Scheduling Data Delivery in Heterogeneous Wireless Sensor Networks

Daeseob Lim[1], Jaewook Shim[2], Tajana Simunic Rosing[1], Tara Javidi[2]
[2]*Dept. of Computer Science and Engineering,* [1]*Dept. of Electrical and Computer Engineering*
University of California, San Diego
La Jolla, CA 92093
{dalim, j4shim, tajana}@ucsd.edu, tara@ece.ucsd.edu

Abstract

In this paper we present a proxy-level scheduler that can significantly improve QoS in heterogeneous wireless sensor networks while at the same time reducing the overall power consumption. Our scheduler is transparent to both applications and MAC in order to take the advantage of the standard off-the-shelf components. The proposed scheduling reduces collisions through a generalized TDMA implementation, and thus improves throughput and QoS, by activating only a subset of stations at a time. Power savings are achieved by scheduling transfer of larger bursts of IP packets followed by longer idle periods during which node's radio can either enter sleep or be turned off. Our simulation and measurement results show significant power savings with an improvement in QoS. On average we get 18% of saturation throughput enhancement for real traffic and 79% of power reduction in a highly loaded network.

1. Introduction

Wireless sensor network (WSN) applications abound from monitoring inventory of large warehouses and tracking temperature distributions of server racks to studying wildlife and aiding military in warfare. Although much of the research on WSNs has been primarily focused on design of homogeneous networks, a number of real life examples of large scale WSNs use heterogeneous sensor nodes. An already deployed High Performance Wireless Research and Education Network (HPWREN) in Southern California [1] is a great example of the needs of such a heterogeneous sensor network.

The HPWREN provides high speed wireless network access for a number of different sensors with varying resource requirements, such as large bandwidth requirements of Palomar observatory, low bandwidth but tight real-time traffic deadlines of seismic and acoustic sensor nodes, and long battery lifetime requirements of small and remotely deployed weather stations. It consists of a clustered sensor network with an additional wireless ad hoc overlay in the form of the HPWREN. Small sensor node data is gathered by larger cluster heads that prepare the data for transmission. The data is transmitted when needed to the HPWREN wireless backbone, which routes it out to the internet. Such organization is appropriate for large scale heterogeneous WSNs as the sensor data is often collected in remote and possibly hazardous locations, and thus needs to be both accessed at the location and also forwarded to the internet backbone for further analysis and storage.

Quality of service (QoS) is an important issue in such a sensor network. Although in recent years the issues of QoS and power/energy efficiency have been considered, most QoS schedulers and protocols either target the wired networks that have higher bandwidth, or require significant MAC layer modifications, making their deployment too costly in the existing networks with legacy devices. As a result, providing an effective solution for fully exploiting the potential resources in wireless sensor networks is amiss under the existing solutions. Our work addresses this gap.

In our work we focus on improving QoS while reducing the power consumption in heterogeneous WSNs. Santa Margarita Ecological Reserve (SMER) network, a subnet of the HPWREN, is a good illustration of QoS issues present in a typical heterogeneous WSN. SMER is hierarchically organized in three layers. The lowest layer of the hierarchy consists of sensor nodes which are connected to the corresponding child cluster heads. Child cluster heads in the middle layer collect data from sensor nodes and send them to a parent cluster head through the IEEE 802.11b. Wireless LAN (WLAN) connectivity was chosen due to its higher throughput,

Proceedings of the Eighth IEEE International Symposium on Multimedia (ISM'06)
0-7695-2746-9/06 $20.00 © 2006 IEEE

since child cluster heads need to transmit large amounts of data. Parent cluster heads form a multihop wireless mesh network, part of the HPWREN backbone. Because a large number of child cluster heads are joined to a parent cluster head, child cluster heads compete for 802.11b bandwidth. The theoretical maximum throughput of IEEE 802.11b is just around 7.8Mbps without considering contention [14]. In [26] and [10], it is shown that an increase in the number of competing clients results in a significant decrease of available bandwidth due to collisions in 802.11b. Therefore, in order to improve the throughput, we need to reduce the MAC layer contentions. Scheduling data delivery is one way to accomplish that. Similarly, in order to reduce the energy consumption, we need to have battery powered nodes spend as little time as possible with the wireless turned on.

In this paper, we present a power-aware QoS scheduling mechanism for wireless sensor networks implemented at a proxy layer—between the existing IEEE 802.11 MAC layer and the application layer. Proxy is a good candidate for implementing the scheduler since deployed systems already use proxies for security purposes. It also provides easy access into the lower layers, while having an interface with the application layer. Our proposed scheme is similar to a TDMA(Time Division Multiple Access)-based scheduler in that it regulates collisions by limiting the number of contenders. The main difference is that our scheduler operates effectively over an 802.11 MAC by allowing for more than one user to access the channel in a time slot. Like a conventional TDMA scheme, however, our proposed scheduler has the following problem. When transmission rate of a node is below a permitted throughput bound, the time slot is not fully utilized. However, in SMER network, we find that most applications on sensor nodes generate continual streams of data. In particular, because child cluster heads aggregate traffic from sensor nodes, traffic pattern from child cluster heads makes the likelihood of such underutilization negligible. Most QoS research assumes exponential arrivals for ease of modeling. Recent work in [12] and [11] shows that much of internet traffic is better modeled with a heavy-tailed distribution such as the *Pareto*. We compare power savings and QoS improvements associated with scheduling when using each of the above models, as well as the model we derive from the real data traces. Our ns-2 [4] simulations show that we can obtain large performance improvements under a variety of network conditions and arrival models. In other words, although sensor node traffic in some cases is better modeled with a heavy-tailed distribution (the *Bounded Pareto*), our scheduling scheme achieves significant improvements in QoS metrics and power consumption across a wide variety of traffic models. This ensures the applicability of our result to a large set of problems and sensor network scenarios. The large power savings we obtain illustrate another advantage of our scheduling methodology. Since the nodes transmit bursts of data at a time, they can either sleep or be turned off for a large fraction of the overall time.

