

# Energy-Efficient Duty Cycle Assignment for Receiver-Based Convergecast in Wireless Sensor Networks

Yuqun Zhang, Chen-Hsiang Feng, Ilker Demirkol, Wendi B. Heinzelman

Department of Electrical and Computer Engineering  
University of Rochester  
Rochester, NY, USA  
Email: {yuzhang, feng, demirkol, wheinzel}@ece.rochester.edu

**Abstract**—Duty cycling is often used to reduce the energy consumption caused by idle listening in Wireless Sensor Networks (WSNs). Most studies on WSN protocols define a common duty cycle value throughout the network to achieve synchronization among the nodes. On the other hand, a few studies propose adaptation of the duty cycle according to uniform traffic conditions, which is beneficial assuming one-to-one traffic patterns that result in evenly distributed packet traffic. In this work, we consider the convergecast communication pattern commonly observed in WSNs. In convergecast communication, the packet traffic observed around the sink node is much higher than the traffic observed far from the sink, i.e., nodes with different distances to the sink node receive and must relay different amounts of traffic. Additionally, we utilize receiver-based protocols, which enable nodes to communicate with no synchronization or neighbor information, and hence do not require all nodes in the network to have the same duty cycle. In this paper, we model the expected energy consumption of nodes utilizing receiver-based protocols as a function of their duty cycle and their distance to the sink node. Using this analysis, we derive a closed-form formula for the duty cycle that minimizes the expected energy consumption at a given distance. Moreover, we propose an adaptation method for the derived distance-based duty cycle, based on local observed traffic. Performance evaluations of the two proposed duty cycle assignment methods show that they greatly improve the energy efficiency without sacrificing packet delivery ratio or delay significantly.

**Index Terms**—Receiver-based protocols, convergecast traffic, adaptive duty cycle, energy-efficiency, wireless sensor networks.

## I. INTRODUCTION

Duty cycling, where a node is periodically placed into the sleep mode, is an effective method of reducing energy dissipation in Wireless Sensor Networks (WSNs). The lower the duty cycle, the nodes can sleep longer and the more energy they will save, whereas the fewer nodes are available to participate in data routing at any given time, which will increase transmission latency and decrease the throughput. Thus, there is a trade-off between energy efficiency, transmission latency, and throughput, determined by the duty cycle used in the network.

Duty cycle is typically fixed throughout the network, with all nodes utilizing the same duty cycle. However, this may not

provide the best overall performance for the network. Many sensor network applications require convergecast communication, where data from sensors are transmitted to a sink in the network. In this type of communication pattern, nodes close to the sink must transmit much more data than nodes far from the sink, and hence the duty cycles of the nodes should be adjusted appropriately to ensure energy efficiency while meeting traffic demands and keeping latency low.

Recently, a new class of protocols, called receiver-based routing, has been proposed as a means of allowing communication when nodes are not aware of the exact duty cycle of their neighbors. In receiver-based routing, receivers contend to be the next-hop router of a packet, and the transmitter selects the “best” receiver under a given optimality criteria to become the next hop for transmission. For example, in the receiver-based protocol Implicit Geographic Forwarding (IGF) [1], all receivers contend to be the next-hop router when they hear a packet route request, and the transmitter selects the receiver that is closest to the sink as the next hop. Specifically, the transmitter initiates communication by sending an RTS packet that indicates the transmitter’s location and the location of the sink. Nodes that hear the RTS packet first determine whether they make forward progress to the sink, and, if so, they calculate their distance to the sink. After a delay proportional to their distance to the sink, nodes send a CTS packet back to the transmitter. The first node that sends a CTS packet is selected as the next hop by the transmitter, and the transmitter forwards the data packet to that node.

