Max-Min Fair Collision-Free Scheduling for Wireless Sensor Networks

Avinash Sridharan, and Bhaskar Krishnamachari
Department of Electrical Engineering-Systems,
University of Southern California,
Los Angeles, CA 90036, USA
{asridhar, bkrishna}@usc.edu, `http://ceng.usc.edu/~anrg`

Abstract—**When the data rates in sensor networks are comparable to the available channel bandwidth, traditional randomized access schemes face the problem of energy inefficiency and reduced throughput due to increased MAC collisions as well as the problem of unfair data delivery. We argue that under such conditions it is preferable to focus on techniques for scheduled access. We present a Linear Programming formulation and corresponding distributed TDMA-based scheduling algorithms to provide max-min fair collision-free bandwidth allocation to all sources. We evaluate the performance of the proposed scheduled flow technique using the Tossim/Nido network simulator for the Berkeley Mote/TinyOS platform. Our results show that under high data rate conditions, the proposed scheme significantly outperforms randomized access based schemes in terms of key metrics such as fairness, energy efficiency, throughput, and delay.**

Keywords: Sensor Networks, Medium Access, Fairness, Flow Scheduling.

I. INTRODUCTION

Since energy resources are scarce in sensor networks, it is desirable to avoid collision and congestion events that can result in wasted packet transmissions. In this paper we propose a max-min fair bandwidth allocation scheme for scheduling flows from multiple sensor sources to a sink, in conjunction with a scheduled TDMA-based MAC that provides collision and congestion avoidance.

Even though these networks may be idle during most of the operation time when nothing significant occurs in the environment, collisions are likely to be a serious issue in sensor networks when events occur that trigger increased sensing activity. Particularly in dense deployments, the data rates during events are likely to be of the same order as the available radio bandwidth. Even when individual reporting rates are low (e.g. 1 packet per second), at high densities, when there are several sources in the network, the combined source rates can easily add up to a value comparable to the available radio bandwidth.

Most of the prior work on medium access techniques for sensor networks has focused on random access schemes based on CSMA, e.g. the schemes proposed in [6] and the SMAC scheme [7]. However such random access schemes are ill-suited for conditions where data rates become comparable to the available radio bandwidth. This is illustrated through some sample simulation results that we present in table I for a representative CSMA protocol, in a scenario where the maximum radio bandwidth is 20kbps [1]. We find that even at moderate data rates, the performance of such arandom ized MAC scheme deteriorates with respect to significant metrics such as throughput, delay, energy efficiency and fairness. This is due to the increased levels of contention and collision experienced as the source rates increase with respect to the available bandwidth. This observation motivates our effort to examine alternativesolutions that would be suitable for the moderate-to-high data rate conditions that would be typical in large scale dense sensor networks during periods of increased sensor activity.

Data Rate	Low (.3 kbps)	High (~5 kbps)
Delivery Ratio	~1	0.06
Delay (ms)	6.9	1799
Fairness/Max-Min Ratio	~1	0.13
Energy (J)	10.86	42.9

TABLE I
ILLUSTRATION OF THE PERFORMANCE DEGRADATION OF A RANDOMIZED ACCESS SCHEME (CSMA WITH OVERHEARING AVOIDANCE) AT HIGH DATA RATES.

Under high-data rate conditions, feedback-based congestion control mechanisms are one way to improve delivery performance. Several recent studies have explored protocols that provide adaptive congestion control, e.g. SPEED [3], ESRT [4], and CODA [5]. What is common to these congestion control techniques that differentiates them from the work proposed in this paper is that they are all *reactive*, and do not assume that the underlying source-sink traffic has any regular/predictable structure. Further, they all assume an underlying MAC which is based on random access and therefore susceptible to collisions. TRAMA [8] is a scheduled access scheme for WSN and has some similarities to the work presented here. The key difference between TRAMA and our work is that we focus explicitly on the issue of *fairness* for long lived flows under high data rates in WSN and present the scheduled access scheme to achieve this.