The rest of this paper is organized as follows. In section 2, we briefly discuss related work. The design of our scheduler follows in section 3. We describe the analysis and modeling of packet traces from the SMER network in section 4. A description of our verification methodology with simulation results follows in section 5. Finally, we conclude in section 6.

2. Related Work

The interest in wireless networks has lead to a growth in research activities focusing on QoS-aware and/or energy efficient techniques. A difficulty in predicting QoS in wireless network has to do with inherent unpredictability of the wireless channel and limitations in design of commonly used MAC protocols. For example, IEEE 802.11 MAC uses a random backoff mechanism when collisions occur, thus reducing the overall throughput, increasing the power-consumption and the delay. Much of recent work focuses on modifying MAC protocol to improve performance in terms of throughput and energy consumption. Generally, altering MAC implies significant changes in hardware, firmware, and device drivers. It cannot be easily applied to the previously deployed networks without significant additional cost. On the other hand, scheduling above MAC layer gives more flexibility. It can be implemented through software modifications; hence it is more cost effective. Some of the recent work focuses on scheduling data delivery above MAC layer, which we summarize in this section.

There is a lot of research focusing on design of different MAC techniques [5][28][25]. I. Aad *et al*. [5] propose three differentiation mechanisms for IEEE 802.11 MAC which assign different contention window (CW), DIFS, and maximum packet length according to the priority level. SMAC [28] is an energy efficient MAC protocol for wireless sensor network. In SMAC, each sensor node has its own schedule to transmit data. By sharing its schedule with neighbors, a sensor node goes to sleep when there is no data to send or receive. J. Choi *et al*. [9] introduce a new scheme to avoid collisions in IEEE 802.11 MAC protocol, which is called Early Backoff Announcement (EBA). In [8], F. Cali *et al*. provide the capacity analysis of IEEE

802.11 MAC protocol. The authors suggest that the capacity of IEEE 802.11 is improved by adjusting the size of contention window. N. H. Vaidya *et al*. [25] introduce a combined protocol of MAC and Self-Clocked Fair Queueing (SCFQ). In this protocol, the arrival packet is stamped with a start tag and a finish tag. Conceptually, a start tag denotes the arrival time of the packet, and a finish tag means the estimated finish time of the service. Before a station starts the backoff process, a backoff interval is computed according to the flow weight and a finish tag of the packet. The mechanism provides fairness by assigning smaller backoff interval to the packet with smaller finish tag. The authors also extend this scheme to avoid waste of time of a station with huge backoff interval when all contenders complete transmission.

While the above policies require modification of MAC protocol, the following schemes do not alter the existing MAC protocol. The distributed Deficit Round-Robin (DDRR) in [20] is a fair queueing scheduler which combines IEEE 802.11 Point Coordination Function (PCF) and Deficit Round-Robin (DRR). Unfortunately, PCF is not implemented in most commercial wireless network cards. Wireless Rether [22] is originated from software based QoS protocol for wired real-time Ethernet (Rether) [27]. In order to allow for implementation without hardware modification, this scheduler is located between device driver and IP layer. To provide differentiated service, it adopts the concept of token ring. However, since clients are not informed when a token will visit, they have to be awake all the time. Thus, the protocol is inefficient in terms of energy consumption.

The TDMA based protocol in [24] gives a simpler control scheme; a server periodically broadcasts a control packet which contains scheduling information of each client station. A client awakes at a predetermined time to transmit a series of data packets after which it transits to a power-save. Since only one station is activated at any given time, it can complete transmission during a short interval and stay in power-save mode for a long time. Nevertheless, this scheduling scheme does not take care of throughput enhancement. Although the total throughput of a wireless network is mostly inversely proportional to the number of contending stations, the absence of contending stations does not guarantee the maximum throughput, which is shown in section 6.

Another important aspect in the design of a scheduler is to understand the characteristics of the data traffic, specifically the distribution of interarrival times and their effect on various QoS parameters. The *Poisson* process is commonly used to model data arrivals in network theory. However, it has been shown that in most cases network traffic is better modeled with a heavy-tailed distribution. In [19], the authors show that only user-initiated sessions (ftp and telnet connection arrival) have exponential interarrival times, whereas other connections do not. A. Feldmann *et al*. in [11] uses a *Weibull* distribution to model the heavy-tailed traffic. In [18], Paxson shows that various connections such as ftp, telnet, and SMTP can be modeled as exponential by introducing hour by hour modeling methodology. The *Pareto* distribution is used in [23] to model idle times in mobile system. To combat the problem with infeasibility of unbounded burst sizes and interarrival times, [13] proposes the *Bounded Pareto*. In section 4, we use the *Bounded Pareto* and exponential distribution to fit to the real network data. We find that heavy tailed behavior is commonly observed on most sensor nodes observed, even though some sources of data exhibit exponential interarrival patterns. We, then, use these models as well as real traffic traces to verify the benefit of scheduling across all traffic conditions.