Researchers have analyzed the performance of receiver-based routing through mathematical models [2] [3] and shown that receiver-based routing protocols perform well in terms of hop distance, energy and latency. Unicast traffic is assumed in these works, hence, for convergecast traffic further studies are required. Extensions to traditional receiver-based routing have included providing information about link quality for making routing decisions [4], and supporting multiple paths by strategically selecting relay nodes and employing adaptive rate control [5]. Utilizing duty cycling with receiver-based routing and convergecast data patterns, it is clear that a network-wide

fixed duty cycle will not provide the optimal trade-off between energy efficiency and latency.

Adapting the duty cycle to the local traffic was proposed in PMAC [6], where the sleep-wakeup schedule is represented by a string of bits that are updated each period using local traffic information available at the node. These schedules are exchanged at the end of each period, so that neighboring nodes are aware of each others' schedules. Another adaptive duty cycle approach, ALPL, adjusts a node's duty cycle according to the node's neighbors' duty cycles in order to support the data flows it receives [7]. However, none of these approaches optimize the duty cycle for convergecast data patterns and receiver-based routing.

In this paper, we derive a mathematical model to determine the energy dissipation of a node as a function of its duty cycle and its distance to the sink for convergecast data patterns and receiver-based routing. Using this model, we find the duty cycle as a function of node distance to the sink to minimize the expected energy dissipation. Additionally, in order to balance energy efficiency and latency, we develop a traffic-adaptive duty cycle approach that begins with the distance-based duty cycle assignment and adapts the duty cycle based on current local traffic patterns observed by the node. In receiver-based protocols, the number of retransmitted RTS packets provides an indication of the traffic. Under heavy traffic, nodes must generate many retransmitted RTS packets. If the number of retransmitted RTS packets outnumbers the number of original RTS packets, nodes should increase their duty cycle in order to alleviate the traffic congestion; otherwise, they should decrease their duty cycle to save energy. This approach allows the duty cycle to be tuned to trade-off energy and latency for observed local traffic patterns.

Specifically, two duty cycle assignment methods are proposed in this paper: Distance-based Duty Cycle Assignment (DDCA), where the duty cycle is assigned to each node based on the node to sink distance, and Traffic-Adaptive Distance-based Duty Cycle Assignment (TDDCA), where the duty cycle is initialized to the one given by the DDCA method and adapted to the traffic as explained above. Simulation results show that DDCA and TDDCA reduce energy consumption compared with the commonly used, network-wide constant duty cycle method. Additionally, TDDCA reduces latency at the expense of a small increase in energy consumption compared with DDCA, which indicates that TDDCA is able to trade-off the lower latency of the network-wide constant duty cycle method and the energy efficiency of the distant-based duty cycle method (DDCA).

## II. DISTANCE-BASED DUTY CYCLE ASSIGNMENT METHODS

The traffic relayed at a node is related to its distance to the sink, the number of source nodes in the network, the packet traffic generated by each source node, and the node density. In this section, we present this relationship analytically, then, given the average traffic observed at a node,

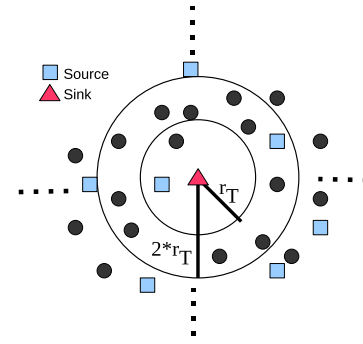


Fig. 1. Sample network topology.

we derive the expected duty cycle for minimizing the expected energy consumption of the node..

### A. Traffic Rate Analysis

For the analysis, we assume a circle area with the sink located in the center and the nodes including the sources uniformly randomly allocated as illustrated in Fig. 1, where  $r_T$  is the transmission range. We define the  $n^{th}$  ring to be the ring whose inner circle is  $(n-1)r_T$  away from the sink with width  $r_T$ . Hence, the  $n^{th}$  ring contains the nodes that are  $n$ -hops away from the sink. Let there be  $N_n$  nodes in this ring. The average traffic that must be relayed by all of the nodes located in the  $n^{th}$  ring per unit time,  $\Gamma_n$ , is the summation of the traffic generated by the source nodes in the  $n^{th}$  ring and within the rings outside of the  $n^{th}$  ring per unit time, i.e.,

$$\Gamma_n = \lambda_g \rho_s \pi (R^2 - [(n-1)r_T]^2), \quad (1)$$

where  $\lambda_g$  is the average traffic generation rate of the source nodes,  $\rho_s$  is the density of source nodes, and  $R$  is the radius of the network area.