In this paper, we focus on applications where the flows from sources to sink are relatively predictable and stable.

This work was supported by NSF under award number 0325875.

---

[1] The complete details of the simulation setup are described later in the paper.

Under these conditions, an alternative approach is structured communication involving bandwidth allocation and scheduled medium access. This shifts the emphasis from reactive congestion control to proactive congestion and collision avoidance. In allocating bandwidths to multiple sources, fairness is a key concern. We use the notion of max-min fairness as our metric in this study.

The rest of the paper is organized as follows. In section II, we formulate the problem of max-min fair flow scheduling in sensor networks as a linear program, taking into account interference between nearby nodes. In section II-B, we present a distributed algorithm for allocating the bandwidths in a max-min fair manner that assumes a collision-free MAC layer. In section II-C, we describe a distributed time-slot allocation scheme to implement the collision-free MAC. The performance of the proposed approach is evaluated with respect to key metrics such as throughput, energy efficiency, delay and fairness using Tossim/Nido simulations (based on the Mote/TinyOS platform) in section III. We present concluding comments in section IV.

II. MAX-MIN FAIRNESS IN SENSOR NETS

The basic notion of fairness is that of equality. However even if all the links are of equal capacity, the number of flows on different links might be different. In this case if we allocate equal bandwidth to all flows there might be links in which the bandwidth has not been utilized completely. Thus flows passing this link can potentially achieve higher bandwidth than other flows. A better metric therefore is the notion of max-min fairness, which is defined as follows. Let $R = R_1, R_2, R_3, \dots R_k$ be av ector that represents the rates allocated to the k sources, it is said to be a *max-min fair* solution if for all other possible allocations R', $\min(R) \geq \min(R')$, where $\min(R)$ represents the minimum rate allocated to any source in the allocation R.

Max-min fairness is a well-studied concept that has been extensively used to implement QoS through resource allocation in wired networks. However in the wireless domain there has been very little work in this field. The most important characteristic of a wireless network that makes it different from a wired network is interference. In a wireless network, because of the broadcast nature of radio communication, all flows within a given radius range share the same medium and even after using medium access schemes such as CSMA/CA (RTS/CTS mechanism in 802.11) they suffer from the exposed node problem. This is unlike wired networks in which flows on different links do not interfere with each other. Thus *while in wired networks the resource constraints correspond to the bandwidth available at each independent link, in wireless networks the resource constraints correspond to the receiver bandwidth at each node.*

In the following section we formulate max-min fairness as ali near programming problem for allocating bandwidths to well-defined source-to-sink flows in wireless sensor networks. We make the following assumptions:

- All nodes have the same maximum receiver bandwidth
- There are multiple source nodes each of which is generating independent, uncorrelated data to be routed to a common sink without aggregation.
- It is assumed that these flows are reasonably long-lived (e.g. involving at least hundred-plus packets per source).
- Since all flows are directed to a common sink, the nodes involved in generating and routing these flows together form a routing tree. The routing tree is assumed to be known (e.g. pre-determined through a routing algorithm such as directed diffusion) before the resource allocation is performed and does not change for the duration of the flows [2]
- The MAC is collision free. This implies that as long as the combined data rate of every transmitter in any portion of the wireless medium is within the available bandwidth range there should be no collision. We will justify this assumption by designing a collision-free TDMA-based MAC scheme in the following section.

Surprisingly, aside from [1] we're not aware of papers that discuss max-min fair allocation of resources in a wireless environment. Our work differs from [1] primarily in that [1] uses a token-based scheduling mechanism for arbitrary-pattern stochastic traffic and assumes the existence of some MAC-mechanism to handle interference without providing one. *Our focus in this paper is to develop a distributed max-min fair scheduling mechanism suited for continuous traffic on a data gathering tree (particularly suitable for sensor networks) and incorporate it with a time-slot based bandwidth allocation scheme to guarantee collision-free traffic.*