In this research, we show how a simple scheduling algorithm can provide significant improvements in terms of QoS and energy consumption. Our scheduler is implemented in a proxy layer, just above the transport layer. The scheduler works with existing MAC (802.11b) and transport layers (TCP, UDP). We propose a simple proxy scheduler enforcing a TDMA like sharing of the wireless channel. There are two main benefits in our scheduling. First, through a reduction in overall packet collisions, our scheduling results in a considerable throughput increase. Secondly, the increase in throughput allows for longer sleep times, and therefore a significant reduction in the power consumption. It has been shown in [16] that the existence of inactive periods during which the users do not contend for the channel can significantly increase the overall network throughput.

3. Proxy Scheduling

In wireless sensor networks, the stations have strict power and bandwidth constraints. The reduction of power consumption is necessary to prolong the lifetime of sensor nodes. Moreover, multiple nodes in the network share limited resources. They contend with each other to gain access to the shared resources. The contention deteriorates the overall throughput of network and increases power consumption due to collisions. In addition, overhearing and idle-listening, which denote receiving packets destined to other stations and listening to an idle channel, are other sources of wasted power [28]. Therefore, if we can avoid collisions, overhearing, and idle-listening, it is

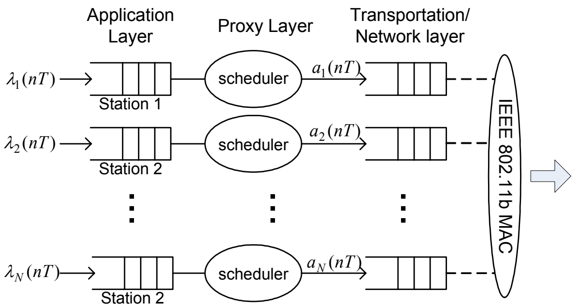


Figure 1. The system model

possible to lower the overall power consumption while improving throughput.

TDMA scheduling can provide these advantages since only the owner of a time slot is allowed to communicate while others nodes are sleeping. Thus, when there are *n* nodes, each node can sleep during (*n*-1) slots. During the allocated time slot, the owner transmits bulk data instead of just a single packet. The larger the time slot, the bigger the size of bulk data and the longer the sleep time. In general, TDMA scheduling has two problems in low traffic scenarios: (1) delay increases in proportion to the number of time slots (or the number of users), and (2) the slots assigned to idle station may be wasted. However, because sensor networks tend to generate steady streams of traffic, we have found that idleness in time slots was not a significant issue. Furthermore, as we show in section 5, in heavy traffic conditions the delay without scheduling is even larger than with scheduling.

As we discussed before, significant changes in the MAC layer are not appropriate for our application due to the cost of replacing the already deployed components and the compatibility with widely available hardware. Transport layer scheduling also requires different implementation for specific protocols. In this case, proxy layer scheduling can be a desirable alternative because it is independent of the application and transport layers. Also, proxy layer scheduling is implemented with minimal cost. In this work, we propose a proxy-level TDMA-type scheduler, shown in figure 1, and investigate its performance. In this model, time is divided into slots whose length is *T*, referred to as a scheduling decision interval. Our scheduler is based on a generalized TDMA scheme. With this scheme, we reduce the contention by activating only a portion of stations in the network, and thus improve the throughput. Since simulations presented in [6] show that maximum throughput in 802.11b is possible with 3 contending stations, we limit the number of concurrently communicating devices to between 2-4.

Scheduler and proxies run as application processes as shown in figure 2. Scheduler determines a new schedule whenever a new child cluster head enters into a network. The schedule is delivered to parent and child heads through parent-side proxy (P-proxy) and child-side proxy (C-proxy). P-proxy is controlled directly by scheduler. It delivers scheduling-related messages to C-proxy. C-proxy executes the actual scheduling command delivered from the P-proxy. It also controls the power states of the wireless network interface card (WNIC). In sleep mode, C-proxy buffers data to the cluster head. P-proxy also buffers the data traffic to the corresponding sensor node. Note that most of traffic from child cluster heads is UDP traffic in the SMER network, while TCP data is a small portion of real traffic. Proxies buffer all the data from applications, thus they can work with any traffic of TCP and UDP. Both proxies send buffered data to each other once the sensor node obtains a communication time slot. The communication channel for control messages between P-proxy and C-proxy uses a long-lived TCP connection as it requires guaranteed delivery of control packets for successful scheduling. This mechanism looks transparent to applications and requires minimum modification of MAC and physical layers. It enables legacy application programs to run without any modification to network socket interface.