Since  $\Gamma_n$  is the average traffic relayed by all nodes per unit time in the  $n^{th}$  ring, a node within that ring relays a traffic with a mean  $\lambda_r = \Gamma_n / N_n$  packets per unit time. A node with a distance  $r$  to the sink resides in the  $n = \lceil \frac{r}{r_T} \rceil$  ring and the number of nodes in the  $n^{th}$  ring is

$$N_n = \rho_r \pi \{ (nr_T)^2 - [(n-1)r_T]^2 \}, \quad (2)$$

where  $\rho_r$  is the density of nodes. Hence, the average traffic rate of a node at distance  $r$ ,  $\lambda_r$ , is

$$\lambda_r = \frac{\lambda_g \rho_s \pi \{ R^2 - [(\lceil \frac{r}{r_T} \rceil - 1)r_T]^2 \}}{\rho_r \pi \{ [(\lceil \frac{r}{r_T} \rceil)r_T]^2 - [(\lceil \frac{r}{r_T} \rceil - 1)r_T]^2 \}}. \quad (3)$$

### B. Duty Cycle for a Given Expected Traffic Rate

The time required for a transmission and the energy efficiency of the network is closely related to the duty cycle values used. Higher duty cycle values provide more nodes available for data routing, such that the possibility to have no relay nodes is decreased and a lower latency is achieved, yet they consume more energy. In this section, we derive the duty cycle that minimizes the energy consumption for a given traffic rate.

In [3], a similar derivation is done for unicast traffic, where every node can be a source or a destination. We adapt the analysis presented in [3] for the following MAC protocol modifications proposed. Although a receiver-based MAC protocol is analyzed in [3], our simulation results showed a high number of collisions and high CTS traffic load for the MAC protocol investigated therein. To reduce the number of collisions and the CTS traffic load, the MAC protocol is modified as follows. In [3], the relay region (locations with geographic advancement to the sink) is divided into  $N_p$  priority regions, and each region is assigned a contention slot such that priority region  $i$  is assigned the  $i$ th slot in the contention window. We assign each priority region  $N_r$  CTS contention slots, such that priority region  $i$  is assigned the slots  $((N_i - 1) \times N_r, N_i \times N_r - 1)$ . This reduces CTS collisions, as all nodes in priority region  $i$  can select one of the  $N_r$  CTS contention slots to send their CTS packet.

The following duty cycle analysis is based on the idea that the expected energy consumption of a sensor node is proportional to the expected total awake time,  $t_l$ , of the node. This is because, the radio idle listening power is approximately the same as the transmission and reception power in WSNs [8]. Hence, a constant power value  $P$  is assumed for idle listening, transmission, and reception.

Let  $N$  denote the average number of nodes within a node's transmission range,  $d$  denote duty cycle, and  $\lambda_r$  denote the average traffic rate of a node located at distance  $r$  to the sink node given in (3). Assuming a Poisson or uniform packet generation rate, the average traffic rate of a node follows the Poisson distribution. The probability that a node detects no traffic can be calculated to be  $e^{-\lambda_r N T_L}$  where  $T_L$  is its listen period at each cycle and  $\lambda_r N$  is the average packet arrival rate within its transmission range. Thus, the probability that a node detects any ongoing traffic is  $p_0 = 1 - e^{-\lambda_r N T_L}$ . If  $\xi$  is the ratio of the relay region (i.e., the region in which nodes make forward progress to the sink) to the transmission area,  $p_0 \xi$  is the probability of a node detecting ongoing traffic and residing in the relay region of that traffic.