A. A linear programming approach to max-min fairness in sensor networks

For all nodes i on the routing tree formed by the paths connecting the various sources to the sink, we define the following variables: B_i is the maximum receiver bandwidth of node i, $B_{in,i}$ which is the sum of the data rates in-coming to $i$from its children; $B_{out,i}$ is the rate of data being sent by node i to its parent node; $B_{noise,i}$ is the sum of data rates from all nodes that are within interference range of i that are not its children. Further, for all *source nodes* i on that routing tree (which need not be all nodes on the tree), we define $B_{gen,i}$ as the rate of data generated by source i. Let N represent the set of all nodes in the routing tree, and let $S \in N$ represent the set of nodes on the routing tree that are source nodes. Then the max-min fair resource allocation problem for a sensor network data gathering tree can be formulated as follows:

$$\begin{array}{lll} \max & B_{min} & \\ \text{s.t.} & B_{in,i} + B_{out,i} + B_{noise,i} \leq B_i & \forall i \in N \\ & B_{out,i} = B_{in,i} + B_{gen,i} & \forall i \in S \\ & B_{out,i} = B_{in,i} & \forall i \in N \setminus S \\ & B_{gen,i} \geq B_{min} & \forall i \in S \end{array} \qquad (1)$$

---

[2] Otherwise the algorithms we propose will need to be refreshed each time the routing tree changes.

B. Distributed algorithm for max-min fair resource allocation

The above linear program can be solved in acom pletely distributed, iterative manner within the network by a very simple algorithm. At each round of this distributed algorithm, source nodes that are not constrained increase their generated data rate by a small incremental value ϵ. Nodes become constrained when the total bandwidth usage (the combination of incoming data, generated data as well as interfering data traffic) at those nodes is within ϵ of the total bandwidth available per-node. Further,(a) all nodes that are in the subtree below a constrained node become constrained and (b) all nodes whose output traffic interferes at a constraint node also become constraint (since they cannot increase their rates without causing constraint violations at the original constrained node). These subtree and interfering nodes are notified that they have become constrained through the propagation of constraint messages from the corresponding original constrained node. The algorithm terminates when all nodes on the routing tree become constrained. The rate available to all sources at this point is the allocated rate.

Theorem: The distributed algorithm described above terminates with an solution that is at most ϵ from the optimal in terms of the minimum bandwidth allocated, and it requires $O(\frac{1}{\epsilon})$ rounds to terminate.

Proof: We need only consider the first node i_1 that becomes constrained. Say this occurs at round k. At this point all source nodes on the routing tree have an equal source-bandwidth of $k\epsilon$. Since all source nodes in the subtree routed at i_1 will not increase their source-bandwidth beyond this point, and all other sources can never get less than this bandwidth (due to monotonic property of this algorithm) the minimum bandwidth in the final allocation produced by this algorithm is $k\epsilon$. There is no possible solution in which any of the source nodes in the subtree routed at i_1 could get a higher source bandwidth than $(k+1)\epsilon$ without either (i) violating the constraint at i_1, or (ii) lowering the bandwidth of data generated by one of the other source nodes in the network. Thus the solution provided by this algorithm is no more than ϵ away from the optimal solution. If B is the total bandwidth available at each node, then $k\epsilon \leq B$, hence the total number of rounds $k \leq \frac{B}{\epsilon}$ is $O(\frac{1}{\epsilon})$. $\square$

C. Distributed algorithm for time-slot allocation

While the above algorithm provides a bandwidth allocation to each source that is max-min fair, it assumes that there exists some "ideal" collision-free MAC which can provide this allocation exactly. We will now present a distributed time-slot allocation (i.e. TDMA) algorithm that provides a realizable bandwidth allocation that maintains the relative ratios of the bandwidths allocated to different sources (though as we shall see, inherent inefficiencies associated with time-slot allocation will not necessarily allow the ideal allocation vector obtained from the distributed MMF algorithm alone.)