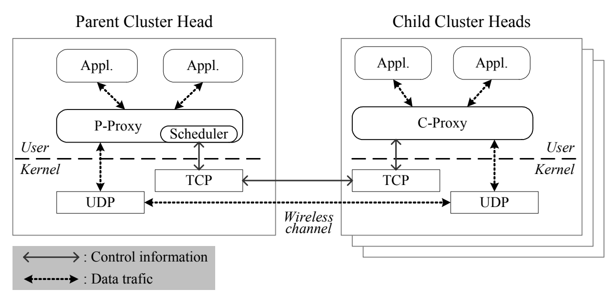


Figure 2. Scheduler and proxy implementation

The only issue on shutting off 802.11 WNIC is a possible loss of packets in a device driver level. To deal with the case where packet transmission is not done, we give a short guard time before actually making WNIC enter a sleep mode. We also note that failure of a child cluster head does not cause any trouble in scheduling. On the other hand, we do assume a parent cluster head does not fail. Our future work will cover this limitation.

4. Traffic Analysis and Modeling

In this section, we analyze the data collected at Santa Margarita Ecological Reserve (SMER). We derive models for packet interarrival times based on our analysis, and verify the models. The models are

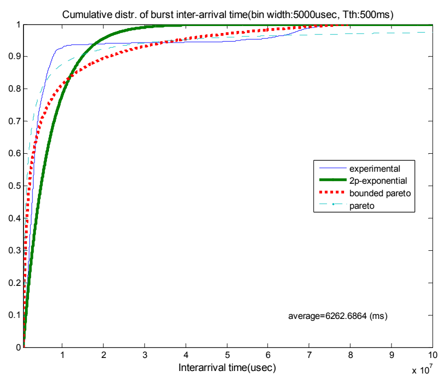


Figure 3. CDF of a cluster head

then used to simulate and to analyze the effectiveness of scheduling algorithm. We use commercial statistical data analysis tool, Minitab, to evaluate Goodness-of-fit between the collected data and the analytic models of 12 different distribution functions such as exponential, Lognormal, Gamma, Weibull, etc.

The data show very non-exponential characteristics, especially for longer idle times that are of interest to us for scheduling purposes. The closest fit we found was to the *2-parameter Exponential* and the *Pareto* distributions. The following equation denotes the cumulative distribution function (CDF) of *2-parameter Exponential* distribution.

$$F(x) = 1 - e^{-\frac{1}{\lambda}(x-\theta)}, \theta \le x \qquad (1)$$

where λ and θ are referred to as *scale* and *threshold*, respectively. The *2-parameter Exponential* function is only valid for x greater than θ.

We also use the *Bounded Pareto* distribution function [13] that is given in (2). There are three parameters that characterize it; k and p are the minimum and the maximum values that we measured in the , respectively. The last parameter, α, is computed by minimizing the mean square error.

$$F(x) = \frac{1-(k/x)^{\alpha}}{1-(k/p)^{\alpha}}, k \le x \le p \qquad (2)$$

We next fit the gathered data from sensor nodes and cluster heads with the *2-parameter Exponential*, the *Pareto*, and the *Bounded Pareto* distributions. Figure 3 shows a cumulative distribution of the burst interarrival time for a cluster head node. The distribution parameters are in table 1. In the figure, the *Bounded Pareto* and the *Pareto* have the good fit to the experimental data. In order to easier simulate the finite arrival times, we use the *Bounded Pareto* in the experiments.

In contrast to camera nodes and cluster heads, the data from seismic sensor nodes can be modeled as a constant bit rate (CBR) of 5.5Kbit/s. In fact, all nodes in our system fall into either of two categories – CBR or heavy tailed *Bounded Pareto*. None can be fit accurately with an exponential distribution. In section 5, we compare the simulation results with our models and the real data.

Table 1. Parameters for cluster head nodes

	k or θ	α or λ	p
Pareto	0.5	0.697	-
Bounded Pareto	0.5	0.421	79.6
2-parameter Exponential	0.5	6.263	-

k, θ, λ & p are in sec

Table 2. Simuation parameter setting in *ns-2*

***ns2* global parameters**	
version of ns2 source files	ns-2.28-snapshot-20050831
Channel model	Channel/WirelessChannel
Propagation model	Propagation/TwoRayGround
Physical model	Phy/WirelessPhy
MAC model	Mac/802_11
Queue model	Queue/DropTail/PriQueue
LL model	LL
Antenna model	Antenna/OmniAntenna
MAC basic rate	1 Mbps
MAC data rate	11 Mbps
RTS threshold	4095 bytes
TCP segment size	1000 bytes

Antenna/OmniAntenna parameters	
X_	0
Y_	0
Z_	1.5
Gt_ (Transmit antenna gain)	1.0
Gr_ (Receive antenna gain)	1.0

Power parameters	
Power in idle mode	0.6698 W
RX power	1.0791 W
TX power	1.7787 W
Power in sleep mode	0.0495 W
Power in mode transition	0.6698 W
Transition time from idle to sleep mode	1 ms
Transition time from sleep to idle mode	100 ms

Phy/WirelessPhy parameters	
L_ (System loss factor)	1.0
freq_ (Channel-13 : 2.472GHz)	2.472e9
bandwidth_ (Data rate)	11Mb
Pt_ (Transmitted signal power)	0.031622777
CPThresh_ (Collision threshold)	10.0
CSThresh_ (Carrier sense power)	5.011872e-12
RXThresh_ (Receive power threshold)	5.82587e-09

Proceedings of the Eighth IEEE International Symposium on Multimedia (ISM'06)
0-7695-2746-9/06 $20.00 © 2006 **IEEE**

5. Results

In this section we compare the standard 802.11b MAC implementation with our scheduler implementation with the various types of traffic. We first experimentally validate our assumptions. Next we compare the effect various intearrival time distributions have on QoS parameters with and without scheduling for three different scenarios: video sensors, seismic sensors and mixed sensors. Finally, we highlight the savings possible with our TDMA scheduling scheme. We use Cisco Aironet WLAN card in all our experiments [1]. Transition times between power states have been obtained experimentally [15].