When a node has a packet to send, it sends an RTS packet and keeps retransmitting the RTS packet until receiving a CTS packet. The expected number of RTS transmissions needed before the first successful RTS/CTS handshake is

$$\sum_{i=1}^{\infty} i(1-p_1)^i p_1 = \frac{1-p_1}{p_1} = \frac{1-e^{-\xi d N}}{e^{\xi d N} - 1} \quad (4)$$

where  $p_1 = 1 - e^{-\xi d N}$  is the probability that at least one node replies to the RTS packet, since the number of nodes residing in an area can be approximated by Poisson distribution for uniformly random deployment [9]. For each retransmission, the node sends out an RTS packet and waits for  $N_p \times N_r$  CTS slot durations. The expected time needed before the first successful RTS/CTS handshake,  $t_H$ , is then

$$t_H = (e^{\xi d N} - 1)^{-1} (T_{RTS} + N_p N_r T_{CTS}), \quad (5)$$

where  $T_{RTS}$  and  $T_{CTS}$  are the transmission delays for RTS and CTS packets, respectively.

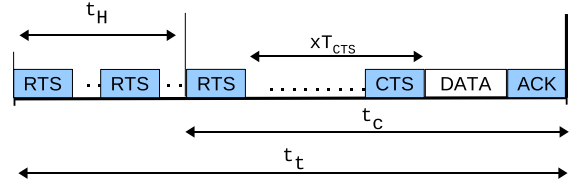


Fig. 2. Representation of packet exchange durations.

As illustrated in Fig. 2, the expected total time for a complete RTS, CTS, DATA and ACK packet communication is

$$t_C = T_{RTS} + xT_{CTS} + T_{DATA} + T_{ACK}, \quad (6)$$

where  $x$  represents the number of CTS contention slots up to and including the first successful CTS packet,  $T_{DATA}$  and  $T_{ACK}$  are the required times for DATA and ACK packets, respectively. The formula for  $x$  can be calculated from a standard CSMA model, and we omit it here for the sake of brevity. Therefore, the expected total time for a node to transmit a packet, including all the failed RTS packets and the successful data exchange is  $t_t = t_H + t_C$ .

The expected time for a node to receive an RTS packet during a listening period is  $\frac{T_L}{2}$ . An approximation for the probability that a node wins the contention and is selected as the relay node is given in [3] as  $\frac{1 - e^{-\xi d N}}{\xi d N}$ . Then, the average active time of a node that receives traffic and that resides in the relay region of the sender node is

$$t_1 = \frac{T_L}{2} + \frac{1 - e^{-\xi d N}}{\xi d N} t_C + \left(1 - \frac{1 - e^{-\xi d N}}{\xi d N}\right) \frac{T_L}{2}. \quad (7)$$

Finally, the expected time a node is awake during one listen period is  $t_l = (1 - p_0 \xi) T_L + p_0 \xi t_1$ , where  $(1 - p_0 \xi)$  represents the probability that either a node hears no traffic or hears some traffic but is not in the relay region, in which case the node is awake for  $T_L$  time.

The expression for the expected energy consumption  $\bar{P}$ , then, can be derived as

$$\begin{aligned} \bar{P} &\simeq P \times \frac{t_l}{T_L} + \lambda_r P t_t \\ &\simeq P \{d + \lambda_r [(e^{\xi d N} - 1)^{-1} (T_{RTS} + N_p N_r T_{CTS}) \\ &\quad + (2 - e^{-\xi d N}) (T_{RTS} + x T_{CTS} + T_{DATA} + T_{ACK})]\} \\ &\simeq P \{d + \lambda_r \{[(e^{\xi d N} - 1)^{-1} (1 + N_p N_r) + 2 + \\ &\quad x] T_{CTS} + 2 T_{DATA}\}\} \\ &\simeq P \{d + \lambda_r [(e^{\xi d N} - 1)^{-1} N_p N_r T_{CTL} + 2 T_{DATA}]\}, \end{aligned} \quad (8)$$

where  $T_{RTS} \simeq T_{CTS} \simeq T_{ACK} = T_{CTL}$ , and  $1 - e^{-\lambda_r N T_L} \simeq \lambda_r N T_L$  when  $\lambda_r N T_L \ll 1$ . Since  $x T_{CTS}$  is dominated by the other components in the formula, it is eliminated as a simplification.