The goal of the time slot allocation is to provide, in each frame, T_i slots for the data originating at each source $i \in S$, such that:

$$\frac{T_i}{\sum\limits_{j \in S} T_j} = \frac{B_{gen,i}}{\sum\limits_{j \in S} B_{gen,j}} \tag{2}$$

Say T_{total} is the total number of time slots allocated to all nodes on the routing tree by the time-slot allocation algorithm. Then the actual bandwidth experienced by a source node i in this time-slot allocation will be

$$\hat{B}_{gen,i} = \frac{T_i}{T_{total}} B_{eff} \tag{3}$$

where the effective bandwidth B_{eff} is determined by the packet size (in bits) and the size (in seconds) of each time slot: $B_{eff}= \frac{S_{pkt}}{t_{slot}}$. In general, due to inefficiencies associated with time slot allocation, it is the case that $T_{total} \geq \sum\limits_{j \in S} T_j$, and further depending the choice of the slot duration t_slot in general $B_{eff} \leq B_i$. Therefore:

$$\frac{\hat{B}_{gen,i}}{B_{gen,i}} = \frac{\sum\limits_{j \in S} T_j B_{eff}}{T_{total} B_i} \leq 1 \tag{4}$$

The time slot allocation algorithm must avoid the hidden-node terminal problem, by ensuring that a time slot allocated to a node does not collide with that allocated to a node that is 2 hops away that might interfere at the recipient of this node's communication. Also since wireless links are unreliable in nature the receivers will have to send ACKs to the sender. Thus the receivers need to ensure that the time slot in which the sender and receiver are exchanging any information, there is no neighbor of the receiver who is receiving data. We assume that at the start of the algorithm, the number of time slots required by each source for its own data is known to that source (obtained based on the max-min fair bandwidth allocation to maintain the same relative ratios as described above), and that each node on the routing tree knows/calculates the total number of slots it needs to send its own data as well as to forward the data of all nodes below it on the tree.

The time slot allocation algorithm proceeds as follows. All nodes in the tree are selected in turn using a breadth-first traversal starting from the root node to ensure that the required number of unique time slots are allocated to each of their children. At each step, for each child, the parent node cycles through all possible slots sequentially, first asking all its neighbors if any one of them has been allocated this time slot or are receiving data from their children in this time slot. It then asks the child node to whom the time slot is being allocated to check with its one hop neighbors if this time slot has been allocated to them. If the child node also verifies that this time slot is free, then the channel is allocated. If the time slot is occupied, it increments and moves to the next slot number. Using the above step a node allocates the desired number of time slots for all its children. At the end of the algorithm all nodes in the network have chosen time slots needed for their transmissions and the highest time slot number is made known to the sink which broadcasts it to all nodes on the routing tree. This suffices for all nodes to know the times when they should transmit. An additional coordination step