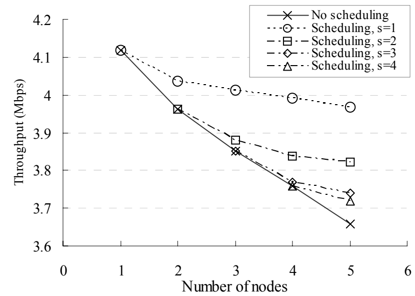
A. Experimental validation

We implemented our TDMA scheduling algorithm on Intel PXA27x developer kit board which has an *XScale* processor. The client boards are equipped with Cisco Aironet 350 PCMCIA wireless LAN adapter. This system represents well the capabilities of sensor node cluster heads. We placed an access point and five client boards in a test room. Each client board runs a client application which sends UDP traffic to the traffic sink. The client's proxy delivers traffic from applications to wireless network, executes the actual scheduling commands and controls the power states of the wireless network interface card.

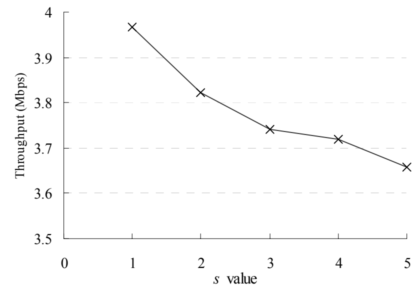
Figure 4 (a) shows the measured aggregate throughput for the case with scheduling for s = 1, 2, 3, and 4, as well as without scheduling. The parameter s specifies the number of clients which can access wireless channel concurrently at a time interval. Clearly, the overall throughput drops as more client machines come into a network. In figure 4 (b), we see the effect of TDMA scheduling for various s values, in particular with the fixed number of clients. We place five client boards and vary only s. The case without TDMA scheduling corresponds to s = 5. In this simulation, it looks like the proper number of scheduled clients in a time slot is very small. Power consumption is shown in figure 4 (c). As expected, the power consumption shows large savings with scheduling as compared to without scheduling.

B. Impact of traffic arrival models on QoS and power

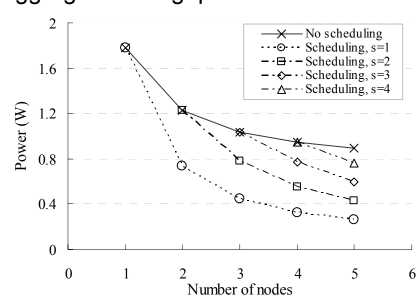
In the next set of results we analyze the impact of sensor data traffic arrival models on various parameters that relate to QoS (throughput, MAC and application layer delay) and power consumption. We simulate and analyze the effect of scheduling on seismic sensor cluster heads, which follow CBR traffic pattern, and general sensor node cluster heads which showed heavy tailed behavior. All simulations are done using NS-2,