We take the derivative of the expected energy consumption function with respect to  $d$  and set it to zero to find the duty cycle that minimizes the expected energy consumption. The duty cycle resulting in  $E_{min}$  is  $d_{opt} = \frac{\log[\frac{\alpha+2+\sqrt{\alpha(\alpha+4)}}{\xi N}]}{\xi N}$  where  $\alpha = \lambda_r \xi N N_p N_r T_{CTS}$ . Finally, the mathematical relation between duty cycle and average traffic rate is derived. The

value of  $\lambda_r$  for a node is found with the analysis presented in Section II-A.

### C. Duty Cycle Assignment Methods Proposed

The Distance-based Duty Cycle Assignment (DDCA) method defines the duty cycle of a node to be the duty cycle based on the analysis presented in Sections II-A and II-B. Since analysis do not take packet contention and collision into consideration, we round up the duty cycle found by DDCA to be  $\lceil \frac{100d}{100} \rceil$ . Although the analysis presented considers expected traffic observed by a node at a given distance, in practice the actual traffic loads vary per node and over time. Moreover, the entire analysis focuses on minimizing energy consumption while leaving the end-to-end delay performance as a later concern. Aiming to solve these problems, we also propose a distance-based duty cycle assignment scheme combined with the actual traffic pattern observed. In general, the receiver-based protocols do not exchange any traffic information between nodes to achieve stateless communication. However, RTS packets can be used to observe the traffic load. The number of retransmitted RTS packets increases either when a node's duty cycle is too low and no relay candidates can be found, or when the traffic load is too high and the high contention of nodes causes collisions of the RTS packets from different transmitters. For either case, increasing the duty cycle would increase the probability of successful communication.

We introduce a piggyback flag to the original packet header of the RTS packet to indicate whether this packet is being retransmitted or not. A counter is also set in every node to record the numbers of the initial and retransmitted RTS packets. If the total number of the received retransmitted RTS packets in the current cycle outweighs the total number of the received initial RTS packets, it indicates severe contention in the neighborhood, and the duty cycle of the node is increased to mitigate the traffic load. Otherwise, the duty cycle is decreased every cycle down to a minimum of 1% to minimize the energy consumption. This method is called Traffic-Adaptive Distance-based Duty Cycle Assignment (TDDCA). TDDCA is expected to improve the latency performance, since it takes into account not only the distance-based duty cycle assignment, but also the spatiotemporal traffic information in a particular network deployment.

## III. PERFORMANCE EVALUATION OF DUTY CYCLE ASSIGNMENT

Simulations are performed using the OPNET simulator to compare the two methods proposed, namely DDCA and TDDCA, with the network-wide constant duty cycle assignment method. In the network-wide constant duty cycle method, the duty cycle is set to the duty cycle found by the DDCA method for the nodes one hop away from the sink, such that a high packet delivery ratio is guaranteed.

The performance metrics evaluated are packet delivery ratio, average energy consumption, and average latency. The radius of the target area  $R$  is set to be 90 m and the transmission range  $r_T$  for all nodes is set to be 30 m. For simplicity,

we assume the relay region ratio is constant and set to 0.4 when determining the DDCA duty cycle, and the power for transmission, reception and idle listening is set to 1 unit. The sink is located in the center of the area, where 400 nodes are uniformly randomly deployed. In TDDCA, the duty cycle is changed by 1% every listening interval based on the observed RTS retransmissions.

Two sets of simulations are performed to investigate the performance of the presented duty cycle assignment methods for a varying number of sources and a varying packet generation rate,  $\lambda_g$ . The effect of the number of sources is investigated for a packet generation rate of 0.5 packet/sec and the effect of the packet generation rate is investigated for 40 sources.