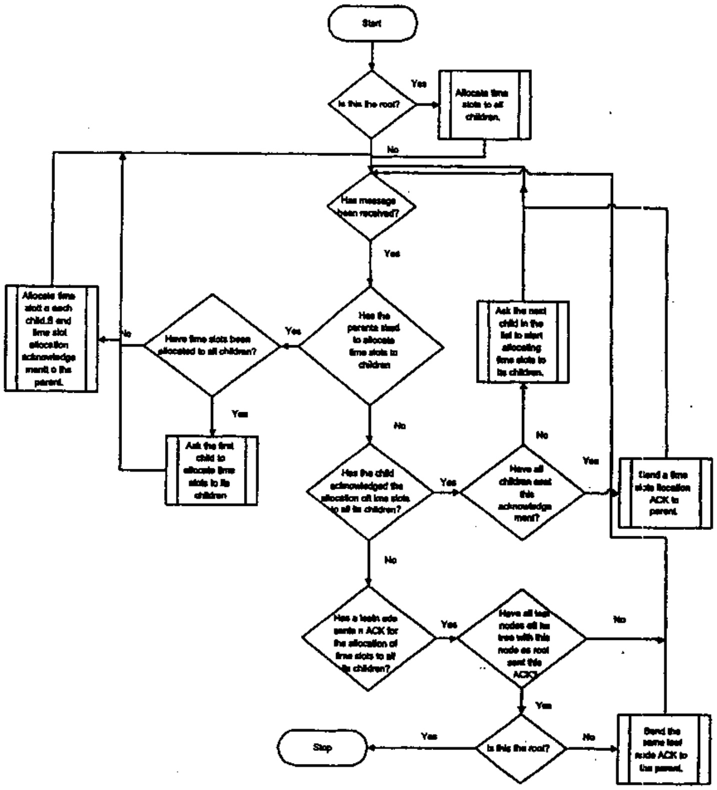

Fig. 1. Flow chart for the distributed time slot allocation algorithm

between parents and children suffices to determine the times when each node needs to be receiving from its children. All other times nodes are guaranteed to be idle, and may choose to go into sleep mode to save energy. Figure 1 shows a flowchart for the distributed time slot algorithm.

III. SIMULATIONS

We will refer to our implementation of the max-min fair solution at the MAC layer as the MMF-TDMA scheme. In our simulations we compare our scheme against two different MAC schemes:

- Overhearing Avoidance MAC: This is similar to the MAC used by S-MAC [7] to benchmark their simulations. The implementation incorporates an RTS/CTS scheme to solve the hidden terminal problem. In this scheme neighbors go to sleep for the duration of data transfer when they receivea CTS from any one of their neighboring nodes.
- Pruned 802.11 MAC: This scheme is very similar to the 802.11 scheme. It incorporates the RTS/CTS mechanism used by 802.11 for solving the hidden node problem. However we have not implemented the ACK's in this scheme (this design decision is common to all three schemes). For our simulations we have assumed a noiseless channel since in a real world scenario the noisy channel can always be made to appear more robust by ACK's.

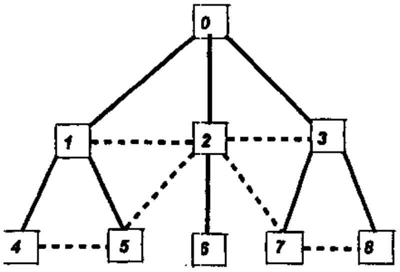

Fig. 2. A 9 node topology used to run the simulation. The dashed links represent the interference in the network. All links are assumed to be bidirectional.

The topology used for running our simulation is shown in Figure 2. The topology is symmetric and the max-min solution for this topology would thus be to allocate equal bandwidth to all sources.

All nodes except the sink are sources. Each source transmits packets of size 40 bytes. Each experiment in the simulation

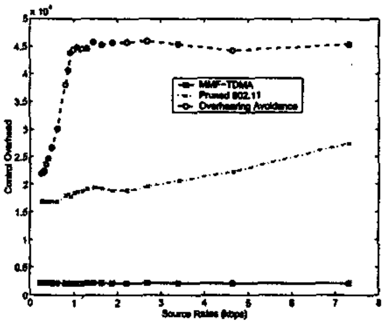

Fig. 3. Control packet overhead vs source rate

consists of all sources generating packets at specified inter arrival times (these inter arrival times are constant). An experiment ends when the sink receives 5000 packets.

The simulator used was Nido (Tossim) [9] the tiny OS simulator provided by UCB.The ability to run the actual nesC code and the ability to provide bit level network simulation that helps us understand the dynamics of packet level collision better, was the motivation for using this simulator.

The channel bandwidth provided by Nido is 20kbps. The MMF-TDMA scheme implementation on Nido initially uses the overhearing avoidance MAC to run the bandwidth allocation algorithm and time slot allocation algorithms on the nodes. Once these algorithms are complete the nodes use the allocated time slot to actually transmit their data.

The metrics presented below help us to compare and evaluate the performance of these schemes. All the metrics have been measured against combined source rates of all sources in the topology. If each source is transmitting at rate x, the combined source rate would be $8x$ (since we have 8 sources in our topology).