(a) Aggregate throughput



(b) Aggregate throughput with different *s* values



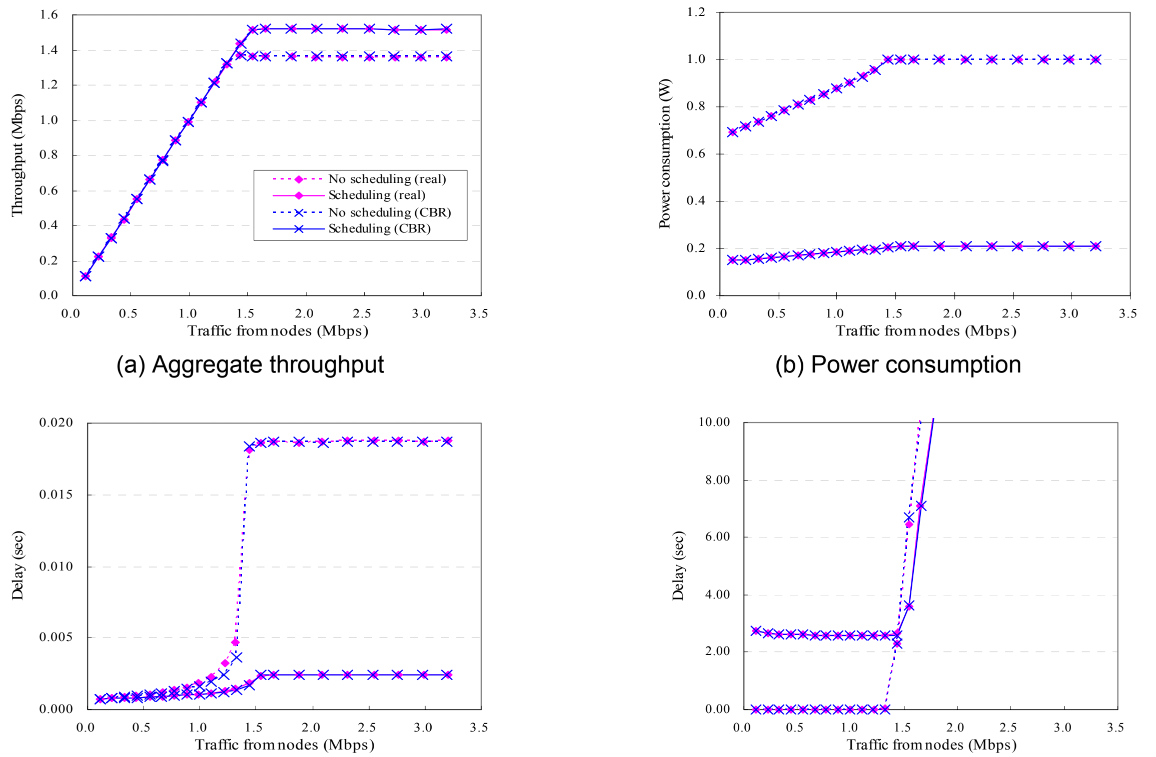
(c) Power consumption per node

Figure 4. Experimental result measured on Intel PXA27x boards

with parameters outlined in Table II. Our findings show that our TDMA based scheduling methodology significantly improves the overall throughput, MAC delay and power consumption irrespective of the specific intearrival time model.

1) CBR traffic: seismic sensors

Much of sensor network consists of nodes that generate relatively low constant bit rate traffic. Some of those nodes also have very tight timing constraints. A good example is seismic sensors which have a constraint of a few seconds for data delivery. In this section we analyze the effect CBR assumption has on both scheduled and unscheduled cases in data delivery of seismic sensors. We change the amount of traffic in the network by varying the number of seismic sensors attached to a cluster head. Since the average packet size for seismic sensors is small, the maximum



(a) Aggregate throughput (b) Power consumption

(c) Delay per packet on MAC layer (d) Delay per packet on application layer

Figure 5. Simulation result for seismic sensors

achievable throughput is also much smaller than ideally possible [17].

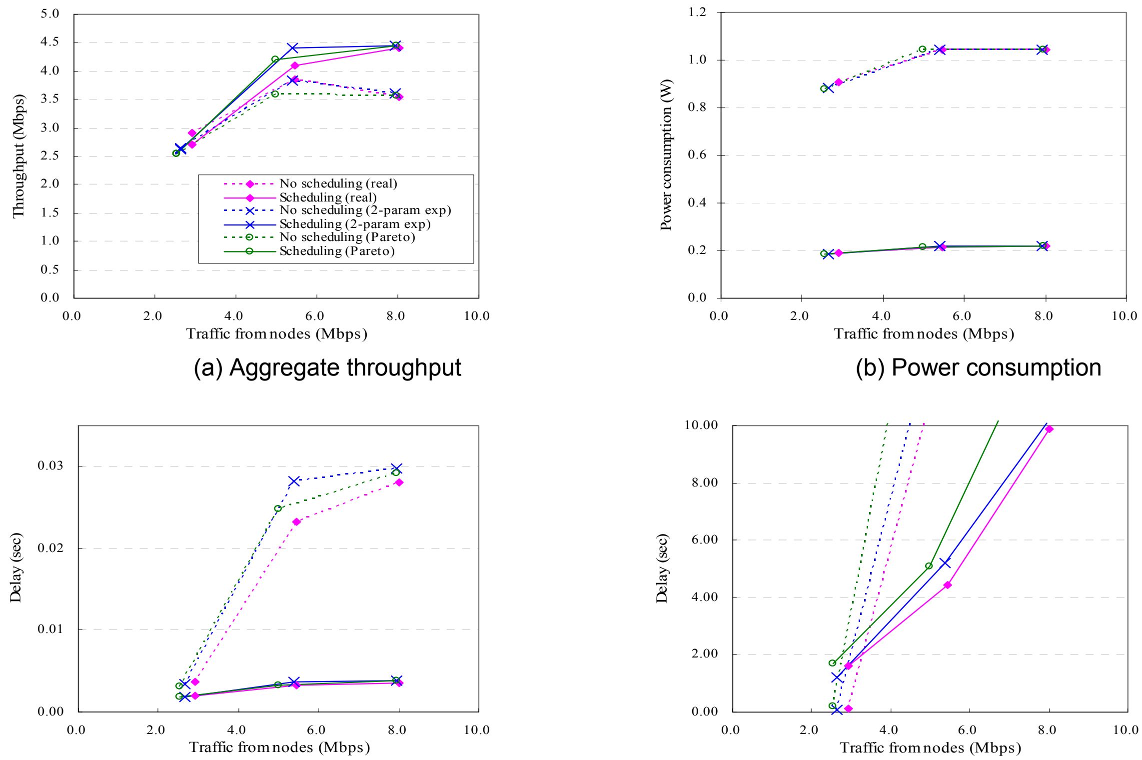
Figure 5 shows the results for seismic sensors. Once the traffic rate reaches saturation point all metrics, except for the application delay, remain stable. MAC layer queues become full nearly at the same time as can be seen by a dramatic increase in MAC collisions represented by an increase in MAC delay as shown in figure 5 (c). At this point, when packet collisions increase dramatically, our TDMA scheduling begins to exhibit its ability to enhance throughput by suppressing packet collisions. The simulation results with CBR traffic model and real traffic data look very similar to each other. The actual traffic log from real seismic sensor nodes show very regular patterns of packet arrival times. Thus, CBR traffic model represents the traffic from seismic sensor nodes very accurately. Power consumption is shown in figure 5 (b). With scheduling 20 nodes, each node stay in sleep mode during 17 time slots per 20 time slots. As a result, the power consumption savings, amounting to 79%, are almost the same regardless of the traffic type

