless sensor networks”. Ph.D. dissertation, May 2003.

[11] Tian He et al. “SPEED: A Stateless Protocol for Real-Time Communication in Sensor Networks”

[12] Yogesh G. Iyer et al. “STCP: A Generic Transport Layer Protocol for Wireless Sensor Networks”

[13] Yogesh Sankarasubramaniam et al., “ESRT: EventtoSink Reliable Transport in Wireless Sensor Networks”

[14] Chieh-Yih Wan, A Resilient Transport System for Wireless Sensor Networks, March 2005

[15] Chieh-Yih Wan, Shane B. Eisenman and Andrew T. Campbell, “CODA: Congestion Detection and Avoidance in Sensor Networks” in SenSys'03, November 5–7, 2003, Los Angeles,California,USA.

[16] H. Zhai, X. Chen, and Y. Fang, “How well can the IEEE 802.11 wireless LAN support quality of service? *IEEE Trans. Wireless Commun.*, vol. 4, no. 6, pp. 3084–3094, Nov. 2005.

[17] Yangfan Zhou et al. “PORT:Price-Oriented Reliable Transport protocol for wieless sensor networks” , November 10, 2005