The packet delivery ratio (PDR) values achieved by the three methods are presented in Fig. 3(a) and Fig. 4(a). In all three methods, the PDR results are very close and higher than 97% for light traffic loads. With an increase in traffic load, the constant duty cycle method performs the best because its higher duty cycle can provide more awake nodes to participate in data routing. The slightly worse performance of TDDCA compared to the constant duty cycle method indicates that the fixed increments and decrements in duty cycle is not efficient in terms of PDR. One alternative is to use varying duty cycle increments and decrements as proposed in [10].

While PDRs are approximately the same using all three methods, Figs. 3(b) and 4(b) both show that TDDCA and DDCA are more energy-efficient than the constant duty cycle method, and that DDCA performs better than TDDCA. DDCA reduces energy dissipation between 21% and 32% compared to the constant duty cycle method, while TDDCA reduces energy dissipation between 12% and 19% compared to the constant duty cycle method. Because the entire network is likely to generate more retransmitted RTS packets than original RTS packets, TDDCA increases duty cycle more often than decreasing it. The reason is as follows: in the area near the sink where traffic is heavy, available nodes that receive the first RTS packet turn to a busy state until they win the contention or receive a CTS packet from another node for the same RTS packet. In this busy state, receivers do not reply RTS packets from other transmitters, which results in retransmitted RTS packets even when there are awake nodes within nodes' transmission ranges. On the other hand, in the area far from the sink where traffic is light, the duty cycles of nodes are low such that it is possible that there are no awake nodes that can hear an RTS packet when it is broadcasted. Thus, retransmitted RTS packets are generated in this case as well. Generally the fact that TDDCA increases the duty cycle more often than decreasing it leads to its larger average energy consumption than DDCA.

Figs. 3(c) and 4(c) show that TDDCA performs the best in terms of latency. In light traffic, TDDCA achieves better latency values compared with DDCA, e.g., latency using TDDCA is 30% less than latency using DDCA when the number of sources is 20. Since nodes are likely to increase their duty cycle rather than to decrease it, in TDDCA there are more nodes available to contend for the channel and latency

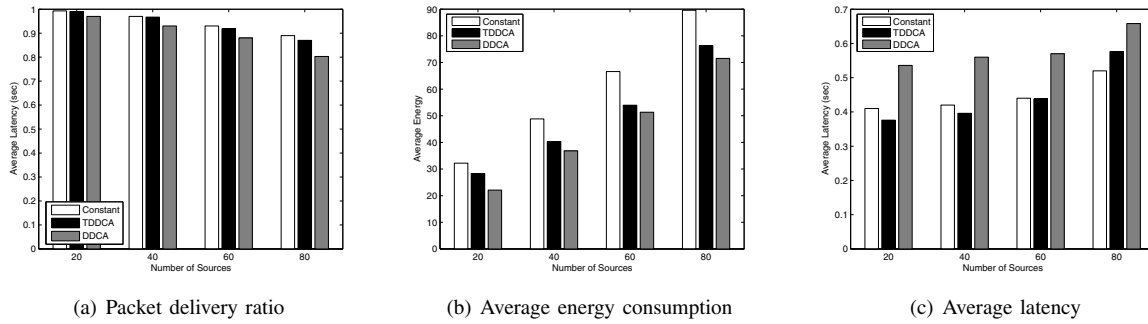


Fig. 3. Simulation results in terms of the number of sources.