2) Heavy tailed traffic: sensor node cluster heads

Typical sensor node cluster heads gather traffic of various sensors, ranging from relatively small amount of CBR data such as temperature readings or seismic data, to larger bandwidth data such as acoustic or video sensors. The next set of simulations compares no scheduling and TDMA scheduling with traffic models for mixed data transmissions at a cluster head. The child cluster head aggregates data from various sensor nodes: video, acoustic, temperature, seismic, etc. We use two different traffic models: *2-parameter Exponential*, and the *Bounded Pareto*, in addition to real traffic. The fitting parameters for the models are shown in section 4. We increase the amount of traffic arriving to a cluster head from various sensors uniformly to see the effect of both scheduling and modeling on QoS and power consumption.

Figure 6 shows the results with traffic models for cluster head. Again we observe that there is not much of a difference between traffic models. In saturated states the improvement in throughput approaches 24% of the maximum with our scheduler. This higher throughput improvement is partially attributable to the overall decrease of throughput in no-scheduling case. The more traffic is generated, the more chances are that MAC queues are full. It also means that the benefit from our scheduling increases since more traffic is fed into network. In figure 6 (b) and (c), the improvements

Proceedings of the Eighth IEEE International Symposium on Multimedia (ISM'06)
0-7695-2746-9/06 $20.00 © 2006 IEEE



(a) Aggregate throughput

(b) Power consumption

(c) Delay per packet on MAC layer

(d) Delay per packet on application layer

Figure 6. Simulation result for cluster heads

in power and MAC delay show the patterns similar to the previous cases.

As can be seen from results presented above, and summarized in the table III, our TDMA scheduling provides improvements in throughput and energy consumption, coupled with reductions in packet collision rate. When the amount of traffic reaches over the MAC queue saturation point, TDMA scheduling shows the most improvement. The benefit in throughput, obtained from reducing collisions, is at minimum more than 10%. Even when the traffic is under the maximum throughput of IEEE 802.11b wireless network, the throughput overhead from TDMA scheduling is very low. In all of our experiments it is within 2% of the overall throughput.

Power saving are about 79%. The more nodes are in present in the network, the more energy saving we obtain as the length of time each node is asleep lengthens. The upper limit of the amount of power savings is primarily limited by QoS requirements of the applications.

6. Conclusion

In this paper, we presented a TDMA-based scheduling to improve throughput and provide power-saving in a heterogeneous wireless sensor network. The proposed scheduler is at the proxy-level and thus is transparent to both application and MAC layers in order to utilize off-the-shelf IEEE 802.11b radio. Our scheduler activates multiple stations simultaneously to maximize aggregate throughput of the network. The optimal number of nodes to be activated has been experimentally obtained and the performance under realistic traffic models has been investigated. Our measurements on an *XScale* DVK verify the

Table III. Maximum and Average gains in saturation

Node type	Traffic model	Throughput	Power consumption	MAC-layer delay	Collision rate
Cluster head node	2-param Exp	23.24 % *(19.05 %)*	-79.01 % *(-78.92 %)*	-87.11 % *(-87.10 %)*	-74.04 % *(-73.59 %)*
	Bounded Pareto	24.84 % *(21.16 %)*	-79.32 % *(-79.07 %)*	-87.08 % *(-86.93%)*	-76.86 % *(-75.23 %)*
	Real traffic	24.34 % *(15.57 %)*	-79.55 % *(-79.24 %)*	-87.14 % *(-86.72 %)*	-76.41 % *(-75.56 %)*
Seismic sensor	CBR	11.36 % *(11.29 %)*	-79.20 % *(-79.19 %)*	-87.05 % *(-87.01 %)*	-73.17 % *(-72.96 %)*
	Real traffic	12.10 % *(11.67 %)*	-79.25 % *(-79.20 %)*	-87.46 % *(-87.09 %)*	-74.93 % *(-73.28 %)*

Average gain is shown in parentheses below the maximum gain.

assumptions we made in modeling and simulation. To ensure the robustness of our scheduler with respect to the range of arrival models, we have simulated with both real traffic data and analytic models using *ns-2*.

Through simulations, we demonstrated that the proposed TDMA-based scheduling improves saturation throughput, providing significant power saving and delay reduction under heavy traffic condition. The average throughput enhancement has been 13.27% for real traffic, 18.51% for *2-parameter Exponential* traffic model and 16.21% for the *Bounded Pareto*. The average power-saving has been around 79.3% for all traffic models, similar to a reduction of 75.6% (on average) in collision rates across all models. There is less benefit to scheduling in lower traffic loads, motivating the future work in adaptive scheduling schemes.