Control Packet Overhead: The Figure 3 shows the number of control packets that are required to be sent with increasing source rates. As can be seen the control overhead of the MMF-TDMA scheme remains constant for all inter arrival times. The reason for such a graph is that the only control overhead in the MMF-TDMA scheme is during the bandwidth allocation and time slot allocation algorithms. This would make the number of control packets generated independent of the source rates.

Data Delivery Ratio: Figure 4 shows the data delivery ratio for these schemes. This is the ratio of the number of packets received at the sink (5000) to the number of packets transmitted by the sources. The MMF-TDMA scheme performs much better than the other two randomized MAC schemes because there are no collisions in the MMF-TDMA scheme. The data delivery ratio remains nearly 1 for a large range. This implies that the there are no collisions taking place. However at much higher data rates, the data delivery ratio for MMF-TDMA decreases. However,e ven in this range, no collisions occur, since at high rates the excess packets in MMF-TDMA are dropped at the source itself, avoiding congestion and conserving energy in the process.

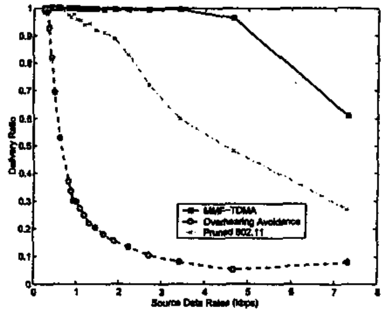

Fig. 4. Data delivery ratio vs source rate

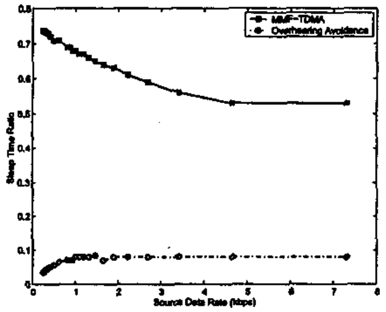

Fig. 5. Sleep time ratio vs source rate

Sleep Time Ratios: As shown in Figure 5 the ratio of the time when nodes are asleep to the total simulation time is much higher for MMF-TDMA than that compared to the overhearing avoidance MAC scheme. Since the nodes follow a pre-conformed schedule of sleep cycles, the sleep times are predetermined. On the other hand the overhearing MAC, fails above a certain value of source(1.5Kbps). The graph for the overhearing avoidance scheme warrants some explanation. When the source rates are really low, neighboring nodes do not hear too many CTS packets and hence cannot sleep too long either. As the source rate increases there is a larger number of CTS packets being heard by the neighbors and hence the sleep ratio improves. However around 1.5kbps this scheme fails, when the network becomes congested. In the baseline pruned 802.11 MAC case, none of the nodes sleep and hence there is no plot representing this MAC on the graph.

Average Throughput: As seen in figure 6, the average throughput (i.e. the incoming data rate at the sink) of the MMF-TDMA is much better than either of the MAC schemes when the source data rates are high. However the throughput values of all three schemes are comparable at lower data rates.

Max-Min Fairness: In figure 7, each point is plotted by obtaining the ratio of the minimum throughput to maximum throughput of all nodes for each source rate value. As can be seen in figure 7, the MMF-TDMA scheme maintains this max-min ratio close to 1 across all data rates, as promised. On the