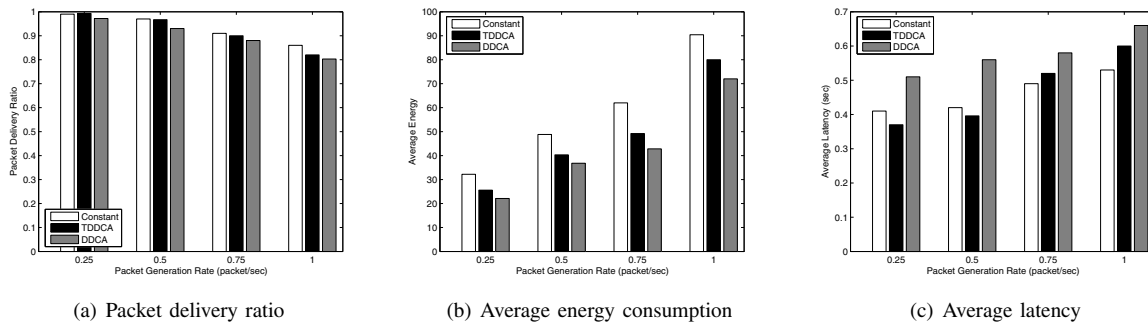


Fig. 4. Simulation results in terms of source packet generation rate  $\lambda_g$ .

is reduced compared with DDCA. It is also shown that in heavy traffic, TDDCA performs worse in terms of latency compared with the constant duty cycle method. This is because under the severe impact of packet collisions and contention, traffic patterns vary between every listening interval such that a simple comparison between the number of original RTS packets and retransmitted RTS packets cannot reflect the current level of traffic accurately enough. Hence, the method of changing duty cycles by 1% in each listening interval is not effective to achieve a low latency in high traffic conditions.

In summary, both DDCA and TDDCA are more energy-efficient than the constant duty cycle method, while achieving similar packet delivery ratio performance. Compared with DDCA, TDDCA has an advantage in terms of latency.

#### IV. CONCLUSION

In this paper, we derived the duty cycle for a node as a function of its distance to the sink to minimize expected energy consumption for convergecast traffic patterns and receiver-based routing. Based on our analysis, we developed two duty cycle assignment algorithms. Simulation results show that both methods decrease energy consumption compared with the constant duty cycle method by up to 32% for the scenarios investigated. The traffic-adaptive distance-based duty cycle assignment method achieves energy improvements without sacrificing from the latency and throughput significantly. The analysis can be extended as future work to improve the performance of distance-based duty cycle assignment in heavy

traffic scenarios, by taking the packet collisions and contention into account.

#### REFERENCES

- [1] B. Blum, T. He, S. Son, and J. Stankovic, "IGF: A state-free robust communication protocol for wireless sensor networks," in *Proc. 3rd IEEE Workshop on Applications and Services in Wireless Networks*, 2003.
- [2] M. Zorzi and R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: multihop performance," *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 337–348, Oct.-Dec. 2003.
- [3] M. Zorzi and R. R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Energy and latency performance," *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 349–365, 2003.
- [4] I. Akyildiz, M. Vuran, and O. Akan, "A cross-layer protocol for wireless sensor networks," in *Proc. of CISS 2006*, March 2006.
- [5] E. Rozner, J. Seshadri, Y. A. Mehta, and L. Qiu, "SOAR: Simple opportunistic adaptive routing protocol for wireless mesh networks," *IEEE Trans. on Mobile Computing*, vol. 8, no. 12, pp. 1622–1635, 2009.
- [6] T. Zheng, S. Radhakrishnan, and V. Sarangan, "PMAC: An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proc. IPDPS '05*. IEEE Computer Society, 2005, p. 237.1.
- [7] R. Jurdak, P. Baldi, and C. Videira Lopes, "Adaptive low power listening for wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 8, pp. 988–1004, 2007.
- [8] Crossbow Technology, "http://www.xbow.com/." [Online]. Available: <http://www.xbow.com/>
- [9] I. Demirkol, C. Ersoy, F. Alagoz, and H. Delic, "The impact of a realistic packet traffic model on the performance of surveillance wireless sensor networks," *Computer Networks*, vol. 53, no. 3, pp. 382 – 399, 2009.
- [10] C. J. Merlin and W. B. Heinzelman, "Duty cycle control for low power listening MAC protocols," in *In Proc. MASS08*, 2008.