7. Acknowledgments

We would like to thank the HPWREN project and NSF for their support. The HPWREN is based on work sponsored by the National Science Foundation and its ANIR division under Grant Number 0426879. This work has also been supported through the Center for Networked Systems at UCSD. Finally, the measurements on *XScale* platforms would not have been possible without a generous donation from Intel Corp.

8. References

[1] Cisco Systems, *Data sheet for Cisco Aironet 802.11 a/b/g CardBus Wireless LAN Client Adapter*, http://www.cisco.com.

[2] High Performance Wireless Research and Education Network(HPWREN), http://hpwren.ucsd.edu.

[3] Santa Margarita Ecological Reserve(SMER) Network, http://fs.sdsu.edu/kf/reserves/smer.

[4] Network Simulator 2(*ns-2*) can be found at http://www.isi.edu/nsnam/ns

[5] I. Aad and C. Castelluccia, "Differentiation mechanisms for IEEE 802.11," in *Proceedings of INFOCOM*, April 2001.

[6] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," in *IEEE Journals on Selected Areas in Communications.*, vol. 18, no. 3, pp 535-547, March 2000.

[7] H. Bruneel, "Message Delay in TDMA Channels with Contiguous Output," in *IEEE Transactions of Communications*, Vol, COM-34, No. 7, July 1986.

[8] F. Cali, M. Conti, and E. Gregori, "IEEE 802.11 Wireless LAN: Capacity Analysis and Protocol Enhancement," in *Proceedings of INFOCOM,* 1998.

[9] J. Choi, J. Yoo, and S. Choi, "EBA: An Enhancement of the IEEE 802.11 DCF via Distributed Reservation," in *IEEE Transaction on Mobile Computing*, Vol. 4, No. 4, July 2005.

[10] S. Choi, K. Park, and C. Kim, "On the performance of WLANs: Revisited," in *Proceedings of the 2005 ACM SIGMETRICS*, 2005.

[11] A. Feldmann, "Characteristics of TCP Connection Arrivals," Technical Report, AT&T Labs Research, 1998.

[12] M. Harchol-Balter, "Network Analysis without Exponentiality Assumptions," Ph.D. Thesis. UC Berkeley, December 1996.

[13] M. Harchol-Balter, M. Crovella, and C. Murta, "On choosing a task assignment policy for a distributed server system.," in *IEEE Journal of Parallel and Distributed Computing*, 59:204 – 228, 1999.

[14] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, "Performance anomaly of 802.11b," in *Proceedings of INFOCOM*, March-April 2003.

[15] K. Jamieson, "Implementation of a Power-Saving Protocol for Ad Hoc Wireless Networks," Master degree thesis, MIT, February 2002.

[16] T. Javidi, M. Liu and R. Vijayakumar, "Revisiting Saturation Throughput in 802.11," in *Proc. of the 43th Annual Allerton Conference on Communication, Control and Computing*, October 2005.

[17] J. Jun, P. Peddabachagari, and M. Sichitiu, "Theoretical Maximum Throughput of IEEE 802.11 and its Applications," in *Proceedings of Second IEEE International Symposium on Network Computing and Applications*, April 2003.

[18] V. Paxson, "Empirically-Derived Analytic Models of Wide-Area TCP Connections," in *IEEE/ACM Transactions on Networking*, Vol.2 pp 316-336, August 1994.

[19] V. Paxson and S. Floyd, "Wide-Area Traffic: The Failure of Poisson Modelling," in *IEEE/ACM Transactions on Networking*, Vol.3 pp 226-244, June 1995.

[20] R. S. Ranasinghe, D. Everitt, and L. L. H. Andrew, "Fair Queueing Scheduler for IEEE 802.11 Based Wireless Multimedia Networks," in *Proceedings of MMT*, 1999.

[21] A. Rao and I. Stoica, "An Overlay MAC Layer for 802.11 Networks," in *Proceedings of MobiSys*, June 2005.

[22] S. Sharma, K. Gopalan and N. Zhu, "Implementation Experiences of Bandwidth Guarantee on a Wireless LAN," in *Proceedings of MMCN*, 2002.

[23] T. Simunic, L. Benini, P. Glynn, G. D. Micheli, "Dynamic Power Management of Portable Systems," in *Proceedings of MobiCom*, August 2000.

[24] J. Snow, W. Feng, and W. Feng, "Implementing a Low Power TDMA Protocol Over 802.11," in *Proceedings of WCNC*, March 2005.

[25] N. H. Vaidya, P. Bahl, and S. Gupta, "Distributed Fair Scheduling in a Wireless LAN," in *Proceedings of Sixth Annual International Conference on Mobile Computing and Networking*, August 2000.

[26] A. Vasan and U. Shankar, "An empirical characterization of instantaneous throughput in 802.11b WLANs," Technical Report CR-TR-4389, UMIACS-TR-2002-69, University of Maryland College Park, 2002.

[27] C. Venkatramani and T. Chiueh, "Supporting Real-Time Traffic on Ethernet," in *Proceedings of IEEE Real-Time Systems Symposium*, 1994.

[28] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," in *Proceedings of INFOCOM,* 2002.

[29] H. Zhu and G. Cao, "A Power-Aware and QoS-Aware Service Model on Wireless Networks," in *Proceedings of INFOCOM*, March 2004.

[30] High Performance Wireless Research and Education Network (HPWREN), http://hpwren.ucsd.edu/