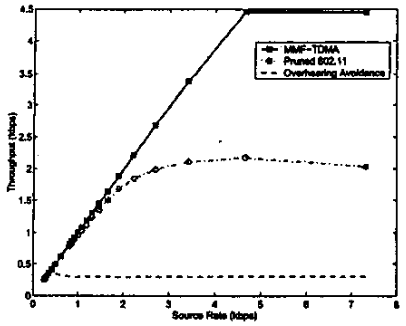

Fig. 6. Average throughput vs source rate

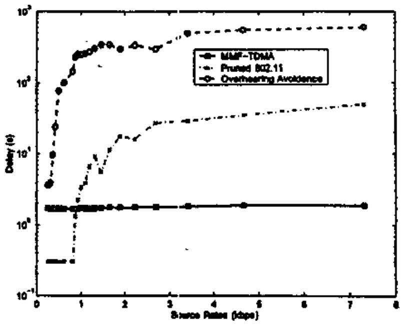

Fig. 8. Average delay vs source rate

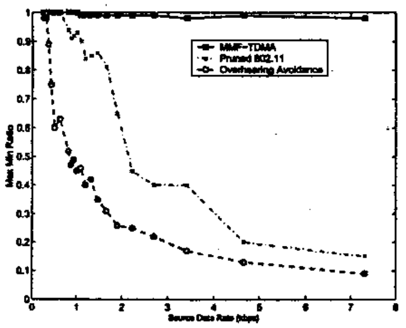

Fig. 7. Max-min Ratio vs source rate

other hand the pruned 802.11 and the overhearing avoidance scheme falter at high data rates although they may be able to achieve this max-min ratio at the lower data rates, when there is no congestion.

Average delay: Figure 8 shows the variation in delay as the source data rates are varied. The Y-axis is in log scale. The overhearing avoidance scheme performs quite poorly to both the other schemes. Between the MMF-TDMA scheme and the pruned 802.11, at lower data rates, the pruned 802.11 outperforms the MMF-TDMA. However as the data rates are increased, MMF-TDMA outperform the pruned 802.11 scheme by avoiding collisions and maintaining a bandwidth allocation mechanism that avoids congestion and reduces delay.

Together these results show convincingly the benefits of using our scheduled MMF-TDMA scheme, particularly at moderate-to-high data rates.

IV. CONCLUSION

We have developed a linear programming formulation for max-min fair scheduling of flows in sensor networks. We have also presented a distributed iterative solution for this problem, which works with a TDMA-based bandwidth allocation scheme to provide fair, collision-free access to bandwidth resources for all sources. Through our Tossim/Nido simulations, we have shown that under high data-rate conditions in sensor networks, the use of such a scheduled access scheme can provide very significant improvements over randomized access schemes in terms of key metrics such as energy, fairness, throughput, and delay.

Future topics related to this work would be to study the scalability of the proposed algorithms and develop algorithms that explicitly handle asymmetric/bidirectional links.

REFERENCES

[1] L. Tassiulas, S. Sarkar, "Max-min Fair Scheduling inW ireless Networks". *In Proceedings IEEE INFOCOM, 2002.*

[2] J. Ros and W. K. Tsai, "An Optimal Distributed Protocol for Fast Convergence to Maxmin Rate Allocation". *Technical Report, University of California Irvine, June 2000.*

[3] T. He, J. A. Stankovic, C. Lu, and T. F. Abdelzaher, "SPEED: A Stateless Protocol for Real-Time Communication in Sensor Networks.". In *International Conference on Distributed Computing Systems (ICDCS 2003).*

[4] Y. Sankarasubramaniam,. B. Akan,I. F. Akyildiz, "ESRT: event-to-sink reliable transport in wireless sensor networks". In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing (Mobihoc 2003).*

[5] Chieh-Yih Wan, Shane B. Eisenman and Andrew T. Campbell, CODA: Congestion Detection and Avoidance in Sensor Networks. In *Proceedings of the First International Conference on Embedded Networked Sensor Systems,(SensSys) Nov 2003.*

[6] A. Woo and D. E.Culler, "A transmission Control Scheme for Media Access in Sensor Networks", Intel Research Labs, IRB-TR-01-003.

[7] W. Ye, J. Heidemann and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks,". In *Proceedings of the 21st International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02), New York, June 2002.*

[8] V. Rajendran, K.Obraczka, J. J. Gracia-Luna-Aceves, "Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks".In *Proceedings of the First International Conference on Embedded Networked Sensor Systems,(SensSys) Nov 2003.*

[9] P. Levis, N. Lee, M. Welsh,A. Woo, and D. Culler, TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications. In *Proceedings of the First International Conference on Embedded Networked Sensor Systems,(SensSys) Nov 2003.*